R4000 Processor Interrupts

15

The R4000 processor supports the following interrupts: six hardware interrupts, one internal "timer interrupt," two software interrupts, and one nonmaskable interrupt. The processor takes an exception on any interrupt.

This chapter describes the six hardware and single nonmaskable interrupts. A description of the software and the timer interrupts can be found in Chapter 5. CPU exception processing is also described in Chapter 5.

Floating-point exception processing is described in Chapter 6.

15.1 Hardware Interrupts

The six CPU hardware interrupts can be caused by external write requests to the R4000SC, R4000MC, and R4000PC, or can be caused through dedicated interrupt pins. These pins are latched into an internal register by the rising edge of **SClock**. The R4000MC and R4000SC packages support a single interrupt pin, **Int*(0)**. The R4000PC package supports six interrupt pins, **Int*(5:0)**.

15.2 Nonmaskable Interrupt (NMI)

The nonmaskable interrupt is caused either by an external write request to the R4000 or by a dedicated pin in the R4000. This pin is latched into an internal register by the rising edge of **SClock**.

15.3 Asserting Interrupts

External writes to the CPU are directed to various internal resources, based on an internal address map of the processor. When **SysAD[6:4]** = 0, an external write to any address writes to an architecturally transparent register called the *Interrupt* register; this register is available for external write cycles, but not for external reads.

During a data cycle, **SysAD[22:16]** are the write enables for the seven individual *Interrupt* register bits and **SysAD[6:0]** are the values to be written into these bits. This allows any subset of the *Interrupt* register to be set or cleared with a single write request. Figure 15-1 shows the mechanics of an external write to the *Interrupt* register.

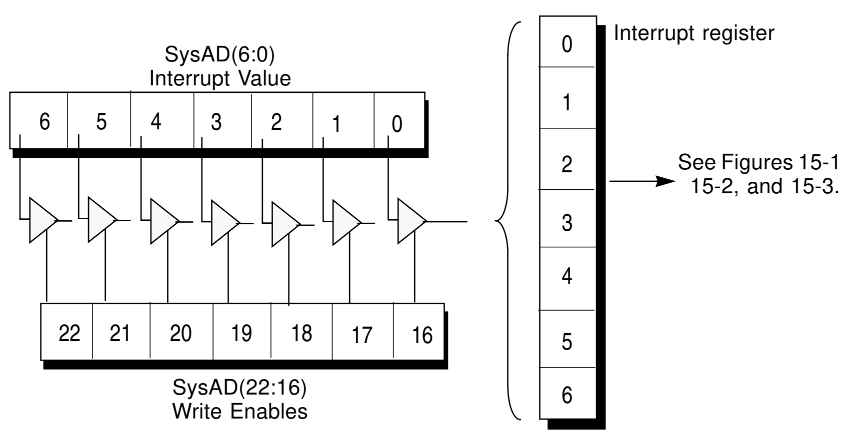

Figure 15-1 Interrupt Register Bits and Enables

Figure 15-2 shows how the R4000SC and R4000MC interrupts are readable through the *Cause* register.

- Bit 5 of the *Interrupt* register in the R4000SC and R4000MC is multiplexed with the **TimerInterrupt** signal and the result is directly readable as bit 15 of the *Cause* register.
- Bits 4:1 of the *Interrupt* register are directly readable as bits 14:11 of the *Cause* register.
- Bit 0 of the *Interrupt* register is latched into the internal register by the rising edge of **SClock**, then ORed with the **Int*(0)** pin, and the result is directly readable as bit 10 of the *Cause* register.

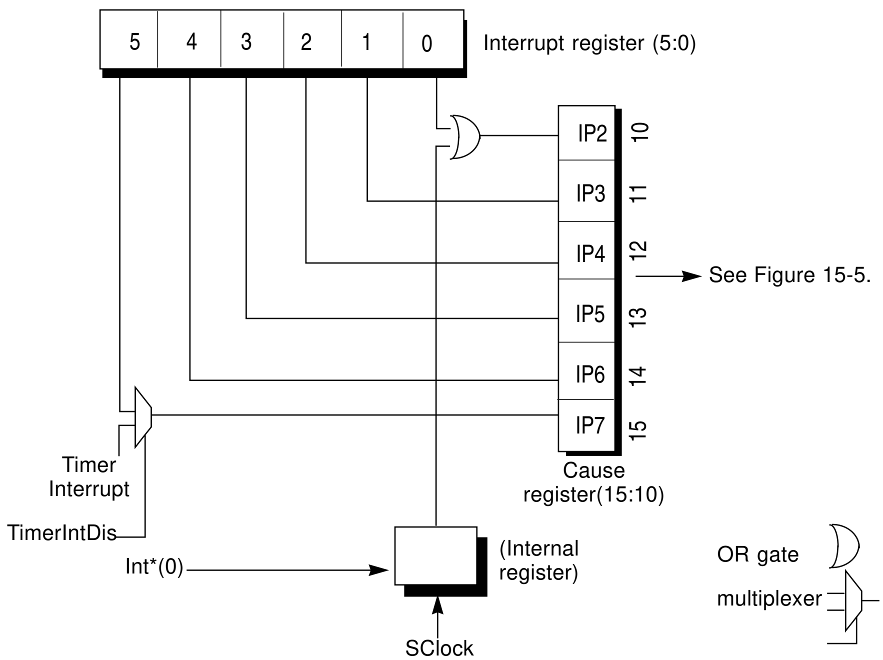

Figure 15-2 R4000SC/MC Interrupt Signals

The select line for the Timer Interrupt multiplexer is enabled by boot-mode bit 19, **TimerIntDis**, as described in Chapter 9. The Timer Interrupt input to the multiplexer is asserted when the *Count* register equals the *Compare* register.

Figure 15-3 shows how the R4000PC interrupts are readable through the *Cause* register. The interrupt bits, **Int*(5:0)**, are latched into the internal register by the rising edge of **SClock**.

- Bit 5 of the *Interrupt* register in the R4000PC is ORed with the **Int*(5)** pin and then multiplexed with the **TimerInterrupt** signal. This result is directly readable as bit 15 of the *Cause* register.
- Bits 4:0 of the *Interrupt* register are bit-wise ORed with the current value of the interrupt pins **Int*[4:0]** and the result is directly readable as bits 14:10 of the *Cause* register.

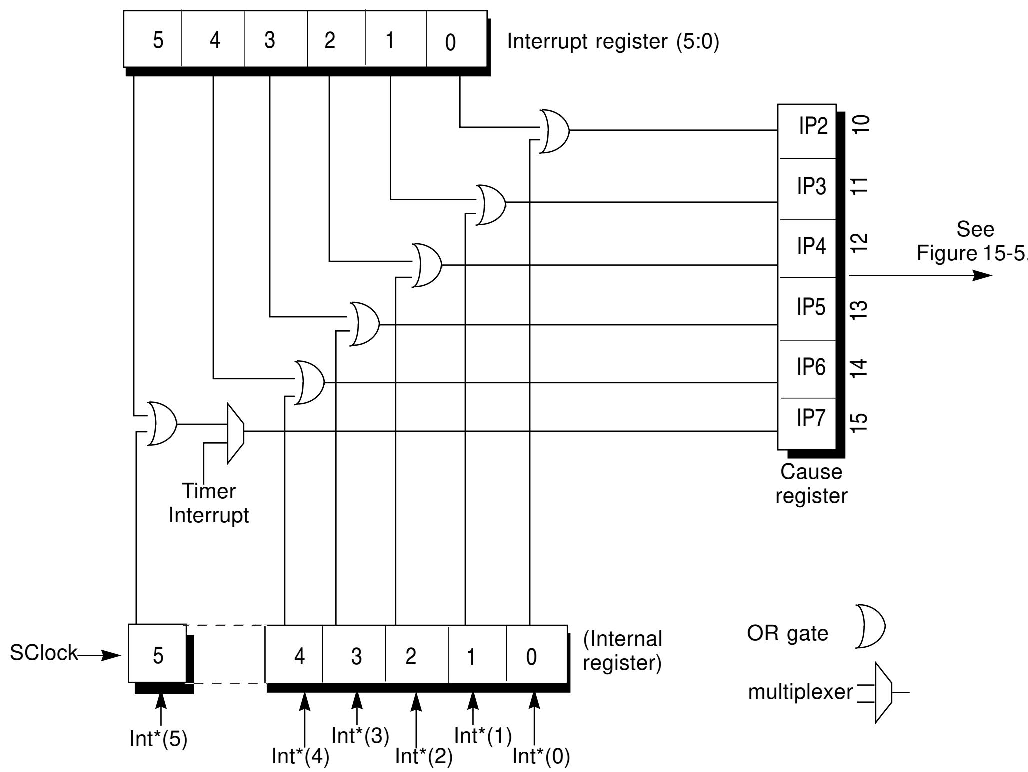

Figure 15-3 R4000PC Interrupt Signals

Figure 15-4 shows the internal derivation of the **NMI** signal, for all versions of the R4000 processor.

The **NMI*** pin is latched by the rising edge of **SClock**, however the NMI exception occurs in response to the falling edge of the **NMI*** signal, and is not level-sensitive.

Bit 6 of the *Interrupt* register is then ORed with the inverted value of **NMI*** to form the nonmaskable interrupt.

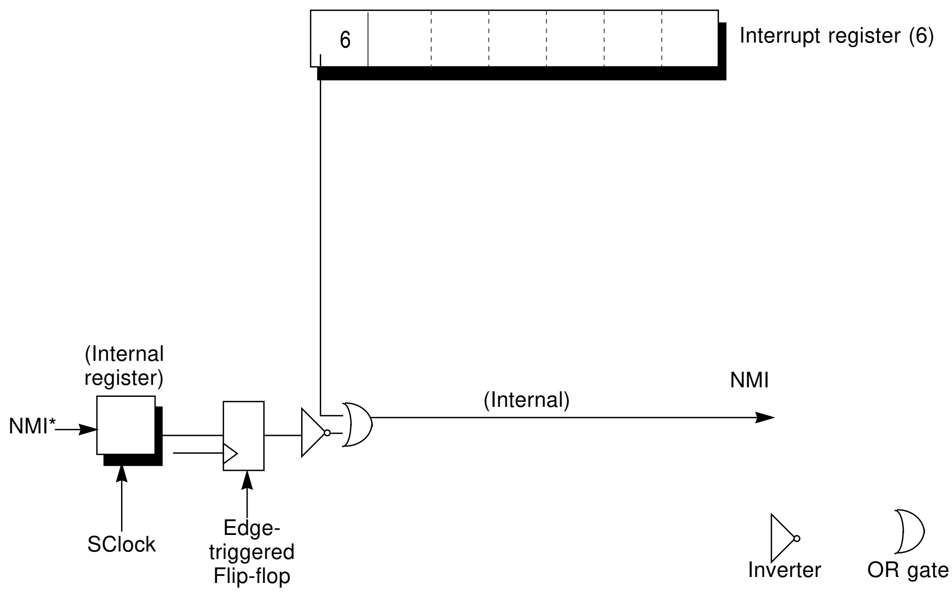

Figure 15-4 R4000 Nonmaskable Interrupt Signal

Figure 15-5 shows the masking of the R4000 interrupt signal.

- *Cause* register bits 15:8 (IP7-IP0) are AND-ORed with *Status* register interrupt mask bits 15:8 (IM7-IM0) to mask individual interrupts.
- *Status* register bit 0 is a global Interrupt Enable (IE). It is ANDed with the output of the AND-OR logic to produce the R4000 interrupt signal.

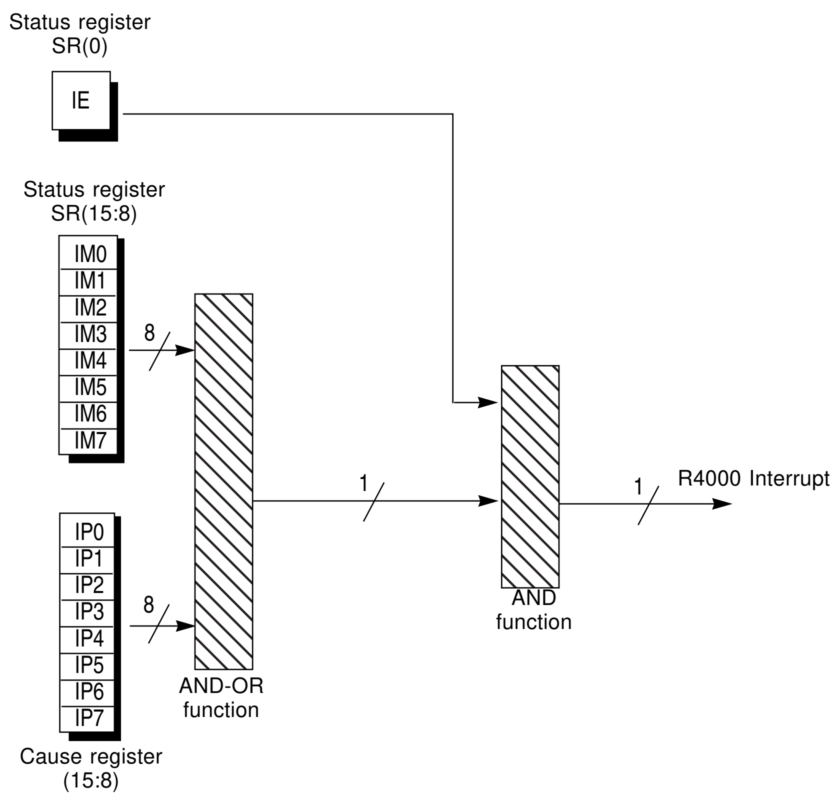

Figure 15-5 Masking of the R4000 Interrupt