HITACHI MICROCOMPUTER TECHNICAL UPDATE

DATE	25 January 2001	No.	TN-SH7-298A/E
THEME	NMI usage notice		
CLASSIFICATION	☐ Spec change ■ Limitation on Use ☐ Supplement of Documents		
PRODUCT NAME	SH7750,SH7750S,SH7751		Lot No.etc.
REFERENCE DOCUMENTS	SH7750 Series Hardware Manual SH7751 Hardware Manual	Rev.	EffectiveDate From

There is a case NMI does not work correctly. Please notice the following NMI usage.

1. Phenomenon

NMI detection logic may not work correctly in the following condition.
When the plural NMI are requested via the external pin within a certain time (it depend on the CPU and external-BUS state.), the CPU may not accept all kind of interrupt.

In following cases, NMI works correctly.
- The system guarantees enough interval time between two NMI requests (*).
- The system uses interrupts other than NMI, such an IRL.

(*) On condition CPU can be execute at least one instruction between two NMI requests under SR.BL=0.

2. Workaround

There are three workarounds available to avoid this phenomenon.

(1) Please provide enough interval time between two NMI requests as above (*).
In addition, a hazard on NMI pin may violate this interval time requirement and cause this phenomenon described above. Thus, the external circuits must pay attention to avoid a hazard (**).

(**) Each HIGH/LOW voltage level width must be more than 5 CKIO, the transition must not include noise pulse.

(2) Please use IRL interrupt instead of NMI interrupt.

(3) Software workaround

This phenomenon can be avoided by the software workaround which inserts following instruction sequence (***)(****) to the NMI exception handler routine.

Notes:

(***) When SR.BL bit is changed in NMI exception handler routine,
please insert this instruction sequence at the location before SR.BL changes.

(****) R0 to R3 can be replace with other registers.
If the register store/restore are necessary, please add them to the beginning and the ending of this instruction sequence.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;; R0  :  tmp              R1  :  Original SR
;; R2  :  Original ICR      R3  :  ICR Address
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
NMIH:
; (1) Set SR.IMASK = H'F
            stc         SR, R1              ;           SR store
            mov         R1,R0
            or          #H'F0,R0
            ldc         R0, SR

; (2) Reverse ICR.NMIE
            mov.l   #ICR, R3
            mov.w       @R3, R2             ;           ICR store
            mov.w       #H'0100, R0         ;
            xor         R2, R0              ;
            mov.w       R0, @R3             ;           ICR.NMIE (Reverse and dummy write)
            bra NMIH1
            nop
            .pool
            .align      4
NMIH2:
; (3)
            mov.w       @R3, R0             ;           dummy read
            mov.w       R2, @R3             ;           ICR.NMIE (Write)
            stc         SR, R0
            ldc         R0, SR
            ldc         R0, SR
            ldc         R0, SR
            ldc         R0, SR
            ldc         R0, SR
            ldc         R0, SR
            ldc         R0, SR
            ldc         R0, SR
            ldc         R0, SR
            ldc         R1, SR              ;           SR restore
            bra NMIH3
            nop
NMIH1:
            bra NMIH2
            nop
NMIH3:
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```