Mask Set Errata for Mask 0N78S

This report applies to mask 0N78S for these products:

- MPC5748G
- MPC5747G
- MPC5746G
- MPC5748C
- MPC5747C

Mask Specific Information

JTAG identifier	0x0988201D

Table 1. Errata and Information Summary

Erratum ID	Erratum Title
e10542	DSPI: Transmit, Command, and Receive FIFO fill flags in status register is not cleared when DMA is improperly configured
e10452	eDMA: When master ID replication is enabled, the stored ID and privilege level will change if read by another master.
e9978	eMIOS: Unexpected channel flag assertion during GPIO to MCB mode transition
e10594	ENET: The ENET1 (Ethernet) module does not function unless the Peripheral control register (MC_ME_PCTL6) is enabled .
e7991	FLASH: Rapid Program or Erase Suspend fail status
e10595	FlexCAN: FLEXCAN1-7 modules will not work unless the Fast External Oscillator (FXOSC) clock source is enabled
e10368	FlexCAN: Transition of the CAN FD operation enable bit may lead FlexCAN logic to an inconsistent state.
e10620	FlexRay: The FS80 clock source should not be selected for the FlexRay protocol clock when the MCU clocking is configured for Linear Dynamic Frequency Scaling
e8770	FlexRAY: Missing TX frames on Channel B when in dual channel mode and Channel A is disabled
e8180	HSM: e200z0 Nexus interface DQTAG implemented as variable length field in DQM message
e10762	HSM: When the Hardware Security Module (HSM) is enabled, execution of selftest will lead to the HSM Destructive Reset Flag being set
e10849	IAHB: Programming of PCM Pending Read Enable can lead to a master stalling or receiving incorrect or spurious data

Table continues on the next page...

Table 1. Errata and Information Summary (continued)

Erratum ID	Erratum Title
e8933	LINFlexD: Inconsistent sync field may cause an incorrect baud rate and the Sync Field Error Flag may not be set
e10141	LPU: LPU_RUN mode system clock must be preconfigured for undivided FIRC prior to LPU_STANDBY entry
e10609	MC_CGM: CLKOUT_0 and CLKOUT_1 may stop if the clock selection is changed when configured for divide by 2
e10603	NPC: Nexus Port Controller (NPC) must be enabled to allow mode changes during debug
e10723	NPC: Repeated Nexus3 Debug Status messages can be observed if more than one master (including a device core) is active and the core is subsequently disabled
e10340	NZxC3: ICNT and HIST fields of a Nexus message are not properly reset following a device reset
e10396	PASS: Password challenge to PASS fails while program erase ongoing in any block in memory partition 0
e9873	PFLASH: Calibration remap to flash memory not supported on 16KB and 32KB flash blocks in address range 0x00F90000-0x00FBFFFF
e10789	PFLASH: EEPROM ECC error suppression is not supported on 16KB and 32KB flash blocks in the address range 0x00F90000-0x00FBFFFF
e11096	SAI: Internal bit clock is not generated when RCR2[BCI]=1 or TCR2[BCI]=1
e11150	SAI: Internally generated receive or transmit BCLK cannot be re-enabled if it is first disabled when RCR2[DIV] or TCR2[DIV] > 0
e10815	STCU2: Offline LBIST execution will toggle GPIO[29] ,GPIO[30], GPIO[62] and GPIO[63]
e10721	STCU: If the Auxiliary Clock 9 Select Control Register CGM_AC9_SC[SELCTL] = 1, offline-BIST will fail leading to a STCU watchdog timeout
e10577	STCU: Device may get stuck in reset if the Fast External Oscillator (FXOSC) is lost during self test
e10200	STM: Reading the System Timer Module Count register may return an incorrect value

Table 2. Revision History

Revision	Changes
1. February 2017	Initial revision
2, January 2018	The following errata were added. • e11096 • e10620 • e10849 • e10542 • e10815 • e10396 • e11150 The following errata were revised. • e10721 • e10609 • e10200 • e10789

e10542: DSPI: Transmit, Command, and Receive FIFO fill flags in status register is not cleared when DMA is improperly configured

Description: The Deserial/Serial Peripheral Interface Transmit, Receive, and Command First In/First Out (FIFO) buffers can request additional information to be transferred via the Direct Memory Access (DMA) module when either the Transmit, Receive, or Command FIFO Fill/Drain Flags are set in the DSPI Status Register (SR[TFFF/RFDF/CMDFFF]). However, the Command/ Transmit Fill Flag only indicates that at least 1 location in the FIFO is available to be written. It does not indicate that the FIFO is empty. Similarly, Receive FIFO fill flag only indicates at least 1 location of the FIFO is available to be read. It does not indicate that the FIFO is full. If the DMA is configured to transfer more than 1 FIFO location size of data, the FIFO Fill Flags may not be properly cleared indicating that the FIFO is not full even when the FIFO is actually full (for Transmit and Command FIFO) and not empty when the FIFO is actually empty (for Receive FIFO).

Workaround: Properly configure the DMA to fill the Transmit, Receive, and Command FIFOs only one FIFO location, in other words, up to 2 bytes, at a time to each of the FIFOs.

Use the DMA loop to transfer more data if needed.

e10452: eDMA: When master ID replication is enabled, the stored ID and privilege level will change if read by another master.

Description: When master ID replication is enabled (DMA_DCHMIDn[EMI]=1), the DMA_DCHMIDn[PAL] and DMA_DCHMIDn[MID] fields should reflect the privilege level and master ID respectively of the master that wrote the DMA_TCDn_CSR[DONE:START] byte. However, if a different master reads the DMA_TCDn_CSR[DONE:START] byte, the master ID and privilege level will incorrectly change to this read access.

Workaround: Only allow the intended master ID replication core to access the DMA_TCDn_CSR[DONE:START] byte.

e9978: eMIOS: Unexpected channel flag assertion during GPIO to MCB mode transition

Description: When changing an Enhanced Modular IO Subsystem (eMIOS) channel mode from General Purpose Input/Output (GPIO) to Modulus Counter Buffered (MCB) mode, the channel flag in the eMIOS Channel Status register (eMIOS_Sn[FLAG]) may incorrectly be asserted. This will cause an unexpected interrupt or DMA request if enabled for that channel.

Workaround: In order to change the channel mode from GPIO to MCB without causing an unexpected interrupt or DMA request, perform the following steps:

(1) Clear the FLAG enable bit in the eMIOS Control register (eMIOS_Cn[FEN] = 0).

(2) Change the channel mode (eMIOS_Cn[MODE]) to the desired MCB mode.

(3) Clear the channel FLAG bit by writing ‘1’ to the eMIOS Channel Status register FLAG field (eMIOS_Sn[FLAG] = 1).

(4) Set the FLAG enable bit (eMIOS_Cn[FEN] = 1) to re-enable the channel interrupt or DMA request reaction.

e10594: ENET: The ENET1 (Ethernet) module does not function unless the Peripheral control register (MC_ME_PCTL6) is enabled .

Description: The Ethernet module ENET1 does not function unless the MLB Peripheral control register (MC_ME_PCTL6) is enabled. This does not apply to ENET0.

Workaround: Enable MC_ME_PCTL6 if using the ENET1 module.

e7991: FLASH: Rapid Program or Erase Suspend fail status

Description: If a flash suspend operation occurs during a 5us window during a verify operation being executed by the internal flash program and erase state machine, and the suspend rate continues at a consistent 20us rate after that, it is possible that the flash will not exit the program or erase operation. A single suspend during a single program or erase event will not cause this issue to occur.

Per the flash specification, a flash program or erase operation should not be suspended more than once every 20 us, therefore, if this requirement is met, no issue will be seen. IF the suspend rate is faster than 20 us continuously, a failure to program/erase could occur.

Workaround: When doing repeated suspends during program or erase ensure that suspend period is greater than 20us.

e10595: FlexCAN: FLEXCAN1-7 modules will not work unless the Fast External Oscillator (FXOSC) clock source is enabled

Description: FLEXCAN modules 1-7 will not work unless the Fast External Oscillator (FXOSC) clock source is enabled on the device.

Workaround: The FXOSC clock should be enabled before using FLEXCAN1-7 modules by setting the Oscillator Enable bit (FXOSCON) in the active mode configuration register (MC_ME_xxxx_MC).

e10368: FlexCAN: Transition of the CAN FD operation enable bit may lead FlexCAN logic to an inconsistent state.

Description: The activation or deactivation of the CAN FD operation by setting or clearing the FDEN bit of the CAN_MCR register or by setting the FlexCAN soft reset bit (SOFTRST) of the CAN_MCR register when the FDEN bit is enabled may cause an internal FlexCAN register to become metastable. As result, the first CAN frame, transmitted or received, may have corrupted data (ID and payload). However, even though the data is corrupted, a valid CAN frame is transmitted because the Cyclic Redundancy Check (CRC) calculation is based on the corrupted data. During reception the data is corrupted internally after the CRC bits have been checked and therefore this corrupted data may be stored in a reception message buffer. After the first CAN frame, all subsequent frames are transmitted and received correctly.

Workaround: Perform the following steps to set the FDEN bit:

1. If FlexCAN is already in freeze mode, go to step 3, otherwise set the HALT and FRZ bits of the CAN_MCR register.

2. Wait the FRZACK bit of the CAN_MCR register to be set by the hardware.

3. Set the LPB (Loop Back Mode) bit of the CAN_CTRL1 register.

4. Configure only one message buffer to be transmitted. The frame should be a classical one (non-FD) with IDE =0, RTR =1 DLC =0x5 and STD_ID =0x682.

5. Set the FDEN bit of the CAN_MCR register.

6. Clear the HALT bit of the MCR register to leave freeze mode.

7. Wait the FRZACK bit of the CAN_MCR register to be cleared by the hardware.

8. Wait the respective bit of the CAN_IFLAG register to be set (successfully transmission in loop back mode).

9. Clear the respective bit of the CAN_IFLAG register by writing 1.

10. Set the HALT and FRZ bits of the CAN_MCR register.

11. Wait the FRZACK bit of the CAN_MCR register to be set by the hardware.

12. Clear the LPB (Loop Back Mode) bit of the CAN_CTRL1 register.

Perform the following steps to apply a soft reset or clear the FDEN bit:

1. If FlexCAN is already in freeze mode, go to step 3, otherwise set the HALT and FRZ bits of the CAN_MCR register.

2. Wait the FRZACK bit of the CAN_MCR register to be set by the hardware.

3. Set the SOFTRST bit of the CAN_MCR register.

4. Wait the SOFTRST bit of the CAN_MCR register to be cleared by the hardware.

5. Set again the SOFTRST bit of the CAN_MCR register.

6. Wait the SOFTRST bit of the CAN_MCR register to be cleared by the hardware.

e10620: FlexRay: The FS80 clock source should not be selected for the FlexRay protocol clock when the MCU clocking is configured for Linear Dynamic Frequency Scaling

Description: The FlexRay module protocol clock can be selected from the FXOSC clock (default) or the FS80 clock and this is configured at FR_MCR[CLKSEL]. When the MCU clock configuration is changed from the default state to the Linear DFS (Dynamic Frequency Scaling) clock mode, the FS80 must not be selected as the source for the FlexRay protocol clock.

Workaround: Prior to configuring the Linear DFS clock mode the user must select FXOSC for the FlexRay protocol clock.

e8770: FlexRAY: Missing TX frames on Channel B when in dual channel mode and Channel A is disabled

Description: If the FlexRay module is configured in Dual Channel mode, by clearing the Single Channel Device Mode bit (SCM) of the Module Control register (FR_MCR[SCM]=0), and Channel A is disabled, by clearing the Channel A Enable bit (FR_MCR[CHA]=0) and Channel B is enabled, by setting the Channel B enable bit (FR_MCR[CHB]=1), there will be a missing transmit (TX)

frame in adjacent minislots (even/odd combinations in Dynamic Segment) on Channel B for certain communication cycles. Which channel handles the Dynamic Segment or Static Segment TX message buffers (MBs) is controlled by the Channel Assignment bits (CHA, CHB) of the Message Buffer Cycle Counter Filter Register (FR_MBCCFRn). The internal Static Segment boundary indicator actually only uses the Channel A slot counter to identify the Static Segment boundary even if the module configures the Static Segment to Channel B (FR_MBCCFRn[CHA]=0 and FR_MBCCFRn[CHB]=1). This results in the Buffer Control Unit waiting for a corresponding data acknowledge signal for minislot:N in the Dynamic Segment and misses the required TX frame transmission within the immediate next minislot:N+1.

Workaround: 1. Configure the FlexRay module in Single Channel mode (FR_MCR[SCM]=1) and enable Channel B (FR_MCR[CHB]=1) and disable Channel A (FR_MCR[CHA]=0). In this mode the internal Channel A behaves as FlexRay Channel B. Note that in this mode only the internal channel A and the FlexRay Port A is used. So externally you must connect to FlexRay Port A.

2. Enable both Channel A and Channel B when in Dual Channel mode (FR_MCR[CHA=1] and FR_MCR[CHB]=1). This will allow all configured TX frames to be transmitted correctly on Channel B.

e8180: HSM: e200z0 Nexus interface DQTAG implemented as variable length field in DQM message

Description: The Hardware Security Module (HSM) core (e200z0) implements the Data Tag (DQTAG) field of the Nexus Data Acquisition Message (DQM) as a variable length packet instead of an 8-bit fixed length packet. This may result in an extra clock (“beat”) in the DQM trace message depending on the Nexus port width selected for the device.

Workaround: Tools should decode the DQTAG field as a variable length packet instead of a fixed length packet.

e10762: HSM: When the Hardware Security Module (HSM) is enabled, execution of selftest will lead to the HSM Destructive Reset Flag being set

Description: When the HSM is enabled, a Logic Built in Self Test (LBIST) execution will cause the flag indicating an HSM Destructive Reset event in the Destructive Event Status Register (MC_RGM_DES[F_HSM_DEST]) to be set even though there has been no actual HSM destructive reset event triggered. Only the flag is set during LBIST, no actual reset occurs.

Workaround: Since the HSM does not function during LBIST, user software should not treat the the setting of the MC_RGM_DES[F_HSM_DEST] flag during LBIST execution as a security/safety issue. Users should clear this flag as part of the post LBIST software routine.

e10849: IAHB: Programming of PCM Pending Read Enable can lead to a master stalling or receiving incorrect or spurious data

Description: Overriding the default configuration of bus traffic optimization for specific masters on the device (Ethernet, eDMA, USB_0, USB_1, MLB, and uSDHC,Z2) can cause those masters to stall, receive wrong read data, or get a spurious read access when uncorrectable ECC errors are received from slaves.

Workaround: To avoid this situation, following configuration fields in the Platform Configuration Module (PCM) should not be modified:

PCM_IAHB_BE0[PRE_ENET]

PCM_IAHB_BE0[PRE_DMA]

PCM_IAHB_BE1[PRE_uSDHC]

PCM_IAHB_BE1[PRE_USB_1]

PCM_IAHB_BE1[PRE_USB_0]

PCM_IAHB_BE1[PRE_MLB]

PCM_IAHB_BE2[PRE_Z2_DATA]

PCM_IAHB_BE2[PRE_Z2_INST]

e8933: LINFlexD: Inconsistent sync field may cause an incorrect baud rate and the Sync Field Error Flag may not be set

Description: When the LINFlexD module is configured as follows:

1. LIN (Local Interconnect Network) slave mode is enabled by clearing the Master Mode Enable bit in the LIN Control Register 1 (LINCR1[MME] = 0b0)

2. Auto synchronization is enabled by setting LIN Auto Synchronization Enable (LINCR1[LASE] = 0b1)

The LINFlexD module may automatically synchronize to an incorrect baud rate without setting the Sync Field Error Flag in the LIN Error Status register (LINESR[SFEF]) in case Sync Field value is not equal to 0x55, as per the Local Interconnect Network (LIN) specification.

The auto synchronization is only required when the baud-rate in the slave node can not be programmed directly in software and the slave node must synchronize to the master node baud rate.

Workaround: There are 2 possible workarounds.

Workaround 1:

When the LIN time-out counter is configured in LIN Mode by clearing the MODE bit of the LIN Time-Out Control Status register (LINTCSR[MODE]= 0x0]):

1. Set the LIN state Interrupt enable bit in the LIN Interrupt Enable register (LINIER[LSIE] = 0b1)

2. When the Data Reception Completed Flag is asserted in the LIN Status Register (LINSR[DRF] = 0b1) read the LIN State field (LINSR[LINS])

3. If LINSR[LINS]= 0b0101, read the Counter Value field of the LIN Time-Out Control Status register (LINTCSR[CNT]), otherwise repeat step 2

4. If LINTCSR[CNT] is greater than 0xA, discard the frame.

When the LIN Time-out counter is configured in Output Compare Mode by setting the LINTCSR[MODE] bit:

1. Set the LIN State Interrupt Enable bit in the LIN Interrupt Enable register (LINIER[LSIE])

2. When the Data Reception Completed flag bit is asserted in the LIN Status Register (LINSR[DRF] = 0b1), read the LINSR[LINS] field

3. If LINSR[LINS]= 0b0101, store LINTCSR[CNT] value in a variable (ValueA), otherwise repeat step 2

4. Clear LINSR[DRF] flag by writing LINSR[LINS] field with 0xF

5. Wait for LINSR[DRF] to become asserted again and read LINSR[LINS] field

6. If LINSR[LINS] = 0b0101, store LINTCSR[CNT] value in a variable (ValueB), else repeat step 4

7. If ValueB – ValueA is greater than 0xA, discard the frame

Workaround 2:

Do not use the auto synchronization feature (disable with LINCR1[LASE] = 0b0) in LIN slave mode.

e10141: LPU: LPU_RUN mode system clock must be preconfigured for undivided FIRC prior to LPU_STANDBY entry

Description: If the LPU_RUN mode system clock is selected to be FXOSC or divided-FIRC when LPU_STANDBY mode is entered then the MCU may not return to LPU_RUN mode on a wake-up event.

In LPU_RUN mode the FXOSC or divided-FIRC can be used as the system clock, but the user must ensure that the undivided FIRC is selected as the system clock before the LPU_STANDBY mode transition is initiated.

Workaround: Prior to entering LPU_STANDBY select undivided FIRC as the LPU System Clock by configuring LPU_RUN_CF[SYS_CLK_SEL] = 0 and FIRC_CTL[FIRCDIV] = 5'b0.

e10609: MC_CGM: CLKOUT_0 and CLKOUT_1 may stop if the clock selection is changed when configured for divide by 2

Description: If the clock out functionality is enabled on either CLKOUT_0 and/or CLKOUT_1 and is configured for divide by 2 (via MC_CGM_AC6_DC0[DE] and/or MC_CGM_CLKOUT1_DC0[DE] = 0b1), then if the clock selection for CLKOUT_0/CLKOUT_1 is changed via MC_CGM_AC6_SC[SELCTL]/MC_CGM_CLKOUT1_SC[SELCTL] register respectively or a Destructive, Functional (long/short) reset occurs then the clock out may stop. The following clock sources when selected are affected:

- FXOSC
- FXOSC divided
- FXOSC ANA Clk
- SXOSC
- SXOSC divided
- SIRC
- SIRC divided
- PLL_CLKOUT1
- PLL_CLKOUT2
- RTC_CLK
- CAN0 CHI clk (when driven by FXOSC, not affected when driven by FS80)
- CAN0 PE clk (when driven by FXOSC, not affected when driven by F40)

Workaround: Changing CLKOUT_0/CLKOUT_1 clock source selection value via software, resets all its corresponding dividers and recovers them.

Apply the following sequence after each reset for enabled CLKOUT_0/CLKOUT_1 clock dividers that are to be configured to divide by 2 for the application.

1. Disable the CLKOUT_0 and/or CLKOUT_1 clock divider by writing to MC_CGM_AC6_DC0[DE] and/or MC_CGM_CLKOUT1_DC0[DE] = 0b0

2. Change the CLKOUT_0 and/or CLKOUT_1 clock source selection to FIRC (MC_CGM_AC6_SC[SELCTL] = 0b0001 and/or MC_CGM_CLKOUT1_SC[SELCTL] = 0b1001).

3. Select the desired clock source as the CLKOUT_0 and/or CLKOUT_1 clock source (e.g. for FXOSC: MC_CGM_AC6_SC[SELCTL] = 0b0000 and/or MC_CGM_CLKOUT1_SC[SELCTL] = 0b1000).

4. Configure and enable the corresponding CLKOUT_0 and/or CLKOUT_1 clock divider by writing to MC_CGM_AC6_DC0[DE] and/or MC_CGM_CLKOUT1_DC0[DE] = 0b1.

e10603: NPC: Nexus Port Controller (NPC) must be enabled to allow mode changes during debug

Description: The Nexus Port Controller (NPC) must be enabled to allow mode changes via the Mode Entry module in debug mode. The e200zx core generates some Nexus trace messages automatically even when trace is not enabled if a Nexus Enable instruction is executed (typically used for Nexus read/Write access of memory by a tool). As a result, if the NPC is not enabled, the core will still see messages pending and never complete the requested mode change.

Workaround: Enable the NPC by enabling the Message Clock Output (MCKO_EN = 1) in the NPC Port Configuration Register (NPC_PCR).

e10723: NPC: Repeated Nexus3 Debug Status messages can be observed if more than one master (including a device core) is active and the core is subsequently disabled

Description: This errata applies to the condition where there is more than one master active on the Nexus Port Controller (NPC) module, and one or more of these masters is a device core. In this situation, if a mode transition is initiated to a mode where that device core is disabled, with the clock gated (as configured in the relevant core control register MC_ME_CCTLx for the requested mode) then message data can be left pending on the interface until the core clock resumes. This causes status message to be repeated several times and no other message from any other Nexus3 client can be transmitted causing potential debugger problems.

Workaround: While transitioning to a low power mode(STOP, STANDBY, LPU_RUN), use the NPC Handshake by clearing NPC_1 PCR [LP1_SYNC] bit. The debugger can then disable the Nexus3 tracing of the core before it acknowledges that the transition into a low-power mode may proceed . For a non-low power mode transition (DRUN, RUNx) , do not disable device core but instead use the Power Architecture ‘wait' instruction to move the device core to the wait state.

Alternatively, transmit repeated or more than one TCODE messages from the active masters.

e10340: NZxC3: ICNT and HIST fields of a Nexus message are not properly reset following a device reset

Description: Following reset, if instruction trace is enabled in the Nexus e200zx core Class 3 trace client (NZxC3), the e200zx core transmits a Program Trace – Synchronization Message (PT-SM). The PT-SM includes the full execution address and the number of instructions executed since the last Nexus message (ICNT) information. However, the ICNT and the Branch History field (HIST), if Branch History trace is enabled, are not properly cleared when this message is transmitted. This may cause unexpected trace reconstruction results until the next Nexus Program Trace Synchronization Message (Program Trace – Direct Branch Message with Sync, Program Trace – Indirect Branch Message with Sync, or Program Trace – Indirect Branch History Message with Sync).

In Branch History mode, the first indirect branch following the reset (and the initial PT-SM) will contain the branch history prior to the reset plus the branch history after reset. However, there is no way to determine which branches occurred prior to reset and which followed reset.

Workaround: If not using branch history trace mode, to recreate the proper trace, the tool should take into account that the ICNT field is not cleared by the first PT-SM. The previous ICNT will be added to new ICNT value in the subsequent Nexus message. This may require extra processing by the tool.

If using branch history mode, then an accurate reconstruction of the executed code just before and just after reset may not be possible. Trace reconstruction can be recovered after the next indirect branch message.

On devices that bypass the Boot Assist Flash (BAF) or Boot Assist Module (BAM) after reset (in other words, the System Status and Configuration Module [SSCM] boots directly to user code if a valid Reset Configuration Half-Word is found), perform an indirect branch instruction shortly after reset to reset the ICNT (and HIST if Branch History mode is enabled). A full program trace synchronization message will be generated after 256 direct branches even if there is no indirect branches. This will allow the tool to recover the trace reconstruction from that point onward.

On devices that always execute the BAF or BAM, an indirect branch will occur during the BAF/BAM execution and the tool trace will be re-synchronized prior to the execution of user code.

e10396: PASS: Password challenge to PASS fails while program erase ongoing in any block in memory partition 0

Description: If the device is in a Censored state (enabled by programming the censorship DCF in UTEST) and a JTAG password is configured to enable device debug access, then the password challenge to the PASS module would be initiated by programming the Challenge Selector Register (PASS_CHSEL) to determine the password group, then programming the Challenge Input Registers (PASS_CINn) with the correct password. Programming the correct password would then allow enabling of debug interface access.

However, this operation will fail if a program or erase operation is ongoing on any flash block in memory partition 0, since this is shared with the UTEST block where the JTAG password resides.

Workaround: Users should ensure that no program or erase operations are occuring on any memory partitions shared with the UTEST block before initiating a password challenge. This can be monitored through the flash module configuration register program and erase status bits (C55FMC_MCR[PGM], C55FMC_MCR[ERS]).

e9873: PFLASH: Calibration remap to flash memory not supported on 16KB and 32KB flash blocks in address range 0x00F90000-0x00FBFFFF

Description: The PFLASH module supports calibration remapping of a flash access to another on-chip flash address. UTEST flash, BAF, and secure flash blocks cannot be remapped nor can accesses to other flash blocks be rerouted to addresses in UTEST flash, BAF, or secure flash. Flash blocks of size 16kB and 32KB in address range 0x00F90000-0x00FBFFFF do not support calibration remap to flash memory. All other flash blocks of size 32KB, 64KB 256KB in address range 0x00FC0000-0x0157FFFF can be overlaid using the mirrored address range.

Workaround: When using the calibration remapping of flash feature, the user must select flash blocks of size 32KB, 64KB 256KB in address range 0x00FC0000-0x0157FFFF.

e10789: PFLASH: EEPROM ECC error suppression is not supported on 16KB and 32KB flash blocks in the address range 0x00F90000-0x00FBFFFF

Description: The PFLASH module supports the suppression of ECC event reporting on secure data flash blocks in the address range 0x00F80000 – 0x00F87FFF.For more information see MPC5748G Reference Manual section “ECC on data flash accesses”. Flash blocks of size 16KB and 32KB in address range 0x00F90000 – 0x00FBFFFF do not support suppression of error reporting on ECC events.When reading from the address range 0x00F90000-0x00FBFFFF any non-correctable ECC error will be reported as a bus error to the requesting master. Both correctable and non-correctable ECC errors are reported to the MEMU.

Workaround: The application software must handle bus errors due to non-correctable ECC errors in the flash memory region 0x00F90000 – 0x00FBFFFF.

e11096: SAI: Internal bit clock is not generated when RCR2[BCI]=1 or TCR2[BCI]=1

Description: When the SAI transmitter or receiver is configured for internal bit clock with BCI = 1, the bit clock is not generated for either of the following two configurations:

a) SYNC = 00 and BCS = 0

b) SYNC = 01 and BCS = 1

Workaround: When the SAI transmitter or receiver is configured for internal bit clock with BCI=1, use only one of the following two configurations:

a) SYNC = 01 and BCS = 0

b) SYNC = 00 and BCS = 1

e11150: SAI: Internally generated receive or transmit BCLK cannot be re-enabled if it is first disabled when RCR2[DIV] or TCR2[DIV] > 0

Description: If the receive or transmit bit clock (BCLK) is internally generated, enabled with DIV > 0 and is then disabled, due to software or Stop mode entry, and the BCLK is enabled again, the clock is not generated.

Workaround: If the receive or transmit BCLK is internally generated and a DIV value greater than 0 is used, the SAI must be reset before the BCLK is re-enabled. This is achieved by writing the SR bit in the respective RCSR or TCSR register first to 1 and then immediately to 0.

e10815: STCU2: Offline LBIST execution will toggle GPIO[29] ,GPIO[30], GPIO[62] and GPIO[63]

Description: When the Self Test Control Unit (STCU2) is configured to execute Offline Logic Built-in Self Test (LBIST) GPIO[29] ,GPIO[30], GPIO[62] and GPIO[63] will toggle during the LBIST execution phase. All other GPIO are static during LBIST execution. Note MPC5748G 0N78S is factory programmed with STCU2 DCF records to execute offline MBIST (Memory Built-in Self Test) and LBIST by default. During MBIST execution all GPIO are static.

Workaround: The user should select the most appropriate workaround for offline LBIST:

1. Disable offline MBIST and LBIST. The STCU2_CFG DCF can be programmed with the NIL pointer. The user should program the following DCF {0x7F000000, 0x0008000C} = (STCU2_CFG data, STCU2_CFG DCF client address).

2. Disable offline LBIST only, MBIST will continue to execute. The STCU2_MB_CTRL75 can be programmed with the NIL pointer. The user should program the following DCF {0x7F000000, 0x0008072C} = (STCU2_MB_CTRL75 data, STCU2_MB_CTRL75 DCF client address).

3. The user can asses the impact of the GPIO[29] ,GPIO[30], GPIO[62] and GPIO[63] toggles to understand whether it will affect the application. If no impact the LBIST can execute.

Note: User DCF records should be programmed to the first erased memory location in the UTEST flash region beyond 0x00400_0300.

Note: Option (1) and (2) should not be used in safety-critical applications.

e10721: STCU: If the Auxiliary Clock 9 Select Control Register CGM_AC9_SC[SELCTL] = 1, offline-BIST will fail leading to a STCU watchdog timeout

Description: When STCU offline-BIST is enabled (default) and the Auxilliary Clock 9 Select Control Register CGM_AC9_SC[SELCTL] = 1 (default = 0) the offline-BIST will not complete leading to an STCU watchdog timeout. STCU watchdog timeout duration depends on STCU_WDG DCF record programmed before the STCU_RUN DCF record.

Workaround: To avoid the STCU offline-BIST failure leading to the STCU watchdog timeout the user must select 1 of the following workarounds:

1. If there is a requirement for CGM_AC9_SC[SELCTL] = 1 (FXOSC) the user can avoid STCU offline-BIST failure by disabling FlexCAN_0 MBIST by programming STCU_MB_CTRL40 DCF (0xBA000000000806A0). This will reduce BIST coverage as FlexCAN_0 MBIST is excluded.

2. If STCU offline-BIST is required to be executed without a reduction in coverage the user must select CGM_AC9_SC[SELCTL] = 0 (FS80).

3. If there is a requirement for CGM_AC9_SC[SELCTL] = 1 (FXOSC) the user can disable STCU offline-BIST by programming STCU_CFG DCF (0x7F0000000008000C).

e10577: STCU: Device may get stuck in reset if the Fast External Oscillator (FXOSC) is lost during self test

Description: If the fast external oscillator (FXOSC) is used as a reference clock for the PLL for online self test and the FXOSC is lost permanently while self test is in progress, the device will be stuck in the reset state.

Workaround: Use the Fast Internal Oscillator (FIRC) as the reference clock for the PLL when running online self test.

e10200: STM: Reading the System Timer Module Count register may return an incorrect value

Description: The erratum may only occur when the STM is configured to use the FXOSC clock source selected at STM_CR[CSL]. For the case application software reads the STM Count register (STM_CNT) the value returned may be incorrect. However, the user should be assured that STM interrupts will continue to be triggered at the expected STM_CNT value.

Note the default clock source for the STM is the FS80 (divided system clock) and this configuration is not impacted by this erratum.

Workaround: To avoid the erratum condition the user should select the FS80 clock source for the STM. However, for the case the FXOSC is required to clock the STM and the STM Count register is to be read, the following sequence must be executed:

1. Disable the STM via STM_CR[TEN]

2. Read STM Counter register STM_CNT

3. Re-enable the STM via STM_CR[TEN]

How to Reach Us:

Home Page:
nxp.com

Web Support:
nxp.com/support