

Device Errata **MC68360 QUICC**

Revision C.1. Masks 1E68C, 0G24F.

April 4, 1996

These errata items are valid on C.1 silicon. Please note that any errata listed in this document also applies to older revisions, unless otherwise stated. In addition to this document, please see the "Changes in Rev. C.1" document which mentions the differences between revision C.1 and previous versions of the 68360.

Revision C.2 mentioned in earlier sheets has been renamed E.1 Revision E.1 is mentioned in this document but is not sampling at the time of writing. Revision D and Revision E.0 do not exist.

All items changed since the January 26, 1996 Rev C.1 errata sheet are in *Italics*.

CPU32+ Errata

no errata

SIM60 Errata

1. Corrupt access after DMA bus error

The address of a bus cycle can be corrupted if the following circumstances occur:

A DMA bus cycle (from either IDMA unit or any of the SDMA channels) is terminated by BERR* and the following cycle from the CPU is a non-aligned access (and thus takes two bus cycles to complete).

If this sequence of events occurs, the address of the second CPU bus cycle will be corrupted (it will be the address of the second CPU bus cycle logically ANDed with the address of the first bus cycle).

Work-arounds:

* Use 16-bit mode of the IDMA rather than 32-bit mode. (This will not work-around the SDMA case. However, SDMA bus errors typically do not occur in properly performing systems)

OR

* Don't use the QUICC bus monitor. Use external circuitry to implement the bus monitor. If the bus cycle to be terminated is a CPU cycle (FC(3)=0), the circuitry should assert BERR*. If the bus cycle to be terminated is a DMA cycle (FC(3)=1), generate DSACK* and assert BERR* on the clock following DSACK* negation.

OR

This document contains information on a product under development. Motorola reserves the right to change or discontinue this product without



MOTOROLA



* Set the SHEN bits in the MCR register to '1x' (show cycles) and have external circuitry assert BR* with the following formula: $BR^* = !(FC(3) \& !AS^*)$. This will generate a one clock idle on the internal bus and avoid the problem.

This item will be fixed in Rev. E.1.

2. PIT and Software Watchdog not deterministic

The prescalers in the periodic interrupt timer (PIT) and the software watchdog (SWT) are not affected by a chip reset. Therefore, the first PIT interrupt or the SWT timeout could occur up to 512 system clocks early if the part has undergone a reset other than power-up. This will be fixed in Rev E.1.

3. Parallel I/O pins are not three-stated immediately after Reset

When the QUICC is reset, the parallel I/O pins (Port A, B, and C) are not three stated until the PLL locks. Thus, any port pins that are programmed as outputs may continue driving until the PLL re-locks after reset. This problem will also occur during power-up reset. In this case, the parallel I/O pins will be in an indeterminate state until the PLL locks. The fix for this has not yet been scheduled.

4. Hardware Breakpoint Generation Error

The hardware breakpoint logic will not detect/generate a breakpoint on any address in an 8-bit or 16-bit memory device. The fix has not yet been scheduled.

CPM Errata

1. Transparent SCC Bit Reversal

If you are running an SCC in transparent mode AND you have the REVD bit in the GSMR set to 1 AND you are transmitting data from a Buffer Descriptor that has its TC bit set to 0 AND its L bit set to 1, the bytes from the last long word will not be reversed. The workaround is to add the value \$FFFFFFFF (which is the same as an idle pattern) to the end of the buffer and to increase the data count by 4. Alternately, the user could manually reverse the data in the last longword rather than padding the buffer. This will be fixed in Rev. E.1.

2. Cannot avoid transmitting BREAK characters in SMC UART

If the SMC port is in UART mode and the STOP TRANSMIT command is executed and the value of the BRKCR register is zero, the SMC UART transmitter will transmit 65536 break characters instead of transmitting no break characters. This will be fixed in Rev E.1. A microcode patch is available that will fix the problem on earlier revisions of silicon.

3. SMC UART Short Start Bit Recognition

If an SMC is being used in UART mode and the received character has a start bit that is shorter than 15/16ths of a bit time, the character will not be received. The limit for start bit recognition will be changed to 9/16ths in Rev E.1.

4. Timer Inaccurate

If timer2 is programmed to be a free-run timer and the user turns timer 1 on and off repeatedly, timer 2 will not increment properly and will count more slowly. This phenomenon will also occur if timer 4 is programmed to be a free-run timer and the user turns timer 3 on and off repeatedly. The work around is to place the free running timers on one pair of timers (1 and 2) and place the timers that are manipulated frequently onto the other pair of timers.(3 and 4). This will be fixed in Rev E.1.



5. Spurious SI Sync Pulse Behavior

If:

1) The Time Slot Assigner is programmed to a ram division mode of "00" (One TDM with Static Frames)

AND

2) The Time Slot Assigner is currently processing an entry in the SI RAM past the 32nd entry

AND

3) A pulse is received on the L1SYNC pin (i.e. before the TSA expects a new frame to begin)

The TSA will reset to the first entry in the SI RAM. This will cause the QMC protocol to lose frame synchronization. The workaround is to not use more than 32 entries in the SI RAM. This will be fixed in a future revision.

JTAG Errata

no errata