Initialization Interface

9

This chapter describes the R4000 Initialization interface. This includes the reset signal description and types, initialization sequence, with signals and timing dependencies, and boot modes, which are set at initialization time.

Signal names are listed in bold letters—for instance the signal **VCCOk** indicates +5 voltage is stable. Low-active signals are indicated by a trailing asterisk, such as **ColdReset***, the power-on/cold reset signal.

9.1 Functional Overview

The R4000 processor has the following three types of resets; they use the **VCCOk, ColdReset***, and **Reset*** input signals.

- **Power-on reset**: starts when the power supply is turned on and completely reinitializes the internal state machine of the processor without saving any state information.
- **Cold reset**: restarts all clocks, but the power supply remains stable. A cold reset completely reinitializes the internal state machine of the processor without saving any state information.
- **Warm reset**: restarts processor, but does not affect clocks. A warm reset preserves the processor internal state.

The operation of each type of reset is described in sections that follow. Refer to Figures 9-1, 9-2, and 9-3 later in this chapter for timing diagrams of the power-on, cold, and warm resets.

The Initialization interface is a serial interface that operates at the frequency of the **MasterClock** divided by 256: (**MasterClock**/256). This low-frequency operation allows the initialization information to be stored in a low-cost EPROM.

9.2 Reset Signal Description

This section describes the three reset signals, **VCCOk, ColdReset***, and **Reset***.

VCCOk: When asserted[†], **VCCOk** indicates to the processor that the +5 volt power supply (Vcc) has been above 4.75 volts for more than 100 milliseconds (ms) and is expected to remain stable. The assertion of **VCCOk** initiates the reading of the boot-time mode control serial stream (described in Initialization Sequence, in this chapter).

ColdReset*: The **ColdReset*** signal must be asserted (low) for either a power-on reset or a cold reset. The clocks **SClock**, **TClock**, and **RClock** begin to cycle and are synchronized with the deasserted edge (high) of **ColdReset***. **ColdReset*** must be deasserted synchronously with **MasterClock**.

Reset*: the **Reset*** signal must be asserted for any reset sequence. It can be asserted synchronously or asynchronously for a cold reset, or synchronously to initiate a warm reset. **Reset*** must be deasserted synchronously with **MasterClock**.

ModeIn: Serial boot mode data in.

ModeClock: Serial boot mode data out, at the **MasterClock** frequency divided by 256 (**MasterClock**/256).

† *Asserted* means the signal is true, or in its valid state. For example, the low-active **Reset*** signal is said to be asserted when it is in a low (true) state; the high-active **VCCOk** signal is true when it is asserted high.

Power-on Reset

The sequence for a power-on reset is listed below.

1. Power-on reset applies a stable Vcc of at least 4.75 volts from the +5 volt power supply to the processor. It also supplies a stable, continuous system clock at the processor operational frequency.
2. After at least 100 ms of stable Vcc and **MasterClock**, the **VCCOk** signal is asserted to the processor. The assertion of **VCCOk** initializes the processor operating parameters. After the mode bits have been read in, the processor allows its internal phase locked loops to lock, stabilizing the processor internal clock, **PClock**, the **SyncOut**-**SyncIn** clock path (described in Chapter 10), and the master clock output, **MasterOut**. Note that when JTAG is not used, **JTCK** must be tied low at the rising edge of **VCCOk** for the processor to properly reset. If JTAG is used, **JTCK** may be toggled during power-up.
3. **ColdReset*** is asserted for at least 64K (2^{16}) **MasterClock** cycles after the assertion of **VCCOk**. Once the processor reads the boot-time mode control serial data stream, **ColdReset*** can be deasserted. **ColdReset*** must be deasserted synchronously with **MasterClock**.
4. The deassertion of **ColdReset*** synchronizes the rising edges of **SClock** and **TClock** with the rising edge of the next **MasterClock**, aligning **SClock**, **TClock**, and **RClock** (which is 90 degrees ahead of phase with **SClock** and **TClock**) of all processors in a multiprocessor system. However, these clocks are only guaranteed to be stabilized 64 **MasterClock** cycles after **ColdReset*** is deasserted.
5. After **ColdReset*** is deasserted synchronously and **SClock**, **TClock**, and **RClock** have stabilized, **Reset*** is deasserted to allow the processor to begin running. (**Reset*** must be held asserted for at least 64 **MasterClock** cycles after the deassertion of **ColdReset***.) **Reset*** must be deasserted synchronously with **MasterClock**.

NOTE: ColdReset* must be asserted when **VCCOk** asserts. The behavior of the processor is undefined if **VCCOk** asserts while **ColdReset*** is deasserted.

Cold Reset

A cold reset can begin anytime after the processor has read the initialization data stream, causing the processor to start with the Reset exception. For information about saving processor states, see the description of the Reset exception in Chapter 5.

A cold reset requires the same sequence as a power-on reset except that the power is presumed to be stable before the assertion of the reset inputs and the deassertion of **VCCOk**.

To begin the reset sequence, **VCCOk** must be deasserted for a minimum of at least 64 MasterClock cycles before reassertion.

Warm Reset

To execute a warm reset, the **Reset*** input is asserted synchronously with **MasterClock**. It is then held asserted for at least 64 **MasterClock** cycles before being deasserted synchronously with **MasterClock**. The processor internal clocks, **PClock** and **SClock**, and the System interface clocks, **TClock** and **RClock**, are not affected by a warm reset. The boot-time mode control serial data stream is not read by the processor on a warm reset. A warm reset forces the processor to start with a Soft Reset exception. For information about saving processor states, see the description of the Soft Reset exception in Chapter 5.

The master clock output, **MasterOut**, can be used to generate any reset-related signals for the processor that must be synchronous with **MasterClock**.†

After a power-on reset, cold reset, or warm reset, all processor internal state machines are reset, and the processor begins execution at the reset vector. All processor internal states are preserved during a warm reset, although the precise state of the caches depends on whether or not a cache miss sequence has been interrupted by resetting the processor state machines.

† Since **MasterOut** is undefined until after the serial PROM is read, reset logic must not depend on **MasterOut** before the boot PROM is read.

9.3 Initialization Sequence

The boot-mode initialization sequence begins immediately after **VCCOk** is asserted. As the processor reads the serial stream of 256 bits through the **ModeIn** pin, the boot-mode bits initialize all fundamental processor modes (the signals used are described in Chapter 8).

The initialization sequence is listed below.

1. The system deasserts the **VCCOk** signal. The **ModeClock** output is held asserted.
2. The processor synchronizes the **ModeClock** output at the time **VCCOk** is asserted. The first rising edge of **ModeClock** occurs 256 **MasterClock** cycles after **VCCOk** is asserted.
3. Each bit of the initialization stream is presented at the **ModeIn** pin after each rising edge of the **ModeClock**. The processor samples 256 initialization bits from the **ModeIn** input.

Figures 9-1, 9-2, and 9-3 on the next three pages show the timing diagrams for the power-on, warm, and cold resets.

Power-on Reset (POR)

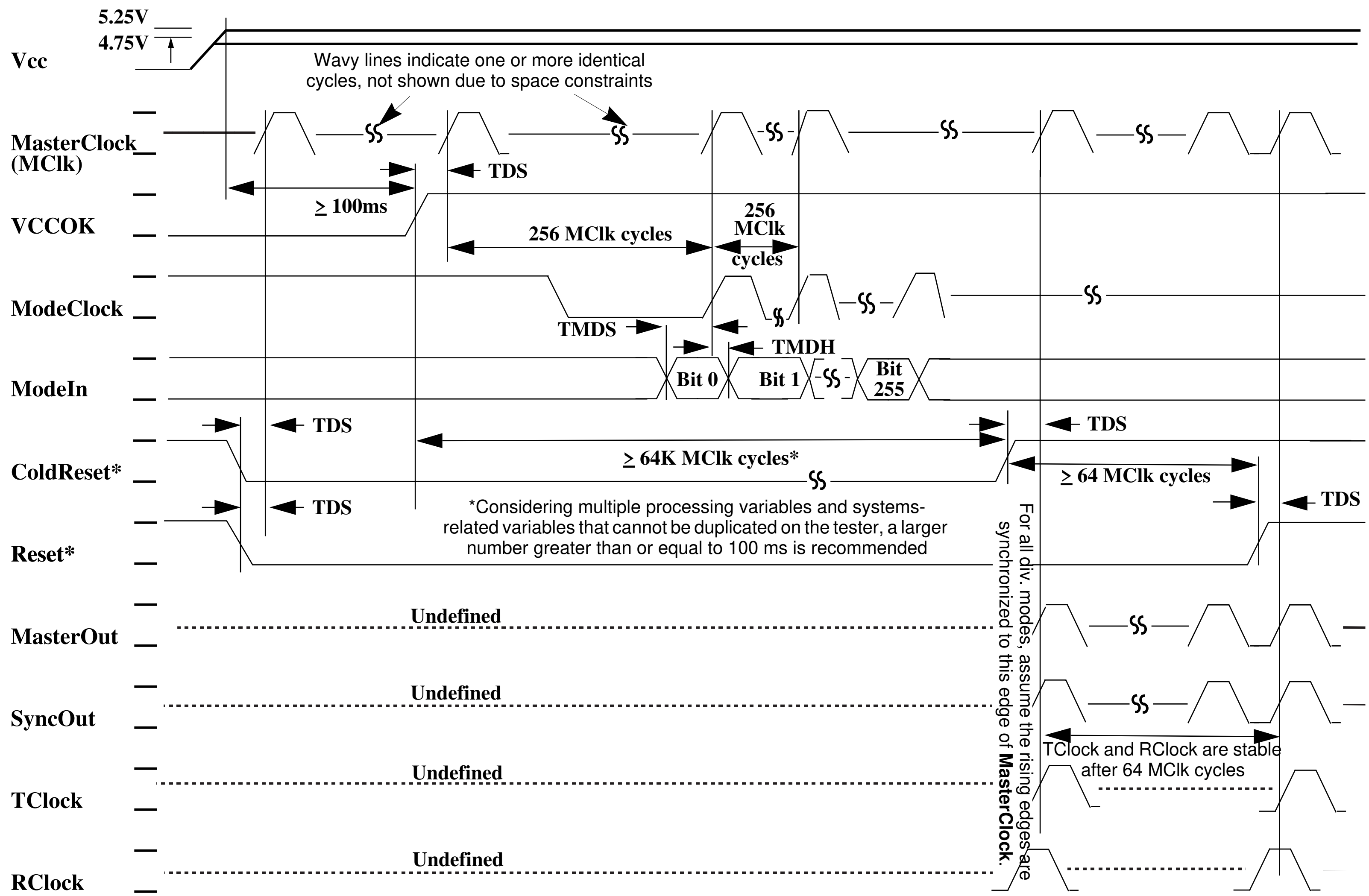

Figure 9-1 Power-on Reset

Cold Reset

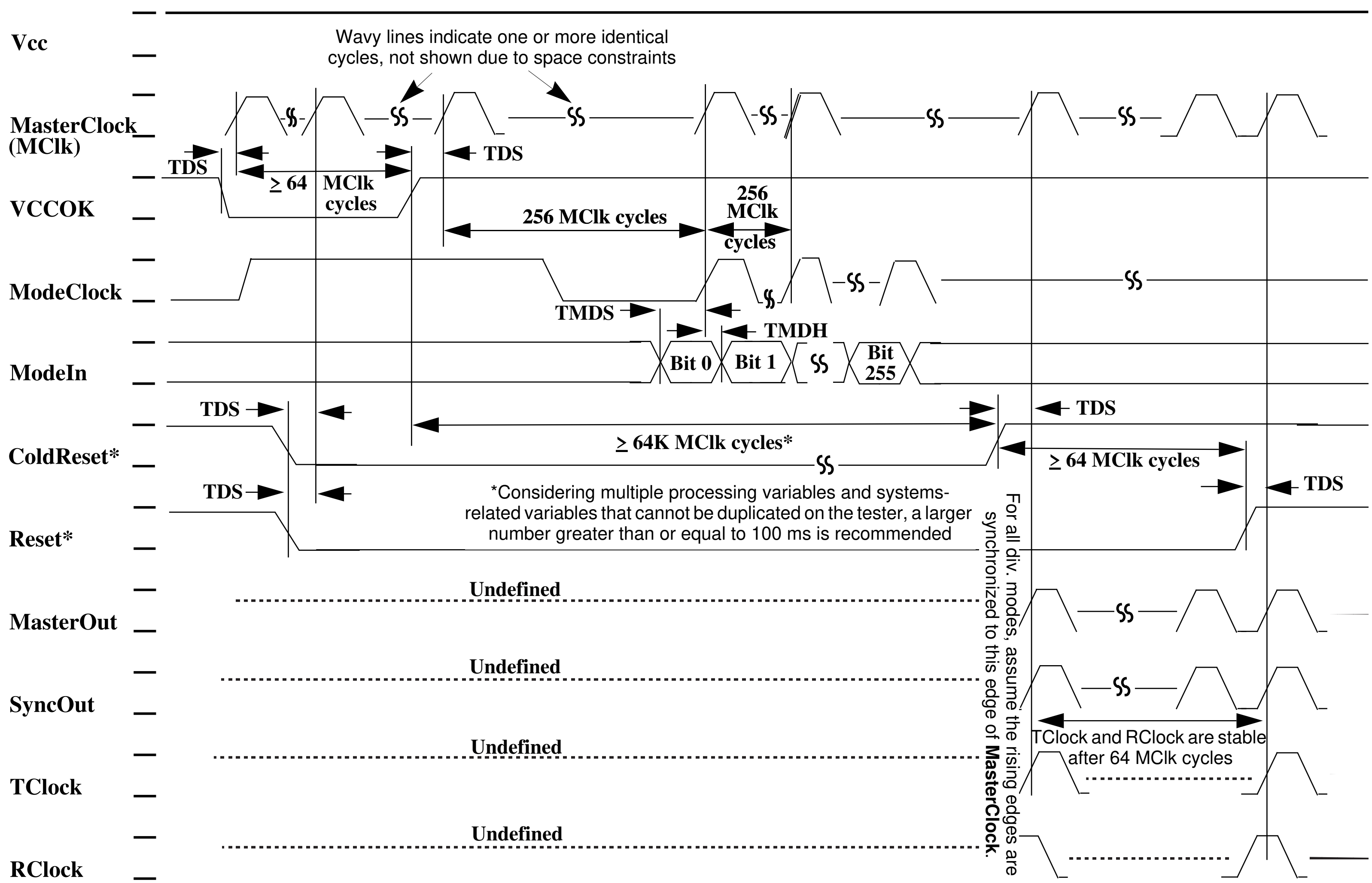

Figure 9-2 Cold Reset

Warm Reset

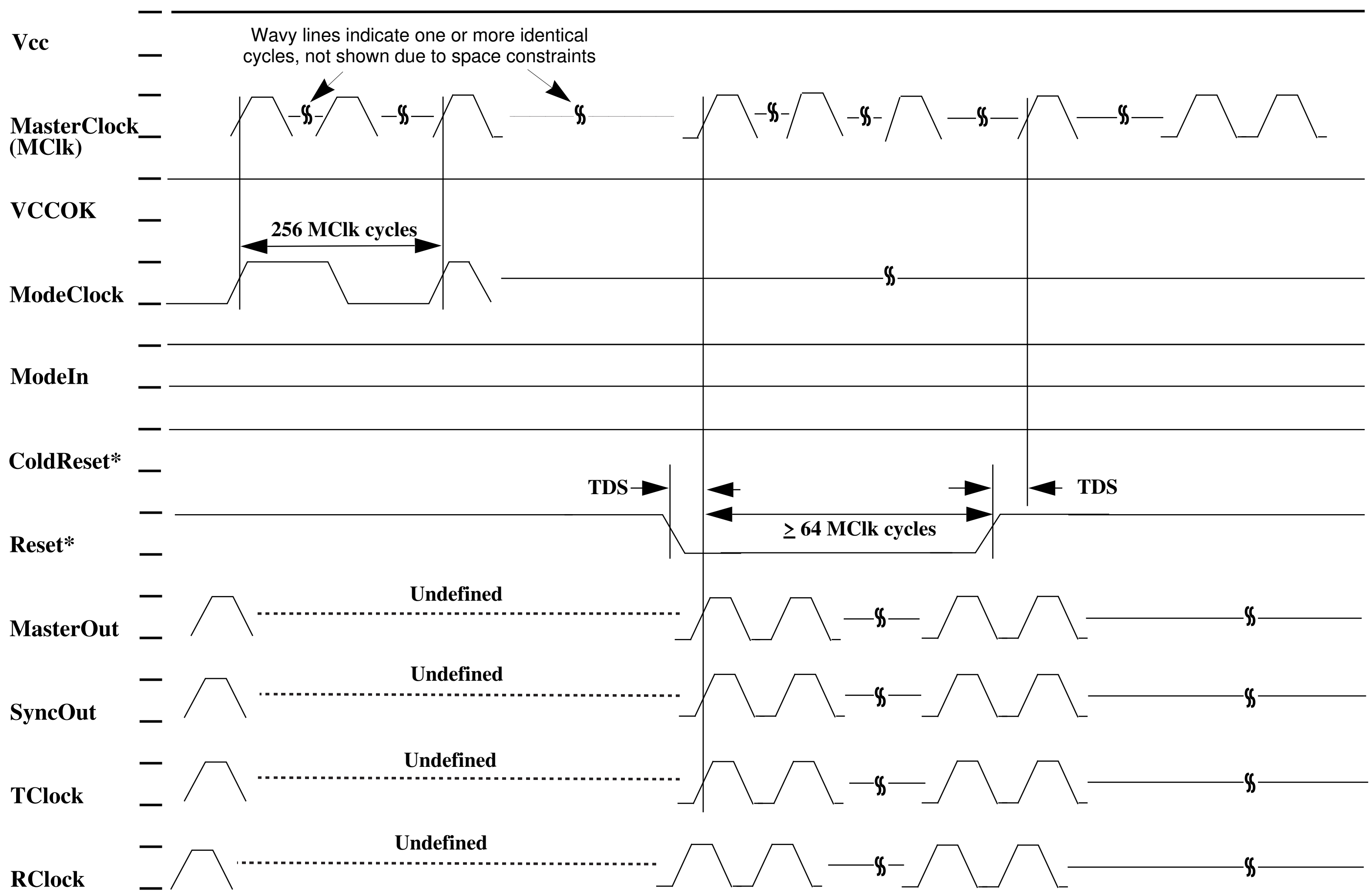

Figure 9-3 Warm Reset

9.4 Boot-Mode Settings

Table 9-1 lists the processor boot-mode settings. The following rules apply to the boot-mode settings listed in this table:

- Bit 0 of the stream is presented to the processor when **VCCOk** is first asserted.
- Selecting a reserved value results in undefined processor behavior.
- Bits 65 to 255 are reserved bits.
- Zeros must be scanned in for all reserved bits.

Table 9-1 Boot-Mode Settings

Serial Bit	Value	Mode Setting
0	**BlkOrder:** Secondary Cache Mode block read response ordering	
	0	Sequential ordering
	1	Subblock ordering
1	**EIBParMode:** Specifies nature of System interface check bus	
	0	Single error correcting, double error detecting (SECDED) error checking and correcting mode
	1	Byte parity
2	**EndBIt:** Specifies byte ordering	
	0	Little-endian ordering
	1	Big-endian ordering
3	**DShMdDis:** Dirty shared mode; enables the transition to dirty shared state on a successful processor update	
	0	Dirty shared mode enabled
	1	Dirty shared mode disabled
4	**NoSCMode:** Specifies presence of secondary cache	
	0	Secondary cache present
	1	No secondary cache present
5:6	**SysPort:** System Interface port width, bit 6 most significant	
	0	64 bits
	1-3	Reserved
7	**SC64BitMd:** Secondary cache interface port width	
	0	128 bits
	1	Reserved
8	**EISpltMd:** Specifies secondary cache organization	
	0	Secondary cache unified
	1	Secondary cache split

Table 9-1 (cont.) Boot-Mode Settings

Serial Bit	Value	Mode Setting
9:10	**SCBlkSz:** Secondary cache line length, bit 10 most significant	
	0	4 words
	1	8 words
	2	16 words
	3	32 words
11:14	**XmitDatPat:** System interface data rate, bit 14 most significant	
	0	D
	1	DDx
	2	DDxx
	3	DxDx
	4	DDxxx
	5	DDxxxx
	6	DxxDxx
	7	DDxxxxxx
	8	DxxxDxxx
	9-15	Reserved
15:17	**SysCkRatio:** PClock to SClock divisor, frequency relationship between SClock, RClock, and TClock and PClock, bit 17 most significant	
	0	Divide by 2
	1	Divide by 3
	2	Divide by 4
	3	Divide by 6 (R4400 processor only)
	4	Divide by 8 (R4400 processor only)
	5-7	Reserved
18	**SIMasterMd**: Master/Checker Mode (see mode bit 42); used in R4400 only.	
19	**TimIntDis:** Timer Interrupt enable allows timer interrupts, otherwise the interrupt used by the timer becomes a general purpose interrupt	
	0	Timer Interrupt enabled
	1	Timer Interrupt disabled
20	**PotUpdDis:** Potential update enable allows potential updates to be issued. Otherwise, only compulsory updates are issued	
	0	Potential updates enabled
	1	Potential updates disabled
21:24	**TWrSUp:** Secondary cache write deassertion delay, T_{WrSup} in PCycles, bit 24 most significant	
	0-2	Undefined
	3-15	Number of PClock cycles: Min 3, Max 15

Table 9-1 (cont.)   Boot-Mode Settings

Serial Bit	Value	Mode Setting
25:26	**TWr2Dly**: Secondary cache write assertion delay 2, T_{Wr2Dly} in PCycles, bit 26 most significant	
	0	Undefined
	1-3	Number of PClock cycles: Min 1, Max 3
27:28	**TWr1Dly:** Secondary cache write assertion delay 1, T_{Wr1Dly} in PCycles, bit 28 most significant	
	0	Undefined
	1-3	Number of PClock cycles; Min 1, Max 3
29	**TWrRc:** Secondary cache write recovery time, T_{WrRc} in PCycles, either 0 or 1 cycle	
	0	0 cycle
	1	1 cycle
30:32	**TDis:** Secondary cache disable time, T_{Dis} in PCycles, bit 32 most significant	
	0-1	Undefined
	2-7	Number of PClock cycles: Min 2, Max 7
33:36	**TRd2Cyc:** Secondary cache read cycle time 2, T_{RdCyc2} in PCycles, bit 36 most significant	
	0-1	Undefined
	2-15	Number of PClock cycles: Min 2, Max 15
37:40	**TRd1Cyc:** Secondary cache read cycle time 1, T_{RdCyc1} in PCycles, bit 40 most significant	
	0-3	Undefined
	4-15	Number of PClock cycles: Min 4, Max 15
41	**NoMPmode:** Secondary cache line is not invalidated	
	0	**NoMPmode** off: after a secondary cache miss, the existing valid cache line is invalidated (following writeback if necessary)
	1	**NoMPmode** on: after a secondary cache miss, the existing valid cache line is not invalidated. Available on the R4000SC and R4400SC, to improve performance.

Table 9-1 (cont.) Boot-Mode Settings

Serial Bit		Value	Mode Setting
42	**SCMasterMd**: selects the type of Master/Checker mode (also see description of mode bit 18). Used in R4400 only.		
	SCMasterMd (Bit 42)	**SIMasterMd (Bit 18)**	**Mode**
	0	0	Complete Master (required for single-chip operation)
	1	1	Complete Listener (paired with Complete Master)
	1	0	System Interface Master (SIMaster)
	0	1	Secondary Cache Master (SCMaster, paired with SIMaster)
43:45	0		Reserved
46	**Pkg179:** R4000 Processor Package type		
	0	Large (447 pin)	
	1	Small (179 pin)	
47:49	**CycDivisor:** This mode determines the clock divisor for the reduced power mode. When the *RP* bit in the *Status* register is set to 1, the pipeline clock is divided by one of the following values. Bit 49 is the most significant.		
	0	Divide by 2	
	1	Divide by 4	
	2	Divide by 8	
	3	Divide by 16	
	4-7	Reserved	
50:52	**Drv0_50, Drv0_75, Drv1_00:** Drive the outputs out in *n* x **MasterClock** period. Bit 52 is the most significant. Combinations not defined below are reserved.		
	1	Drive at 0.50 x **MasterClock** period	
	2	Drive at 0.75 x **MasterClock** period	
	4	Drive at 1.00 x **MasterClock** period	
53:56	**InitP:** Initial values for the state bits that determine the pull-down $\Delta i/\Delta t$ and switching speed of the output buffers. Bit 53 is the most significant.		
	0	Fastest pull-down rate	
	1-14	Intermediate pull-down rates	
	15	Slowest pull-down rate	

Table 9-1 (cont.) Boot-Mode Settings

Serial Bit	Value	Mode Setting
57:60	**InitN:** Initial values for the state bits that determine the pull-up Δi/Δt and switching speed of the output buffers. Bit 57 is the most significant.	
	0	Slowest pull-up rate
	1-14	Intermediate pull-up rates
	15	Fastest pull-up rate
61	**EnblDPLLR:** Enables the negative feedback loop that determines the Δi/Δt and switching speed of the output buffers during ColdReset.	
	0	Disable Δi/Δt mechanism
	1	Enable Δi/Δt mechanism
62	**EnblDPLL:** Enables the negative feedback loop that determines the Δi/Δt and switching speed of the output buffers during ColdReset and during normal operation.	
	0	Disable Δi/Δt control mechanism
	1	Enable Δi/Δt control mechanism
63	**DsblPLL:** Disables the phase-locked loops (PLLs) that match **MasterClock** and produce **RClock**, **TClock, SClock**, and the internal clocks.	
	0	Enable PLLs
	1	Disable PLLs
64	**SRTristate:** Controls when output-only pins are tristated	
	0	Only when **ColdReset*** is asserted
	1	When **Reset*** or **ColdReset*** are asserted
65:255	Reserved. Scan in zeros.	