FREESCALE SEMICONDUCTOR, MICROCONTROLLER DIVISION
CUSTOMER ERRATA AND INFORMATION SHEET
Part: MPC555.M
General Business Use
Report Generated: Wed Oct 27, 2010, 13:43:26

MPC555.M

ERRATA AND INFORMATION SUMMARY

AR_686 MPC556: Operate code compression at 32Mhz
AR_1082 TPU ROM: Channels with the COMM ROM function affect other channels
AR_678 Additional current on KAPWR
AR_697 Revised operating currents
AR_381 New Features on MPC555 mask revision J76N and later
AR_412 Avoid instruction fetches from IMB/UIMB memory map
AR_593 MASKNUM field in USIU is 0x40
AR_597 AC timing changes
AR_750 Pad hardware and software changes on revision K62N
AR_851 Documentation: COLIE is bit 10 of COLIR Register
AR_1070 MPC555: Part Marking is MPC555LF8MZP40 or MPC555LF8CZP40
AR_600 Updated Flash Programming Algorithm and Control Registers
AR_485 Disable of TPU emulation mode while MISC enabled corrupts data in RAM
AR_1143 L2U: Care required when changing a slave MCU’s mode in multi-master systems
AR_810 MIOS: Synchronize writes to DASM B channel in OPWM mode.
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_953 Pads: TEA_b requires an external pull up
AR_940 JTAG: Do Not Switch All Pads Simultaneously When JTAG Enabled
AR_982 PADS: QADC64 Port A (and MUX Out) have CMOS Digital Outputs
AR_1019 RCPU: Don’t execute overflow type before update type MUL/DIV instruction
AR_1077 RCPU: Do not run multi-master compressed application with Show Cycles and BTB
AR_1138 RCPU: Data breakpoint exception may occur even if conditions are not met
AR_1076 RCPU: Treat VF queue flush information value of 6 as 2
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_563 QSM/QSMCM/QADC64 corrupts data after an IACK cycle in CISC parts.
AR_754 QADC64:Do not use queue1 in external gated mode with queue2 in continuous mode.
AR_768 QADC64: Queue2 activity may reset Queue1 ExtGates Single Scan SSE
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_563 QSM/QSMCM/QADC64 corrupts data after an IACK cycle in CISC parts.
AR_1151 SCI: TXD pin reverts to output immediately when SCCxR1[TE] is cleared
AR_584 QSMCM: Do not use link baud and ECK modes
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_577 TPU3 - TCR2PSCK2 bit does not give TCR2 divide ratios specified.
AR_896 UIMB: Avoid external code in addresses 0xZ[3,7,B,F]0_7F80 to 0xZ[3,7,B,F]0_7FFC
AR_985 USIU: Do not use ORx[EHTR] with Dual Mapping
AR_925 USIU: TEXP feature does not function when VDD supply is off
AR_909 USIU: Do not assert cr_b to abort pending store reservation access
AR_910 USIU: PITRTC Clock may not work when SCCR[RTDIV] is 0
AR_984 USIU: Setting of SCCR[EBDF] may slow execution of code
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_1152 USIU: PORESET must always be asserted before the 2.6V supplies reach 0.5V

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_389 Little Endian modes are not supported
AR_442 Avoid loss of clock during HRESET
AR_594 USIU: Changing PLL MF to 1:1 mode can have 180 degree phase shift
AR_598 USIU: Ensure proper configuration for proper startup
AR_1109 USIU: Do not write zero value to the SYPCR[BMT]
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_686 Customer Erratum MPC555.M

MPC556: Operate code compression at 32Mhz

DESCRIPTION:
Code compression logic has speed paths which prevent the part from running at full frequency over the full voltage range while code compression is enabled. Operation while code compression is disabled is not affected. Parts will be tested to ensure that code compression operates at 32MHz over the full temperature and voltage specification.

WORKAROUND:
Operate the part at 32MHz while code compression is enabled.

CDR_AR_1082 Customer Erratum MPC555.M

TPU ROM: Channels with the COMM ROM function affect other channels

DESCRIPTION:
The TPU COMM ROM Function causes problems in other channels. When the Host Service request is set to 0b11, all channels that do not use the COMM function will be forced to outputs and a random state will be selected.

WORKAROUND:
Either: 1) Re-initialize all other channels after the COMM function has been initialized; or 2) If a fixed COMM TPU function is required, download an updated TPU ROM image into the DPTRAM and use the TPU in emulation mode.

CDR_AR_678 Customer Information MPC555.M

Additional current on KAPWR

DESCRIPTION:
KAPWR current exceeds the initial design targets. During operation, KAPWR may be 8ma. Currents during power-down modes have not been fully characterized, and should be assumed to be the same value.

WORKAROUND:
Design KAPWR supply to handle the additional current. Characterize the current consumption in the final application board.

CDR_AR_697 Customer Information MPC555.M

Revised operating currents

DESCRIPTION:
Characterization of silicon indicates that the operating current specifications must be updated. The total current is not anticipated to change significantly, but will be redistributed amongst VDDL, VDDI, KAPWR, VDDSRAM, VDDSYN, and VDDF.

WORKAROUND:
Refer to electrical specification 3.3 or later for revised values.

CDR_AR_381 Customer Information MPC555.M

New Features on MPC555 mask revision J76N and later

DESCRIPTION:
Several new features were added to the MPC555 starting with mask revision 00J76N and also included in revisions K02A. In the USIU, the DBCT and DBSLR clock control bits were added ("Disable backup clock for timers", "Disable clock switch in loss of lock and reset"). In the USIU, a mode was added to allow the WE pins to also assert on reads, allowing the usage of some SRAMS. An additional "MTS" function has been multiplexed onto the IRQ2/CR/SGPIO2 pin. The MTS pin allows for sharing of additional types of devices in a multi-master system. In addition, the CMF FLASH programming control has changed. The recommended connection of the VSSSYN pin has changed. The recommended connection of the crystal has changed (resistor is now internal).

WORKAROUND:
Consult a revised users manual (15 September 1998 or later) to determine how to use these features. Use the latest version of the FLASH programming tools (version 1.1 or later of CMF_DEMO routines).

CDR_AR_412 Customer Information MPC555.M

Avoid instruction fetches from IMB/UIMB memory map

DESCRIPTION:
Instruction fetches on the IMB or to UIMB control registers may result in improper operation, possibly requiring reset to continue.

WORKAROUND:
Avoid instruction fetches from the IMB/UIMB memory map. Program the IMPU to disable instruction accesses to the IMB/UIMB memory map.

CDR_AR_593 Customer Information MPC555.M

MASKNUM field in USIU is 0x40

DESCRIPTION:
MASKNUM field in USIU has been changed to 0x40, and will change on future revisions.

WORKAROUND:
Modify software to expect new value (0x40) for the MASKNUM field.

CDR_AR_597 Customer Information MPC555.M

AC timing changes

DESCRIPTION:
Some of the AC timing specifications have changed. Refer to electrical specification 3.3 or later for new values. See CDR_AR_524 for AC timing specification 3.0. In addition, the following electrical specifications have changed to the following new values: {sp7, sp7a, sp7b, sp7c, sp7d} 4ns, {sp8a, sp8c, sp8d} 14ns, sp8b 15ns, sp10 14ns, sp11 14ns, sp15 12ns, sp15b 8ns, sp22 9ns, sp28 9ns, sp41 18ns. D(0:31) has been moved from sp7 and sp8 to sp7d and sp8d.

WORKAROUND:
Ensure external devices are matched to these updated electrical specifications.

CDR_AR_750 Customer Information MPC555.M

Pad hardware and software changes on revision K62N

DESCRIPTION:
The user should be aware that fields in the USIU PDMCR have changed. Bit 8 was previously a reserved bit. It now holds the control for the pull devices on the t2clk pins (previously, they were always enabled). Other bits in the PDMCR are still reserved for additional control on future devices.The default value for the ENGCLK was previously defined as system clock divided by 2 (20Mhz if system frequency is 40mhz). The ENGCLK default is now set to the system clock divided by 64. The default is 625KHz if the system frequency is at 40Mhz.The ENGCLK pad driver is now sized to drive loads of 25pf or 50pf, selectable by software. The CLKOUT pad driver is now sized to drive loads of 30pf or 90pf, selectable by software.

WORKAROUND:
PDMCR bit 8 of should be written to a logic "0" if the user wishes for the t2clk pull devices to behave as on previous versions of the chip. To have best software compatibility with future devices, PDMCR[9:13] should be programmed to the same value as PRDS (PDMCR[6]). PDMCR[16:17] should be programmed to the same value as SPRDS (PDMCR[7]). The future function of PDMCR[14:15] has not been determined, and should be programmed to 0. For this revision, software should ignore the read values of PDMCR[9:15].If greater frequencies are required on ENGCLK, the user must write to SCCR in the USIU (see ENGDIV bits).The drive strength of ENGCLK and CLKOUT should be selected based upon the external load, along with EMC considerations.

CDR_AR_851 Customer Information MPC555.M

Documentation: COLIE is bit 10 of COLIR Register

DESCRIPTION:
The COLIE (Change of Lock Interupt Enable) bit was incorrectly listed as bit 9 of the COLIR register. The correct location is bit 10.

WORKAROUND:
Software should be updated to reflect the proper bit location. Refer to Reference Manuals dated October 15,2000 or later for the correct bit location.

CDR_AR_1070 Customer Information MPC555.M

MPC555: Part Marking is MPC555LF8MZP40 or MPC555LF8CZP40

DESCRIPTION:
On devices that no longer show the mask set in the part marking, the part number marked on this revision is MPC555LF8MZP40 for full automotive temperature grade parts (-40 to +125C) and MPC555LF8CZP40 for commercial grade temperature range (-40 to +85C). Note that the 8 may not be present on some devices.

WORKAROUND:
Expect new marking on these devices. If the 8 is not present on the device, read the Mask Number field of the Internal Memory Mapping Register (IMMR[MASKNUM]) to determine the revision of the MPC555.

CDR_AR_600 Customer Information CMF.192KB_CDR1UBUS_05_0

Updated Flash Programming Algorithm and Control Registers

DESCRIPTION:
The flash programming and erase algorithms were changed. Consult the latest flash programming algorithm as published in the 15 Octonber 2000 MPC555 Reference Manual (or later) for the number and method of applying pulses. Additional control bits and modes have been added for use during programming and erase. Using the previous flash programming or erase algorithm will subject the part to additional stress, which must be avoided.

WORKAROUND:
Update the erase and program pulse widths and number of pulses to the algorithm published in the 15 October 2000 (or later) Reference Manual Use the latest programming driver from Freescale that incorpoates the version 6.0 or 6.1 programming algorithm. These are available in the Freescale Do NOT program with the previous flash programming or erase algorithms.

CDR_AR_485 Customer Information DPTRAM.6K_CDR1IMB3_03_0

Disable of TPU emulation mode while MISC enabled corrupts data in RAM

DESCRIPTION:
If the TPU emulation mode is negated while MISC is enabled, the DPTRAM data may be corrupted.

WORKAROUND:
In test mode / TPU development mode, disable the MISCEN (DPTMCR) before negation of TPEMEM in the TCR. In normal mode, disable MISCEN prior to performing a soft reset of the TPU (TPUMCR2).

CDR_AR_1143 Customer Information L2U.CDR1LBUSUBUS_02_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_810 Customer Erratum MIOS1.CDR1IMB3_04_2

MIOS: Synchronize writes to DASM B channel in OPWM mode.

DESCRIPTION:
In some cases of using DASM in OPWM mode, when writing B register, the output of the OPWM channel will remain asserted when it was expected to be negated. When B register of an OPWM channel is updated in the same system clock that a match on B is expected, the match will be ignored, and the OPWM output will remain asserted until the match on the new value of B. The problem sequence is: (1) OPWM output is asserted when A match occurs. (2) This match may trigger a SW task (e.g., via interrupt on A match) that updates B register (B is double buffer in this mode, i.e., host writes a new compare value to B, the OPWM output negates when the timer matches to the original value of B, then the new value is copied to the comparator to be used in the next PWM cycle) (3) If the write to B occurs at the same system clock that B match is expected, the match will not be recognized. (4) In this case, the OPWM output will remain asserted until the next B match that follows the next A match.

WORKAROUND:
Follow one of the following procedures: (1) read the relevant counter value (such as MCSM) before doing the DASM write. If the counter value is "just below" the old DASM B value, then the B register update should be delayed. (2) write to B register, and then check if the value of the relevant counter (such as MCSM) is bigger than the OLD B value. If so, it is required to force the pin value to it's desired state (FORCB). (3) When using DASM interrupt to update B register, verify that the PWM pulse width is larger than the interrupt latency. (4) keep the B value constant and only perform writes to the A register to alter the pulse width. In this case the value of A is updated after the interrupt from the previous channel A match. For this to work the pulse period needs to be greater than the interrupt latency, so that the new A value is written before the next A channel compare is enabled. (Note that the A register is not double buffered).

CDR_AR_929 Customer Erratum MIOS1.CDR1IMB3_04_2

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 MIOS1.CDR1IMB3_04_2

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_953 Customer Erratum PKPADRING.555_CDR1_03_0

Pads: TEA_b requires an external pull up

DESCRIPTION:
The internal weak pull-up of the tea_b pin can never be enabled.

WORKAROUND:
Always use an external pull-up on the TEA_b pin, as specified in the spec.

CDR_AR_940 Customer Information PKPADRING.555_CDR1_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_982 Customer Information PKPADRING.555_CDR1_03_0

PADS: QADC64 Port A (and MUX Out) have CMOS Digital Outputs

DESCRIPTION:
The 15 October 2000 MPC555 User Manual incorrectly lists the QADC64 Port A (A:PQA[0:7], A:MA[0:2], B:PQA[0:7], B:MA[0:2]) pins as open drain when selected as digital outputs. This is not correct. These pins are normal CMOS drivers in digital output mode.

WORKAROUND:
Expect the QADC64 Port A pins (and multiplexor out pins when enabled) to drive both high and low when selected as digital outputs.

CDR_AR_1019 Customer Erratum RCPU.CDR1LBUSIBUS_13_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_1077 Customer Erratum RCPU.CDR1LBUSIBUS_13_0

RCPU: Do not run multi-master compressed application with Show Cycles and BTB

DESCRIPTION:
If instruction show cycles (ICTRL[ISCT_SER] not equal to 0x7) and BTB are enabled in a compressed application with interrupts and another master (READI or External Bus master) initiates internal accesses on UBUS, the RCPU may execute incorrect instructions.

WORKAROUND:
Do not enable instruction show cycles together with BTB while running compressed application with interrupts if a UBUS master (READI or External Master) other than the RCPU or the L2U operated by the RCPU, accesses MCU internal resources through the UBUS.

CDR_AR_1138 Customer Erratum RCPU.CDR1LBUSIBUS_13_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_1076 Customer Erratum RCPU.CDR1LBUSIBUS_13_0

RCPU: Treat VF queue flush information value of 6 as 2

DESCRIPTION:
When the RCPU fetches instructions from zero wait state slaves on UBUS (Internal flash or SIU when in enhanced burst mode), the VF queue flush information may have the reserved value of 6.

WORKAROUND:
If a VF instruction queue flush value of 6 is shown on the VF pins, tools should treat this value as 2 for program tracking purposes.

CDR_AR_907 **Customer Information** **RCPU.CDR1LBUSIBUS_13_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.CDR1LBUSIBUS_13_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.CDR1LBUSIBUS_13_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.CDR1LBUSIBUS_13_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_563 Customer Erratum QADC64.CDR1IMB3_03_0

QSM/QSMCM/QADC64 corrupts data after an IACK cycle in CISC parts.

DESCRIPTION:
This problem does not affect parts that do not run IACK cycles (i.e. RISC CPUs). The Common BIU state machine, used by the QSM/QSMCM/QADC64, mis-tracks an IACK cycle if an interrupt is issued while an IACK cycle for the same level is in progress. In this case, the next access on the IMB3 will be corrupted by the QSM/QSMCM/QADC64. On CPU32 based parts (or CPU32X parts where the FASRAM is not used for the stack), the first access after an IACK cycles is the stacking of the vector offset The risk to the system by corrupting this stacked value is very low, since it is not used by the processor or most interrupt service routine software. On CPU32X based parts which have the stack located in the FASRAM, however, the first IMB3 access is the fetch from the vector table. Corruption of this value (handler address) causes the processor to jump to an incorrect location or to produce an address error.

WORKAROUND:
Workarounds exist for both CPU32 and CPU32X based parts. On CPU32 based parts the first access after an IACK cycles is the stacking of the vector offset The risk to the system by corrupting this stacked value is very low, it is not used by the processor. On CPU32X based parts which have the stack located in the FASRAM the first IMB3 access is the fetch from the vector table. Corruption of this value (handler address) causes the processor to jump to an incorrect location or to produce an address error. The suggested workarounds for the QSM/QSMCM/QADC64 are listed below. For CPU32 based parts: - assign the QSM/QSMCM/QADC64 it’s own interrupt levels separate from any other modules if the corruption of the vector offset in the stack frame is an issue. For CPU32X based parts: (a) assign the QSM/QSMCM/QADC64 its own interrupt levels separate from any other module in the system or (b) move the stack out of the FASRAM.

CDR_AR_754 Customer Erratum QADC64.CDR1IMB3_03_0

QADC64:Do not use queue1 in external gated mode with queue2 in continuous mode.

DESCRIPTION:
When the gate for queue1 opens when queue2 is converting the last word in its queue, queue1 completion flag will immediately set and no conversions will occur. Queue1 will remain in a hung state for the duration of the gate (no conversions will occur regardless of how long the gate is open). This failure will only occur when the QADC64 is configured with queue1 in external gated mode (continuous or single scan) and queue2 is in continuous mode. The failure mode can be detected if it is known that the gate for queue 1 is shorter than the length of the queue, and the completion flag becomes set. The failure can also be detected as follows: software writes invalid results to the result register (3ff when it is known the input will never go to full scale); after the gate has closed if the invalid result is still in result space 0, then the failure has occured.

WORKAROUND:
There are 2 workarounds: (1) Do not use queue 2 if queue1 is set for external gated mode. Or, (2) SETUP: (a) queue 2 mode : ’Interval Timer Single-Scan Mode’ (MQ2 = 11000) so the interval is (1/(2MHz/2048)) = 1.024ms (b) Pause bit set in CCW60 (c) Pause bit set in CCW61. FUNCTIONALITY: SSE2 bit gets set, the timer starts, and the internal trigger comes after 1.024ms. queue2 will then start converting and will continue until it sees the pause bit in CCW60. So, a reset could occur every 2ms, and the SSE2 bit should be set allowing the queue to begin again never having reached an end of queue. If ’Task jitter’ does occur, and the queue does not get reset before another internal trigger is created, then it will do a one word conversion and immediately pause again due to the pause bit set in CCW61. Even if there is enough ’Task jitter’ to allow this sub-queue to begin, it will be paused after only one conversion and will not reach the end of queue. Finally, it is assumed that it would not be possible to have enough uncertainty for another level of sub-queues to be needed.

CDR_AR_768 Customer Erratum QADC64.CDR1IMB3_03_0

QADC64: Queue2 activity may reset Queue1 ExtGates Single Scan SSE

DESCRIPTION:
If queue1 is in External Gated Single Scan mode, the SSE bit is written, and the queue is awaiting a trigger, an EOQ condition on queue2 will cause the queue1 SSE bit to be reset. This causes queue1 to not acknowledge any trigger, unless the SSE bit is again set.

WORKAROUND:
Do not let queue2 reach a EOQ while queue1 is in External Gated single scan mode, with SSE bit set, and awaiting a trigger. This may be done by breaking queue2 into sub-queues,and not allowing it to run to the end.

CDR_AR_1125 Customer Erratum QADC64.CDR1IMB3_03_0

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 QADC64.CDR1IMB3_03_0

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_563 Customer Erratum QSMCM.CDR1IMB3_02_2

QSM/QSMCM/QADC64 corrupts data after an IACK cycle in CISC parts.

DESCRIPTION:
This problem does not affect parts that do not run IACK cycles (i.e. RISC CPUs). The Common BIU state machine, used by the QSM/QSMCM/QADC64, mis-tracks an IACK cycle if an interrupt is issued while an IACK cycle for the same level is in progress. In this case, the next access on the IMB3 will be corrupted by the QSM/QSMCM/QADC64. On CPU32 based parts (or CPU32X parts where the FASRAM is not used for the stack), the first access after an IACK cycles is the stacking of the vector offset The risk to the system by corrupting this stacked value is very low, since it is not used by the processor or most interrupt service routine software. On CPU32X based parts which have the stack located in the FASRAM, however, the first IMB3 access is the fetch from the vector table. Corruption of this value (handler address) causes the processor to jump to an incorrect location or to produce an address error.

WORKAROUND:
Workarounds exist for both CPU32 and CPU32X based parts. On CPU32 based parts the first access after an IACK cycles is the stacking of the vector offset The risk to the system by corrupting this stacked value is very low, it is not used by the processor. On CPU32X based parts which have the stack located in the FASRAM the first IMB3 access is the fetch from the vector table. Corruption of this value (handler address) causes the processor to jump to an incorrect location or to produce an address error. The suggested workarounds for the QSM/QSMCM/QADC64 are listed below. For CPU32 based parts: - assign the QSM/QSMCM/QADC64 it’s own interrupt levels separate from any other modules if the corruption of the vector offset in the stack frame is an issue. For CPU32X based parts: (a) assign the QSM/QSMCM/QADC64 its own interrupt levels separate from any other module in the system or (b) move the stack out of the FASRAM.

CDR_AR_1151 Customer Erratum QSMCM.CDR1IMB3_02_2

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_584 Customer Information QSMCM.CDR1IMB3_02_2

QSMCM: Do not use link baud and ECK modes

DESCRIPTION:
Reads of the SCI control and status registers do not read correctly when using the link baud or the external clock source feature of the QSMCM. These modes are enabled by the SCCxR0 control register bits 0 and 1 (OTHR and LNKBD). These modes are not fully operational.

WORKAROUND:
Do not use the link baud or external clock modes of the QSMCM. The OTHR bit in the SCCxR0 control register 0 must be set = 0 to use normal mode operation only.

CDR_AR_1144 Customer Erratum TOUCAN.CDR1IMB3_04_0

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.CDR1IMB3_04_0

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.CDR1IMB3_04_0

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.CDR1IMB3_03_0

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_577 Customer Information TPU3.CDR1IMB3_03_0

TPU3 - TCR2PSCK2 bit does not give TCR2 divide ratios specified.

DESCRIPTION:
The TCR2PSCK2 bit was originally specified to cause the TCR2 timebase to be divided by 2. Actually, it causes the TCR2 timebase to be divided as follows: The /16 of external clock and /128 of internal clock are eliminated and /3, /7, /15 of the external clock and /24, /56, /120 of the internal clock are added.

WORKAROUND:
When the TCR2PSCK2 is set, instead of the specified divides of /16, /32, /64, /128, expect the internal clock source to be /8, /24, /56 and /120 for TCR2 Prescaler values of 00, 01, 10 and 11, respectively. Likewise, for the external clock source expect /1, /3, /7, /15 instead of /2, /4, /8, /16.

CDR_AR_896 Customer Erratum UIMB.CDR1UBUSIMB3_03_0

UIMB: Avoid external code in addresses 0xZ[3,7,B,F]0_7F80 to 0xZ[3,7,B,F]0_7FFC

DESCRIPTION:
When two UBUS cycles are precisely pipelined, such that the first cycle is to an address within the IMB address range, (Internal memory map base address + 0x300000:0x307F7F), and the 2nd cycle is a fetch to an external address in which A[10:29] match A[10:29] of any unimplemented register of the UIMB module, then the IMB cycle will be tagged with an error resulting in a machine check exception. During operation, the pipelining of fetches relative to an IMB access will vary if an interrupt occurs between the last change of flow and the IMB access.

WORKAROUND:
1) Do not place instructions which might be fetched after an IMB access in external memory which matches A[10:24] = 0x60FF. In other words, the instruction address must not fall in the ranges: 0xZY0_7F80 to 0xZY0_7FFC, where ZY is in the external address space, Z=0x00 to 0xFF and Y is 3,7,B, or F. Or 2) Ensure that an external fetch is not pipelined with an IMB access by (a) running from internal memory, (b) running in serialized mode.

CDR_AR_985 Customer Erratum USIU.CDR1UBUS_06_1

USIU: Do not use ORx[EHTR] with Dual Mapping

DESCRIPTION:
When an access is matched through the Dual Mapping registers (DMBR/DMOR), extended hold time (from a previous access region) or Burst length (from the new access region) may cause execution of wrong code.

WORKAROUND:
1) Do not set ORx[EHTR] while a dual mapping region is enabled. Or: 2) Do not enable dual mapping if an extended hold time is required for any memory in the system.

CDR_AR_925 Customer Erratum USIU.CDR1UBUS_06_1

USIU: TEXP feature does not function when VDD supply is off

DESCRIPTION:
The TEXP function does not work if the main power supplies are powered down. Whenever VDD (low voltage supplies other than KAPWR and VDDSRAM) is powered down, hreset_b will be asserted by the chip and low power mode exited. The TEXP pin will never be asserted.

WORKAROUND:
The TEXP pin will never be asserted if VDD is powered down. Use an external counter to indicate the length of power down. As an alternate solution, put the part into Deep Sleep mode to reduce power consumption and leave the power supplies powered.

CDR_AR_909 Customer Erratum USIU.CDR1UBUS_06_1

USIU: Do not assert cr_b to abort pending store reservation access

DESCRIPTION:
If an external cancel reservation (cr_b) is asserted then a pending store reservation may show on the external bus. This may occur with or without transfer start (ts_b), and will terminate after 1 clock. If the region is in the memory controller of the chip generating the store with reservation, then no chip-select or other memory controller attributes will assert on the bus, and the memory will not be altered.

WORKAROUND:
1) Do not assert cr_b; or 2) following assertion of cr_b, external logic must prevent the erroneous store with reservation bus cycle from altering memory, and must not assert ta_b to terminate the erroneous store with reservation bus cycle.

CDR_AR_910 Customer Erratum USIU.CDR1UBUS_06_1

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_984 Customer Erratum USIU.CDR1UBUS_06_1

USIU: Setting of SCCR[EBDF] may slow execution of code

DESCRIPTION:
If the SCCR[EBDF] is greater than 0 and the RCPU is running not serialized, the USIU may issue external read bus cycles that are not complete. The TS_B will assert with an address, but without a chip select or STS_B assertion. These cycles may cause a delay in execution of application code. These cycles will self terminate in 1 to 3 clocks, depending on the TS_B signal negation rate, defined by the external pull up strength and board capacitance.

WORKAROUND:
There are two possible workarounds: 1) In a program with critical timing, do not run from external memory with the SCCR[EBDF] set to a value greater than 0. Or 2) If external logic is used as a memory controller, define the logic to disregard these extra bus cycles.

CDR_AR_1134 Customer Erratum USIU.CDR1UBUS_06_1

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.CDR1UBUS_06_1

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.CDR1UBUS_06_1

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.CDR1UBUS_06_1

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_1152 Customer Erratum USIU.CDR1UBUS_06_1

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_1154 Customer Erratum USIU.CDR1UBUS_06_1

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.CDR1UBUS_06_1

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_389 Customer Information USIU.CDR1UBUS_06_1

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_442 Customer Information USIU.CDR1UBUS_06_1

Avoid loss of clock during HRESET

DESCRIPTION:
The chip may fail to switch to backup clock. This mode may occur if the input reference clock fails to toggle during hreset while switching from normal clock to backup clock. This condition may occur while switching from backup clock to normal clock (during hreset) if the PLL is not locked and there is no reference clock. In order to resume operation, the part may require the input reference clock to resume (for 1-2 more clocks) or for PORESET to be asserted.

WORKAROUND:
Avoid loss of the reference clock during hreset; ensure that the PLL is locked before switching to PLL clock. Do not enable reset upon loss of lock if limp mode is enabled, instead enable an change of lock interrupt by setting the COLIE bit (COLIR).

CDR_AR_594 Customer Information USIU.CDR1UBUS_06_1

USIU: Changing PLL MF to 1:1 mode can have 180 degree phase shift

DESCRIPTION:
After software changes MF from >1 to MF = 1, a 180 degree skew between EXTCLK and CLKOUT could occur.

WORKAROUND:
If synchronization between EXTCLK and CLKOUT is required, set MODCK to boot in 1:1 mode, and do not alter the MF bits to exit 1:1 mode.

CDR_AR_598 Customer Information USIU.CDR1UBUS_06_1

USIU: Ensure proper configuration for proper startup

DESCRIPTION:
In some systems, the PLL does not lock on power-up, or the system does not properly execute software out of reset. This issue occurs on some board designs, and not on others. Locking may be improved by board design and component selection, and can be resolved by paying attention to the design and setup, and ensuring that the PLL and Oscillator components are correct and as noise free as possible.

WORKAROUND:
First, make sure that the PLL and reset circuitry is correct: ensure that the PLL components are properly selected and that the PLL power (VDDSYN) is not noisy. Refer to appendix E of the users manual, "Clock and Board Guidelines". Verify that the XFC capacitor is connected to VDDSYN. Validate that the TRST pin is asserted upon power-up. Do not connect TRST to HRESET or SRESET. Validate that all power supplies are stable and all MODCK pins are at the correct levels in time for the PLL and Oscillator to be stable prior to PORESET rising above VIL. Verify that the proper reset configuration word is used. Validate the reset and post reset pin state for each pin controlled by the reset configuration word, and ensure there is not a conflict with an external driver. Preferably use the internal reset configuration word. If using an external reset configuration word, do not rely on the internal pull-downs to operate (refer to CDR_AR_454) and ensure that RSTCONF is asserted until SRESET is negated. After the part exits reset with the system running via the backup clock, validate the clock control registers settings and the PLL status. If the PLL is slow on locking, or the register settings indicate the MODCK pins are incorrect, address the board issues listed above. To avoid risk of system failure for no start, enable limp mode, allowing the system to boot using the backup clock even though lock is not yet indicated. After booting, switch from backup clock to PLL clock under software control after the PLL has gained lock.

CDR_AR_1109 Customer Information USIU.CDR1UBUS_06_1

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_1137 Customer Information USIU.CDR1UBUS_06_1

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.CDR1UBUS_06_1

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.