Hitachi America, Ltd.

TN-0168

Application Engineering
TechNote

Carol Jacobson

Handling Multiple Interrupts from a Single Source

When working with interrupts, if the time required to complete the interrupt routine for an interrupting source is longer than the time between interrupt requests from the same source one of two conditions may occur. (See H8/300 H.W.M.s and TN-0107 for more information on interrupt operation).

1. After completing the interrupt routine, the CPU does not return to the main routine. The program counter returns to the beginning of the interrupt routine immediately after the RTE instruction.

This condition arises when the CPU receives a second interrupt request while executing the interrupt routine for the same source. If the second request occurs after the interrupt condition is cleared but before the end of the first interrupt cycle, the second request remains pending. When the RTE command clears the I flag in the CCR, the CPU recognizes and responds to the second request before returning to the main routine. The program counter returns to the start of the interrupt routine. As long as the interrupt response time remains longer than the interval between interrupt requests the CPU will continue in this loop.

Under many circumstances the programmer may not want to make corrections for this situation. The CPU is behaving normally by responding to all requests. The only way to change the condition is to either shorten the interrupt handling routine or lengthen the time between interrupts. Verify the interrupting source is behaving as expected.

If the second interrupt must be serviced immediately (as with a serial port), the only solution is to shorten the interrupt cycle time. A frequent mistake in calculating the cycle time comes from not taking into account the interrupt latency (time from interrupt request to start of the interrupt routine). The latency period can be up as much as 53 CPU cycles when executing from off-chip memory or 29 CPU cycles for on-chip execution. This overhead time must be added to the time required to complete the interrupt handling routine when determining the total cycle time.

This first condition is also caused by not clearing the interrupt request before exiting the interrupt routine. After RTE, before continuing with the next instruction, the CPU checks for pending requests. If the interrupting condition was not cleared during the interrupt cycle, the CPU responds as though a second valid request remains pending. The program counter continues to return to the beginning of the interrupt routine until the condition is cleared.

2. During the interrupt routine, further interrupts from the same source are ignored.

As above, if the duration of the interrupt routine is longer than the interval between requests, the second interrupt from the same source remains pending (see above). If the interrupt routine clears the interrupt condition after the second interrupt has occurred, the second pending request is also cleared and lost. When the CPU returns from the interrupt cycle the second interrupt is not recognized. Subsequent requests made after RTE are valid.

If the programmer wants to recognize a second interrupt from an on-chip source the interrupting condition must be cleared before the second request. A second interrupt from an external source will be recognized as long as the IRQx input remains valid.