To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 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 1st, 2010
Renesas Electronics Corporation

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

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

RENESAS

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.

RENESAS

Customer Notification

QB-V850ESFX2™

In-Circuit-Emulator

Operating Precautions

Target Device

V850ES/FE2
V850ES/FF2
V850ES/FG2
V850ES/FJ2
V850ES/HE2
V850ES/HF2
V850ES/HG2
V850ES/HJ2
µPD703229Y
µPD70F3229Y

Global Document No. U18081EE4V0IF00 (4th edition)
Document No. TPS-HE-B-3213
Date Published November 2005

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 NEC 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 NEC 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 agree and acknowledge 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.

Applicable Law:

The law of the Federal Republic of Germany applies to all information provided by NEC to the Customer under this Operating Precaution document without the possibility of recourse to the Conflicts Law or the law of 5th July 1989 relating to the UN Convention on Contracts for the International Sale of Goods (the Vienna CISG agreement).

Düsseldorf is the court of jurisdiction for all legal disputes arising directly or indirectly from this information. NEC is also entitled to make a claim against the Customer at his general court of jurisdiction.

If the supplied goods/information are subject to 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 Corporation and also includes its direct or indirect owned or controlled subsidiaries.

2. "PRODUCT(S)" means 'NEC semiconductor products' (NEC semiconductor products means any semiconductor product developed or manufactured by or for NEC) and/or 'TOOLS' (TOOLS means 'hardware and/or software development tools' for NEC semiconductor products' developed, manufactured and supplied by 'NEC' and/or 'hardware and/or software development tools' supplied by NEC but developed and/or manufactured by independent 3rd Party vendors worldwide as their own product or on contract from NEC)

(A) Product Version

1. Product Code: QB-V850ESFX2

Control Code[Note]	EVA Chip	I/O Chip
A	uPD703195 ES1.0	uPD70F3239, ES2.0
B	uPD703195A ES1.0	uPD70F3239, ES2.0
C	uPD703195A ES1.0 or uPD703195A ES1.1	uPD70F3239, ES3.0

2. The Control Code is indicated by the letter appearing at the 2nd position from the left in the serial number of the product.

Caution: In conjunction with the usable exec version a qualified device file (Dxxxxx.800) is additionally necessary for the corresponding device, which has to be emulated. Make sure that you use the appropriated version of the device file.

(B) Table of Operating Precautions

No.	Outline	Control-Code[Note]	QBV850ESFX2		
			A	B	C
1	Access of UAnRX register during break (Specification change notice)		✗	✗	✗
2	Access of CBnRX register during break (Specification change notice)		✗	✗	✗
3	Access of CnRGPT register during break (Specification change notice)		✗	✗	✗
4	Access of CnTGPT register during break (Specification change notice)		✗	✗	✗
5	Access of CnGNCTRL register during break (Specification change notice)		✗	✗	✗
6	DMA transfer forcible termination (Specification change notice)		✗	✓	✓
7	Program execution and DMA transfer in internal RAM (Specification change notice)		✗	✗	✗
8	Emulator hangs up on internal RESET (Direction of use)		✗	✓	✓
9	Emulator hangs up while downloading data or setting software break (Specification change notice)		✗	✗	✗
10	External RAM connection (Technical limitation)		✗	✗	✗
11	POC circuit and clock monitor (Technical limitation)		✗	✗	✗
12	Flash mask option (Technical limitation)		✗	✗	✗
13	aFCAN: Rx limitation (Technical limitation)		✗	✗	✓
14	Illegal break during program execution in internal RAM (1) (Specification change notice)		✗	✗	✗
15	Reset input during break (Technical limitation)		✗	✓	✓
16	Entering and releasing STOP mode when the RESET pin is masked (Specification change notice)		✗	✗	✗
17	A/D conversion function during a break (Specification change notice)		✗	✗	✗

			QBV850ESFX2		
No.	**Outline**	**Control-Code**[Note]	A	B	C
18	Illegal break during program execution in internal RAM (2) (Specification change notice)		✗	✗	✗
19	Address not retained during external bus access (Direction of use)		✗	✗	✗

✓ Not applicable

✗ Applicable

Note: The Control Code is indicated by the letter appearing at the 2nd position from the left in the serial number of the product.

(C) Description of Operating Precautions

No. 1

Access of UAnRX register during break (n = 0...3)
(Specification change notice)

Details

An overrun error occurs under the following conditions (a) to (c):

(a) If a break occurs after reading the UART receive buffer register (UAnRX) and the UAnRX register is displayed in the I/O register window of the debugger, an overrun error occurs when UART reception is performed for the next time.

(b) If a software break occurs immediately after reading the UART receive buffer register (UAnRX), an overrun error occurs when UART reception is performed the next time regardless of whether or not the UAnRX register is displayed in the I/O register window.

(c) If a DMA transfer from the UART receive buffer register (UAnRX) is performed during a break[NOTE], an overrun error occurs when UART reception is performed the next time.

Note: Including breaks by the RAM monitor function or DMM function. However the realtime RAM monitor function does not cause this behaviour since it does not set breaks.

Remark: An overrun error also occurs when the UART receives data multiple times during a break (This complies with the specification of the emulator).

Workaround

(a) Do not display the UAnRX register in the I/O register window.
(b) Set a hardware break when setting a break immediately after reading the UAnRX register
(c) There is no workaround.

No. 2

Access of CBnRX register during break (n = 0...3)
(Specification change notice)

Details

When the CSIBn receive data register (CBnRX) is read, it usually starts the next reception operation. Under the following conditions (a) and (b), however, the next reception operation is not started even if CBnRX is read.

(a) If a software break occurs immediately after reading the CSIBn receive register (CBnRX).
(b) If a DMA transfer from the CSIBn receive data register (CBnRX) is performed during a break[NOTE]. As a result the communication stops or the DMA controller stops.

Note: Including breaks by the RAM monitor function or DMM function. However the realtime RAM monitor function does not cause this behaviour since it does not set breaks.

Workaround

(a) Set a hardware break when setting a break immediately after reading the CBnRX register.
(b) There is no workaround.

No. 3 Access of CnRGPT register during break (n = 0...3)
(Specification change notice)

Details

Under the following conditions (a) and (b), the read pointer (RGPT) that should be incremented is not incremented and the same data as previously read is read again.

(a) If a software break occurs immediately after reading the CANn module receive history list register (CnRGPT)
(b) If a DMA transfer from the CANn module receive history list register (CnRGPT) is performed during a break[NOTE].

Note: Including breaks by the RAM monitor function or DMM function. However the realtime RAM monitor function does not cause this behaviour since it does not set breaks.

Workaround

(a) Set a hardware break when setting a break immediately after reading the CnRGPT register.
(b) There is no workaround.

No. 4 Access of CnTGPT register during break (n = 0...3)
(Specification change notice)

Details

Under the following conditions (a) and (b), the read pointer (TGPT) that should be incremented is not incremented and the same data as previously transmitted is transmitted again.

(a) If a software break occurs immediately after reading the CANn module transmit history list register (CnTGPT).
(b) If a DMA transfer from the CANn module transmit history list register (CnTGPT) is performed during a break[NOTE].

Note: Including breaks by the RAM monitor function or DMM function. However the realtime RAM monitor function does not cause this behaviour since it does not set breaks.

Workaround

(a) Set a hardware break when setting a break immediately after reading the CnTGPT register.
(b) There is no workaround.

No. 5	Access of CnGNCTRL register during a break (n = 0...3) (Specificatin change notice)

Details

When a register access is performed in the following sequence, an unexpected forcible shutdown may occur after the sequence is complete.

Sequence :
(1) The EFSD bit of the CANn module control register (CnGMCTRL) is set.
(2) The I/O register[NOTE] is accessed.
(3) The GOM bit of the CANn mode control register (CnGMCTRL) is cleared.

Note: I/O register access except for clearing the GOM bit of the CnGMCTRL register

The conditions under which a forcible shutdown takes place are shown below:
(a) If a break occurs immediately after the I/O register access in (2) occurs.
(b) If a break by the RAM monitor function or the DMM function occurs immediately after the I/O register access in (2) occurs.
(c) Stepwise execution is performed for the I/O register access in (2).

Workaround

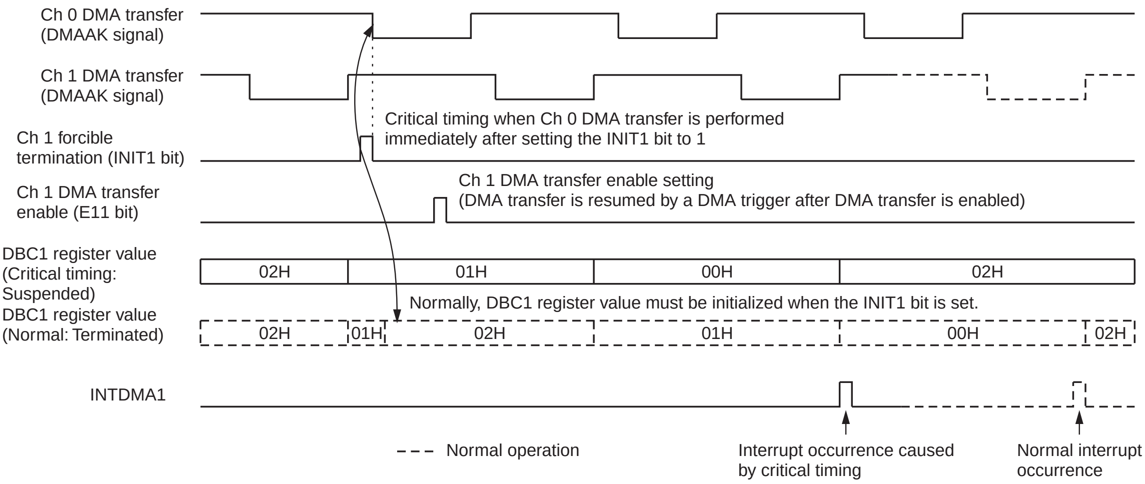
Be sure to set the EFSD bit and clear the GOM bit successively when executing a forcible shutdown. Do not perform a register access in the above sequence when not performing a forcible shutdown.

No. 6	DMA transfer forcible termination (Specification change notice)

Details

When terminating a DMA transfer by setting the corresponding INITn bit of the DCHCn register, the transfer may not be terminated, but just suspended, even though the INITn bit is set (1). As a result, when the DMA transfer of a channel that should have been terminated is resumed, the DMA transfer will terminate after an unexpected number of transfers are completed and a DMA transfer completion interrupt may occur. In addition, a DMA transfer of a channel n for which the INITn bit is set after forcible termination may be performed once again with the initialized value (n = 0 to 3).
The critical situation occurs if a DMA transfer is executed immediately after a forcible termination is set (by setting the INITn bit), refer to figure below.
The critical timing does not depend on the number of transfer channels, transfer type, transfer target, transfer mode, or trigger, and can occur with any combination of the above elements that can be set under the specifications. In addition, another channel may affect the occurrence of this critical timing.

Operation example: Both DMA channels, ch 0 and ch 1, are in single transfer mode, and ch 1 DMA transfer count is 3 (DBC1 register value = 02H).


Ch 0 DMA transfer (DMAAK signal)
Ch 1 DMA transfer (DMAAK signal)
Ch 1 forcible termination (INIT1 bit)
Ch 1 DMA transfer enable (E11 bit)
DBC1 register value (Critical timing: Suspended)
DBC1 register value (Normal: Terminated)
INTDMA1
Critical timing when Ch 0 DMA transfer is performed immediately after setting the INIT1 bit to 1
Ch 1 DMA transfer enable setting (DMA transfer is resumed by a DMA trigger after DMA transfer is enabled)
02H
01H
00H
02H
Normally, DBC1 register value must be initialized when the INIT1 bit is set.
02H
01H
02H
01H
00H
02H
Normal operation
Interrupt occurrence caused by critical timing
Normal interrupt occurrence

The following registers are buffer register with a 2-stage FIFO configuration of master and slave:

- DMA source address register (DSAnH, DSAnL)
- DMA destination address register (DDAnH, DDAnL)
- DMA transfer count register (DBCn)

If these registers are overwritten during a DMA transfer, or in the DMA suspended status, the value is written to the master register, and reflected in the slave register when the DMA transfer of the overwritten channel is terminated.
The “initialization” in the figure above means that the contents of the master register are reflected in the slave register.

No. 6	DMA transfer forcible termination (Specification change notice)

(cont.)

Workaround

The critical situation can be avoided by implementing any of the following procedures.

<1> Stop all transfers from DMA channels temporarily.

The following measure is effective if the program does not assume that the TCn bit of the DCHCn register is 1 except for the following workaround processing. (Since the TCn bit of the DCHCn register is cleared (0) when it is read, execution of the following procedure b) under <5> clears this bit.)

Procedure to avoid the critical timing:

<1> Disable interrupts (DI state)

<2> Read the DMA restart register (DRST) and transfer the ENn bit of each channel to a general purpose register (value A).

<3> Write 00H to the DMA restart register (DRST) twice[Note].
By executing twice[Note], the DMA transfer is definitely stopped before proceeding to <4>.

<4> Set (1) the INITn bit of the DCHCn register of the channel that should be terminated forcibly.

<5> Perform the following operations for value A read in (2) to obtain value B.
a) Clear (0) the bit of the channel that is not terminated forcibly.
b) If the TCn and ENn bits of the channel that is not terminated forcibly are 1,
clear (0) the bit of the channel.

<6> Write value B in <5> to the DRST register.

<7> Enable interrupts (EI state)

Note: Execute three times if the transfer target (transfer source or transfer destination) is the internal RAM.

Remarks: **1.** Be sure to execute <5> to prevent the ENn bit from being set illegally for channels that are terminated normally during the period of <2> and <3>

2. n = 0 to 3

<2> Repeat setting the INITn bit until the forcible DMA transfer termination is correctly performed (n = 0 to 3)

Procedure to avoid the critical timing:

<1> Copy the initial transfer count of the channel that should be terminated forcibly to a general-purpose register.

<2> Set (1) the INITn bit of the DCHCn register of the channel that should be terminated forcibly.

<3> Read the value of the DMA transfer count register (DBCn) of the channel that should be terminated forcibly and compare the value with the one copied in <1>. If the value do not match, repeat <2> and <3>.

Remarks: **1.** When the DBCn register is read in procedure <3>, the remaining transfer count will be read if the DMA is stopped due to this bug. If the forcible DMA termination is performed correctly, the initial transfer count will be read.

2. Note that it may take some time for forcible termination to take effect if this workaround is implemented in an application in which DMA transfer other than for channels subject to forcible termination are frequently performed.

No. 7

Program execution and DMA transfer in internal RAM
(Specification change notice)

Details

When a DMA transfer for the internal RAM and a bit manipulation instruction (SET1, CLR1, or NOT1) allocated in the internal RAM or a data access instruction for a misaligned address are executed simultaneously, the CPU may deadlock due to conflict between the internal bus operations. At this time, only a reset can be acknowledged, an NMI or an maskable interrupt cannot be acknowledged any more.

Unaffected cases

The critical situation does not occur if no instruction is executed in the internal RAM, or no DMA transfer is performed on the internal RAM.

Workaround

Implement any of the following workarounds.

- Do not perform a DMA transfer for the internal RAM when an instruction allocated in the internal RAM is being executed.
- Do not execute an instruction allocated in the internal RAM when a DMA transfer for the internal RAM is being performed.

No. 8

Emulator hangs up on internal reset
(Technical limitation)

Details

The emulator may hang up when a reset is generated by watchdog timer 2 or the low-voltage detector (LVI).

Workaround

There is no workaround. This behaviour has been corrected with control code B or later.

No. 9	Emulator hangs up while downloading data or setting software break (Specification change notice)
	Details The emulator may hang up when an active signal is connected to the WAIT or HLDRQ pin during program download or when a software break point is set to the internal ROM. *Workaround* When WAIT and HLDRQ are not used mask these signals using the pinmask function of the debugger. When WAIT and HLDRQ are used do not connect an active signal to the WAIT or HLDRQ input pin during download or when a software break point is set to the internal ROM. The above behaviour can be avoided using the following software components: - ID850: Use the version 2.81 or later of the ID850 debugger. - Green Hills Multi: Use the V1.57 or later of the exec.dll.

No. 10	External RAM connection (Technical limitation)
	Details When external RAM on the target system is connected to the CS0 area (0x100000 - 0x1fffff) and the bus control pins are active the data in this area may be overwritten by downloading data to the internal ROM area or by setting a software breakpoint in this area. *Workaround* Initialize the data in external RAM by program run after downloading data to the CS0 area or use a hardware break for the external RAM. The above behaviour can be avoided using the following software components: - ID850: Use the version 2.81 or later of the ID850 debugger. - Green Hills Multi: Use the V1.57 or later of the exec.dll.

No. 11	POC circuit and clock monitor (Technical limitation)
	Details Emulation of the POC circuit and the clock monitor is not possible. *Workaround* There is no workaround.

No. 12	Flash mask option (Technical limitation)
	Details Overwriting the data of the option data area (0x0000007A) for the flash mask option is possible. However, independent of the value written the emulator operates as if the setting of the above address was 0x00. *Workaround* There is no workaround. The above behaviour can be avoided using the following software components: - ID850: Use the version 2.81 or later of the ID850 debugger. - Green Hills Multi: Use the V1.57 or later of the exec.dll.

No. 13	aFCAN: Rx limitation (Technical limitation)

Details

RX Limitation

The aFCAN macro may store an incoming message although this message was interrupted by a bus error frame. Thus, the incomplete reception causes that a message buffer is updated with old or incorrect data or that the message is even stored at an incorrect location.

This unexpected behaviour affords that the bus error occurs in a certain relation to the currently present message on the bus. The critical time window starts at the sample point of the LSB of the DLC-field and lasts for the duration of an internal process in the aFCAN macro (RX-search). This time window usually lasts for a few bit times only. The actual length depends on the clock supply for the AFCAN, the CPU accesses during this period, the baud rate and the number of message buffers of the particular AFCAN macro.
In this time window the RX-search evaluates the received identifier of the current message. When the bus error is detected within this window and when the RX-search has just scanned buffer #n for reception and found it is matching, the message will unexpectedly be treated as a received message. As the time window is limited as described above, only a stuff bit error occurring right in this window can cause this behaviour.
There are two types of unexpected behavior for the RX limitation depending on the presence of pending transmission request (TRQ) for any other message buffer.

1. Behaviour at pending TRQ (TRQi = 1)

When the host processor has already submitted a transmit request (TRQ) for at least one buffer, the unexpected reception of the message will take place into the message buffer found by internal RX-search. This is the correct location to store the message i.e. the acceptance filter criteria are correctly fulfilled. However the data part will be updated with the contents of the shift register of the CAN protocol core. As this register is immediately stopped at detection of the bus error, the data provided to the message buffer can not be interpreted by the host processor.

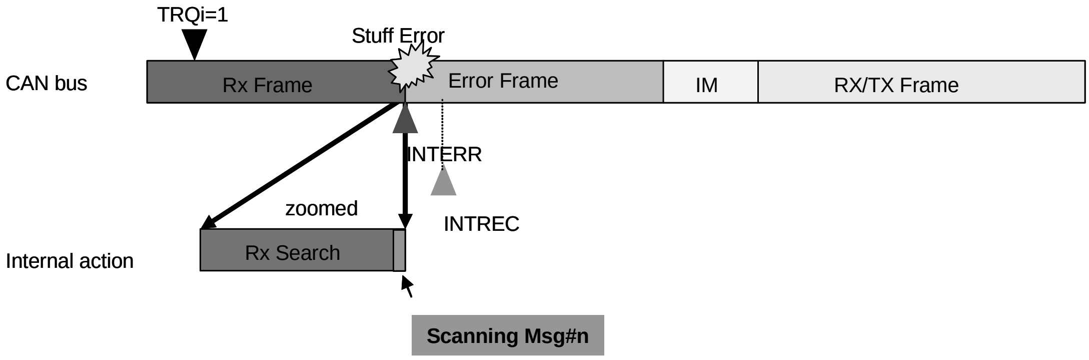


Figure 1: Behavior at pending TRQ

As during a regular reception, the RX-interrupt (if enabled) is generated and the application processes the message object.

No. 13	aFCAN: Rx limitation (Technical limitation)

2. Behavior without pending TRQ (TRQi = 0)

In case the host processor has not submitted a transmit request (TRQ) for any buffer before the detection of the bus error but submits TRQ = 1 after that point in time (see figure below) before the re-transmission of the message interrupted by the stuff bit error started, the unexpected reception of the message will take place.

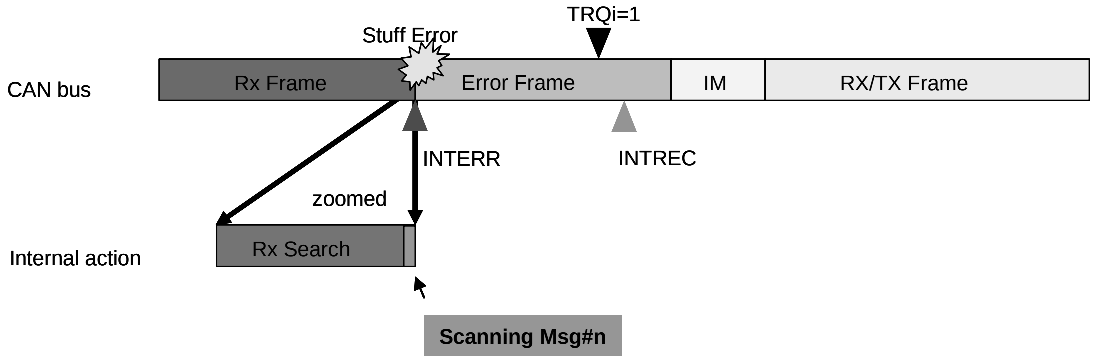


Figure 2: Behavior without pending TRQ

The unexpected storage of the message is issued in the particular message buffer that matches the acceptance filter criteria at the time where the bus error is detected (as described in 1.) or the message buffer #0 is overwritten independently of its configuration.

Impact on application

In typical applications the RX-limitation will lead to transiently incorrect data. In the vast majority of cases the message interrupted by a bus error is repeated by the transmitter right away. Then the application receives correct data shortly after the unexpected reception.

In scenarios where the message buffer #0 is overwritten, the impact for the application depends on the usage of that buffer. If it is configured as a receive buffer, the application receives a message at an unexpected location and will interpret the data to belong to the identifier originally programmed for that buffer. The message buffer #0 needs to be re-configured in order to receive the originally intended message object again.

In case of a transmit message buffer the unexpected storage may falsify a transmit object; i.e. when the unexpected behavior occurs after preparation of the message data but before the actual start of transmission. This scenario is even less likely than the scenario described in 1, which itself has a low probability. However the transmission of a falsified message can lead to repetitive transmission attempts when the original provider of that message (identifier) tries to send its message at the same time. Then the messages most likely will differ in their data part and a bit error is detected. This repetition resumes until one of the nodes enters error passive or bus off state. Then the situation is resolved as all pending TRQ are send with delay or are cancelled (in case of bus off state).

No. 13

aFCAN: Rx limitation
(Technical limitation)

Workaround

NEC will update the affected products.
NEC does not recommend a S/W workaround as first choice as it is fairly complex. On the one hand it is based on the control of submitting transmission requests only when the bus is idle. On the other hand a less complex algorithm can be used which does not prevent the unexpected reception but detects it safely and discards the unexpected reception in the CAN S/W driver. Any of these algorithms require that message buffer #0 is not used or that a 'dummy' TRQ in an unused buffer is set. This prevents behaviors as described in 2.

No. 14

Illegal break during program execution in internal RAM (1)
(Specification change notice)

Details

An illegal break may occur when a peripheral I/O register is accessed during program execution in internal RAM.

Workaround

Cancel the fail-safe break setting for the internal RAM in the debugger.
- In ID850QB debugger:
Click the button “Detail” in the “Fail-safe Break” field in the “Configuration” window and clear the check box for “Internal RAM”.
- In the MULTI debugger:
Cancel the fail-save break for “ramgrd” and “ramgrdv” using the target command “flsf”

No. 15

Reset input during break
(Technical limitation)

Details

The QB-V850ESFX2 may hang up if a break occurs when the RESET pin is active (low level).

Workaround

Mask the RESET pin using the pin mask function of the debugger.

No. 16	Entering and releasing STOP mode when the RESET pin is masked (Specification change notice)

Details

When the RESET pin is masked using the pin mask function of the debugger and watchdog timer 2 is used in reset mode, the CPU’s internal operating clock is switched to the internal ring oscillator clock after STOP mode is released, depending on the timing for entering and releasing STOP mode (one of (1) to (4) of the below table). After the clock is switched to the ring oscillator clock, the CPU continues the operation with the ring oscillator clock until a reset is ececuted by the debugger.

No.	Operating clock for watchdog timer 2	Timing at which CPU operation clock switches to ring oscillator clock
1	Main clock	STOP mode is entered during the period from when a reset of watchdog timer 2 occurs until the reset is released Note.
2	Subclock	STOP mode is entered during the period from when a reset of watchdog timer 2 occurs until the reset is released Note.
3	Ring oscillator clock	STOP mode is entered during the period from when a reset of watchdog timer 2 occurs until the reset is released Note.
4		The internal oscillation clock is stopped during the the period from when a reset of watchdog timer 2 occurs until the rset is released Note, and then STOP mode is entered.

Note The period in which watchdog timer 2 generates a reset signal while the reset signal of watchdog timer 2 is masked as a result of masking RESET using the pin mask function of the debugger.

Workaround

Do not use watchdog timer 2.
To generate a reset of watchdog timer 2, do not mask the RESET pin using the pin mask function of the debugger.

No. 17	A/D conversion function during a break (Specification change notice)

Details

(1)
A/D conversion does not start if any of the following conditions (a) to (c) is satisfied in peripheral break mode (in peripheral break mode the peripheral functions are stopped during a break). In addition, no interrupt requests are generated upon completion of the A/D conversion.

(a)
A break occurs in the time frame from when an A/D conversion start trigger is generated [Note 1] until the execution of the following two instructions ends [Note 2].
Example: In software trigger mode

```
<1>      set1 0x7,ADA0M0
<2>      nop
<3>      nop
<4>      nop
```

A/D conversion does not start if a break occurs during <1> to <3>. If a break occurs after <4>, A/D conversion starts normally (Caution: Behaviour described in below paragraphs (2) and (3) may still apply).

(b)
If execution is started during an A/D conversion start instruction in software trigger mode and a software break or a break before execution is set to this instruction.
Example:

```
         set1 0x7, ADA0M0
```

A/D conversion does not start if a software break or a break before execution is set to this line.

(c)
A break occurs while an A/D conversion operation is stopped and an attempt is made to start A/D conversion during this break [Note 3].

(2)
If a break occurs during A/D conversion in peripheral break mode, the A/D conversion result immediately after restart is invalid.

(3)
If a break occurs [Note 2] during A/D conversion in peripheral break mode, a write access is performed [Note 5] on an A/D related register [Note 4], and the A/D conversion is restarted, then conversion is performed once or twice with the values before this write access.
- If the break occurs in normal conversion operation mode: Once or twice,
- If a break occurs during A/D conversion in high speed conversion mode and the ADA0CE bit is cleared and re-set during the break: Twice,
- Other: Once.

After this conversion is completed, A/D conversion starts with the values after the writing. Consequently, an invalid A/D conversion result is obtained ant it seems as though invalid interrupts occur once or twice for the operation (Normally, re-conversion is performed immediately after re-execution with values newly set to the A/D-related register).

Notes:
1. Starting conversion by DMA transfer, external trigger and timer trigger are included in this condition in addition to starting conversion triggered by instruction execution.

No. 17	(continued)
	2. Includes the following break sources: - Step execution, - Fail safe break, - RAM monitoring (does not apply for realtime RAM monitoring), - DMM - Change of event while program is running. Among these sources, RAM monitoring, DMM and a change of event while the program is running is implemented through an instantaneous break, so the actual break point cannot be specified and thus the A/D conversion becomes invalid. **3.** DMA transfer, external trigger and timer trigger are included in this condition in addtition to a write access to the ADA0CE bit in the I/O register window. **4.** A/D-related registers: ADA0M0, ADA0M1, ADA0M2, ADA0S, ADA0PFT and ADA0PFM. **5.** Cases such that the write setting is applied in the I/O register window or through DMA transfer. *Workaround* Do not set peripheral break mode if you want to avoid this behaviour entirely or observe all of the following points: - Do not set breaks between the A/D conversion start trigger and the end of A/D conversion. - Do not perform step execution of an A/D conversion start instruction in software trigger mode. - Do not perform a write access to an A/D related register during a break. - Disable the RAM monitoring function. - Do not use DMM - Do not change events while the program is running.

No. 18 Illegal break during program execution in internal RAM (2)
(Specification change notice)

Details

A non-map break occurs if all of the following conditions are satisfied, even if the program itself is correct:
- A program is executed in the internal RAM.
- Data access for the internal RAM area is performed twice successively.
- A branch occurs to the internal ROM area using a JR or JARL instruction immediately after the above successive data access or one NOP instruction after the above successive data access.

Workaround

Implement either one of the following workarounds.

1.
- When using ID850QB:
Click the “Detail” button in the fail save break field in the configuration window and clear the check box for “Internal RAM”.
- When using MULTI:
Cancel the fail save break for “ramgrd” and “ramgrdv” using the target command “flsf”.

2.
Insert two or more NOP instructions between the successive data access for the internal RAM area and the instruction to branch to the internal ROM area.

No. 19 Address not retained during external bus access
(Direction of use)

Details

When the multiplexed bus mode is selected for the external bus and the data bus size is 8-bit, the address is not retained after the T2 state of the bus cycle. A low level is output instead.

Workaround

There is no workaround.

The above behaviour can be avoided by using the following or later device file versions:
- When using V850ES/FE2, V850ES/FF2, V850ES/FG2, V850ES/FJ2: Use DF703239 V2.11 or later.
- When using uPD703229Y or uPD70F3229Y: Use DF703229 V2.01 or later.

(D) Valid Specification

Item	Date pulished	Document No.	Document Title
1	September 2005	SUD-CD-05-0129	QB-V850ESFX2 Preliminary User’s Manual
2	November 2005	U17830EE1V0UM00	V850ES/Fx2 User’s Manual
3	November 2005	U17834EE1V0DS00	V850ES/FE2 Data Sheet
4	November 2005	U17833EE1V0DS00	V850ES/FF2 Data Sheet
5	November 2005	U17832EE1V0DS00	V850ES/FG2 Data Sheet
6	November 2005	U17831EE1V0DS00	V850ES/FJ2 Data Sheet
7	April 2004	U15943EJ3V0UM00	V850ES Architecture Manual

(E) Revision History

Item	Date pulished	Document No.	Comment
1	June, 2004	TPS-HE-B-3210	First release
2	August, 2004	TPS-HE-B-3211	Added item 29
3	August, 2004	TPS-HE-B-3212	Added Control Code ’B’, new evachip version; modified 23, 24, 26
4	November, 2005	TPS-HE-B-3213	Removed items 1 to 14 (origninal numbering of listed items) as these items have been added to the new version of the User’s Manual; removed items 27 and 28 (of origninal numbering) as these relate to the device; Introduced new numbering of listed items; added items 14 to 19; Modified item 7 (of new numbering).