FREESCALE SEMICONDUCTOR, MICROCONTROLLER DIVISION
CUSTOMER ERRATA AND INFORMATION SHEET
Part: MPC565.D
General Business Use
Report Generated: Wed Oct 27, 2010, 13:46:18

MPC565.D

ERRATA AND INFORMATION SUMMARY

AR_865 USIU: Do not rely on the VDDSRAM Low Voltage Detect Circuit
AR_912 Bit 15 of the Reset Configuration Word is not documented
AR_1035 MPC565: Masknum is 0x20
AR_1036 REV field in the Readi is 0x2
AR_1072 MPC565: Part marking is Rev D
AR_1097 MPC565.D: New functional features (SST, 8-beat burst, BBC options) added
AR_1098 TPUROM: ID is now 0x3
AR_904 BBC2: Branch targets must be 4 sequential instructions before MTSPR BBC SPR.
AR_870 BBC2: Do not use debug mode with BTB enabled, if code has 0x2F30 branch target.
AR_1079 BBC2: Flush the BTB if instructions in a region are changed during execution
AR_1121 BBC2: Do not run software from the DECRAM that modifies the DECRAM contents
AR_793 BBC2 Compression: No Compressed Code in Addresses $FFF00000 to $FFFFFFFF
AR_1146 UC3F: Contents of 0x00-0x1F may be invalid after HRESET if using RCW from flash
AR_973 UC3F: Frequent Suspend/Resume Operations may cause Program or Erase Timeouts
AR_1104 UC3F: The UC3FMCRE[FLASHID] field is 0x04
AR_1132 UC3F: UC3F registers can only be accessed when in supervisor mode
AR_1146 UC3F: Contents of 0x00-0x1F may be invalid after HRESET if using RCW from flash
AR_973 UC3F: Frequent Suspend/Resume Operations may cause Program or Erase Timeouts
AR_1104 UC3F: The UC3FMCRE[FLASHID] field is 0x04
AR_1132 UC3F: UC3F registers can only be accessed when in supervisor mode
AR_1087 DLCMD2: do not allow RxFIFO to overflow or read using a high priority interrupt
AR_1126 DLCMD2: Transmitter may need to be reset after losing arbitration
AR_755 DLCMD2 switching into 1x-4x mode
AR_772 DLCMD2 Equating Parameter Values
AR_785 DLCMD2 RFIFO Polling
AR_771 DLCMD2 SEL bit is not lockable.
AR_1143 L2U: Care required when changing a slave MCU’s mode in multi-master systems
AR_929 MIOS: Problem with DASM Duty Cycle Change to 0%
AR_1127 MIOS: MDASMSCR polarity bit has no effect when open-drain mode selected
AR_1131 PADS: The weak internal pull device remains enabled on RSTCONF/TEXP
AR_922 PADS: VOH2.6 Spec changed to -1 mA
AR_932 PADS: Leakage higher when 2.6V pads are pulled above 2.6V supply during reset
AR_940 JTAG: Do Not Switch All Pads Simultaneously When JTAG Enabled
AR_1018 Execute memory write prior to slave read for slave predischarge
AR_1019 RCPU: Don’t execute overflow type before update type MUL/DIV instruction
AR_1138 RCPU: Data breakpoint exception may occur even if conditions are not met
AR_907 RCPU: Issue ISYNC command when entering debug mode
AR_440 RCPU: Execute any IMUL/DIV instruction prior to entering low power modes.
AR_211 Do not set breakpoint on mtspr ICTRL instruction
AR_214 Only negate interrupts while the MSR[EE] disables interrupts (MSR[EE]=0)
AR_949 QADC64E: Write CCW[EOQ] to 0x3F for the End of Queue
AR_915 QADC64E: Conversion Clock cannot be shared between Master/Slave Modules
AR_1125 QADC64: Don’t change both BQ2 and MQ2 while Q2 is running
AR_420 QADC64: Don’t change BQ2 with a set of SSE2 without a mode change.
AR_1048 QADC64E: Sample Time is 8 QCLKs instead of 16 when CCW[IST]=1 in Enhanced Mode
AR_1151 SCI: TXD pin reverts to output immediately when SCCxR1[TE] is cleared
AR_1118 READI: Program Flow Tracking Error Under Rare Condition
AR_846 READI: Synchronize the MCKI input clock to the MCKO output clock.
AR_698 READI Input message requires 2 MCKI idle after READI Enabled.
AR_1096 READI: New features to enhance Nexus debug
AR_1144 TouCAN: Transmit buffers may freeze or indicate missing frame
AR_1045 CAN: Bus Off recovery not ISO compliant

AR_1142 TouCAN: Writing to an active receive MB may corrupt MB contents
AR_627 TPU: (Microcode) Add neg_mrl with write_mer and end_of_phase
AR_910 USIU: PITRTC Clock may not work when SCCR[RTDIV] is 0
AR_1134 USIU: RTC, DEC, TB and PIT counters may not count after PORESET or HRESET
AR_1158 USIU: Stop Time Base to write new value
AR_287 USIU: System to Time Base frequency ratio must be greater than 4
AR_479 USIU: The MEMC does not support external master burst cycles
AR_1135 USIU: Disable USIU burst in debug mode if READI R/W feature is used
AR_1152 USIU: PORESET must always be asserted before the 2.6V supplies reach 0.5V
AR_867 USIU: Do not operate USIU burst on a burst-inhibited memory region
AR_1154 SIU: RTSEC register not documented; May affect the initial increment of the RTC
AR_1113 USIU: Ensure HRESET/SRESET negation time is longer than 3 CLKOUT periods
AR_869 USIU: Do not enable BRx[SST] with SCCR[EBDF]>0
AR_895 USIU: Do not assert TEA pin on fetch while SIUMCR[BURST_EN] bit set.
AR_965 USIU: Program All Chip Selects with the same Burst Length
AR_1153 USIU: Sleep and Deep-Sleep modes require power to all 2.6V supplies
AR_389 Little Endian modes are not supported
AR_1109 USIU: Do not write zero value to the SYPCR[BMT]
AR_1120 USIU: Interrupt Controller may generate vector 0x0 or has no request indication
AR_1137 USIU: RSR[LLRS] can be set even though no loss of lock reset has occurred
AR_1155 SIU: TEA for external access must be negated within 1 system bus clock

DETAILED ERRATA DESCRIPTIONS

CDR_AR_865 Customer Information MPC565.D

USIU: Do not rely on the VDDSRAM Low Voltage Detect Circuit

DESCRIPTION:
At temperatures above room ambient (25C), the VDDSRAM low voltage detect circuit may not always indicate that the VDDSRAM voltage has dropped below the minimum data retention level for the SRAM.

WORKAROUND:
Utilize an external mechanism for detecting when the voltage supplied to the VDDSRAM pin(s) is below the minimum data retention voltage. Refer to manual dated on or after May 2003.

CDR_AR_912 Customer Information MPC565.D

Bit 15 of the Reset Configuration Word is not documented

DESCRIPTION:
Bit 15 of the Reset Configuration Word is not documented in the Reference Manual. Bit 15 is the Interlock Write Select (IWS). This bit determines which interlock write operation should be used during the clear censorship operation. 0 = Interlock write is defined as a write to any UC3F array location. 1 = Interlock write is a write to the UC3FMCR register. This bit always comes from the shadow row of the flash module being accessed and never comes from the External Reset Configuration Word.

WORKAROUND:
Consult the latest version of the Reference Manual (dated June 29,2001 or later). The net affect is that if bit 15 is cleared (0), then the censor bits can only be cleared while running in an uncensored mode. If bit 15 is set (1), then the censor bits can also be cleared in censored mode by performing a clear censorship operation, which erases the flash module. Refer to manual dated after June 2001.

CDR_AR_1035 Customer Information MPC565.D

MPC565: Masknum is 0x20

DESCRIPTION:
The Masknum field of the IMMR register was updated from 0x11 to 0x20 to indicate a minor revision of the device mask set.

WORKAROUND:
Update software that reads the IMMR register to read 0x3320 instead of 0x3311 for the PARTNUM:MASKNUM.

CDR_AR_1036 Customer Information MPC565.D

REV field in the Readi is 0x2

DESCRIPTION:
The REV field of the Readi Device ID register has been changed to 0x2 and may change for future revisions.

WORKAROUND:
Modify software to expect a new value (0x2) for the REV field of the Readi DID.

CDR_AR_1072 Customer Information MPC565.D

MPC565: Part marking is Rev D

DESCRIPTION:
On devices that no longer show the Mask Set in the part marking, the part marking may show the part Revision, for example, Rev D. Note: During this part marking change from Mask Set to Revision, some parts may be marked with a D as the part suffix (i.e. MPC565MZP56D - 56 MHZ).

WORKAROUND:
If the part marking does not indicate the revision, read the Mask Number field of the Internal Memory Map Register (IMMR[MASKNUM]) to determine the revision of the MPC565.

CDR_AR_1097 Customer Information MPC565.D

MPC565.D: New functional features (SST, 8-beat burst, BBC options) added

DESCRIPTION:
New features have been added to the MPC565.D. These features are (1) an optional 8 beat burst from external execution memories and (2) a short Setup Time option in the Memory Controller for external memories. By default, both of these new features are disabled after reset. (3) The BBCMCR[EN_COMP] can now be written by supervisor code (MPC566 only), previously, it could only be set by the Reset Configuration Word.

WORKAROUND:
Bit 28 of the SIUMCR register is now BURST_EN (Burst Enable) to enable the enhanced burst feature of the USIU. Bit 25 of the BRx registers is now BL (Burst Length). The value of BRx[BL] bit determines the USIU burst length, 4 word (default) or 8 word. Note that the BBCMCR[BE] should be disabled (0) if the USIU burst is enabled. Bit 22 of the BRx registers is now SST to control the Short Setup Time feature. Short Setup time gives additional setup time for memories at the cost of an additional cycle between memory bursts. Compression can now be enabled in software, even if the Reset Configuration Word had it disabled. Note: this does not put the RCPU into compressed mode, it only enables the capability of switching to compressed code. See updated Reference Manuals dated January 2003 or later for additional information on these features.

CDR_AR_1098 Customer Information MPC565.D

TPUROM: ID is now 0x3

DESCRIPTION:
A new multi-function has been added to the TPUROM utilizing the previously unused function 7 in bank 1. The new functions include: running average speed, a PWM analyzer, and a 32-bit free running counter that can be either incrementing or decrementing.

WORKAROUND:
Read the version of the TPUROM using the ID function before using the new function. Refer to updated documentation on the use of these new functions. Note this version of the TPU is 0x3 and also contains the fixed COMM function (See CDR_AR_1082).

CDR_AR_904 Customer Erratum BBC2.4K_CDR3UBUS_09_0

BBC2: Branch targets must be 4 sequential instructions before MTSPR BBC SPR.

DESCRIPTION:
A user application may crash when a BBC SPR is written in a program loop, if the MTSPR is within 4 instructions of a branch target.

WORKAROUND:
1) Make sure that a "mtspr" instruction writing to any BBC SPR register is preceded by four instructions that are not the target of any branch and followed by "isync" instruction, or 2) Disable BTB. Refer to manual dated on or after May 2003.

CDR_AR_870 Customer Erratum **BBC2.4K_CDR3UBUS_09_0**

BBC2: Do not use debug mode with BTB enabled, if code has 0x2F30 branch target.

DESCRIPTION:
The BTB (Branch Target Buffer) incorrectly matches in debug mode if there was a change of flow address of 0x2F30 and code from the address resides in the valid BTB buffer when the part enters debug mode. The address 0x2F30 is the debug port instruction register (SPR) address that the core issues to the BBC in debug mode for instruction fetches. The debug tool may lose communication with the part since the debug port will not assert the "ready" status (DSDO pin "low") until reset.

WORKAROUND:
Do not use debug mode on applications running with the BTB enabled if there is a branch with a target address of 0x2F30. Alternatively, either do not enable the BTB in debug mode or do not put any code at 0x2F30.

CDR_AR_1079 Customer Erratum **BBC2.4K_CDR3UBUS_09_0**

BBC2: Flush the BTB if instructions in a region are changed during execution

DESCRIPTION:
If an IMPU region register has the BTB inhibit bit set (MI_RAx[BTBINH] = 1), the BTB inhibit function does not work for the first branch pointing into the region. These instructions will be stored in a vacant BTB table entry. Any following branches in the same region will NOT be stored in the BTB. This is the correct operation. In addition, the instructions following a branch out of the region will not be stored in the BTB table. This issue will only cause problems if there is a possibility that the instructions at a cached address are changed after they have been executed once.

WORKAROUND:
1) Disable the BTB if the caching from a memory region is undesirable; or 2) Flush the BTB by disabling and then re-enabling the BBCMCR[BTEE] any time the contents of a memory changes, prior to executing from that memory.

CDR_AR_1121 Customer Erratum **BBC2.4K_CDR3UBUS_09_0**

BBC2: Do not run software from the DECRAM that modifies the DECRAM contents

DESCRIPTION:
When executing code from the DECRAM, a store instruction with destination address in the DECRAM may result in an overwrite of that code area.

WORKAROUND:
Do not perform data writes to the DECRAM while also executing code from DECRAM. The DECRAM should only be loaded while executing from a different memory.

CDR_AR_793 Customer Information BBC2.4K_CDR3UBUS_09_0

BBC2 Compression: No Compressed Code in Addresses $FFF00000 to $FFFFFFFF

DESCRIPTION:
IMPU translates addresses in compression mode regardless of address form. Note that this may have a minor application impact. It will cause a failure ONLY if the compressed address space covers $FFF00000 to $FFFFFFFF and the BBCMCR[ETRE] and BBCMCR[EIR] are set, enabling Exception Table Relocation and Enhanced External Interrupt Relocation.

WORKAROUND:
Do not put compressed code at addresses $FFF00000 to $FFFFFFFF if Exception Table Relocation or Enhanced External Interrupt Relocation are enabled by BBCMCR[ETRE] and BBCMCR[EIR]. Refer to manual dated on or after May 2003.

CDR_AR_1146 Customer Erratum C3FARRAY_A.512KCDR3_04_0

UC3F: Contents of 0x00-0x1F may be invalid after HRESET if using RCW from flash

DESCRIPTION:
When using the Reset Configuration Word (RCW) located in internal flash, addreses 0x00-0x1F of the UC3F may be incorrectly read from the shadow row instead of the data array after HRESET. This will continue to occur until an instruction is fetched from an address outside the 0x00-0x1F range. For example, when the internal flash RCW is programmed with exception table relocation enabled (IP = 1 and ETRE = 1) and the vector table base address is 0x0 (OERC = 0b00), the reset vector absolute branch may not be correctly fetched and the application may not start properly.

WORKAROUND:
Option 1) Use the external RCW or the default internal RCW. Option 2) When using the internal flash RCW: A) Ensure the first instruction fetch from the flash is outside of 0x00-0x1F by locating the reset vector outside of 0x00-0x1F. This can be done by setting IP = 0 or ETRE = 0 (reset vector at 0x100), or by setting IP = 1, ETRE = 1, and OERC != 0b00 (reset vector at 0x10008, 0x80008 or 0x3FE008). The vector table can be relocated later in the application if required. OR B) Program the same data/instructions to addresses 0x00-0x1F of both the shadow row and the data array.

CDR_AR_973 Customer Information C3FARRAY_A.512KCDR3_04_0

UC3F: Frequent Suspend/Resume Operations may cause Program or Erase Timeouts

DESCRIPTION:
Frequent use of the suspend or resume feature can cause a premature timeout of the program or erase time. The internal program and erase state machine has a counter for the maximum allowable program or erase time. This counter is incremented prior to the actual program or erase pulse. If the operation is suspended at a high frequency, it is possible that the upper count will be reached prior to completion of the program or erase operation. This will primarily be seen during erase operations.

WORKAROUND:
Do not suspend a program or erase at a high frequency (more than approximately once per millisecond). Refer to manual dated on or after May 2003.

CDR_AR_1104 Customer Information C3FARRAY_A.512KCDR3_04_0

UC3F: The UC3FMCRE[FLASHID] field is 0x04

DESCRIPTION:
The UC3FMCRE[FLASHID] has been changed to 0x04 and may change for future revisions of the flash module.

WORKAROUND:
Modify software to expect a new value (0x04) for the FLASHID field of the UC3FMCRE.

CDR_AR_1132 Customer Information C3FARRAY_A.512KCDR3_04_0

UC3F: UC3F registers can only be accessed when in supervisor mode

DESCRIPTION:
All UC3F registers are accessible only when the device is in supervisor mode. Any attempt to access the UC3F registers in user mode will terminate the cycle with a data error exception.

WORKAROUND:
Ensure all UC3F registers are accessed only when in supervisor mode. Refer to manual dated after September 2003.

CDR_AR_1146 Customer Erratum C3FARRAY_B.512KCDR3_04_0

UC3F: Contents of 0x00-0x1F may be invalid after HRESET if using RCW from flash

DESCRIPTION:
When using the Reset Configuration Word (RCW) located in internal flash, addreses 0x00-0x1F of the UC3F may be incorrectly read from the shadow row instead of the data array after HRESET. This will continue to occur until an instruction is fetched from an address outside the 0x00-0x1F range. For example, when the internal flash RCW is programmed with exception table relocation enabled (IP = 1 and ETRE = 1) and the vector table base address is 0x0 (OERC = 0b00), the reset vector absolute branch may not be correctly fetched and the application may not start properly.

WORKAROUND:
Option 1) Use the external RCW or the default internal RCW. Option 2) When using the internal flash RCW: A) Ensure the first instruction fetch from the flash is outside of 0x00-0x1F by locating the reset vector outside of 0x00-0x1F. This can be done by setting IP = 0 or ETRE = 0 (reset vector at 0x100), or by setting IP = 1, ETRE = 1, and OERC != 0b00 (reset vector at 0x10008, 0x80008 or 0x3FE008). The vector table can be relocated later in the application if required. OR B) Program the same data/instructions to addresses 0x00-0x1F of both the shadow row and the data array.

CDR_AR_973 Customer Information C3FARRAY_B.512KCDR3_04_0

UC3F: Frequent Suspend/Resume Operations may cause Program or Erase Timeouts

DESCRIPTION:
Frequent use of the suspend or resume feature can cause a premature timeout of the program or erase time. The internal program and erase state machine has a counter for the maximum allowable program or erase time. This counter is incremented prior to the actual program or erase pulse. If the operation is suspended at a high frequency, it is possible that the upper count will be reached prior to completion of the program or erase operation. This will primarily be seen during erase operations.

WORKAROUND:
Do not suspend a program or erase at a high frequency (more than approximately once per millisecond). Refer to manual dated on or after May 2003.

CDR_AR_1104 Customer Information C3FARRAY_B.512KCDR3_04_0

UC3F: The UC3FMCRE[FLASHID] field is 0x04

DESCRIPTION:
The UC3FMCRE[FLASHID] has been changed to 0x04 and may change for future revisions of the flash module.

WORKAROUND:
Modify software to expect a new value (0x04) for the FLASHID field of the UC3FMCRE.

CDR_AR_1132 Customer Information C3FARRAY_B.512KCDR3_04_0

UC3F: UC3F registers can only be accessed when in supervisor mode

DESCRIPTION:
All UC3F registers are accessible only when the device is in supervisor mode. Any attempt to access the UC3F registers in user mode will terminate the cycle with a data error exception.

WORKAROUND:
Ensure all UC3F registers are accessed only when in supervisor mode. Refer to manual dated after September 2003.

CDR_AR_1087 Customer Erratum DLCMD2.CDR3IMB3_04_0

DLCMD2: do not allow RxFIFO to overflow or read using a high priority interrupt

DESCRIPTION:
The DLCMD2 RxFIFO status register (STAT) bit 6 will become stuck high (1), indicating that there is a completion code in the RxFIFO. This occurs when a completion code is being read from the RxFIFO and a completion code is being written into the RxFIFO simultaneously. This occurance is highly unlikely, but may occur over time if the RxFIFO is allowed to repeatedly overflow, causing completion codes to be continually written to the RxFIFO.

WORKAROUND:
Implement a high priority service routine to insure that the DLCMD2 RxFIFO is read before another message completes and is put into the RxFIFO. This will prevent the possibility that completion codes will be written and read simultaneously.

CDR_AR_1126 Customer Erratum DLCMD2.CDR3IMB3_04_0

DLCMD2: Transmitter may need to be reset after losing arbitration

DESCRIPTION:
The DLCMD2 transmitter may incorrectly remain idle if it loses arbitration and an Incomplete Byte Received error or a Bit Timing Error is received before an End Of Frame is detected, and a Terminate Auto Retry (TAR) command was not previously issued. The Receiver Status Register (STAT) will indicate that the TxFIFO is full, but no transmission will occur until the transmitter is reset and a new message loaded into the TxFIFO.

WORKAROUND:
Reset the transmitter by issuing an Abort Transmission Now command (CMD[7:5] = 0b111) if either of the following cases occur when reading a valid completion code (CC[6] = 0). CASE 1: The completion code indicates a Transmitter Lost Arbitration status, and either an Incomplete Byte Received error or a Bit Timing Error (CC[7:0] = 0b1010xx01, or 0b1010xx10). CASE 2: STAT indicates the TxFIFO is full (STAT[1:0] = 0b11), and the completion code indicates Transmitter Not Involved for the transmitter action status (CC[5:4] = 0b00).

CDR_AR_755 Customer Information DLCMD2.CDR3IMB3_04_0

DLCMD2 switching into 1x-4x mode

DESCRIPTION:
If 4x mode is entered before the symbol counter value reaches the normal mode TIFS value but after the counter has passed the 4x mode TIFS value, the module will hang. Before a transmitter can send an SOF (which resets the symbol counter) it must wait for either of the two following conditions. One, TIFS must have been reached. Two, REOF and a rising edge from another module must have been detected. The second condition means that if another module tries to access the bus before TIFS and after REOF then we can also contend and try and gain access to the bus. If no other module is trying to access the bus then condition two won’t occur. The symbol counter does not reset when the mode is changed. This means that if the module is put into 4x mode before the normal mode TIFS value has been detected (which would signal an SOF and reset the counter) the module will keep counting until it reaches its max value and holds. Since the counter is stuck at its max value the module can never detect any symbols on the bus so it will hang until reset.

WORKAROUND:
TIFS must be waited for before changing to 4x mode. To wait for TIFS the difference between the normal mode REOF and TIFS values must be found manually based on their values in the SDATA register. Once that value is determined, wait for bus_idle (REOF), which can be polled for, plus (TIFS - REOF) amount of time. Assuming all cycle counts for normal mode parameters are greater than or equal to the cycle counts for the 4x parameters, no delay (waiting for TIFS) is needed to transition from 4x to normal mode. Note that the bus should be idle when the transition takes place (regardless as to whether the DLCMD2 has detected idle or not) as it is always a bad idea to transition between modes during a message.

CDR_AR_772 Customer Information DLCMD2.CDR3IMB3_04_0

DLCMD2 Equating Parameter Values

DESCRIPTION:
If certain symbols are assigned the same counter value, one of the symbols might never be recognized. Instead the other symbol with the same value would be detected. For example, if RMIN=TSHA than an active RMIN will never be recognized. In the hardware implementation the parameter values are checked with a priority scheme. Once a parameter matches, no further checking is done until the next clock cycle when the counter value has changed. For an active pulse the following parameters are checked in order: TSOF, TSHA, TLNA, TBRK, RMIN, RSH, RLN, and REOF. In order for each parameter to be detectable they must all have unique values. Similarly, during a passive pulse the following parameters are checked in order: TSHP, TLNP, TIFR, TIFS, RMIN, RSH, RLN, REOF. Again, this set of passive parameters must be distinct. Keeping these parameters distinct is almost guaranteed due to the timing requirements of the J1850 specifications. Certain round trip delays in the transceiver may suggest that some parameters should be equated. In this case either parameter should be adjusted to make them unique. The real risk of equation parameters occurs in test mode when parameters are set as small as possible to accelerate testing.

WORKAROUND:
Parameters in the active set must all be distinct and parameters in the passive set must all be distinct.

CDR_AR_785 Customer Information DLCMD2.CDR3IMB3_04_0

DLCMD2 RFIFO Polling

DESCRIPTION:
Reads from the staus/rdata word result in valid data being popped from the RFIFO. If the FIFO is empty no pop occurs. This behavior is implemented as follows. First, the status and data at the head of the FIFO is returned. If the FIFO is currently empty, the status register will indicate this and the FIFO data returned will be invalid. After the read has completed, the FIFO will be popped if not empty. The problem occurs if data has been pushed since the status register read indicated an empty FIFO. In this case, when the pop is requested, the FIFO contains valid data which is then popped and lost.

WORKAROUND:
When polling for data, access the status register with a byte read access. Upon finding valid data present in the FIFO, access both the status and data with a word read access.

CDR_AR_771 Customer Information DLCMD2.CDR3IMB3_04_0

DLCMD2 SEL bit is not lockable.

DESCRIPTION:
The spec states that if LCK=1 writes to the SEL bit are disabled. This is not the case. The SEL bit can still be written when LCK=1. This allows the user to read both the 1x SDATA parameters and the 4x SDATA parameters. If writes to the SEL bit were not allowed, after the LCK bit was set it would only be possible to view one set of SDATA parameters depending on the state of the SEL bit wen the LCK bit was set.

WORKAROUND:
This is the desired operation of the DLCMD2. The reference manual will be updated to reflect this operation.

CDR_AR_1143 Customer Information L2U.CDR3LBUSUBUS_03_0

L2U: Care required when changing a slave MCU’s mode in multi-master systems

DESCRIPTION:
If an external master changes the mode of a slave MCU from slave to peripheral mode by setting EMCR[PRPM], and then accesses addresses on the slave MCU’s LBUS at the same time as the slave MCU’s RCPU accesses addresses over the UBUS for data, a deadlock may occur. The slave MCU may lock up until reset assertion.

WORKAROUND:
Ensure the slave MCU’s RCPU does not perform data accesses over the UBUS when an external master changes the slaves MCU mode from slave to peripheral mode, and then accesses the slave MCU’s LBUS (i .e. CALRAM). Use interrupts or other notification mechanisms to prevent the slave MCU’s RCPU from writing/reading data over UBUS.If the slave MCU changes its own mode, ensure any subsequent load/store instruction over the UBUS is at least 6 instructions after the write to EMCR[PRPM], or that they are separated by an ISYNC instruction.

CDR_AR_929 Customer Erratum MIOS14.CDR3IMB3_03_0A

MIOS: Problem with DASM Duty Cycle Change to 0%

DESCRIPTION:
If the MIOS DASM is used for a OPWM function, a problem occurs if the duty cycle is changed to 0%. The module finishes the current cycle and then it generates one period with 100% duty cycle before it switches to 0% duty cycle.

WORKAROUND:
There are three cases required for the workaround (case 3 has two different solutions): 1.) When changing from a duty cycle >0% to a different duty cycle >0%, change the dataB register to the new value. 2.) When changing from a duty cycle >0% to a duty cycle of 0%, change the dataA register to equal the value currently in the dataB register. 3a.) When changing from a duty cycle of 0% to a duty cycle >0%, change the dataA register and dataB register by doing a 32-bit write that writes both registers to new values. OR 3b.) When changing from a duty cycle of 0% to a duty cycle >0%, write the dataA register to a value that will never match, then write the dataB register to its new value, then write the dataA register to its new value. An alternative implementation of 0% or 100% duty cycle can be achieved using the FORCE bits. For 0% duty cycle (100% if EDPOL=1), stop the associated MMCSM counter by writing 00 to the MMCSMSCR CLS bits, then write the MDASM FORCB bit to 1. For 100% duty cycle (0% if EDPOL=1), stop the associated MMCSM counter by writing 00 to the MMCSMSCR CLS bits, the write the MDASM FORCA bit to 1.

CDR_AR_1127 Customer Information MIOS14.CDR3IMB3_03_0A

MIOS: MDASMSCR polarity bit has no effect when open-drain mode selected

DESCRIPTION:
MDASMSCR[EDPOL] does not change the polarity of the MDA pin when MDASMSCR[WOR] = 1. This only applies to the MDASM output modes (OCB, OCAB and OPWM).

WORKAROUND:
Do not rely on MDASMSCR[EDPOL] to change the output polarity when open-drain mode is selected for an MDASM pin in output mode. Refer to the latest version of the Reference Manual (dated August 2003 or later).

CDR_AR_1131 Customer Information PADRING.565_CDR3_03_0

PADS: The weak internal pull device remains enabled on RSTCONF/TEXP

DESCRIPTION:
If PDMCR[SPRDS] is set by software after reset negates, the weak internal pull device on RSTCONF/TEXP remains enabled. This has no impact on applications designed according to the specification.

WORKAROUND:
There is no workaround required for applications designed according to the specification. If PDMCR[SPRDS] is set, and the pin is set to input mode, the existing external pull device or driver will overdrive the weak internal pull device (maximum = 130uA) on RSTCONF/TEXP.

CDR_AR_922 Customer Information PADRING.565_CDR3_03_0

PADS: VOH2.6 Spec changed to -1 mA

DESCRIPTION:
The IOH specification for all 2.6 volt outputs has been changed from -2 mA to -1 mA to insure a VOH2.6 of 2.3 volts. An additional specification has been added for VOH2.6A for a -2.0 mA load with a minimum VOH of 2.1 volts.

WORKAROUND:
2.6 volt outputs will only drive -1.0 mA with a VOH of 2.3 volts and will drive -2.0 mA with an output voltage of 2.1 volts minimum. Refer to manual dated after June 2001.

CDR_AR_932 Customer Information PADRING.565_CDR3_03_0

PADS: Leakage higher when 2.6V pads are pulled above 2.6V supply during reset

DESCRIPTION:
2.6V and 2.6V/5V pads will leak up to 10uA during reset. The leakage will occur to the QVDDL supply from each 2.6V pad which has a voltage greater than QVDDL. This additional leakage does not occur on 5V only pads.

WORKAROUND:
Cumulative input leakage of all pins pulled above the 2.6V supply must be utilized by loads on the 2.6V supply during low power modes while reset is applied. Refer to manual dated on or after May 2003.

CDR_AR_940 Customer Information PADRING.565_CDR3_03_0

JTAG: Do Not Switch All Pads Simultaneously When JTAG Enabled

DESCRIPTION:
JTAG mode puts all output pins in fast slew rate mode. The power supply pins of the device cannot supply enough current to allow all pins to be changed at the same time in fast slew rate mode. During normal operation, this is not an issue since all pins on the device do not switch at the same time.

WORKAROUND:
When using JTAG, all pins should not be switched simultaneously. Refer to manual dated on or after May 2003.

CDR_AR_1018 Customer Information PADRING.565_CDR3_03_0

Execute memory write prior to slave read for slave predischarge

DESCRIPTION:
When using multiple processors on a common bus with an external device that outputs voltages exceeding 3.1v, the predischarge cycle will not occur if the processor that initiated the read is different than the processor that initiated the write.

WORKAROUND:
Perform a write access to external memory to discharge the external bus, or read a value of 0x0 from the external device prior to accessing another MPC56x device on the same bus. Refer to manual dated on or after May 2003.

CDR_AR_1019 Customer Erratum RCPU.CDR3LBUSIBUS_17_0

RCPU: Don’t execute overflow type before update type MUL/DIV instruction

DESCRIPTION:
When an integer overflow type non multiply or divide instruction (designated by an ’o’ in the instruction mneumonic, such as addo) starts to execute before a previously started Condition Register 0 (CR0) update type integer multiply or divide instruction (designated by a ’.’ in the instruction mneumonic, such as divw.) completes, the CR0[SO] bit may be wrongly updated from the XER[SO] bit earlier changed by the overflow type instruction. For example, instruction sequence "divw. Rx,Ry,Rz , subfo Rt,Ru,Rv" may cause this problem. It does not happen if the overflow type instruction is also a CR0 update type instruction (designated by ’o.’ in the instruction mneumonic, such as addo.), or if register dependencies exist.

WORKAROUND:
Do any one of the following: 1) Keep a gap of at least 1 instruction between a CR0 update type integer multiply instruction and an overflow type instruction or a gap of 4 integer or 6 other instructions between a CR0 update integer divide instruction and an overflow type instruction; 2) Use the CR0 update type for both instructions; 3) Run the RCPU in serialized mode; 4) Place a "sync" instruction between the integer multiply/divide instruction and the overflow type instruction; 5) Don’t use the update form of integer multiply or divide instructions; or 6) Don’t use overflow type integer instructions. (Note: most compiler vendors do not generate the error case.)

CDR_AR_1138 Customer Erratum RCPU.CDR3LBUSIBUS_17_0

RCPU: Data breakpoint exception may occur even if conditions are not met

DESCRIPTION:
The RCPU may incorrectly take a second data breakpoint exception, if a data breakpoint occurs on a load/store instruction with a load following within five instructions in the RCPU program flow. This extra exception will only be taken if very specific internal bus timing occurs during the instruction sequence and the data breakpoint state remains set after the first data breakpoint exception is taken. In this condition, any load/store instruction executed with breakpoints enabled will cause the second data breakpoont exception.The additional exception sets SRR0 to the effective address of the instruction after the second load/store instruction, but the BAR register remains set to the effective address of the first load/store instruction that met the data breakpoint conditions.If the processor is in a non-recoverable state (MSR[RI] = 0) and breakpoints are not masked (LCTRL2[BRKNOMASK] = 1), the first load/store instruction within the data breakpoint exception handler (usually saving CPU context) will cause the second exception, handler re-entrance and loss of program tracking.

WORKAROUND:
1) Run RCPU in serialized mode. 2) Create conditions for an exception during the data breakpoint exception handler execution after saving SRR0/1 on the stack, for example, use ’SC’ instruction inside the handler, or a floating point instruction if the Floating Point Unit is disabled, or an unimplemented instruction. This exception will reset the internal data breakpoint state, eliminating the false data breakpoint exception.

CDR_AR_907 Customer Information **RCPU.CDR3LBUSIBUS_17_0**

RCPU: Issue ISYNC command when entering debug mode

DESCRIPTION:
If the ICTRL[29] bit is set (non-serialized mode) then the RCPU issues two instruction fetch requests into the instruction pipeline after entering debug mode. The debug port and the debug tool may get confused when processing an "mtspr DPDR,Rx" instruction. The debug tool loses synchronization with debug port and receives the wrong data for the "Rx" register. The typical case is when the debug tool tries to save scratch registers or read the debug mode cause.

WORKAROUND:
Issue an ISYNC instruction to the debug port prior to any other instructions when the RCPU enters debug mode after running code. Refer to manual dated on or after May 2003.

CDR_AR_440 Customer Information **RCPU.CDR3LBUSIBUS_17_0**

RCPU: Execute any IMUL/DIV instruction prior to entering low power modes.

DESCRIPTION:
There is a possibility of higher than desired currents during low power modes. This is caused by a possible contention in the IMUL/DIV control area. This contention may only exist prior to the execution of any IMUL/DIV instruction.

WORKAROUND:
Execute a MULLW instruction prior to entering into any low power mode (anytime after reset, and prior to entering the low power mode). Refer to manual dated on or after May 2003.

CDR_AR_211 Customer Information **RCPU.CDR3LBUSIBUS_17_0**

Do not set breakpoint on mtspr ICTRL instruction

DESCRIPTION:
When a breakpoint is set on an "mtspr ICTRL,Rx" instruction and ICTRL[IIFM] = 1, the result will be unpredictable. The breakpoint may or may not be taken on the instruction and value of the IIFM bit can be either 0 or 1.

WORKAROUND:
Do not put a break point on mtspr ICTRL, Rx instruction when ICTRL[IIFM] is set to 1. Refer to manual dated on or after May 2003.

CDR_AR_214 Customer Information RCPU.CDR3LBUSIBUS_17_0

Only negate interrupts while the MSR[EE] disables interrupts (MSR[EE]=0)

DESCRIPTION:
If the MSR[EE] bit is set and an external interrupt request to the RCPU is negated before the external interrupt vector is issued, the RCPU may become unpredictable until the device is reset. This interrupt event may be generated by software while managing peripheral modules in the MCU, or external devices connected to external interrupt request pins of the MCU or the MCU interrupt controller. This issue may occur when performing USIU operations like masking interrupt requests, clearing interrupt flags, masking or changing interrupt logic in the interrupt controller, or switching on/off enhanced interrupt control if available.

WORKAROUND:
Do not clear an interrupt that is not being serviced by software while MSR[EE]=1. Software should disable interrupts (MSR[EE]=0) in the RCPU before clearing or masking any interrupt source from the USIU, IMB or external pin. For external interrupt request pins, it is recommended that edge triggered interrupts be used. No delay time is required before re-enabling interrupts (MSR[EE]=1). Refer to manual dated on or after May 2003.

CDR_AR_949 Customer Erratum QADC64E.CDR3IMB3_03_0B

QADC64E: Write CCW[EOQ] to 0x3F for the End of Queue

DESCRIPTION:
Using 0x7F as an EOQ (end of queue) causes a conversion of VRL to occur when the EOQ is reached. In single or continuous scan modes, this conversion is underway when the queue wraps back to the first word, and the first conversion is not performed. The result for the conversion of VRL gets written in the result space for the first conversion word. The queue and conversions then proceed on correctly.

WORKAROUND:
Always use 0x3F instead of 0x7F as an EOQ in the CCW for both Legacy and Enhanced modes of QADC64E operation. Refer to an updated Reference Manual dated after January 2003.

CDR_AR_915 Customer Erratum QADC64E.CDR3IMB3_03_0B

QADC64E: Conversion Clock cannot be shared between Master/Slave Modules

DESCRIPTION:
In a multiple QADC64E module configuration it is not possible to operate the modules on synchronous conversion clocks. The conversion clock of a module configured as Master cannot be input to the Slave .

WORKAROUND:
If simultaneous conversions are required, the customer can trigger both QADC64E modules SIMULTANEOUSLY using the external trigger inputs (ETRIG1 or 2), however, the conversions will not be performed SYNCHRONOUSLY. There is no workaround to allow synchronous QADC64E module operation. Do not set EXTCLK of the QADCMCR register to use the conversion clock of a master QADC (don’t set to 1). References to this feature are removed in updated Reference Manuals dated after January 2003.

CDR_AR_1125 Customer Erratum QADC64E.CDR3IMB3_03_0B

QADC64: Don’t change both BQ2 and MQ2 while Q2 is running

DESCRIPTION:
There exists a window of 2 system clocks in the conversion cycle during which a change to the Queue2 trigger mode (QACR2[MQ2]) along with a change to the Queue2 start location (QACR2[BQ2]) while Queue2 is active will cause the new value for BQ2 to be ignored. The new trigger mode takes place and conversions continue to be stored in Q2 as defined by the previous BQ2. Hence the locations following the new BQ2 will not contain results.

WORKAROUND:
Before changing the Queue2 mode, disable Q2 (MQ2=0b0000), then update MQ2 and BQ2.

CDR_AR_420 Customer Information QADC64E.CDR3IMB3_03_0B

QADC64: Don’t change BQ2 with a set of SSE2 without a mode change.

DESCRIPTION:
Changing BQ2 and setting SSE2 with no mode change will cause Q2 to begin but not recognize the change in BQ2. Further, changes of BQ2 after SSE2 is set, but before Q2 is triggered are also not recognized. All other sequences involving a change in BQ2 are recognized.

WORKAROUND:
Be sure to do mode change when changing BQ2 and setting SSE2. Recommend setting BQ2 first then setting SSE2. Refer to manual dated on or after May 2003.

CDR_AR_1048 Customer Information QADC64E.CDR3IMB3_03_0B

QADC64E: Sample Time is 8 QCLKs instead of 16 when CCW[IST]=1 in Enhanced Mode

DESCRIPTION:
If the QADC64E is in enhanced mode, the documentation says that the Input sample time is 16 QCLKs when CCW[IST]=1. Actually the Input sample time is 8 QCLKs. On the MPC561-564, enhanced mode is enabled by setting QADCMCR[FLIP]=1. The MPC565 is always in enhanced mode.

WORKAROUND:
Always expect the Input Sample time to be 8 QCLKs when CCW[IST]=1 when operating in enhanced mode. Refer to manual dated after January 2003.

CDR_AR_1151 Customer Erratum QSMCM.CDR3IMB3_04_0

SCI: TXD pin reverts to output immediately when SCCxR1[TE] is cleared

DESCRIPTION:
When the Transmiter Enable bit of the SCI Control Register 1 is cleared (SCCxR1[TE]=0), the Transmit Data pin, TXD, reverts immediately to general purpose output mode, and the pin will be driven high or low as determined by the PortQS Data Register, PORTQS. If the transmitter is not idle when SCCxR1[TE] is cleared, any data still being output on the TXD pin will be lost.

WORKAROUND:
Ensure SCCxR1[TE] is only cleared after the Transmit Complete bit of the SCI status Register is set (SCxSR[TC]=1).

CDR_AR_1118 **Customer Erratum** **READI.CDR3LBUSUBUS_03_0**

READI: Program Flow Tracking Error Under Rare Condition

DESCRIPTION:
Under certain conditions, the program trace information output by the READI module may not accurately reflect the actual program flow. This condition requires ALL of the following conditions: 1) Either the BTB or code compression is enabled. 2) A double branch instruction sequence must occur where: the first branch is indirect and its condition is already determined or is non-conditional, the second branch is conditional and is miss-predicted and then corrected due to a long (execution time) instruction. And 3) the BBC must be held off the U-bus so that U-bus show cycle addresses are delayed.

WORKAROUND:
Either disable code compression and the BTB, or accept erroneous trace reconstruction under this rare condition.

CDR_AR_846 **Customer Information** **READI.CDR3LBUSUBUS_03_0**

READI: Synchronize the MCKI input clock to the MCKO output clock.

DESCRIPTION:
The READI module may not properly receive input messages if the input clock is not synchronous with the output clock.

WORKAROUND:
Synchronize the MCKI input clock to the MCKO output clock. Refer to manual dated on or after May 2003.

CDR_AR_698 **Customer Information** **READI.CDR3LBUSUBUS_03_0**

READI Input message requires 2 MCKI idle after READI Enabled.

DESCRIPTION:
If an input message is sent to the READI immediately after deassertion of RSTI_B (enabling READI) the READI may not recognize the start of the message and will ignore it. This behavior could cause the tool to get out of sync with the READI.

WORKAROUND:
Do not send an input message until at least 2 MCKI after READI is enabled, or better, until the DID message is received from the READI. Refer to manual dated on or after May 2003.

CDR_AR_1096 Customer Information READI.CDR3LBUSUBUS_03_0

READI: New features to enhance Nexus debug

DESCRIPTION:
New features have been added to the READI to provide enhanced Nexus debug capabilities. (1) The Nexus message queue has been increased from 16 locations to 32 locations. (2) A new register READI_MC has been added with bits to control the following: Program Trace Sync Mode (PTSM) which optionally adds an I-CNT packet to message, Queue Flush Mode (QFM) that allows the user to control the message buffer action when a message overflow occurs, and Program Trace Mode (PTM) that allows Nexus trace with an ICTRL[ISCT_SER]=0x6. (3) Nexus Public Messages have been added or modified. (4) The manufacturer ID in the READI_DID has been corrected to 0xE. (5) A change was made to the data trace message to allow tools to differentiate between 8-bit data reads and 16 bit data reads.

WORKAROUND:
Tools must be updated to allow use of the new features. Consult updated Reference Manuals dated after November 2002.

CDR_AR_1144 Customer Erratum TOUCAN.CDR3IMB3_05_1A

TouCAN: Transmit buffers may freeze or indicate missing frame

DESCRIPTION:
If a received frame is serviced during reception of a second frame identified for the same MB (message buffer) and a new Tx frame is also initiated during this time, the Tx MB can become frozen and will not transmit while the bus is idle. The MB remains frozen until a new frame appears on the bus. If the new frame is a received frame, the frozen MB is released and will arbitrate for external transmission. If the new frame is a transmitted frame from another Tx MB, the frozen MB changes its C/S (control status word) and IFLAG to indicate that transmission has occurred, although no frame was actually transmitted. The frozen MB occurs if lock, unlock and initiate Tx events all occur at specific times during reception of two frames. The timing of the lock event affects the timing window of the unlock event as follows: Situation A) Rx MB is locked during the second frame. A frozen Tx MB occurs if: 1) Both of these events occur in either a-then-b or b-then-a order: a) A new transmission is initiated by writing its C/S between CRC6 (sixth bit of CRC field) and EOF7 (seventh bit of end of frame) of the second frame. b) The Rx MB is locked by reading its C/S after EOF6 of first frame and before EOF6 of second frame. 2) The Rx MB is unlocked between EOF7 and intermission at end of the second frame. Notice in this situation that if the lock/unlock combination happens close together, the lock must have been just before EOF6 of the second frame, and therefore the system is very close to having an overrun condition due to delayed handling of received frames. Situation B) Rx MB was locked before EOF6 of the first frame; in other words, before its IFLAG is set. This is a less likely situation but provides a larger window for the unlock event. A frozen Tx MB occurs if: 1) The Rx MB is locked by reading its C/S word before EOF6 of the first frame. 2) Both of these events occur in either a-then-b or b-then-a order: a) A new transmission is initiated by writing its C/S word sometime between CRC6 and EOF7 of the second frame. b) The Rx MB is unlocked between CRC6 and intermission at end of the second frame. Notice in this situation that if the unlock event occurs after EOF6, the first frame would be lost and the second frame would be moved to the Rx MB due to the delayed handling of received frames. Situation C) Rx unlocked during bus idle. A frozen/missing Tx occurs if: 1) An Rx MB is locked before EOF6 of an incoming frame with matching ID and remains locked at least until intermission. This situation would usually occur only if the received frame was serviced after reception of a second frame. 2) An internal arbitration period is triggered by writing a C/S field of an MB. 3) The locked Rx MB is unlocked within two internal arbitration periods (defined below) before or after step 2). 4) 0xC is written to the C/S of a Tx MB within these same two arbitration periods. This step is optional if 0xC was writin in step 2) above. Two internal arbitration periods are calculated as ((2 * number of MBs) + 16) IMB clocks. Additional Notes: 1) The received frames can be transmitted from the same node, but they must be received into an Rx MB. 2) When the frozen Tx MB’s IFLAG becomes set, an interrupt will occur if enabled. 3) The timestamp of the missing Tx will be set to the same timestamp value as the last reception before it was frozen. 4) If the user software locks the Rx MB before a frame is received, situation A can occur with a single received frame. 5) The issue does not occur if there were any additional pending Tx MBs before CRC6. 6) If multiple Tx MBs are initiated within the CRC6/EOF7 window (situation A and B) or two internal arbitration windows (situation C), they all become frozen.

WORKAROUND:
If received frames can be handled (lock/unlocked) before EOF6 of the next frame, situations A and C are avoided. If they are handled before CRC6, or lock times are below 23 CAN bit times, situation B is avoided. If these conditions cannot be guaranteed, situation A and B are avoided by inserting a delay of at least 28 CAN

bit times between initiating a tranmission and unlocking an Rx MB, and vice versa. Typically a system would use a mechanism to selectively add the necessary delay. For example, software might use a global variable to record an external timer value (the TouCAN timer can't be used as that would unlock) when initiating a new Tx or unlocking an Rx, and then add the required delay before performing second action. Situation C can be avoided by inserting a delay of at least two internal arbitration periods between writing 0xC and unlocking the locked Rx MB.

CDR_AR_1045 Customer Information TOUCAN.CDR3IMB3_05_1A

CAN: Bus Off recovery not ISO compliant

DESCRIPTION:
The Bus Off recovery is not ISO compliant on the FlexCAN and TouCAN modules. The ISO specification indicates that the CAN node should remain inactive until user intervention restarts it. The FlexCAN and TouCAN modules both include an automatic recovery mechanism for the Bus Off condition.

WORKAROUND:
The Bus Off condition interrupt should be enabled and an interrupt service routine implemented to disable the CAN. The user's software should then determine when the CAN should be re-activated.

CDR_AR_1142 Customer Information TOUCAN.CDR3IMB3_05_1A

TouCAN: Writing to an active receive MB may corrupt MB contents

DESCRIPTION:
Deactivating a TouCAN receive message buffer (MB) may cause corruption of another active receive MB, including the ID field, if the following sequence occurs. 1) A receive MB is locked via reading the Control/Status word, and has a pending message in the temporary receive serial message buffer (SMB). 2) A second frame is received that matches a second receive MB, and is queued in the second SMB. 3) The first MB is unlocked during the time between the CRC field and the 6th bit of end of frame (EOF) of the second frame. 4) The second MB is deactivated within 20 IMB clock cycles of the 6th bit of EOF, resulting in corruption of the first MB.

WORKAROUND:
Do not write to the Control/Status word after initializing a receive MB. If a write (deactivation) is required to the Control/Status field of an active receive MB, either FREEZE the TouCAN module or insert a delay of at least 27 CAN bit times plus 21 IMB clock cycles between unlocking one MB and deactivating another MB. This will avoid MB corruption, however frames may still be lost.

CDR_AR_627 Customer Information TPU3.CDR3IMB3_03_0A

TPU: (Microcode) Add neg_mrl with write_mer and end_of_phase

DESCRIPTION:
Incorrect generation of 50% duty cycle is caused by the command combination "write_mer, end". If the write_mer is the last instruction together with the end, this may create an additional match using the old contents of the match register (which are in the past now and therefore handled as an immediate match)

WORKAROUND:
Add neg_mrl together with the last write_mer and with end-of-phase. The negation of the flag overrides the false match which is enabled by write_mer and postpones the match effect by one micro-instruction. In the following micro-instruction the NEW MER value is already compared to the selected TCR and no false match is generated. The neg_mrl command has priority over the match event recognition, separating the write_mer and the end command. This gives enough time for the new MER to update before the channel transition re-enables match events.

CDR_AR_910 Customer Erratum USIU.CDR3UBUS_12_0

USIU: PITRTC Clock may not work when SCCR[RTDIV] is 0

DESCRIPTION:
The RCPU RTC/PIT may not count in all operating conditions if the ratio of System clock to the PITRTC Clock is less than or equal to 4. This may happen if the SCCR[RTDIV] is set to 0 and either 1) the part is running on the limp clock, or 2) the PLPRCR[MF] = 0 and both the System PLL and the PITRTC Clock use the same clock source (EXTCLK or the crystal oscillator).

WORKAROUND:
Keep the System Clock to PITRTC clock frequency ratio greater than 4. This can be done the easiest by setting the SCCR[RTDIV] to a value of 1 (reset value).

CDR_AR_1134 Customer Erratum USIU.CDR3UBUS_12_0

USIU: RTC, DEC, TB and PIT counters may not count after PORESET or HRESET

DESCRIPTION:
The Real-Time Clock (RTC), Timebase (TB), Decrementer (DEC) and Periodic Interrupt Timer (PIT) may not count during the time between PORESET or HRESET negation and the time at which the PLL is programmed by application software and becomes locked to the target frequency.

WORKAROUND:
Always program the PLL to the target operating frequency (by changing the PLPRCR[MF] or PLPRCR[DIVF] bits) before referencing the TB, RTC, DEC, or PIT in an application after a PORESET or a HRESET.

CDR_AR_1158 Customer Erratum USIU.CDR3UBUS_12_0

USIU: Stop Time Base to write new value

DESCRIPTION:
The RCPU Time Base registers may become corrupted if a new value is written (with a mttbl or mttbu instruction) to the Time Base Upper (TBU) or Time Base Lower (TBL) registers while the Time Base clock is enabled in the Time Base Control and Status Register (TBSCR[TBE]=1).

WORKAROUND:
Disable the Time Base clock by clearing the Time Base Enable bit in the TBLSCR (TBSCR[TBE]=0) prior to any write to the TBU or TBL registers.

CDR_AR_287 Customer Erratum USIU.CDR3UBUS_12_0

USIU: System to Time Base frequency ratio must be greater than 4

DESCRIPTION:
The Time Base and Decrementer may not count properly if the ratio of the System clock to Time Base Clock is 4 or less.

WORKAROUND:
Keep the ratio of the System Clock to the Time Base clock above 4. Always set SCCR[TBS] = 1 when running on the limp clock. Refer to manual dated on or after May 2003.

CDR_AR_479 Customer Erratum USIU.CDR3UBUS_12_0

USIU: The MEMC does not support external master burst cycles

DESCRIPTION:
The MTS function of the Memory Controller (MEMC) will not work properly to control external devices when an external master initiates a burst.

WORKAROUND:
Use external logic to control devices which can have burst accesses from multiple masters. Refer to manual dated on or after May 2003.

CDR_AR_1135 Customer Erratum USIU.CDR3UBUS_12_0

USIU: Disable USIU burst in debug mode if READI R/W feature is used

DESCRIPTION:
If the RCPU is in debug mode and USIU burst mode is enabled (SIUMCR[BURST_EN]=1), READI R/W accesses may cause the RCPU to stop fetching instructions. The device must be reset before the RCPU will fetch and execute instructions.

WORKAROUND:
Use a BDM debugger or when using a Nexus debugger, disable the USIU burst when debugging (SIUMCR[BURST_EN]=0). Alternately, debuggers could disable the USIU burst when entering debug mode (and re-enable upon exiting debug mode) before using READI R/W accesses (i.e. with BDM messages), or the debugger could use BDM messages to perform all read/write accesses instead of using READI R/W accesses.

CDR_AR_1152 Customer Erratum USIU.CDR3UBUS_12_0

USIU: PORESET must always be asserted before the 2.6V supplies reach 0.5V

DESCRIPTION:
When exiting low power modes where the 2.6V supplies (VDD, QVDDL, NVDDL and VDDSYN) are off (Power-down and SRAM Standby modes), correct operation cannot be guaranteed if the 2.6V supplies are above 0.5V before PORESET is asserted. For example, the CALRAM or flash contents may be corrupted.

WORKAROUND:
Ensure PORESET is asserted before ramping the 2.6V supplies above 0.5V in any power-up sequence.

CDR_AR_867 Customer Erratum USIU.CDR3UBUS_12_0

USIU: Do not operate USIU burst on a burst-inhibited memory region

DESCRIPTION:
If the SIUMCR[BURST_EN] is set and a burst-inhibited code region (memory) is accessed (either by asserting bi_b pin or setting the BI bit in the corresponding ORx memory controller register) then the CPU will not execute code properly.

WORKAROUND:
Do not set the SIUMCR[BURST_EN] while trying to run code from an external non-burstable memory.

CDR_AR_1154 Customer Erratum USIU.CDR3UBUS_12_0

SIU: RTSEC register not documented; May affect the initial increment of the RTC

DESCRIPTION:
The Reference Manuals have an incomplete statement in the description of the Real-Time Clock register (RTC). In addition, the reserved Real-Time Clock Predivider Register (RTSEC) is not documented and may affect the initial increment of the RTC (seconds) counter. In the Reference Manual, the statement "A write to the RTC resets the seconds timer to zero." is incorrectly worded. A better statement that fully describes the this action would be: "A write of 0 to the RTC must be performed to reset the RTC (seconds) timer to zero." The RTSEC register is the predivider to the RTC (seconds) timer. The RTC, the RTSEC, and the Real Time Clock Alarm (RTCAL) registers, as well as the Real-Time Clock Enable [RTE] and the Real-Time Clock Source [4M] bits of the Real Time Clock Control and Status Register (RTCSC), are not affected by any reset (unchanged) and power up in a random state. This will cause the initial increment of the RTC to be between one system clock and 26143 PITRTCLK clocks. All of these bits and registers must be initialized the first time they are used or if known start points are required. RTSEC is implemented as an 18-bit counter that is left justified in a 32-bit word at address 0x2F_C228. The RTC Alarm itself is always disabled by reset, but RTCAL should be initialized to the desired alarm time, if required, before the Alarm Interrupt Enable (ALE) in the RTCSC is enabled (RTCSC[ALE]=0b1).

WORKAROUND:
To properly initialize the RTC timer to a completely known state with the most accurate startup, the following sequence must be used. 1) The Real-Time Clock Enable [RTE] and the Real-Time Clock Source [4M] bits must be configured in the Real-Time Clock Control and Status Register (RTCSC) after any true power on reset (if KAPWR is powered up) prior use of the RTC. The bits must be initialized since they are not affected by any reset and can be in a random state after the power up. For the most accuracy in the start value of the RTC, RTE should be cleared during this step. For the most accuracy in the start value of the RTC, RTE should be cleared during this step. 2) In order to guarantee that the first increment of the RTC register occurs in approximately 1 second (depending on whether a 4 MHz or 20 MHz crystal is being used), the reserved register RTSEC must also be initialized by writing either 0x0F42_4000 (if using a 4 MHz crystal) or 0x4C4B_4000 (if using a 20 MHz crystal). Alternately, RTSEC could be written to 0 and RTSEC will be updated automatically to these values, but will then immediately (within one PITRTCLK clock) increment the RTC when the RTC is enabled. 3) If a known starting point is desired (like 0), a value must be written to the Real-Time Clock register (RTC). 4) RTE bit should be then be set in the RTCSC register to enable RTC operationNote that the RTCSC, the RTC, and the RTSEC registers are locked following all resets and must be unlocked. The RTSEC can be unlocked by writing 0x55CC_AA33 to address 0x2F_C328.

CDR_AR_1113 **Customer Information** **USIU.CDR3UBUS_12_0**

USIU: Ensure HRESET/SRESET negation time is longer than 3 CLKOUT periods

DESCRIPTION:
If either HRESET or SRESET are externally re-asserted after a negation time of less than 3 CLKOUT clocks, and after an initial assertion of more than 512 CLKOUT periods, the MCU will remain in that reset until PORESET is applied. In the case of SRESET being the cause, then HRESET can also clear the locked condition. In the case of HRESET being the cause then SRESET will be held asserted internally by the MCU. The SWT (Software Watchdog Timer) will not clear the locked condition.

WORKAROUND:
Do not re-assert HRESET/SRESET within 3 CLKOUT periods of the previous HRESET/SRESET negation; Or apply PORESET.

CDR_AR_869 **Customer Information** **USIU.CDR3UBUS_12_0**

USIU: Do not enable BRx[SST] with SCCR[EBDF]>0

DESCRIPTION:
When EBDF>0 an external burst access with short setup timing will corrupt any USIU register load/store

WORKAROUND:
Do not enable BRx[SST] while EBDF>0. Refer to manual dated on or after MAy 2003.

CDR_AR_895 **Customer Information** **USIU.CDR3UBUS_12_0**

USIU: Do not assert TEA pin on fetch while SIUMCR[BURST_EN] bit set.

DESCRIPTION:
If USIU burst is enabled (SIUMCR[BURST_EN]) a transfer error indication (internal by Bus Monitor, Chip Select Region access attribute mismatch, or external by TEA signal) may cause the chip to stop executing until reset. In some cases the address read from the SRR0 register in Machine Check Exception handler may not precisely indicate the exact faulty bus address causing the exception. The RCPU will pre-fetch code from the memory address after the last address of the Chip Select region. When the pre-fetch causes a Transfer Error Acknowledge, either by the Bus Monitor or by a Supervisor/User attribute mismatch in the Memory Controller Region, the RCPU will hang up even if the pre-fetch was unnecessary for code execution.

WORKAROUND:
1. Do not assert TEA on the external bus for instruction fetch while the SIUMCR[BURST_EN] bit is set. 2. Do not place code at the 8 last words of a memory controller region while the SIUMCR[BURST_EN] bit is set. Refer to manual dated on or after May 2003.

CDR_AR_965 Customer Information USIU.CDR3UBUS_12_0

USIU: Program All Chip Selects with the same Burst Length

DESCRIPTION:
When the enhanced burst mode is activated in the USIU by setting SIUMCR[BURST_EN] bit and the burst length bits in the BRx[BL] of the active chip selects are programmed differently, the USIU may provide the wrong instruction to the RCPU core.

WORKAROUND:
Before programming the Enhanced Burst feature (SIUMCR[BURST_EN] = 1), make sure that all of the BRx[BL] bits of active chip selects are programmed to the same value; and do not load instructions from an external memory region that is not covered by any chip selects if any of the BRx[BL] bits is set to 1.

CDR_AR_1153 Customer Information USIU.CDR3UBUS_12_0

USIU: Sleep and Deep-Sleep modes require power to all 2.6V supplies

DESCRIPTION:
The reference manual table 8-5 Power Mode Descriptions has incorrect votlage requirements for Sleep and Deep-Sleep modes. Sleep and Deep-Sleep modes require that VDD, QVDDL, NVDDL and VDDSYN all remain powered-up.

WORKAROUND:
Maintain power to all 2.6V supplies during Sleep or Deep-Sleep low power mode.

CDR_AR_389 Customer Information USIU.CDR3UBUS_12_0

Little Endian modes are not supported

DESCRIPTION:
The little Endian modes are not functional.

WORKAROUND:
Do not activate little endian modes. The reference manual will be updated to remove all little endian mode references.

CDR_AR_1109 Customer Information USIU.CDR3UBUS_12_0

USIU: Do not write zero value to the SYPCR[BMT]

DESCRIPTION:
If the BMT (Bus Monitor Timing) field of the SYPCR register is written as zero, the external bus activity may not be available after SRESET assertion even if the bus monitor is disabled by BME bit. The MCU will assert TEA which will terminate any external bus cycle with a data error.

WORKAROUND:
Always write a non-zero value to the BMT field of the SYPCR register.

CDR_AR_1120 Customer Information USIU.CDR3UBUS_12_0

USIU: Interrupt Controller may generate vector 0x0 or has no request indication

DESCRIPTION:
When software masks interrupt requests, clears interrupt flags, stops or disables a module, or masks or changes interrupt logic in the UIMB or the USIU interrupt controller while MSR[EE] = 1, the interrupt request may disappear during or after the RCPU has acknowledged the external interrupt exception. It may also occur after re-enabling interrupts in the RCPU. This may cause the following: 1. When external interrupt relocation is enabled, the BBC may issue a vector offset of 0x0. 2. The SIPEND registers will not contain set bits, and if the service routine polls for a set bit it may hang. 3. The SIVEC register may contain a value of zero which could cause software to branch to an unmapped location.

WORKAROUND:
Follow the workarounds in AR_214, however, note that a time delay is required prior to re-enabling interrupts. Before clearing an interrupt related register, ensure that MSR[EE] = 0. Expect a vector offset of 0x0 if an interrupt is cleared or disabled while MSR[EE] = 1. This vector should be handled as if no interrupt has occurred, i.e. perform an RFI. After clearing an interrupt source, sufficient time must occur before re-enabling interrupts in the RCPU. This time should take longer than the time needed for a load of the same register that was just cleared. If unsure, include this load instruction before the instruction that re-enables interrupts in the RCPU. Refer to manual dated on or after May 2003.

CDR_AR_1137 Customer Information USIU.CDR3UBUS_12_0

USIU: RSR[LLRS] can be set even though no loss of lock reset has occurred

DESCRIPTION:
If the Loss of Lock Reset Enable bit in the PLPRCR register is set when the PLL Multiplication or Division Factor value is changed (PLPRCR[MF] or PLPRCR[DIVF]), the Loss of Lock Reset Status bit in the RSR register will be set (RSR[LLRS] = 1), even though a reset does not occur.

WORKAROUND:
Enable PLPRCR[LOLRE] after setting PLPRCR[MF] and PLPRCR[DIVF] values, or if PLPRCR[LOLRE] is already enabled, clear RSR[LLRS] after changing the value of PLPRCR[MF] or PLPRCR[DIVF].

CDR_AR_1155 Customer Information USIU.CDR3UBUS_12_0

SIU: TEA for external access must be negated within 1 system bus clock

DESCRIPTION:
When accessing external memory and the SIU bus monitor terminates the cycle with a Transfer Error Acknowledge (TEA), the SIU may produce unexpected results on subsequent accesses to the SIU address space, including SIU internal registers reads. This condition occurs when the TEA signal (pin) is not negated within 1 system clock of the time that the MCU stops asserting the TEA signal. While TEA is asserted by the MCU, it must be negated by the required external pull-up resistor. While the TEA negation requirement (1 clock) is documented in the Reference Manual, it may not be obvious that internally terminated accesses of an external memory space require the use of the external pull-up resistor. The value of the resistor should be small enough to pull the TEA line up to VIH level faster than one system clock and depends on the TEA line/board wire capacitance. Circuitry inside the MCU generates an actively driven TEA for accesses to internal non-existent memory spaces and does not rely on the external pull-up resistor to negate the cycle.

WORKAROUND:
Insure that the external pull-up resistor on the TEA pin is sufficient to negate TEA within one system clock. A value of 1K is recommended.