8. Initialization

This section describes initialization of the R10000 processor, including initialization of logical registers.

Initialization of the processor occurs during a reset sequence. The processor supports three separate reset sequences:

- Power-on reset
- Cold reset
- Soft reset

These sequences are described in this chapter.

Also described are the mode bits.

8.1 Initialization of Logical Registers

After a power-on or cold reset sequence, all logical registers (both in the integer and the floating-point register files) must be written before they can be read. Failure to write any of these registers before reading from them will have an unpredictable result.

8.2 Power-On Reset Sequence

The Power-on Reset sequence is used to reset the processor after the initial power-on, or whenever power or **SysClk** are interrupted.

The Power-on Reset sequence is as follows:

- The external agent negates **DCOk**.
- The external agent asserts **SysReset***.
- The external agent negates **SysGnt***.
- The external agent negates **SysRespVal***.
- Once **Vcc**, **VccQ[SC,Sys]**, **Vref[SC,Sys]**, **Vcc[Pa,Pd]**, and **SysClk** stabilize, the external agent waits at least 1ms and then asserts **DCOk**.
- At this time, the System interface resides in slave state and all internal state is initialized.
- The **SysClkDiv** mode bits default to divide-by-1.
- The **SCClkDiv** mode bits default to divide-by-3.
- After waiting at least 100 ms for the internal clocks to stabilize, the external agent loads the mode bits into the processor by driving the mode bits on **SysAD[63:0]**, waiting at least two **SysClk** cycles, and then asserting **SysGnt*** for at least one **SysClk** cycle.
- After waiting at least another 100 ms for the internal clocks to restabilize, the external agent synchronizes all clocks internal to the processor. This is performed by asserting **SysRespVal*** for one **SysClk** cycle.
- After waiting at least 100 ms for the internal clocks to again restabilize, (a third 100 ms restabilization period) the external agent negates **SysReset***.
- The external agent must retain mastership of the System interface, refrain from issuing external requests or nonmaskable interrupts, and ignore the system state bus until the processor asserts **SysReq***. The assertion of **SysReq*** indicates the processor is ready for operation. In a cluster arrangement, all processors must assert **SysReq***, indicating they are ready for operation.

If the **virtual SysClk** is used during the reset sequence, the mode bits, **SysGnt***, **SysRespVal***, and **SysReset*** should all be referenced to the **virtual SysClk** that is created with **SysCyc***. This approach will cause the R10000 to come out of reset synchronously with the **virtual SysClk**, which will allow repeatable and lock-step operation (see Chapter 6, the section titled "Support for Hardware Emulation," for description of **virtual SysClk** operation). (See page 148 of the Errata.)

During a Power-on Reset sequence, all internal state is initialized. A Power-on Reset sequence causes the processor to start with the Reset exception.

Figure 8-1 shows the Power-on Reset sequence.

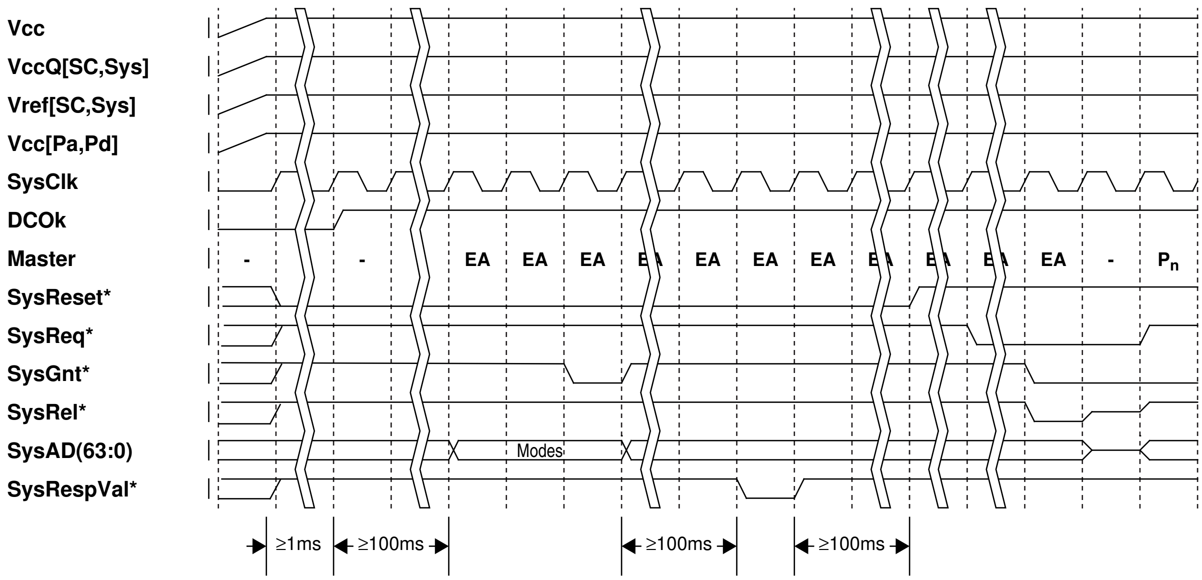

Figure 8-1 Power-On Reset Sequence

8.3 Cold Reset Sequence

The Cold Reset sequence is used to reset the entire processor, and possibly alter the mode bits while power and **SysClk** are stable.

The Cold Reset sequence is as follows:

- The external agent negates **SysGnt*** and **SysRespVal***.
- After waiting at least one **SysClk** cycle, the external agent asserts **SysReset***.
- After waiting at least 100 ms, the external agent loads the mode bits into R10000. This is performed by driving the mode bits on **SysAD[63:0]**, waiting at least two **SysClk** cycles, and then asserting **SysGnt*** for at least one **SysClk** cycle.
- After waiting at least another 100 ms for the internal clocks to restabilize, the external agent synchronizes all processor internal clocks by asserting **SysRespVal*** for one **SysClk** cycle.
- After waiting at least 100 ms for the internal clocks to again restabilize, (a third 100 ms restabilization period) the external agent negates **SysReset***.
- The external agent must retain mastership of the System interface, refrain from issuing external requests or nonmaskable interrupts, and ignore the system state bus until the processor asserts **SysReq***. The assertion of **SysReq*** indicates the processor is ready for operation. In a cluster arrangement, all processors must assert **SysReq***, indicating they are ready for operation.

During a Cold Reset sequence all processor internal state is initialized. A Cold Reset sequence causes the processor to start with a Reset exception.

Figure 8-2 shows the cold reset sequence.

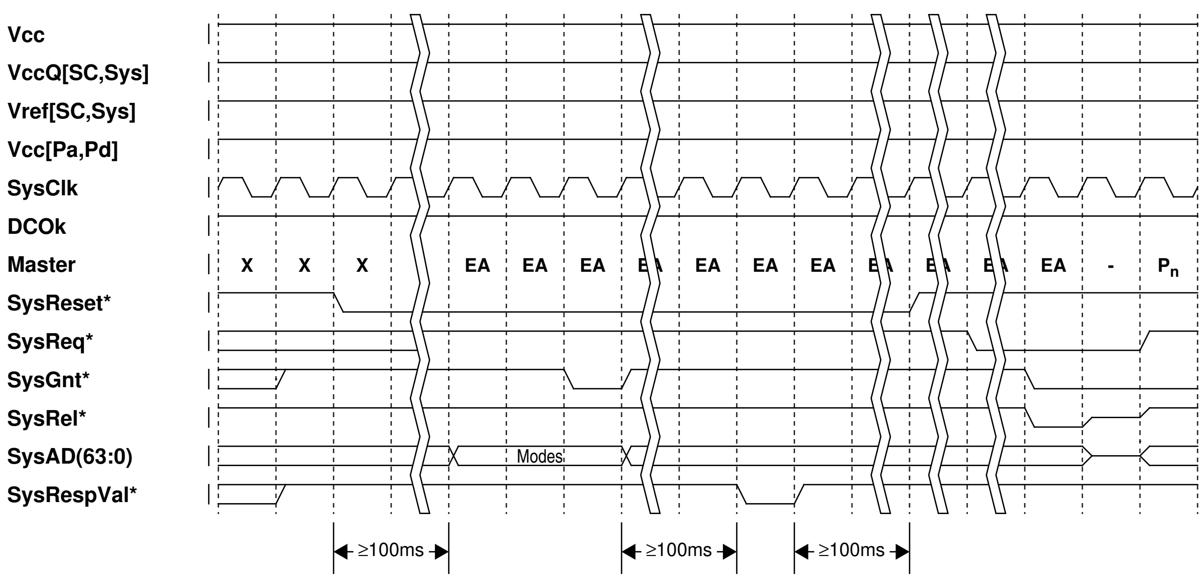

Figure 8-2 Cold Reset Sequence

8.4 Soft Reset Sequence

A Soft Reset sequence is used to reset the external interface of the processor without altering the mode bits while power and **SysClk** are stable.

The Soft Reset sequence is as follows:

- The external agent negates **SysGnt*** and **SysRespVal***.
- After waiting at least one **SysClk** cycle, the external agent asserts **SysReset*** for at least 16 **SysClk** cycles.
- The external agent must retain mastership of the System interface, refrain from issuing external requests or nonmaskable interrupts, and ignore system state bus until the processor asserts **SysReq***. The assertion of **SysReq*** indicates the processor is ready for operation. In a cluster arrangement, all processors must assert **SysReq***, indicating they are ready for operation.

During a Soft Reset sequence, all external interface state is initialized. The internal and secondary cache clocks are not affected by a Soft Reset sequence. The general purpose, CP0, and CP1 registers are preserved, as well as the primary and secondary caches.

A Soft Reset sequence causes a Soft Reset exception, in which the Soft Reset exception handler executes instructions from uncached space and uses CACHE instructions to analyze and dump the contents of the primary and secondary caches. To resume normal operation, a Cold Reset sequence must be initiated.

Figure 8-3 presents the Soft Reset sequence.

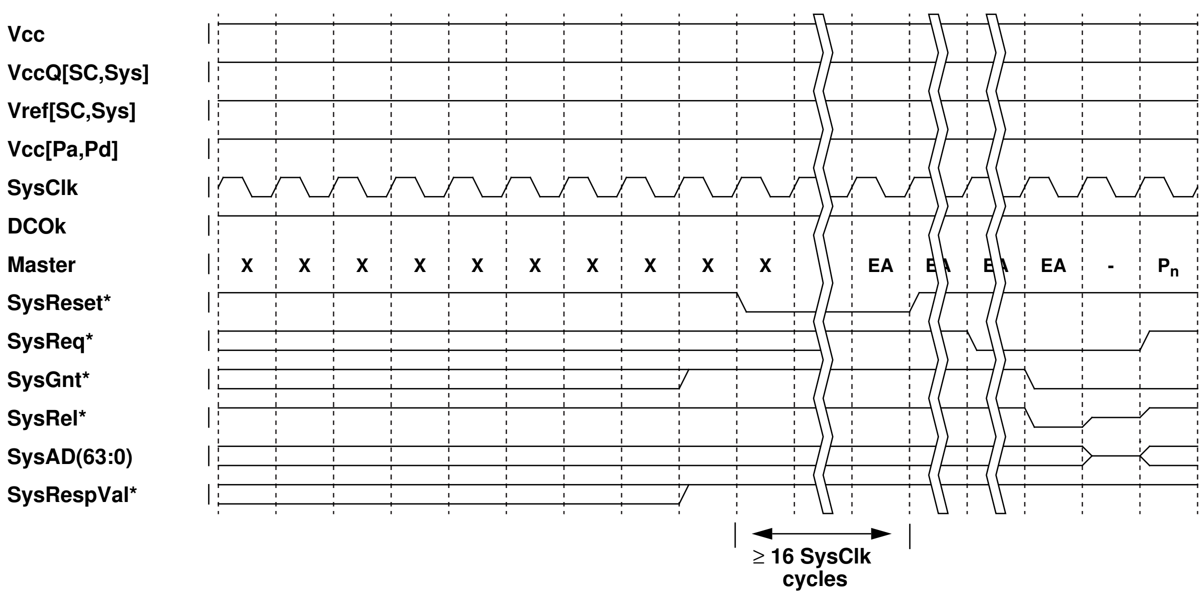

Figure 8-3 Soft Reset Sequence

8.5 Mode Bits

The R10000 processor uses mode bits to configure the operation of the microprocessor. These mode bits are loaded into the processor from the **SysAD[63:0]** bus during a power-on or cold reset sequence while **SysGnt*** is asserted. The **SysADChk[7:0]** bus does not have to contain correct ECC during mode bit initialization. During the reset sequence, the mode bits obtained from **SysAD[24:0]** are written into bits 24:0 of the CP0 *Config* register.

The mode bits are described in Table 8-1.

Table 8-1 Mode Bits

SysAD Bit	Name and Function	Value	Mode Setting
2:0	**Kseg0CA** Specifies the *kseg0* cache algorithm.	0 1 2 3 4 5 6 7	Reserved Reserved Uncached Cacheable noncoherent Cacheable coherent exclusive Cacheable coherent exclusive on write Reserved Uncached accelerated
4:3	**DevNum** Specifies the processor device number.	0-3	
5	**CohPrcReqTar** Specifies the target of processor coherent requests issued on the System interface by the processor.	0 1	External agent only Broadcast
6	**PrcElmReq** Specifies whether to enable processor eliminate requests onto the System interface by the processor.	0 1	Disable Enable
8:7	**PrcReqMax** Specifies the maximum number of outstanding processor requests allowed on the System interface by the processor.	0 1 2 3	1 outstanding processor request 2 outstanding processor requests 3 outstanding processor requests 4 outstanding processor requests

Table 8-1 (cont.) Mode Bits

SysAD Bit	Name and Function	Value	Mode Setting
12:9	**SysClkDiv** Sets **PClk** to **SysClk** ratio; determines the System interface clock frequency; see Chapter 7, the section titled “System Interface Clock and Internal Processor Clock Domains	0 1 2 3 4 5 6 7 8 9 A B C D E F	Reserved Result of division by 1 Result of division by 1.5 Result of division by 2 Result of division by 2.5 Result of division by 3 Result of division by 3.5 Result of division by 4 Reserved Reserved Reserved Reserved Reserved Reserved Reserved Reserved
13	**SCBlkSize** Specifies the secondary cache block size.	0 1	16-word 32-word
14	**SCCorEn** Specifies the method of correcting secondary cache data array ECC errors.	0 1	Retry access through corrector Always access through corrector
15	**MemEnd** Specifies the memory system endianness.	0 1	Little endian Big endian
18:16	**SCSize** Specifies the size of the secondary cache.	0 1 2 3 4 5 6 7	512 Kbyte 1 Mbyte 2 Mbyte 4 Mbyte 8 Mbyte 16 Mbyte Reserved Reserved
21:19	**SCClkDiv** Sets **PClk** to **SCClk** ratio; determines the secondary cache clock frequency; see Chapter 7, the section titled “System Interface Clock and Internal Processor Clock Domains	0 1 2 3 4 5 6 7	Reserved Result of division by 1 Result of division by 1.5 Result of division by 2 Result of division by 2.5 Result of division by 3 Reserved Reserved
24:22	Reserved	0	

Table 8-1 (cont.) Mode Bits

SysAD Bit	Name and Function	Value	Mode Setting
28:25	**SCClkTap** Specifies the alignment† of **SCClk[5:0]** and **SCClk*[5:0]** relative to the internal secondary cache clock.	0 1 2 3 4 5 6 7 8 9 A B C D E F	**SCClk** same phase as internal clock **SCClk** 1/12 **PClk** period earlier than internal clock **SCClk** 2/12 **PClk** period earlier than internal clock **SCClk** 3/12 **PClk** period earlier than internal clock **SCClk** 4/12 **PClk** period earlier than internal clock **SCClk** 5/12 **PClk** period earlier than internal clock undefined undefined **SCClk** 6/12 **PClk** period earlier than internal clock **SCClk** 7/12 **PClk** period earlier than internal clock **SCClk** 8/12 **PClk** period earlier than internal clock **SCClk** 9/12 **PClk** period earlier than internal clock **SCClk** 10/12 **PClk** period earlier than internal clock **SCClk** 11/12 **PClk** period earlier than internal clock undefined undefined
29	Reserved	0	
30	**ODrainSys** Specifies whether or not to configure select‡ System interface bidirectional and output signals as open drain.	0 1	Push-pull Open drain
31	**CTM** Specifies whether or not to enable cache test mode.	0 1	Disable Enable
63:32	Reserved	0	

†Does not include the output buffer delay.

‡ **SysReq***, **SysRel***, **SysCmd[11:0]**, **SysCmdPar**, **SysAD[63:0]**, **SysADChk[7:0]**, **SysVal***, **SysState[2:0]**, **SysStatePar**, **SysStateVal***, **SysCorErr***, **SysUncErr***

Bits 28:25 of Table 8-1; see page 155 of Errata.