MIPS R4400Master_Checker Errata, Processor Revision 1.0

May 26, 1993

MIPS Technologies Inc.
2011 N Shoreline Blvd
PO Box 7311
Mountain View, CA 94039-7311

This document contains information that is proprietary to MIPS Technologies, Inc.

MIPS Technologies, Inc. reserves the right to change any products described herein to improve the function or design. MIPS Technolgies, Inc. does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under patent rights nor imply the rights of others.

"Master/Checker Mode" is used when two R4400s are operated in the lock step. For detail explanation of this mode see "MIPS R4400 Features" document. Additional errata which affect PC or SC designs are listed in the "MIPS R4400 PC, R4400SC Errata". Change bars in the left column indicate corrections or changes from the last version of the errata.

1. Processors might not function in "lock-step" properly because the timer in CP0 (Count Register) may not synchronize across multiple processors at reset. As a result, the timers may increment on different clock edges across multiple parts causing either a timer interrupt to occur on a different cycle, or the value read from the count register to be off by one least significant bit.

Workaround: Do not use the Count Register as a timer when running in the master/checker mode..

2. The two chips running in the master/checker mode may go out of step, due to a circuit hazard in accessing the cache, under following condition:

Condition: As a result of a jump or eret instruction, there is a switch from unmapped to mapped space and the processor takes a TLB refill exception due to a miss at the destination address.

Symptom: The "out of step" behavior may appear, during the SCache access, as one extended cycle.

The operation of the processor is not affected by this behavior. User programs will work properly since they are normally run from the mapped space. However, the kernel must avoid the situation, discussed above, from occurring.

Workaround: When jump or eret switches the operation from unmapped to mapped space then the line in the line primary ICache that maps to the destination must be marked invalid. This can be done by using the Cache Index Invalidate instruction for the destination index address and also making sure that the jump or eret is not at the same cache index address as the destination index address.