# SPC574K70x, SPC574K72x Errata sheet

SPC574K70x, SPC574K72x device errata
JTAG_ID = 0x0AF07041

## Introduction

This errata sheet describes all the functional and electrical problems known in the revision 2.0 of the SPC574Kxx devices, identified with the JTAG_ID = 0x0AF07041.

All the topics covered in this document refer to SPC574Kxx reference manual (RM0334 rev 5) and SPC574Kxx datasheet rev 5 (see *A.1: Reference document*).

Device identification:

- Package device marking mask identifier: BA
- JTAG_ID = 0x0AF07041
- MIDR register:
  - MAJOR_MASK: 1
  - MINOR_MASK: 0

This errata sheet applies to SPC574Kxx devices in accordance with *Table 1*.

**Table 1. Device summary**

| Part number | Package |
|---|---|
| SPC574K70E5 | eTQFP144 |
| SPC574K72E5 | |
| SPC574K70E7 | eLQFP176 |
| SPC574K72E7 | |

# Contents

**1 Functional problems . . . . . 13**

1.1 ERR003881: MC_CGM: Core 2 must be used to change the clock ratio between the cores and the Cross Bar (XBAR) interfaces . . . . . 13

1.2 ERR003970: NAR: Trace messages include a 6-bit Source Identification field instead of 4-bits . . . . . 13

1.3 ERR004136: XOSC and IRCOSC: Bus access errors are generated in only half of non-implemented address space of XOSC and IRCOSC, and the other half of address space is mirrored . . . . . 14

1.4 ERR004697: TDM: Diary base address must be 16 KB aligned . . . . . 14

1.5 ERR005749: SDADC: New conversion data is discarded if the overflow (DFORF) status bit is set . . . . . 14

1.6 ERR005947: SARADC: ADC may miss a GTM trigger pulse if width of pulse is less than 1 AD Clk cycle . . . . . 15

1.7 ERR006350: LINFlexD: WLS feature cannot be used in buffered mode . 15

1.8 ERR006409: GTM: ATOM Force Update does not activate a comparison when in SOMC mode . . . . . 16

1.9 ERR006410: GTM: Write to ATOM_CH_CTRL sets WRF if CCU0 compare match has already occurred, but CCU1 compare match is pending, in ATOM SOMC mode . . . . . 16

1.10 ERR006411: GTM: Incorrect Input Signal Characteristics when the TIM channel is in TIEM, TPWM, TIPM, TPIM or TGPS mode, when ECNT is selected as the captured GPRi value. . . . . 17

1.11 ERR006412: GTM: Incorrect Input Signal Characteristics when the TIM channel is in TBCM mode and ECNT is selected as the captured GPR0 value. . . . . 17

1.12 ERR006538: LINFlexD: Stop mode request may be ignored if requested before the end of a frame . . . . . 18

1.13 ERR006639: GTM: A compare match event does not clear WR_REQ when ATOM is in SOMC mode . . . . . 18

1.14 ERR006640: GTM: Valid edge after Timeout event ignored by TIM . . . . 19

1.15 ERR006642: GTM: THVAL not available immediately after inactive trigger in DPLL . . . . . 19

1.16 ERR006643: GTM: Incorrect timestamp captured in CNTS when TIM operates in TPWM or TPIM modes if CMU_CLK is not equal to system clock . . . . . 20

1.17 ERR006644: GTM: Incorrect duty cycle in TOM PCM mode . . . . . 20

1.18 ERR006645: GTM: Clearing of DPLL PCM1/2 bits after the Missing Pulse Correction Values calculations delayed . . . . . 21
1.19 ERR006720: SIUL2: Logic state of LVDS input pads cannot be read via GPDI registers. . . . . 21
1.20 ERR006828: e200zx: Local Instruction and Data Memories are not accessible during corresponding e200z4/e200z7 core reset. . . . . 22
1.21 ERR006835: LFAST: Full 320Mbps may give bit errors . . . . . 22
1.22 ERR006850: SSCM: Nexus enable required for mode changes when a debugger is attached . . . . . 22
1.23 ERR006906: SDADC: Invalid conversion data when output settling delay value is less than 23 . . . . . 23
1.24 ERR006932: NAR: Nexus timestamps are not implemented for the Nexus clients in the e200zx cores . . . . . 23
1.25 ERR006967: eDMA: Possible misbehavior of a preempted channel when using continuous link mode . . . . . 23
1.26 ERR006994: XBIC: XBIC may trigger false FCCU alarm . . . . . 24
1.27 ERR007058: STCU2: Nexus interface of Peripheral Core_2 is not reset after an LBIST execution . . . . . 24
1.28 ERR007083: GTM: The DPLL's SORI, TORI, MTI, and MSI interrupts may not be asserted . . . . . 24
1.29 ERR007084: GTM: An active edge input, that is rejected by the DPLL trigger plausibility check, does not assert a Missing Trigger Interrupt . . . . . 25
1.30 ERR007085: GTM: A TIM timeout occurs when the TDU is re-enabled . 25
1.31 ERR007086: GTM: TIM PWM and PIM modes may capture the wrong timestamp . . . . . 26
1.32 ERR007087: GTM: The DPLL's Address Pointer Extension value is added to the Address Pointer when the Address Pointer Status bit is 0 . . . . . 26
1.33 ERR007088: GTM: When ATOM is in SOMP mode the SR0/SR1 registers could be updated twice in one PWM period . . . . . 27
1.34 ERR007099: FCCU: Error pin signal length is not extended when the next enabled fault, with its alarm timeout disabled, occurs . . . . . 27
1.35 ERR007103: MC_CGM: Incorrect cause for the latest clock source switch may be reported by the CGM if a safe mode request arrives when the system clock is the IRC . . . . . 28
1.36 ERR007109: I2C: In master receive mode, data remains latched in I2C data I/O register (IBDR) until new data is received . . . . . 28
1.37 ERR007115: DSPI: Mixing 16 and 32 bits frame size in XSPI Mode can cause incorrect data to be transmitted . . . . . 29

1.38 ERR007116: CRC: AutoSAR 4.0 8-bit CRC8 0x2F is not supported in hardware . . . . . 29
1.39 ERR007126: MEMU: Instead of Byte 1 of MEMU CTRL Register, Byte 3 is currently protected . . . . . 30
1.40 ERR007134: RCCU: If any accesses to the I-MEM or D-MEM of the safety and checker core are performed while the cores are disabled, the cores will get out of lockstep when enabled . . . . . 30
1.41 ERR007137: MEMU: incorrect indication when a correctable error is signaled by the e200zx core cache . . . . . 31
1.42 ERR007138: SARADC: Missed conversion after ABORT of the last channel of an injected chain . . . . . 31
1.43 ERR007185: SDADC: Watchdog Crossover event missed if PBRIDGEx_CLK less than SD_CLK . . . . . 32
1.44 ERR007190: GTM: Simultaneous Core and DPLL accesses to RAM Region 2 may lead to the DPLL reading erroneous data . . . . . 32
1.45 ERR007191: GTM: The DPLL's SORI and TORI interrupts are not asserted . . . . . 33
1.46 ERR007203: SENT: In debug mode SENT message data registers appear to lose contents . . . . . 34
1.47 ERR007204: SENT: Number of Expected Edges Error status flag spuriously set when operating with Option 1 of the Successive Calibration Check method . . . . . 34
1.48 ERR007211: MC_ME: Core register IAC8 is cleared during a mode change when the core is reset . . . . . 35
1.49 ERR007222: SARADC: Minimum value of precharge must be greater than or equal to 2 ADC clock cycles . . . . . 36
1.50 ERR007223: FCCU: FCCU_IRQ_EN register is writeable in all operating modes . . . . . 36
1.51 ERR007226: FCCU: the error-out signalling cannot be disabled in non Bi-stable protocols . . . . . 36
1.52 ERR007227: FCCU: FCCU Output Supervision Unit (FOSU) will not monitor faults enabled while already pending . . . . . 37
1.53 ERR007228: LINFlexD: Erroneous transmission in LIN master mode for payload greater than eight bytes . . . . . 38
1.54 ERR007234: PSI5: No transfer error generated for accesses within the unused range of the PSI5 peripheral window . . . . . 38
1.55 ERR007246: SARADC: First conversion after exit from stop mode may be corrupted . . . . . 38

1.56 ERR007259: e200zx: ICNT and branch history information may be incorrect following a nexus overflow . . . . . 39
1.57 ERR007269: LINFlexD: Erroneous timeout interrupt could be generated by LIN in master mode . . . . . 40
1.58 ERR007274: LINFlexD: Consecutive headers received by LIN Slave triggers the LIN FSM to an unexpected state . . . . . 40
1.59 ERR007297: LINFlexD: Response timeout values is loaded in LINOCR[OC2] field instead of LINOCR[OC1] . . . . . 41
1.60 ERR007305: e200zx: JTAG reads of the Performance Monitor Counter registers are not reliable . . . . . 41
1.61 ERR007325: FCCU: Unsuccessful decorated storage access may cause erroneous signaling of FCCU Channel 40 . . . . . 42
1.62 ERR007335: MEMU: ECC errors may be double reported when initiated by the Safety core to local memory of other cores . . . . . 42
1.63 ERR007339: STCU2: STCU2 fault injected by FCCU is self clearing . . . 42
1.64 ERR007352: DSPI: reserved bits in slave CTAR are writable . . . . . 43
1.65 ERR007356: SDADC: The SDADC FIFO does not function correctly when FIFO overwrite option is used . . . . . 43
1.66 ERR007360: FEC: Minimum VDD is 3.15 V instead of 3.0 V . . . . . 44
1.67 ERR007362: SDADC: Additional DMA request generated after single read access . . . . . 44
1.68 ERR007404: SENT: Message overflow in SENT Receiver can lead to stall condition in the MCU . . . . . 45
1.69 ERR007411: PBRIDGE: Incorrect transfer error information on accesses to reserved locations . . . . . 46
1.70 ERR007412: PBRIDGE: Incorrect transfer error information for accesses to TDM and FEC reserved locations . . . . . 46
1.71 ERR007414: PBRIDGE: Incorrect transfer error when accessing reserved locations of the Peripheral Bridge . . . . . 47
1.72 ERR007415: JTAG: PA[9] = JTAG TDO pad is not pull-up during reset . 47
1.73 ERR007416: MEMU: Flexray data RAM and GTM RAMs Error not correctly registered within MEMU during online MBIST . . . . . 48
1.74 ERR007425: SENT: Unexpected NUM_EDGES_ERR error in certain conditions when message has a pause pulse . . . . . 48
1.75 ERR007433: JTAGM: Nexus error bit is cleared by successful RWA . . . 50
1.76 ERR007498: M_(TT)CAN: Transmitted bit in control field is falsified when using extreme bit time configurations . . . . . 51
1.77 ERR007528: GTM: Action not always calculated immediately by DPLL . 51

1.78 ERR007529: GTM: TIM overflow bit is not set and the signal level bit has inverse value when sent to ARU in some cases . . . . . 52
1.79 ERR007530: GTM: New DPLL Position Minus Time data not received . . 53
1.80 ERR007531: GTM: DPLL Position Minus Time result is not sent to the ARU . . . . . 53
1.81 ERR007532: M_TTCAN: Incorrect value of Reference Trigger Offset status for time slaves . . . . . 54
1.82 ERR007538: M_(TT)CAN: Switch between CAN operating modes during transmission or reception may be ignored . . . . . 55
1.83 ERR007587: SSCM: Multi-bit ECC error at RCHW locations will cause device to remain in reset as a security and safety precaution . . . . . 55
1.84 ERR007589: LINFlexD: Spurious timeout error when switching from UART to LIN mode or when resetting LINTCSR[MODE] bit in LIN mode . . . . . 56
1.85 ERR007632: SENT: Incorrect rounding of the Status Nibble, Data Nibbles, and CRC Nibbles . . . . . 57
1.86 ERR007652: M_CAN: TTCAN triggers protect part of memory . . . . . 57
1.87 ERR007701: STCU2: Short Functional Reset reaction and Long Functional Reset reaction of the FCCU does not take effect upon PLL1 Loss-Of-Lock while MBIST ONLINE is running . . . . . 58
1.88 ERR007750: PAD_RING: Incorrect control of internal pull-up and pull-down on GPIO PB[5] and PE[14] . . . . . 58
1.89 ERR007772: PMC: a SWT_2 destructive reset may be forced if the 5 V supply is over voltage for longer than 16 ms . . . . . 59
1.90 ERR007788: SIUL2: A transfer error is not generated for 8-bit accesses to non-existent MSCRs . . . . . 59
1.91 ERR007791: SIUL2: Transfer error not generated if reserved addresses within the range of SIUL BASE + 0x100 to 0x23F are accessed . . . . . 60
1.92 ERR007796: M_CAN: Message reception and transmission directly after detection of Protocol Exception Event . . . . . 60
1.93 ERR007801: WKPU: functional NMI filter enable trigger FCCU fault monitor channel #47 . . . . . 61
1.94 ERR007824: DCI: Avoid asserting system reset when switching JTAG operating modes . . . . . 61
1.95 ERR007840: M_CAN: Change of operation mode during start of transmission . . . . . 62
1.96 ERR007841: M_CAN: Incorrect frame transmission after recovery from Restricted Operation Mode . . . . . 63
1.97 ERR007842: M_CAN: Erroneous Interrupt flag after setting / resetting INIT during frame reception . . . . . 64

1.98 ERR007847: GTM: MCS's CAT status may be incorrect . . . . . 64
1.99 ERR007848: GTM: Bit 0 of TIM edge counter register may not indicate the actual signal level . . . . . 64
1.100 ERR007855: SENT: Integer division during calibration pulse measurement causes reduced robustness . . . . . 65
1.101 ERR007869: FCCU: FOSU monitoring of a fault is blocked for second or later occurrence of the same fault . . . . . 65
1.102 ERR007873: SPC574KxB: Current injection causes leakage path across the DSPI and LFAST LVDS pins . . . . . 66
1.103 ERR007904: PASS: Programming Group Lock bit (PGL) can be de-asserted by multiple masters writing the correct password sections to the CINn registers. . . . . 67
1.104 ERR007905: PIT: Accessing the PIT by peripheral interface may fail immediately after enabling the PIT peripheral clock . . . . . 67
1.105 ERR007911: MC_ME: Decorated accesses not supported to core local memories during reset . . . . . 68
1.106 ERR007932: NAR/SIPI: Part ID for NAR and Debug SIPI does not match the MIDR MINOR_MASK . . . . . 68
1.107 ERR007934: FEC: MDC and MDIO timing requirements and configuration . . . . . 69
1.108 ERR007935: PMC: Accessing the HVD_FLASH divider tap point through the ADC test channel may cause a device reset . . . . . 69
1.109 ERR007947: XOSC: Incorrect external oscillator status flag after CMU event clear . . . . . 69
1.110 ERR007960: FCCU: Channels associated to MEMU cannot be activated through the fake fault register (FCCU_RFF) . . . . . 70
1.111 ERR007981: LBIST: LBIST of the flash may leave flash in an unknown state and stress flash bit cells . . . . . 71
1.112 ERR007996: PSI5: Incorrect SMC message decoding and timestamp generation in case of late last sensor message overlapping with next SYNC period pulse . . . . . 71
1.113 ERR008005: DSMC: Software reset of core0/core0s may trigger FCCU channel #11 event . . . . . 72
1.114 ERR008019: PMC: Escalation of LVD and HVD functional resets should be expected . . . . . 73
1.115 ERR008034: FCCU: Channel 40 fault (COMP_XBIC_DSMC_Monitor) may be reported if a CPU is reset upon a MC_ME mode change . . . . . 74
1.116 ERR008039: SDADC: digital filter and FIFO not disabled when MCR[EN] is cleared . . . . . 75

1.117 ERR008042: FCCU: EOUT signals are active, even when error out signaling is disabled . . . . . 75
1.118 ERR008054: PIT: DMA request stays asserted when initiated by PIT trigger, until PIT is reset . . . . . 76
1.119 ERR008056: LBIST: Flash must be idle during LBIST . . . . . 77
1.120 ERR008062: M_CAN: Frame transmission in DAR mode . . . . . 77
1.121 ERR008082: SENT: A message overflow can lead to a loss of frames combined with NUM_EDGES_ERR being set . . . . . 77
1.122 ERR008089: TDM: Diary updates require PECIE bit of MCR register of flash to be set . . . . . 78
1.123 ERR008117: MC_ME: Restrictions on enabling FlexRay in low power modes . . . . . 79
1.124 ERR008122: GTM: (A)TOM's CCU1 event interrupt is not generated when CM1=0 or 1 and RST_CCU0=1 . . . . . 79
1.125 ERR008131: SPC57BD1: Boundary scan of the interconnect between PD and BD is not available . . . . . 80
1.126 ERR008132: I2C: debug session request may not be correctly synchronized with I2C clock . . . . . 80
1.127 ERR008133: SIUL2: PC[2] reset configuration selects alternate function 5 (FCCU EOUT1 with strong push-pull buffer) . . . . . 81
1.128 ERR008145: MEMU: address registers in the uncorrectable error reporting tables can be written when the corresponding valid bit is not asserted . . 81
1.129 ERR008146: MEMU: ECC error syndrome is not transmitted to MEMU for system and core0 RAMs . . . . . 82
1.130 ERR008225: SDADC: FIFO Flush Reset command requires clearing the Data FIFO Full Flag . . . . . 82
1.131 ERR008229: FCCU: Enabling the programmable glitch filter on EIN may cause a destructive reset . . . . . 82
1.132 ERR008310: XBIC: Crossbar Integrity Checker may miss recording information from an initial fault event in the case of back-to-back faults . 83
1.133 ERR008314: SDADC: Double trigger is generated for conversion when SW trigger is connected to SDADC own HW trigger input . . . . . 83
1.134 ERR008325: MC_ME: Invalid clock configuration not detected for PSI5 peripherals during mode change . . . . . 84
1.135 ERR008343: DCI: EVTO[1:0] outputs remain stuck low if asserted while the system clock source is the IRC . . . . . 84
1.136 ERR008429: GTM: Unexpected TIM CNTS register reset in TPWM OSM mode . . . . . 85

1.137 ERR008430: GTM: DPLL sub-inc generation and action calculations are delayed . . . . . 85
1.138 ERR008438: GTM: Wrong signal level when TIM mode is changed from TBCM to any other mode . . . . . 85
1.139 ERR008439: GTM: TOM and ATOM CM0, CM1 and CLK_SRC register updates may not be triggered . . . . . 86
1.140 ERR008526: LINFlexD: LIN or UART state may be incorrectly indicated by LINSR[LINS] bitfield . . . . . 87
1.141 ERR008561: LINFlexD: Corruption of Tx data in LIN mode with DMA feature enabled . . . . . 87
1.142 ERR008573: LINFlexD: Pre-mature header/response timeout in LIN mode . . . . . 88
1.143 ERR008602: LINFlexD: Tx through DMA can be re-triggered after abort in LIN/UART modes or can prematurely end on the event of bit error with LINCR2[IOBE] bit being set in LIN mode . . . . . 88
1.144 ERR008631: SDADC: low threshold watchdog cannot be used with signed data . . . . . 89
1.145 ERR008640: TSENSE: Temperature sensor ADC readings are inaccurate during over temperature condition . . . . . 89
1.146 ERR008655: M_CAN: Setting the Configuration Change Enable (CCE) bit during a transmission scan can halt CAN transmissions . . . . . 90
1.147 ERR008673: LINFlexD: Module protocol clock must be greater than the eDMA clock to use eDMA transmit functionality . . . . . 91
1.148 ERR008688: GTM: Data lost in ATOM when CMU_CLKx is slower than ARU . . . . . 92
1.149 ERR008689: GTM: F2A stream data are not deleted after stream disabling . . . . . 92
1.150 ERR008730: XBIC: XBIC may store incorrect fault information when a fault occurs . . . . . 93
1.151 ERR008731: LINFlexD: Corruption of Received Rx data in UART mode 93
1.152 ERR008770: FlexRay: Missing TX frames on Channel B when in dual channel mode and Channel A is disabled . . . . . 94
1.153 ERR008884: XBAR: Masters on peripheral shell bus concentrator may stall or fetch data incorrectly . . . . . 95
1.154 ERR008904: M_CAN: Incorrect activation of MRAF interrupt . . . . . 96
1.155 ERR008915: SARADC: wrong behavior when aborting the conversion of a chain . . . . . 96

1.156 ERR008923: M_CAN: FD frame format not compliant to the new ISO/CD 11898-1: 2014-12-11 . . . . . 97
1.157 ERR008933: LINFlexD: Inconsistent sync field may cause an incorrect baud rate and Sync Field Error Flag may not be set . . . . . 97
1.158 ERR008935: JTAGM: write accesses to registers must be 32-bit wide . . 98
1.159 ERR008951: I2C: Attempting a start cycle while the bus is busy may generate a short clock pulse . . . . . 98
1.160 ERR008970: LINFlexD: Spurious bit error in extended frame mode may cause an incorrect Idle State . . . . . 99
1.161 ERR009003: FLASH: wrong address decoding error generation during Read-while-Write operation . . . . . 99
1.162 ERR009060: M_CAN: FD frame abort may cause Protocol exception event and extended Bus Integration state . . . . . 100
1.163 ERR009157: MC_ME: system frequency limitation when resetting IOP core . . . . . 100
1.164 ERR009158: MEMU: System RAM ECC errors on accesses by some masters report to MEMU Peripheral RAM table . . . . . 101
1.165 ERR009215: PAD_RING: Higher output impedance on PC[12] when Ethernet I/O segment is configured for 3.3 V supply . . . . . 101
1.166 ERR009409: PMC: low voltage core regulator sense monitoring through SARADC channel cannot be used . . . . . 102
1.167 ERR009413: M_CAN: SPC574KxB, Data loss when the storage of a received frame is not complete before EOF . . . . . 102
1.168 ERR009415: M_CAN: Message RAM / RAM arbiter not responding in time . . . . . 103
1.169 ERR009420: FCCU: FOSU may give destructive reset when a hardware recoverable fault of width less than one safe clock occurs . . . . . 103
1.170 ERR009436: DSPI: AC timing limitation on DSPI4/DSPI5 for some pins associated to the SCK and SOUT functionality . . . . . 104
1.171 ERR009599: DSPI: AC timing limitation on DSPI2/DSPI4/DSPI5 in LVDS mode . . . . . 105
1.172 ERR009656: DSPI: Frame transfer does not restart in case of SPI parity error in master mode . . . . . 105
1.173 ERR009658: SPI: Inconsistent loading of shift register data into the receive FIFO following an overflow event . . . . . 106
1.174 ERR009664: DSPI: Frame transfer does not restart in case of DSI parity error in master mode . . . . . 106

1.175 ERR009764: SARADC: DMA interface limitation depending on PBRIDGE/SARADC clock ratio . . . . . 107

1.176 ERR009799: SIUL2: Incorrect selection of alternate functions for some pins . . . . . 108

**Appendix A Further information . . . . . 109**

A.1 Reference document . . . . . 109

A.2 Acronyms . . . . . 109

**Appendix B Defect across silicon version . . . . . 112**

**Revision history . . . . . 120**

# List of tables

Table 1. Device summary . . . . . 1
Table 2. Erratum behavior of NUM_EDGES_ERR . . . . . 49
Table 3. Expected behavior, clarification of NUM_EDGES_ERR cases . . . . . 50
Table 4. Acronyms . . . . . 109
Table 5. Defects across silicon version . . . . . 112
Table 6. Document revision history . . . . . 120

# 1 Functional problems

## 1.1 ERR003881: MC_CGM: Core 2 must be used to change the clock ratio between the cores and the Cross Bar (XBAR) interfaces

**Description:**

The Peripheral I/O Processor (IOP/Core 2) is the only core that can successfully change the clock ratio between the cores and the Cross Bar (XBAR) interfaces (including the peripheral bridge frequencies). Therefore only core 2 should be used to modify the System Clock Divider Configuration Registers (CGM_SC_DC 0, 1, & 2). These registers should not be changed while Core 0, core 1 or the HSM (Hardware Security Module) is performing a bus transfer through the interface that is changing frequency.

**Workaround:**

If only Core2 is enabled by the Boot Assist Flash (BAF), then the CGM_SC_DC 0, 1, & 2 registers can be changed by core 2 prior to enabling the other cores, including the HSM, FlexRay, FEC (Fast Ethernet Controller), DMA (Direct Memory Access), and the SIPI (Serial Interprocessor Interface).

However, if the HSM, Core 0 or Core 1 is enabled in the BAF, then the following should be taken into account:

1. Switch clock ratios using only Core 2 and ensure Core 0 and Core 1 are not accessing the bus during the clock ratio switch.
2. If the HSM is enabled but idle, clock ratios between cores and the rest of the system can be changed using core 2 with core0 and core 1 disabled or idle.
3. The HSM must never run faster than the Slow XBAR. Therefore, one should change its clock ratio prior to change the Slow XBAR clock ratio. The HSM must be running from its local SRAM when the MCG register is being changed by core 2.
4. While writing the CGM_SC_DC_0, no access can be done on the Fast XBAR by any core (other than Core 2).
5. While writing the CGM_SC_DC_1, no access can be done on the Slow XBAR by any core (other than the Core 2).

## 1.2 ERR003970: NAR: Trace messages include a 6-bit Source Identification field instead of 4-bits

**Description:**

The source field (SRC) of trace messages from the Nexus Aurora Router are 6-bits in length. All other clients implement a 4-bit SRC field. Per the IEEE-ISTO 5001 Standard (Nexus) the SRC field of all clients on a device should be of the same length. The two most significant bits of the SRC are 0b00.

**Workaround:**

Tools should treat the SRC field as a 4-bit field for all Nexus clients. In addition, tools should ignore the extra 2-bits as an extra field with no meaning. In case of a NAR Error Message (TCODE=8), these two bits are between the 4-bit SRC field and the 4-bit Error Type

(ETYPE) field. For the NAR Watchpoint Message, these bits are between the 4-bit SRC and the 6- bit Watchpoint Hit (WPHIT) field.

## 1.3 ERR004136: XOSC and IRCOSC: Bus access errors are generated in only half of non-implemented address space of XOSC and IRCOSC, and the other half of address space is mirrored

**Description:**

Bus access errors are generated in only half of the non-implemented address space of Oscillator External Interface (40MHz XOSC) and IRCOSC Digital Interface (16MHz Internal RC oscillator [IRC]). In both cases, the other half of the address space is a mirrored version of the 1st half. Thus reads/writes to the 2nd half of address space will actually read/write the registers of corresponding offset in the 1st half of address space.

**Workaround:**

Do not access unimplemented address space for XOSC and IRCOSC register areas OR write software that is not dependent on receiving an error when access to unimplemented XOSC and IRCOSC space occurs.

## 1.4 ERR004697: TDM: Diary base address must be 16 KB aligned

**Description:**

In the Tamper Detection module (TDM), the Diary base address must be 16 KByte aligned. Assigning the Diary Base Address to a non 16 KByte will cause the diary to behave as if the base address is 16 KB aligned with respect to the different diary region and the diary select.

**Workaround:**

Set the base address of the Tamper Detect Module Diary to a 16 KB aligned region.

## 1.5 ERR005749: SDADC: New conversion data is discarded if the overflow (DFORF) status bit is set

**Description:**

The Sigma-Delta Analog-to-Digital Converter (SDADC) stops filling the Converted Data Register (CDR) and FIFO (if enabled) when the FIFO overrun bit in the Status Flag Register (SFR[DFORF]) becomes set.

The SFR[DFORF] bit becomes set when a FIFO or CDR overflow condition occurs, and once this happens the SDADC stops filling the CDR and FIFO, causing new converted data results to be discarded until the software clears the overflow bit. After clearing the overflow, normal operation resumes.

**Workaround:**

If the Global DMA/Interrupt gating feature is not being used then the problem can be avoided by either of two methods.

1. If DMA is being used then continuously transferring the CDR or FIFO data via DMA will prevent overflows.
2. If DMA is not used then use an interrupt service routine (ISR) to clear the overflow bit (SFR[DFORF]) and empty the FIFO/CDR

If the Global DMA/Interrupt gating feature is being used then use an interrupt service routine at the start of the DMA gating window to clear the DFORF bit and empty the FIFO and CDR. This will resume normal filling of the CDR. or FIFO which should then be serviced using DMA or ISR for the duration of the gating window.

## 1.6 ERR005947: SARADC: ADC may miss a GTM trigger pulse if width of pulse is less than 1 AD Clk cycle

**Description:**

The Successive Approximation Register Analog to Digital Converter (SARADC) may miss a trigger (and no conversions will be started) from the Generic Timer Module (GTM) if the pulse width from the GTM Timer Output Module (TOM) or Advanced Routing Unit (ARU) Connected TOM (ATOM) is less than one ADC clock. The GTM Counter Compare Unit registers(CM0/CM1) set the pulse width.

**Workaround:**

The GTM registers Counter Compare Unit registers (CM0 and CM1) should be appropriately programmed such that pulse width of trigger pulses is always greater than one (1) ADC clock cycle.

## 1.7 ERR006350: LINFlexD: WLS feature cannot be used in buffered mode

**Description:**

The Flexible Local Interconnect Network (LINFlex) module may not operate correctly if the Special Word Length (WLS) for enabling 12-bit data length is selected in the Universal Asynchronous Receiver/Transmitter (UART) Mode Control Register (UARTCR) and the module is configured in the receive buffered mode.

**Workaround:**

When WLS mode is required, always use the First In, First Out (FIFO) mode of the UART LINFLEX module by setting the Receive FIFO/Buffer mode bit of the UARTCR (UARTCR[RFBM]=1). In addition, the UART word length bits must be set (UARTCR[WL0 = 0b1] and UARTCR[WL1] = 0b1).

## 1.8 ERR006409: GTM: ATOM Force Update does not activate a comparison when in SOMC mode

**Description:**

(GTM-IP-139)

When the Generic Timer Module (GTM) ARU Connected Timer Output Unit (ATOM) is configured in Signal Output Mode Compare (SOMC) mode, with the Advanced Routing Unit (ARU) Enabled, and if no comparison is active (no valid data was received by ARU, ATOM[i]_CH[x]_STAT.B.DV = 0) a Force Update Request can cause the ATOM to remain in a state waiting for the update event to happen.

A Force Update is requested by first setting CPU Write Request field (WR_REQ) in ATOM[i]_CH[x]_CTRL, then updating the Shadow Counter Registers (ATOM[i]_CH[x]_SRx) and optionally the ATOM Mode Control Bits (ACB) in ATOM[i]_CH[x]_CTRL, and finally updating the Counter Registers (ATOM[i]_CH[x]_CMx) via a Forced Update (ATOM[i]_TGCx_FUPD_CTRnL = 0b11).

Under the above conditions:

- The registers CMx are updated correctly but no new comparison is activated
- The ACBO bits are erroneously not cleared
- The ARU read request is canceled because of WR_REQ=1

**Workaround:**

After the Forced Update, re-write the desired values to CM0 or CM1 to activate the comparison and to reset the ACBO bits.

## 1.9 ERR006410: GTM: Write to ATOM_CH_CTRL sets WRF if CCU0 compare match has already occurred, but CCU1 compare match is pending, in ATOM SOMC mode

**Description:**

(GTM-IP-140)

When the Generic Timer Module (GTM) ARU Connected Timer Output Unit (ATOM) is configured in Signal Output Mode Compare (SOMC) mode, with the Advanced Routing Unit (ARU) Enabled, the capture/compare strategy is 'Serve Last' (ATOM[i]_CH[x]_CTRL.B.ACB42 = 0b1xx), with Counter Compare Unit 0 (CCU0) matched but the CCU1 match is pending, a write access to the ATOM Channel Control register (ATOM[i]_CH[x]_CTRL) will set the "Write Request of CPU Failed for the Last Update" (WRF) field in the Channel Status Register (ATOM[i]_CH[x]_STAT) independently of the status of the CPU Write Request Bit for Late Compare field (WR_REQ) in ATOM[i]_CH[x]_CTRL.

**Workaround:**

If ATOM[i]_CH[x]_CTRL is written during an active 'Serve Last' comparison without the intention of setting WR_REQ, write to ATOM[i]_CH[x]_CTRL and then reset/clear WRF in ATOM[i]_CH[x]_STAT by writing a '1'.

## 1.10 ERR006411: GTM: Incorrect Input Signal Characteristics when the TIM channel is in TIEM, TPWM, TIPM, TPIM or TGPS mode, when ECNT is selected as the captured GPRi value.

**Description:**

(GTM-IP-141)

When a Generic Timer Module (GTM) Timer Input Module (TIM) channel captures a valid rising edge event at TIM[i]_CH[x]_FOUT (post filtering) in TIM Input Event Mode (TIEM), TIM PWM Measurement Mode (TPWM), TIM Input Prescaler Mode (TIPM), TIM Pulse Integration Mode (TPIM) or TIM Gated Periodic Sampling Mode (TGPS), the captured values of the Edge Counter (TIM[i]_CH[x]_ECNT) to the General Purpose Registers (TIM[i]_CH[x]_GPRi) are incorrect. The captured value will be ECNT+2; bit 0 (signal level) will be 0 (Falling Edge). The correct operation would be to capture ECNT+1; bit 1 (signal level) would be 1 (Rising Edge).

This leads to an inconsistency between the ARU signal level bit, bit 0 of the ARU word which shows the captured ECNT value, and TIM[i]_CH[x]_GPRi which shows an inconsistency when comparing GPRi [bits 31:24] to ECNT [bits 7:0].

**Workaround:**

When using the TIMs captured data the correct data can be reconstructed by:

if ARU_SIGNAL_LEVEL ==1 and ARU_DATA[0] == 0 the ARU_DATA = ARU_DATA -1;

When reading TIM[i]_CH[x]_GPRi by the data can be corrected as long as there is no GPR overflow and no new edge by:

if TIM[i]_CH[x]_GPRi[24] == 1 and TIM[i]_CH[x]_GPRi[0] == 0 then
TIM[i]_CH[x]_GPRi[23:0] = TIM[i]_CH[x]_GPRi[23:0] -1;

## 1.11 ERR006412: GTM: Incorrect Input Signal Characteristics when the TIM channel is in TBCM mode and ECNT is selected as the captured GPR0 value.

**Description:**

(GTM-IP-142)

When a Generic Timer Module (GTM) Timer Input Module (TIM) channel captures an input pattern match condition in TIM Bit Compression Mode (TBCM), the values captured by the General Purpose Register TIM[i]_CH0_GPR0 are incorrect under the condition EGPR0_SEL=1 with GPR0_SEL= 0 (use ECNT as input).

Starting at t=0 with counter value ECNT(t=0), the captured values of two consecutive edges can be ECNT(t=0)+2 followed by ECNT(t=0)+2 instead of ECNT(t=0)+1 followed by ECNT(t=0)+2. The captured ECNTs do not increment by 1 as expected, and reading TIM[i]_CH0_GPR0 shows an inconsistency when comparing ECNT [bits 31:24] to GPR0 [bits 7:0].

**Workaround:**

Ignore the captured data via the Advanced Routing Unit (ARU) and construct the edge count value with an independent Multi Channel Sequencer (MCS) counter which increments on each ARU transfer, or when reading TIM[i]_CH0_GPR0 use only TIM[i]_CH0_GPR0[31:24] as EDGE counter; do not use TIM[i]_CH0_GPR0[23:0].

## 1.12 ERR006538: LINFlexD: Stop mode request may be ignored if requested before the end of a frame

**Description:**

The LINFlexD module fails to enter stop mode when the stop mode request is issued before the second data byte of an ongoing frame transfer.

User requests stop mode by setting the appropriate bit of the Peripheral Control Register of the Mode Entry Module (ME_PCTLx).

**Workaround:**

If a LIN transmission/reception is in progress, wait until it reaches the frame boundary (complete current frame transfer) before sending a request to enter in stop mode.

## 1.13 ERR006639: GTM: A compare match event does not clear WR_REQ when ATOM is in SOMC mode

**Description:**

(GTM-IP-146)

If a Generic Timer Module (GTM) ARU Connected Timer Output Module (ATOM) channel is

operating in Signal Output Mode Compare (SOMC) mode, with the Advanced Routing Unit (ARU) enabled and a late compare register update is requested by the CPU by setting WR_REQ (CPU Write request bit for late compare register update in ATOM[i]_CH[x]_CTRL), a late update of the Counter Compare Unit (CCU0/1) Compare registers (CM0/CM1) and/or the compare strategy (i.e. ARU Control Bits (ACB) bits altered) is successfully done, but after final compare match the WR_REQ bit is not reset. As a result no new ARU read request is set up after final compare match.

**Workaround:**

CPU should clear WR_REQ by software after the late update.

## 1.14 ERR006640: GTM: Valid edge after Timeout event ignored by TIM

**Description:**

(GTM-IP-150)

When a Generic Timer Module (GTM) Timer Input Module (TIM) timeout event triggers an Advanced Routing unit (ARU) write request with timeout information "timeout detected without valid edge" (ARU Control Bit 2 (ACB2)=1 and ACB1=0), and this request is acknowledged by the ARU at the same time as a new valid edge occurs, the valid edge is neither acknowledged by setting the bits ACB2=1 and ACB1=1 (timeout detected with subsequent valid edge detected) within the acknowledged transfer nor acknowledged by setting up a subsequent ARU write request for the new valid edge with ACB2=0 and ACB1=0 (valid edge detected).

**Workaround:**

The workaround for this issue requires an additional plausibility check within the Multi Channel Sequencer (MCS) or CPU via FIFO:

Step 1) store the received data (ARUDATA(47:0)) and ACB0 in temporary variables.

Step 2) if an ARU transfer with ACB2=1 and ACB1=0 is received also check whether the previously received ARUDATA(47:0) and ACB0 values are the same:

If they are not the same values then a timeout with subsequent valid edge has occurred, which means ACB1 must be corrected to 1.

## 1.15 ERR006642: GTM: THVAL not available immediately after inactive trigger in DPLL

**Description:**

(GTM-IP-152)

The Generic Timer Module (GTM) Digital PLL (DPLL) Measured TRIGGER hold time value

(THVAL) is calculated correctly for each INVALID trigger slope, but this value is only stored into the THVAL memory location in RAM Region 1a with every new ACTIVE edge of the trigger signal.

**Workaround:**

If the THVAL value is needed immediately with the inactive trigger edge it is necessary to calculate the THVAL value by using Timer Input Module Channels 0/1 (TIM_CH0/1) to obtain the active and inactive slopes in TIM input event mode (TIEM). With these timestamps the CPU is able to calculate the time span.

## 1.16 ERR006643: GTM: Incorrect timestamp captured in CNTS when TIM operates in TPWM or TPIM modes if CMU_CLK is not equal to system clock

**Description:**

(GTM-IP-153)

When a Generic Timer Module (GTM) Timer Input Module (TIM) channel operates in the following configuration, the Timebase Unit Channel 0 (TBU_TS0) value is not captured correctly into the counter register CNTS:

- PWM Measurement Mode (TPWM) or Pulse Integration Mode (TPIM) and
- TBU_TS0 selected as the input to CNTS (CNTS_SEL = 1 in TIM[i]_CH[x]_CTRL) and
- Selected clock (CMU_CLKn) of the TIM channel is not the same frequency as the system clock.

**Workaround:**

Configure the TIM channel to operate on a CMU_CLK (Divider =1) which is identical to the system clock when TBU_TS0 is to be captured in TPWM or TPIM modes.

## 1.17 ERR006644: GTM: Incorrect duty cycle in TOM PCM mode

**Description:**

(GTM-IP-154)

The Generic Timer Module (GTM) Timer Output Module (TOM) duty cycle output in Pulse Count Modulation (PCM) mode is always one count less than the configured value in the Compare Match 1 (CM1) register. For example, if the value 1 is written to CM1, a duty cycle of 0% will be generated and if a duty cycle of 100% was configured by writing the maximum value (0xFFFF) in to the CM1 register, the resultant waveform would be a signal with duty cycle less than 100%. A value of zero in the CM1 register results in 100 % duty cycle.

**Workaround:**

Configure the CM1 value for the targeted duty cycle in the CM1 register with target duty cycle + 1.

To get 0 % duty cycle, CM1 = 1.

To get 100 % duty cycle, CM1 = 0 and CM0 = 0xFFFF. (Setting CM0 = 0x1000 and CM1 = 0xFFFF will also result in a duty cycle of 100 %)

## 1.18 ERR006645: GTM: Clearing of DPLL PCM1/2 bits after the Missing Pulse Correction Values calculations delayed

**Description:**

(GTM-IP-158)

The Generic Timer Module (GTM) Digital PLL (DPLL) Pulse Correction Mode bits (PCM1/2 in DPLL_CTRL_1) are expected to be cleared after the Missing Pulse Correction Values (MPVAL1/2) are used to calculate the number of sub_incs for the next increment and to calculate the add_in values by the DPLL State Machine, however the PCM1/2 bit is transferred with an active edge into the dedicated shadow registers, but cleared some time later. If the PCM1/2 bits are written by the CPU in between the point of time of the transfer to the shadow register and the point of time were the PCM1/2 bits are cleared, the bits are cleared and never used.

**Workaround:**

Do not allow the CPU to write to the PCM1/2 bits until at least 750 system clocks have passed since the previous Trigger Active Slope Detected (TASI) interrupt. This time could be derived by a GTM resource like an ARU connected TOM (ATOM) channel.

## 1.19 ERR006720: SIUL2: Logic state of LVDS input pads cannot be read via GPDI registers.

**Description:**

When two adjacent pads are configured as LVDS inputs, the associated System Integration Unit Lite (SIUL2) General Purpose Data Input (GPDI) registers for each pin are still connected to the CMOS input buffer. Therefore, the actual state of the LVDS input cannot be read since the LVDS logic levels don't align to TTL, CMOS, or Automotive input levels. After settling, these GPDI registers will always read zero when the associated pads are configured as LVDS inputs.

**Workaround:**

There is no work-around available to read the LVDS logic state for diagnostic purposes on this revision of the device. In future revisions, the GPDI register for the lowest numbered port of the LVDS pair will reflect the LVDS logic state of the pin when in LVDS input mode. The SIUL2 Multiplexed Signal Configuration Register Input Buffer Enable (MSCR[IBE]) bit for the associated port must be enabled to read the logic state, in addition to selecting LVDS inputs with the Input Level Selection (MSCR[ILS]) field.

## 1.20 ERR006828: e200zx: Local Instruction and Data Memories are not accessible during corresponding e200z4/e200z7 core reset.

**Description:**

Local Instruction Memory (IMEM) and Data Memories (DMEM) are not accessible while its corresponding e200z4 / e200z7 core is in reset. Write accesses from cores not in reset will be ignored and read accesses will return all zeros (0x0) in the data and Error Correction Code checker bits. This will result in an End-to-End Error Correction Code (E2E ECC) error. The issue occurs when an e200z core reset is triggered by writing to the Mode Entry Core Control register (MC_ME.ME_CCTL0,1,3) RESET bit.

**Workaround:**

Applications should not access the IMEM and DMEM memories for e200z4 / e200z7 cores that are being held in reset via the ME_CCTLx registers.

## 1.21 ERR006835: LFAST: Full 320Mbps may give bit errors

**Description:**

The Low Voltage Differential Signaling (LVDS) Fast Asynchronous Serial Transmission (LFAST) output voltages are not balanced across the full process, voltage, and temperature ranges and are affected by the LFAST LVDS Control Register (LCR) LVDS Data Select (LVCKSS) and LVDS Clock Select (LVCKP) configurations. High bit error rates may be experienced at core supply voltages other than 1.20 V when LVCKSS is set.

**Workaround:**

Clear LVCKSS to 0 when using LFAST communications. A small bit error rate may occur when using full 320 MHz operation. This error rate will decrease with reduced bit rate and should be eliminated completely at 200 Mbps and slower.

## 1.22 ERR006850: SSCM: Nexus enable required for mode changes when a debugger is attached

**Description:**

If the Nexus interface is enabled in the e200zx cores, even if trace (program, data, ownership, watchpoint, data acquisition) is not enabled, the Nexus Aurora Router (NAR) must be enabled to allow mode changes via the Mode Entry module if debug mode is enabled (debugger connected to the MCU). In addition, trace options must be set to guarantee that none of the NAR trace buffers become full and to prevent new messages from being input into the NAR since some Nexus trace messages are automatically generated regardless whether any trace mode is disabled. Nexus is enabled in the core if any Nexus feature is accessed by a tool (executing the Nexus_enable command to use the Nexus Read/Write Access feature to access memory).

**Workaround:**

The Nexus enable bit (NEN) must be set in the NAR Control Register (NAR_CR[NEN]=0b1) when a debugger is connected to allow messages to exit the core Nexus module.

## 1.23 ERR006906: SDADC: Invalid conversion data when output settling delay value is less than 23

**Description:**

In the Sigma Delta Analog to Digital Converter (SDADC), if the Output Settling Delay field of the Output Settling Delay register (OSDR[OSD]) is programmed to a value less than 23 then the initial converted data from SDADC block is "0000" instead of the correct conversion result.

**Workaround:**

Program the OSDR[OSR] value equal to or greater than 23.

## 1.24 ERR006932: NAR: Nexus timestamps are not implemented for the Nexus clients in the e200zx cores

**Description:**

The Nexus clients in the e200zx cores do not implement timestamp packets on the Nexus trace messages.

**Workaround:**

Do not expect timestamps on the Nexus messages from the core, even though timestamps are enabled in the device.

## 1.25 ERR006967: eDMA: Possible misbehavior of a preempted channel when using continuous link mode

**Description:**

When using Direct Memory Access (DMA) continuous link mode Control Register Continuous Link Mode (DMA_CR[CLM]) = 1) with a high priority channel linking to itself, if the high priority channel preempts a lower priority channel on the cycle before its last read/write sequence, the counters for the preempted channel (the lower priority channel) are corrupted. When the preempted channel is restored, it continues to transfer data past its "done" point (that is the byte transfer counter wraps past zero and it transfers more data than indicated by the byte transfer count (NBYTES)) instead of performing a single read/write sequence and retiring.

The preempting channel (the higher priority channel) will execute as expected.

**Workaround:**

Disable continuous link mode (DMA_CR[CLM]=0) if a high priority channel is using minor loop channel linking to itself and preemption is enabled. The second activation of the preempting channel will experience the normal startup latency (one read/write sequence + startup) instead of the shortened latency (startup only) provided by continuous link mode.

## 1.26 ERR006994: XBIC: XBIC may trigger false FCCU alarm

**Description:**

The Crossbar Integrity Checker (XBIC) will incorrectly signal a fault alarm when a system bus request results in a bus error termination from a crossbar client. The Fault Correction and Collection Unit (FCCU) alarm number 40 (for XBIC_0) or number 35 for (XBIC_1) will be signaled.

**Workaround:**

Software should handle faults on FCCU alarm #35 and alarm #40 in case of a system bus error.

## 1.27 ERR007058: STCU2: Nexus interface of Peripheral Core_2 is not reset after an LBIST execution

**Description:**

After completion of an off-line logic built-in self-test (LBIST) via the Self Test Control Unit (STCU2), the Nexus interface of the Peripheral Core_2 is not part of the reset module and may remain in an unknown state.

This will have no effect when debug interface is not used and the associated JCOMP input of the JTAG interface is tied low. It may instead prevent correct code execution in case debug session is active.

**Workaround:**

When debugging the MCU, reset the Nexus interface after an off-line self-test completion via JCOMP reset input: toggle JCOMP pin from high to low and then back to high.

## 1.28 ERR007083: GTM: The DPLL's SORI, TORI, MTI, and MSI interrupts may not be asserted

**Description:**

(GTM-IP-161)

The Generic Timer Module (GTM) Digital PLL (DPLL) Missing Trigger(MTI), Missing State (MSI), Trigger Out Of Range (TORI) and State Out of Range (SORI) interrupts are not set (1) when the 3 following conditions are met:

- The upper 24 bits of Timebase Timestamp Channel 0 (TBU_TS0) are used as the input (DPLL_STATUS[LOW_RES]=1)
- The trigger/state input time stamps have an 8 times higher resolution than TBU_TS0 (DPLL_CTRL1[TS0_HRT/S] = 1)
- The upper three bits of TBU_TS0 are not equal to "000".

The DPLL Lock Status bits for SUBINC1 and SUBINC2 (LOCK1/2) and the MTI/MSI flags in the DPLL_STATUS register are incorrect if the above case occurs.

**Workaround:**

Do not use the configuration DPLL_STATUS[LOW_RES]=1 with DPLL_CTRL1[TS0_HRT/S] = 1.

## 1.29 ERR007084: GTM: An active edge input, that is rejected by the DPLL trigger plausibility check, does not assert a Missing Trigger Interrupt

**Description:**

(GTM-IP-162)

The Generic Timer Module (GTM) Digital PLL (DPLL) Missing Trigger Interrupt (MTI) is not

asserted during a gap in the trigger profile, stored in RAM region 2c, when an active input signal edge is rejected by the Plausibility Value of Trigger (PVT) check.

In this case, the DPLL's internal check for MTI is performed when the invalid active input occurs.

The check for the first and valid active inputs is not done for this gap. As a result, when monitoring the DPLL synchronization using the MTI in a gap, the application may report a synchronization problem which is not real.

**Workaround:**

The activated PVT check is reported by the activation of the Plausibility Window Violation of TRIGGER (PWI) interrupt. This interrupt can be used to check if a gap condition in the profile has occurred. This information can be used to correct the incorrect synchronization information out of the DPLL.

## 1.30 ERR007085: GTM: A TIM timeout occurs when the TDU is re-enabled

**Description:**

(GTM-IP-163)

The Generic Timer Module (GTM) Timer Input Module (TIM) Timeout Detection Unit (TDU)

indicates a timeout event when it is re-enabled and the new TDU Timeout Value (TOV) is lower than the previous TOV.

After stopping the TDU, the Time Out Counter (TO_CNT) field will have an arbitrary value less than or equal to (<=) the timeout value. When TOV is reconfigured to a value less than or equal to (<=) TO_CNT, and the TDU re-enabled an incorrect timeout is signaled. This is because if at the same time as the TDU is enabled the selected clock has an active edge, TO_CNT is greater than or equal to (>=) TOV because TO_CNT is not reset to zero.

**Workaround:**

If the TDU needs to be changed to a TOV value which is less than the previous value either:

1. Postpone disabling the TDU until the TO_CNT is less than (<) the new TOV. Then disable the TDU, configure TOV, and re-enable TDU Unit.
2. Disable the TDU and then write TOV with the value 0xFF. Then enable the TDU unit and reconfigure TOV to the desired value.

## 1.31 ERR007086: GTM: TIM PWM and PIM modes may capture the wrong timestamp

**Description:**

(GTM-IP-164)

When the Generic Timer Module (GTM) Timer Input Module (TIM) channel is configured with Counter Select (CNT_SEL) of the Channel Control Register (CTRL) set, and an input edge is detected by that channel before a rising edge on the clock, the Channel Counter Shadow Register (CNTS) will capture the value of Channel Count Register (CNT) instead of the timebase (TBU_TS0) in PWM measurement mode (TPWM) and pulse integration mode (TPIM).

**Workaround:**

To avoid incorrect timestamp captures in the TIM PWM and PIM mode, the following steps must be taken:

1. Select a TIM clock source which is identical to SYS_CLK.
2. Use the input event mode (TIEM) to capture TBU_TS0 for rising and falling input edges.
3. In TPWM mode, use CNT register as input (CNTS_SEL=0) with CMU_CLK source selected.

With TBU_TS0 selected as the input to General Purpose Register 0 (GPR0), and with CNT

selected as the input to General Purpose Register 1 (GPR1), calculate the correct timestamp: GPR0 - GPR1 + CNTS.

## 1.32 ERR007087: GTM: The DPLL's Address Pointer Extension value is added to the Address Pointer when the Address Pointer Status bit is 0

**Description:**

(GTM-IP-166)

If the Generic Timer Module (GTM) Digital PLL (DPLL) Address Pointer Extension field

(APT_2b_EXT/APS_1c2_EXT) in the Address Pointer Trigger/State Synchronization Register (DPLL_APT_SYNC) is not zero during synchronization, it is added to the Address Pointer for Trigger/State (APT_2b/APS_1c2) in DPLL_APT/DPLL_APS registers regardless of the state of the status bits (APT_2b_STATUS/APS_1c2_STATUS) in DPLL_APT_SYNC.

**Workaround:**

If the pointers should remain unchanged after synchronization, APT_2b_EXT/APS_1c2_EXT must be set to zero before the synchronization is performed.

## 1.33 ERR007088: GTM: When ATOM is in SOMP mode the SR0/SR1 registers could be updated twice in one PWM period

**Description:**

(GTM-IP-167)

When the Generic Timer Module (GTM) Advanced Routing Unit (ARU) Connected Timer Output Module (ATOM) is in Signal Output Mode PWM (SOMP) mode with the ARU enabled, and the channel is configured to be updated by the preceding channel (ARU_EN=1 and RST_CCU0=1 in ATOM[i]_CH[x]_CTRL), an update of the channel's Shadow Registers (ATOM[i]_CH[x]_SR0/SR1) via ARU is requested when Counter 0 (ATOM[i]_CH[x]_CN0) reaches Compare 0 (ATOM[i]_CH[x]_CM0).

In this case, if CN0 reaches CM0, CN0 is not reset and continues counting until it is reset by the trigger of the preceding channel. As a result, the ATOM channel updates the SR0/SR1 registers after the update of CM0/CM1, which is not synchronous with the counter reset. Depending on the time between CM0 and the value of CN0 when it was reset by the trigger, the SR0/SR1 registers may be updated twice in one period.

**Workaround:**

If new data via ARU is provided by Multi Channel Sequencer (MCS) ensure, through software, that only one value per period of new data for SR0/SR1 register can be read. For example, this can be achieved by starting a 'master period' which triggers the reset of CN0 on a time base value and provides the start value and period to the MCS. The MCS can then calculate a minimum time it must wait before providing new ARU data.

If the above workaround is unsuitable (for example if new data via the ARU is provided by the FIFO) do not use SOMP mode with ARU_EN=1 and RST_CCU0=1.

## 1.34 ERR007099: FCCU: Error pin signal length is not extended when the next enabled fault, with its alarm timeout disabled, occurs

**Description:**

In the Fault Collection and Control Unit (FCCU), when the following conditions are met:

- Two faults occur
- The second fault arises with a delay (T_delay) from the first error
- The second fault has its alarm timeout disabled
- T_delay is lower than the FCCU error pin minimum active time (T_min, defined in the Delta T register (FCCU_DELTA_T))

Then the error output signal is not extended and its duration is only T_min, if the faults are cleared before the timer expires.

The expected behavior is to have the error output signal duration of T_min + T_delay, if the faults are cleared before the timer expires.

**Workaround:**

Take into account that the error out signal duration will only be T_min, if the faults are cleared before the timer expires.

The timer count is meaningful only when the Error pin is driven low, which can be checked by reading the pin status FCCU_STAT[ESTAT].

## 1.35 ERR007103: MC_CGM: Incorrect cause for the latest clock source switch may be reported by the CGM if a safe mode request arrives when the system clock is the IRC

**Description:**

If the current system clock source is the Internal RC oscillator (IRC) as reported in the Clock Generation Module System Clock Select Status Register System Clock Source Selection field (CGM_SC_SS.SELSTAT = 0b0000) and the Clock Generation Module System Clock Select Status Register Switch Trigger Cause shows the cause for the latest clock switch as MC_ME succeeded (CGM_SC_SS.SWTRG = 0b001) indicating that a successful Mode Entry mode change was the cause of the last clock change then the CGM_SC_SS.SWTRG will incorrectly continue to show the cause for the latest clock switch as MC_ME succeeded after a safe mode request is generated. If a subsequent safe mode request is generated CGM_SC_SS.SWTRG switches to report the correct status value of 0b100 (switch to system clock source 0 due to SAFE mode request or reset succeeded).

**Workaround:**

If the CGM_SC_SS.SELSTAT shows the system clock as IRC (0b0000), then software should check the Mode Entry Global Status register Current Mode field (ME_GS.CURRENT_MODE) and the Mode Entry Interrupt Status Register Safe mode Interrupt (ME_IS.I_SAFE) to establish the cause of the switch.

## 1.36 ERR007109: I2C: In master receive mode, data remains latched in I2C data I/O register (IBDR) until new data is received

**Description:**

When the Inter-Integrated Circuit (I2C) is configured in master mode and receiving data from a slave which is transmitting data bytes on an irregular basis, there is no way for the master to know if the data received in the I2C data Input/Output register (IBDR) is the old latched data or the new data received from the slave.

**Workaround:**

When slave is configured to transmit data on an irregular basis, in other words intermittently, it should not send 2 consecutive bytes with the same data. When 2 consecutive data bytes are different, a dummy read of the I2C data I/O register (IBDR) can be initiated before the actual read. These 2 bytes can be compared to know if it is the new data or the old data.

## 1.37 ERR007115: DSPI: Mixing 16 and 32 bits frame size in XSPI Mode can cause incorrect data to be transmitted

**Description:**

The Deserial Serial Peripheral Interface (DSPI) features an Extended SPI mode (XSPI) supporting frames of up to 32 bits.

When the XSPI Mode is enabled, transferring a mixture of frames having a size up to 16 bits and those having size above 16 bits can cause an incorrect data transmission to occur. This happens when the First In/First Out (FIFO) queue read pointers roll-over and a frame needs to be extracted from both the bottom of the FIFO and the top of the FIFO when the Frame Size is greater than 16 bits.

**Workaround:**

Even number of Transmit FIFO Register (TXFR) registers:

Do not mix frames that have data size less than 16 bits with those having a size more than 16 bits in XSPI Mode.

Odd number of TXFR registers:

Do not mix frames that have data size less than 16 bits with those having a size more than 16 bits in XSPI Mode.

If the frame size is greater than 16, initially send a dummy frame (a frame with no chip select, but containing data) of less than or equal to 16 bits. Continue sending a dummy frame after each (number of TXFR Registers - 1) / 2 frames.

## 1.38 ERR007116: CRC: AutoSAR 4.0 8-bit CRC8 0x2F is not supported in hardware

**Description:**

The Cyclic Redundancy Check (CRC) module does not implement the 8-bit CRC-8- H2F required to support the Autosar 4.0 specification. The CRC-8-H2F uses a polynomial

generator seed of 0x2F and an equation of x ^5^ + x ^3^ + x ^2^+x+1.

**Workaround:**

Do not set the Polynomial selection to 0b11 in the CRC Configuration register (CRC_CFG). The 8-bit CRC-8-H2F function must be written in software to support AuroSAR 4.0.

## 1.39 ERR007126: MEMU: Instead of Byte 1 of MEMU CTRL Register, Byte 3 is currently protected

**Description:**

The Memory Error Management Unit (MEMU) Control (CTRL) register has Byte 3 protected instead of Byte 1 (using Byte numbering 3 to 0). Therefore the Software Reset bit (SWR) does not have register protection features.

**Workaround:**

Write software such that it does not rely on register protection features for MEMU CTRL Byte 1 SWR. Note that if the SWR bit is inadvertently set, only the status flags and overflow register will be cleared. The reporting tables are not cleared when SWR is set.

## 1.40 ERR007134: RCCU: If any accesses to the I-MEM or D-MEM of the safety and checker core are performed while the cores are disabled, the cores will get out of lockstep when enabled

**Description:**

If any accesses to the local Instruction Memory (I-MEM) or Data Memory (D-MEM) of the safety (Main Core_0) and checker core (Checker Core_0s) are performed while the cores are disabled, the cores will get out of lockstep when enabled. A disablement of Main Core_0 and Checker Core_0s is defined to be when the corresponding status bits of the Core Status Register (ME_CS) are set to 0. Most typically, this will happen after a destructive RESET where the system is booted up with the I/O Processor (IOP) and Hardware Security Module (HSM) enabled, but the safety and checker cores are disabled.

Memory Built In Self Test (MBIST) and Logic Built In Self Test (LBIST) are not affected by this errata.

**Workaround:**

Do not read or write the I-MEM or D-MEM of the safety core while it is disabled.

Insure that the Main Core_0 and Checker Core_0s are enabled and executing out of some memory (I-MEM / D-MEM, system RAM, Flash, ROM, external memory) when any bus master is accessing the safety core I-MEM / D-MEM.

## 1.41 ERR007137: MEMU: incorrect indication when a correctable error is signaled by the e200zx core cache

**Description:**

The Memory Error Management Unit (MEMU) incorrectly reports a non correctable error in case of a double bit error in the data array of the instruction or data cache of the e200zx core.

There are two types of errors in the e200zx core cache that the MEMU should detect as a correctable error:

1. Single bit error and not an address error
2. Double bit error in data array and not an address error

In this revision of the device, the MEMU can only detect the first type as a correctable error.

When an error is a double error in data array, MEMU is not able to report it as correctable error.

*Note: For the second type, the expected behavior for cache double bit error in the data array is the e200zx core logic detects the error and invalidates the cache line. This is defined as a correctable error event and should be reported as such to MEMU in normal operation.*

**Workaround:**

Do not expect the MEMU to contain a correctable error report for a double bit error in the cache data array. Use the e200zx cache correction / auto-invalidation and cache line lockout features to achieve the desired cache memory error management. Correction, auto-invalidation, and lockout features are configured in the L1 Cache Control and Status Registers (L1CSR[0,1]).

## 1.42 ERR007138: SARADC: Missed conversion after ABORT of the last channel of an injected chain

**Description:**

In the Successive Approximation Register Analog-to-Digital Converter (SARADC), when a chain conversion is injected over a normal chain conversion and an abort conversion command is initiated by setting the abort conversion bit of the Main Configuration Register (SARADC.MCR[ABORT]), when the last channel of the injected chain is in the sampling phase then a conversion will be missed.

The conversion of the next normal channel is skipped when the normal conversion chain is resumed.

Expected behavior: the conversion of the last injected channel is aborted, and the normal chain is resumed.

Errata behavior: the conversion of the last injected channel is correctly aborted, but the conversion of the next normal channel is incorrectly aborted.

For example: If channels 0, 1, 2, 3, 4, and 5 are to be converted in Normal chain and channels 6, 7, 8, and 9 are injected when the channel 3 conversion was ongoing the following behavior occurs:

nch0 --> nch1 --> nch2 --> nch3 (aborted by injected conversion chain) --> jch6 --> jch7 --> jch8 -- > jch9 (MCR[ABORT] set at this time) --> nch3 (restarted after end of injected chain) [(a)] --> nch5

**Workaround:**

Do not issue a conversion abort request when the conversion of the last channel of an injected chain is on going.

The user can read the Channel under measure address field (CHADDR) of the SARADC Main Status register (SARADC_MSR) to identify the channel under conversion.

## 1.43 ERR007185: SDADC: Watchdog Crossover event missed if PBRIDGEx_CLK less than SD_CLK

**Description:**

In the Sigma-Delta Analog-to-Digital Converter (SDADC), the watchdog monitor Lower and Higher threshold crossover events may get missed if the peripheral bridge clock (PBRDIGEx_CLK) is lower than the SDADC clock (SD_CLK). Therefore, the Watchdog Upper Threshold Cross Over Event (WTHH) and Watchdog Lower Threshold Cross Over Event (WTHL) bits of the Status Flag Register (SDADC.SFR) may not be set and the corresponding Direct Memory Access (DMA) or interrupts are not triggered.

**Workaround:**

When setting the different clocks in the Clock Generation Module (MC_CGM), ensure that PBRIDGEx_CLK is greater than SD_CLK.

## 1.44 ERR007190: GTM: Simultaneous Core and DPLL accesses to RAM Region 2 may lead to the DPLL reading erroneous data

**Description:**

(GTM-IP-168)

A core or DMA access to the Generic Timer Module (GTM) RAM Region 2 at the same time as a Digital PLL (DPLL) accesses that memory may result in the DPLL reading erroneous data. As a result, calculations of the DPLL may be wrong and this may lead to loss of synchronization.

For example, if the DPLL accesses RAM Region 2 to read a value from the profile (for calculation of TRIGGER) and at the same time the core or DMA has initiated a second read/write operation of RAM Region 2 via the Automotive Electronics Interface (AEI), it is possible that the output data of the core or DMA RAM read/write request is used as read data for the DPLL initiated read instead of the Number of Real and Virtual Events to be Considered for the Current or Last Increment (syn_t and syn_t_old) values.

**Workaround:**

Option 1: Synchronize core and DMA accesses to phases where the DPLL is not accessing RAM Region 2.

a. The conversion of the normal channel 4 is missing.

Example 1: synchronize DPLL RAM2 accesses to the TRIGGER signal using the TRIGGER Active Slope Interrupt (TASI) and checking that the RAM2 access is finished before the next active TRIGGER edge.

Example 2: use an Advanced Routing Unit (ARU) Connected Timer Output Module (ATOM)

channel in Signal Output Mode PWM (SOMP) one shot mode to output a 200 SYS_CLK pulse when the DPLL calculation starts. Use a Timer Input Module (TIM) channel to generate an active edge interrupt based on the DPLL calculation synchronized ATOM output. With the ATOM's Counter Compare Unit 1 (CCU1) interrupt (200 SYS_CLKs later) the DPLL sub increment calculation should be complete. At start of the ATOM interrupt service routine check if any action calculation is ongoing in the DPLL by reading the Calculation of Actions In Progress (CAIP2) flag in the DPLL_STATUS register. If this flag is zero (0), RAM Region 2 access by the core and DMA are safe to do.

Option 2: allow core and DMA accesses in tooth profile phases where DPLL is not accessing RAM2.

For example: use a Multi-channel Sequencer (MCS) to calculate and set flags that indicate the non critical phases the tooth profile when the DPLL does not access RAM2.

## 1.45 ERR007191: GTM: The DPLL's SORI and TORI interrupts are not asserted

**Description:**

(GTM-IP-169)

The Generic Timer Module (GTM) Digital PLL (DPLL) Trigger Out Of Range (TORI) and State Out of Range (SORI) interrupts are not set (1) when the following conditions are met:

- The upper 24 bits of Timebase Timestamp Channel 0 (TBU_TS0) are used as the input (DPLL_STATUS[LOW_RES]=1)
- The trigger/state input time stamps have an 8 times higher resolution than TBU_TS0, and the estimated time point value is multiplied by 8 (DPLL_CTRL1[TS0_HRT/S] = 0)

**Workaround:**

In this configuration use a Timer Output Module (TOM) or ARU Connected TOM (ATOM) to generate an interrupt on the time out of TRIGGER/STATE.

With every TRIGGER/STATE edge, adapt the (A)TOM period to the current speed and reset

Counter 0 (CN0). If CN0 is not reset by the next TRIGGER/STATE event, (A)TOM raises an edge interrupt at the end of the period.

## 1.46 ERR007203: SENT: In debug mode SENT message data registers appear to lose contents

**Description:**

The message read registers [Channel ‘n' Fast Message Data Read Register (n = 0 to (CH-1)) (CHn_FMSG_DATA), Channel ‘n' Fast Message CRC Read Register (n = 0 to (CH-1)) (CHn_FMSG_CRC), Channel ‘n’ Fast Message Time Stamp Read Register (n = 0 to (CH-1)) (CHn_FMSG_TS), Channel ‘n’ Serial Message Read Register (Bit 3) (n = 0 to (CH-1)) (CHn_SMSG_BIT3), Channel ‘n’ Serial Message Read Register (Bit 2) (n = 0 to (CH-1)) (CHn_SMSG_BIT2), Channel ‘n’ Serial Message Time Stamp Read Register (n = 0 to (CH-1)) (CHn_SMSG_TS), DMA Fast Message Data Read Register (DMA_FMSG_DATA), DMA Fast Message CRC Read Register (DMA_FMSG_CRC), DMA Fast Message Time Stamp Read Register (DMA_FMSG_TS), DMA Slow Serial Message Bit3 Read Register (DMA_SMSG_BIT3), DMA Slow Serial Message Bit2 Read Register (DMA_SMSG_BIT2) and DMA Slow Serial Message Time Stamp Read Register (DMA_SMSG_TS)] will appear to lose their contents in the following conditions:

a) The very first message is being received but not yet completely received and the MCU enters debug or freeze mode, the current message reception will get discarded and message read registers (as mentioned above) will read zeros

b) Auto clear functionality is enabled (GBL_CTRL[FAST_CLR] = 1). In this case, when first message is read, it will get clear the message read registers (due to auto clear functionality being enabled). On reading again, the message read registers (as mentioned above) might read zeros.

**Workaround:**

If the MCU requests entry to debug or stop mode, the message being received currently by SENT Receiver is discarded and the MCU enters debug/stop mode immediately. This does not affect the messages received completely, prior to entering debug mode and these messages will still be present on the message buffer and registers until they are read out.

Thus allow one message to be received completely and do not enable "Auto Clear" (GBL_CTRL[FAST_CLR] = 0), to allow messages to be read in debug or stop mode.

## 1.47 ERR007204: SENT: Number of Expected Edges Error status flag spuriously set when operating with Option 1 of the Successive Calibration Check method

**Description:**

When configuring the Single Edge Nibble Transmission (SENT) Receiver (SRX) to receive message with the Option 1 of the successive calibration pulse check method (CHn_CONFIG[SUCC_CAL_CHK] = 1), the number of expected edges error (CHn_STATUS[NUM[EDGES_ERR]) gets randomly asserted.
Option 2 is not affected as the number of expected edges are not checked in this mode.

The error occurs randomly when the channel input (on the MCU pin) goes from idle to toggling of the calibration pulse.

*Note: The Successive Calibration Pulse Check Method Option 1 and Option 2 are defined as follows:*

*Option 2: Low Latency Option per SAE specification*

*Option 1: Preferred but High Latency Option per SAE specification*

**Workaround:**

To avoid getting the error, the sensor should be enabled first (by the MCU software) and when it starts sending messages, the SENT module should be enabled in the SENT Global Control register (by making GBL_CTRL[SENT_EN] = 1). The delay in start of the two can be controlled by counting a fixed delay in software between enabling the sensor and enabling the SENT module. The first message will not be received but subsequent messages will get received and there will be no false assertions of the number of expected edges error status bit (CHn_STATUS[NUM[EDGES_ERR]).

Alternatively, software can count the period from SENT enable (GBL_CTRL[SENT_EN] = 1) to the first expected calibration pulse. If the number of expected edges error status bit

(CHn_STATUS[NUM[EDGES_ERR]) is asserted, software can simply clear it as there have no messages which have been completely received.

Alternatively, the software can clear this bit at the start and move ahead. When pause pulse is enabled, then NUM_EDGES will not assert spuriously for subsequent messages which do not have errors in them or cause overflows.

## 1.48 ERR007211: MC_ME: Core register IAC8 is cleared during a mode change when the core is reset

**Description:**

If a core is reset (ME_CADDR[0,1,2].RMC =1) in the Core Address register during a Mode Entry module (MC_ME) mode change then the Instruction Address Compare 8 (IAC8) register within the core which receives the reset will be cleared. In this implementation IAC8 is used as the Security watchdog service address. If a watchdog time-out occurs after this mode change and no valid service address exists, the core will attempt to execute code from the invalid address potentially resulting in an exception.

The watchdog (SWT) associated with that core is not reset by this change and retains its configuration. If fixed address execution is configured by the Service Mode in the software watchdog control register (SWT_CR.SMD= 0b10) when IAC8 is cleared to 0, it will not be possible to update IAC8 with the correct value. For other service modes the IAC8 register will be cleared to 0, but can be updated.

**Workaround:**

If the software watchdog mode is in fixed address execution (SWT_CR.SMD= 0b10), do not reset the corresponding core upon mode change. For all other modes, IAC8 must be updated by software immediately after the mode transition is completed.

## 1.49 ERR007222: SARADC: Minimum value of precharge must be greater than or equal to 2 ADC clock cycles

**Description:**

The Successive Approximation Register Analog-to-Digital Converter (SARADC) requires a minimum valid value of the Precharging phase duration field (PRECHG) of the Conversion Timing Register (SARADC_x.CTRz[PRECHG]) must be 2, meaning the precharge phase duration is two (2) SARADC clock cycles. This is incorrectly defined as '1' in some revisions of the documentation.

**Workaround:**

Take into account the minimum value of 2 when configuring the precharge duration in the SARADC_x.CTRz[PRECHG].

## 1.50 ERR007223: FCCU: FCCU_IRQ_EN register is writeable in all operating modes

**Description:**

In the Fault Collection and Control Unit (FCCU), the FCCU Interrupt Enable register (FCCU_IRQ_EN) is writable (and readable) in all states (NORMAL, CONFIG, FAULT and ALARM) while in some revisions of the documentation it is stated "This register is writable only in the CONFIG state".

**Workaround:**

Take into account that FCCU_IRQ_EN register can be written in all the states of the FCCU.

Please ignore the following text in the description of FCCU_IRQ_EN register if you find it in the documentation revision in hand: "This register is writable only in the CONFIG state."

## 1.51 ERR007226: FCCU: the error-out signalling cannot be disabled in non Bi-stable protocols

**Description:**

In the Fault Collection and Control Unit (FCCU) module, the error out signalling can only be disabled when using the Bi-stable protocol and not all of the protocols.

The Error Out (EOUT) Signaling Enable Register (FCCU_EOUT_SIG_ENx) registers, which can be used for disabling error out signalling, are applicable only for bi-stable mode and do not affect the other protocols. (Various protocols like Bi-stable, Dual rail and other protocols are selected by programming the Fault Output Mode (FOM) field of the Configuration Register (FCCU_CFG)).

**Workaround:**

Do not expect and program the EOUT_SIGN_ENx register to support anything other than bi-stable protocol.

The EOUT Signaling Enable (EOUTENx) field description of FCCU_EOUT_SIG_EN should be read as follows:

0 = EOUT signaling is disabled for error Recoverable Fault (RF) source x for bi-stable protocol. Error pins behave as if in Non-Faulty state.

1 = EOUT signaling is enabled for error RF source x for bi-stable protocol.

## 1.52 ERR007227: FCCU: FCCU Output Supervision Unit (FOSU) will not monitor faults enabled while already pending

**Description:**

The Fault Collection and Control Unit (FCCU) Output Supervision Unit (FOSU) will not monitor the FCCU reaction to fault inputs that are enabled with an already pending notification.

The FOSU monitoring is triggered by an edge from a fault input. The edge detection will be

blocked in the following cases:

1. When a fault input is disabled in the FCCU and a fault occurs,
2. When a fault input is enabled in the FCCU and a fault occurs in the CONFIG state. FOSU edge detection remains blocked until it gets initialized by a FCCU reaction or a destructive reset.

**Workaround:**

Apply the following procedure when enabling fault inputs in the FCCU in order to ensure the correct monitoring by the FOSU:

1. Check for FCCU pending faults and clear them.
2. Configure the FCCU as desired. In addition enable fault input for interrupt reaction (software recovery mode) to an injected error on this input.
3. Immediately on exiting the CONFIG state, check for FCCU pending faults. If there is a fault status set then initiate a destructive reset.
4. Clear the FOSU status by injecting a fault on the FCCU fault input configured for software recovery mode. This will generate a FCCU reaction that will clear the FOSU edge detection logic.

Apply the following procedure when exiting FCCU CONFIG state in order to ensure the correct monitoring by the FOSU:

- Check for FCCU pending faults. If there is a fault status set then initiate a destructive reset.

## 1.53 ERR007228: LINFlexD: Erroneous transmission in LIN master mode for payload greater than eight bytes

**Description:**

When the LINFlexD module is configured as follows:

- LIN (Local interconnect network) master mode is enabled by setting the MME (Master Mode Enable) bit in the LINCR1 (LIN Control Register 1);

if an extended frame is transmitted (DFL (Data Field Length) bits in BIDR (buffer identifier register) is bigger than 7) and the data buffer is refilled immediately after the DBEF (Data Buffer Empty Flag) bit in the LINSR (LIN Status Register) is set, then the DATA7 (DATA byte 7) field of BDRM (Buffer Data Register Most Significant) register is overwritten by the received data byte and sent instead of the desired value.

**Workaround:**

Once the DBEF (Data Buffer Empty Flag) in LINSR (LIN Status Register) is set, add a delay of 3 sample time duration before refilling the data buffer and then clear the DBEF flag.

Note: One sample time duration is (1/16*Baud Rate) seconds.

## 1.54 ERR007234: PSI5: No transfer error generated for accesses within the unused range of the PSI5 peripheral window

**Description:**

The Peripheral Sensor Interface (PSI5) uses 4 Kbytes of the 16 Kbytes range of the peripheral bridge slot assigned to it.

Accesses after the 4 Kbytes (from offset 0x1000 to offset 0xFFFF) will not generate a transfer error.

*Note: Accesses to unimplemented locations within the 4 Kbyte window will correctly generate a transfer error.*

**Workaround:**

Take into account that no transfer error will be generated outside the 4 Kbyte region used by the PSI5 module.

In case such accesses must be detected, use the memory protection unit (MPU) to limit accesses.

## 1.55 ERR007246: SARADC: First conversion after exit from stop mode may be corrupted

**Description:**

In the Successive Approximation Analog to Digital Converter (SARADC), if a chain conversion is on going and a transition to stop mode request is done, the result of the first conversion after exit of the stop mode (in other words, the conversion that was interrupted when going into stop mode) will be corrupted in the following cases:

Case A: the peripheral bridge clock (PBRIDGEx_CLK) becomes lower than the SARADC clock (SAR_CLK)

*Note: This might be the case if the input of the PBRDIGEx_CLK divider is changed during the mode transitions (from the output of the PLL to the internal RC Oscillator for example)*

Case B: the PBRIDGEx_CLK is resumed after the SAR_CLK, with a delay greater than 10 cycles of SAR_CLK

**Workaround:**

The following workarounds are possible:

1. Disable PBRIDGEx_CLK during stop mode and enable it only with a configuration such that it is greater than SAR_CLK.

OR

2. Verify that no analog conversion is ongoing before issuing a stop mode request by reading the ADC status field of the Main Status Register (SARADC_x.MSR[ADCSTATUS] = 0b000, also known as IDLE).

OR

3. Ignore the result of the first conversion after stop mode exit, considering it as corrupted.

## 1.56 ERR007259: e200zx: ICNT and branch history information may be incorrect following a nexus overflow

**Description:**

If an internal Nexus message queue over-flow occurs when the e200zx core is running in branch history mode (Branch Method bit [BTM] in the Development Control register 1 [DC1] is set [1]), the instruction Count (ICNT) and branch history (HIST) information in the first program trace message following the Program Correlation message caused by an over-flow of the internal trace buffers, will contain incorrect ICNT and HIST information.

This can also occur following an overflow of the internal Nexus message queues in the traditional branch mode (BTM in the DC1 is cleared [0]). Traditional branch mode Nexus messages do not include HIST information, since all branches generate a trace message.

**Workaround:**

There are two methods for dealing with this situation.

1. Avoid overflows of the Nexus internal FIFOs by reducing the amount of trace data being generated by limiting the range of the trace area by utilizing watchpoint enabled trace windows or by disabling unneeded trace information, or by utilizing the stall feature of the cores.
2. After receiving an overflow ERROR message in Branch History mode, the ICNT and HIST information from the first Program Trace Synchronization message and the next Program Trace message with a relative address should be discarded. The address information is correct, however, the ICNT and previous branch history are not correct. All subsequent messages will be correct.

In traditional branch mode, the ICNT information should be discarded from the Program Trace Sync message and the next direct branch message.

## 1.57 ERR007269: LINFlexD: Erroneous timeout interrupt could be generated by LIN in master mode

**Description:**

When the LINFlexD (Local Interconnect Network module) is configured in LIN Master mode by setting the Master Mode Enable bit of the LIN Control Register 1 (LINCR1[MME]=1) and the Time-out counter mode bit in the LIN Time-Out Control Status Register is cleared (LINTCSR [MODE]=0), if the LIN Master transmits an header and expects a response from a slave, it is unable to update the OC2[7:0] (Output Compare Value 2) bits in the LINOCR (LIN Output Compare Register) with a timeout value called 'Response Timeout Value', which is used to trigger a timeout interrupt in case no response is received.

Consequently, an erroneous or no timeout interrupt will be generated by the LIN timer.

**Workaround:**

In order to generate a Response Timeout, and consequently an interrupt, the LIN Master mode should only be used with the MODE (Time-out counter mode) bit in the LINTCSR (LIN Time-Out Control Status Register) set.

In addition, the software must program the OC2[7:0] (Output Compare Value 2) bits in the LINOCR (LIN Output Compare Register) with the value of Header Duration plus Response

Timeout value plus the LINTCSR CNT[7:0] before asserting the HTRQ (Header Transmission Request) value in the LINCR2 (LIN Control Register 2).

## 1.58 ERR007274: LINFlexD: Consecutive headers received by LIN Slave triggers the LIN FSM to an unexpected state

**Description:**

As per the Local Interconnect Network (LIN) specification, the processing of one frame should be aborted by the detection of a new header sequence and the LIN Finite State Machine (FSM) should move to the protected identifier (PID) state. In the PID state, the LIN FSM waits for the detection of an eight bit frame identifier value.

In LINFlexD, if the LIN Slave receives a new header instead of data response corresponding to a previous header received, it triggers a framing error during the new header's reception and returns to IDLE state.

**Workaround:**

The following steps have to be performed:

1. Configure slave to Set the MODE bit in the LIN Time-Out Control Status Register (LINTCSR[MODE]) to '0'.
2. Configure slave to Set Idle on Timeout in the LINTCSR[IOT] register to '1'. This causes the LIN Slave to go to an IDLE state before the next header arrives, which will be accepted without any framing error.
3. Configure master to wait for Frame maximum time (T Frame_Maximum as per LIN specifications) before sending the next header.

*Note:* *THeader_Nominal = 34 * TBit*

*TResponse_Nominal = 10 * (NData + 1) * TBit*

*THeader_Maximum = 1.4 * THeader_Nominal*

*TResponse_Maximum = 1.4 * TResponse_Nominal*

*TFrame_Maximum = THeader_Maximum + TResponse_Maximum*

*where TBit is the nominal time required to transmit a bit and NData is the number of bits sent.*

## 1.59 ERR007297: LINFlexD: Response timeout values is loaded in LINOCR[OC2] field instead of LINOCR[OC1]

**Description:**

In the LINFlex module, the response timeout value calculated by hardware is loaded onto the OC2[7:0] (Output Compare 2) bits of LINOCR (LIN Output Compare Register) instead of being loaded into the OC1[7:0] (Output Compare 1) bits of the same register as stated in the documentation.

This applies when the Time-out counter mode is enabled by clearing the MODE (Time-out counter mode) bit in the LINTCSR (LIN Timeout Control Status Register).

**Workaround:**

Expect that OC2[7:0] (Output Compare 2) bits are loaded by hardware with the response timeout value when the MODE (Time-out counter mode) bit in LINTCSR is cleared.

## 1.60 ERR007305: e200zx: JTAG reads of the Performance Monitor Counter registers are not reliable

**Description:**

Reads of the Performance Monitor Counter (PMC0, PMC1, PMC2, and PMC4) registers through the IEEE 1149.1 or IEEE 1149.7 (JTAG) interfaces may return occasional corrupted values.

**Workaround:**

To ensure proper performance monitor counter data at all times, software can be modified to periodically read the PMCx values and store them into memory. JTAG accesses could

then be used to read the latest values from memory using Nexus Read/Write Access or the tool could enable Nexus data trace for the stored locations for the information to be transmitted through the Nexus Trace port.

## 1.61 ERR007325: FCCU: Unsuccessful decorated storage access may cause erroneous signaling of FCCU Channel 40

**Description:**

In rare conditions, a decorated memory reference targeting the system RAM may interfere with the Decorated Storage Memory Controller (DSMC) safety monitor. The result is an erroneous signalling of Channel 40 in the Fault Collection and Control Unit (FCCU). This will only occur if the decorated memory reference is unsuccessful due to either an illegal decoration encoding or a non-correctable Error Correction Code (ECC) event. Although this usually coincides with an exception taken by the core, no exact procedure is known to detect an erroneous signalling of Channel 40.

**Workaround:**

No special workaround required. Software should handle any fault of FCCU Channel 40 according to the chosen fault reaction for this channel.

## 1.62 ERR007335: MEMU: ECC errors may be double reported when initiated by the Safety core to local memory of other cores

**Description:**

Error Correction Code (ECC) events on memory references initiated by the Safety (Main) Core 0 targeting the local memory of either the Main Core 1 or the Input/Output Peripheral Core 2 (IOP) may be reported twice in the Memory Error Management Unit (MEMU) System RAM Reporting Table with the same address. One entry will include syndrome information, the other entry will not include this information.

**Workaround:**

During operation, if the MEMU contains two entries for the same address of an address which is part of a memory for which ECC syndromes are reported, software should check whether one of the two syndromes is 0xFF. If so, this entry should be ignored and deleted. This entry came from another ECC unit which does not report syndromes. As long as the entry with the correct syndrome is stored in the MEMU, entries for the same address without syndrome will not be stored.

## 1.63 ERR007339: STCU2: STCU2 fault injected by FCCU is self clearing

**Description:**

In the Self-Test Control Unit (STCU2), a fault can be injected by the Fault Collection and Control Unit (FCCU) in order to verify the correct behavior of the interface (fake fault).

The STCU_LMBIST_USR_ERR signal, which is connected to the FCCU input #8, generates only a pulse when an error is injected to this signal by the FCCU.

This is different to other signals from STCU2, where injected faults remain asserted until explicitly cleared.

**Workaround:**

Use a software recoverable fault (select-able with FCCU_RF_CFG) for FCCU input #8, when a fault is injected into the STCU2.

## 1.64 ERR007352: DSPI: reserved bits in slave CTAR are writable

**Description:**

When the Deserial/Serial Peripheral Interface (DSPI) module is operating in slave mode (the Master [MSTR] bit of the DSPI Module Configuration Register [DSPIx_MCR] is cleared), bits 10 to 31 (31 = least significant bit) of the Clock and Transfer Attributes Registers (DSPIx_CTARx) should be read only (and always read 0). However, these bits are writable, but setting any of these bits to a 1 does not change the operation of the module.

**Workaround:**

There are two possible workarounds.

Workaround 1: Always write zeros to the reserved bits of the DSPIx_CTARn_SLAVE (when operating in slave mode).

Workaround 2: Mask the reserved bits of DSPIx_CTARn_SLAVE when reading the register in slave mode.

## 1.65 ERR007356: SDADC: The SDADC FIFO does not function correctly when FIFO overwrite option is used

**Description:**

In the Sigma-Delta Analog-to-Digital Converter (SDADC), when the FIFO Over Write Enable bit (FOWEN) of the FIFO Control Register (FCR) is set (FCR[FOWEN]=1), the following flags of the Status Flag Register (SFR) may not reflect the correct status:

- Data FIFO Full Flag (DFFF)
- Data FIFO Empty Flag (DFEF)

When the number of entries received by the FIFO reaches 2x the FIFO size (field FSIZE of FIFO Control Register (FCR)):

- SFR[DFFF] is cleared, incorrectly indicating the FIFO is not full
- SFR[DFEF] is set, incorrectly indicating the FIFO is empty

The expected behavior is that:

- SFR[DFFF] remains set until data is read out of the FIFO
- SFR[DFEF] remains clear until all data is read out of the FIFO

**Workaround:**

Do not use the FIFO Overwrite option to overwrite FIFO contents. Software shall clear the FIFO overrun condition (if necessary) and flush the FIFO contents before expecting valid data in the FIFO.

## 1.66 ERR007360: FEC: Minimum VDD is 3.15 V instead of 3.0 V

**Description:**

The Fast EtherNet Controller (FEC) Reduced Media Independent Interface (RMII) may not operate correctly if the power supply for the Ethernet pins (VDD_HV_IO_FLEX) is less than 3.15 volts, the external load is greater than 15 pF, or for junction temperatures over 150 °C (KGD). In addition, CMOS levels must be selected for the RMII pins instead of TTL in the System Integration Unit Lite (SIUL) Multiplexed Signal Configuration registers. For 150 °C (Junction temperature [Tj] and less) applications and external pin loads of less than 15 pF, there are no restrictions on voltage, but CMOS levels must be used. For 165 °C (Tj) applications, the minimum supply voltage is 3.15 volts regardless of load.

**Workaround:**

For 150 °C Tj applications, regulate VDD_HV_IO_FLEX supply to the FEC IOs to have a minimum voltage of 3.15 V (-5 % of the nominal supply voltage) instead of 3.0 (-10 %), or use an external load of less than 15 pF. Additionally, the use of CMOS voltage levels will be required instead of TTL by setting the Input Level Selection to CMOS (0b11) instead of TTL (0b01) in each of the pins MCSRs, regardless of supply voltage, load, and temperature. For 165 °C Tj applications, VDD_HV_IO_FLEX must be regulated to 3.15 V minimum.

## 1.67 ERR007362: SDADC: Additional DMA request generated after single read access

**Description:**

The Sigma-Delta Analog-to-Digital Converter (SDADC) issues an extra transfer request when the FIFO full Direct Memory Access (DMA) channel is configured to read only 1 data value from the SDADC.

Therefore, when the FIFO (First-In-First-Out) Threshold (FTHLD) field of the SDADC FIFO Control Register (FCR) is 0 (1 conversion) or when the FIFO Enable bit (FE) of FCR is 0 (FIFO is disabled), the extra read request will return invalid data.

The first DMA read access to the SDADC (correct read) returns good data, the second one (extra access, unwanted) returns the contents of the FIFO (undefined, old conversion results).

**Workaround:**

Workaround 1:

Configure the SDADC FIFO threshold to a number N greater than 0 and its FIFO full DMA channel to read at least 2 conversion results at a time.

Workaround 2:

If available (not in use within the application software), use the SDADC watchdog DMA channel, that is not generating extra requests, instead of the SDADC FIFO full channel.

Use the following settings:

- FIFO threshold set to 1
- Watchdog high and low threshold set to 0

## 1.68 ERR007404: SENT: Message overflow in SENT Receiver can lead to stall condition in the MCU

**Description:**

Under certain conditions, the Single Edge Nibble Transmission (SENT) Receiver (SRX) stalls and the Fast Message Data Ready bit for the SENT channel (FMSG_RDY[F_RDYn]) will no longer get set to indicate that a fast message is available. Reads of any of the fast message registers by the MCU core will stall and not complete. The registers affected are:

| Register | Register Name |
|---|---|
| DMA_FMSG_DATA | Direct Memory Access (DMA) Fast Message Data Read Register |
| DMA_FMSG_CRC | DMA Fast Message Cyclic Redundancy Check Register |
| DMA_FMSG_TS | DMA Fast Message Time-stamp Register |
| CHn_FMSG_DATA | Channel Fast Message Data Read Register |
| CHn_FMSG_CRC | Channel Cyclic Redundancy Check Register |
| CHn_FMSG_TS | Channel Fast Message Time-stamp Register |

A stall may occur if an overflow status condition is detected in the SENT Receiver Channel Status register (CHn_STATUS[FMSG_OFLW] = 1).

The overflow occurs when two messages are allowed to queue in the internal buffers of the SENT Receiver.

**Workaround:**

Software should ensure that SENT message overflow does not occur.

If interrupts are used (when the Enable FDMA (FDMA_EN) bit of Fast Message DMA Control Register (SRX_FDMA_CTRL) is set to 0) to read the SENT messages, the interrupt for data reception should be enabled by setting Enable for Fast Message Ready Interrupt (FRDY_IE[n]) bit of Fast Message Ready Interrupt Control Register (SRX_FRDY_IE) for every channel n and the interrupt priority should be such that the software is able to read the message before the next message arrives.

When using eDMA access to access the SENT (when the Enable FDMA (FDMA_EN) bit of Fast Message DMA Control Register (SRX_FDMA_CTRL) is set to 1), the DMA request from SENT should be serviced before the next message arrives.

The minimum duration between the reception of two consecutive messages in one channel is 92 times the utick length (time).

If the stall occurs, a reset will be required to clear the stall condition. A Software Watchdog Timer (SWT) should be enabled to force a reset of the MCU if the device becomes stalled.

## 1.69 ERR007411: PBRIDGE: Incorrect transfer error information on accesses to reserved locations

**Description:**

Some reserved address ranges of memory map (unused areas of the Peripheral Bridges), do not behave as expected with regard to the setting of the Peripheral bus Abort Enable bit (PAE) and the Register bus Abort enable bit (RAE) of the System Status and Configuration Module (SSCM) Error Configuration register (SSCM_ERROR).

Case1: When SSCM_ERROR[PAE] = 1 and SSCM_ERROR[RAE] = X, no transfer error is generated for the ranges:

- 0xFBFB0480 - 0xFBFFFFFF
- 0xFFFB0040 - 0xFFFB007F
- 0xFFFB00C0 - 0xFFFB00FF
- 0xFFFB0140 - 0xFFFB01FF
- 0xFFFB0240 - 0xFFFB02FF

Case2: When SSCM_ERROR[PAE] = 0 and SSCM_ERROR[RAE] = 0, a transfer error is incorrectly generated for accesses in the range 0xFFFE4000 - 0xFFFF3FFF.

Case3: When SSCM_ERROR[PAE] = 0 and SSCM_ERROR[RAE] = 1, a transfer error is incorrectly generated for accesses in the range 0xFFF6C800 - 0xFFF73FFF.

Case4: When SSCM_ERROR[PAE] = 1 and SSCM_ERROR[RAE] = 0, no transfer error is generated for accesses in the range 0xFFF6C800 - 0xFFF73FFF.

**Workaround:**

Do not rely on transfer error information for the above address ranges.

## 1.70 ERR007412: PBRIDGE: Incorrect transfer error information for accesses to TDM and FEC reserved locations

**Description:**

When the Peripheral bus Abort Enable bit (PAE) and the Register bus Abort enable bit (RAE) of the System Status and Configuration Module (SSCM) Error Configuration register (SSCM_ERROR) are set, no transfer error is generated for accesses to reserved locations of the following peripherals:

- Tamper Detection Module (TDM [base address 0xFC0E_4000])
- Fast Ethernet Controller (FEC [base address 0xFC0B_0000])

**Workaround:**

Do not rely on transfer error information for accesses to the reserved locations of the TDM and the FEC.

## 1.71 ERR007414: PBRIDGE: Incorrect transfer error when accessing reserved locations of the Peripheral Bridge

**Description:**

The following locations of the Peripheral Bridge (PBRIDGE) do not behave as expected:

- Accesses to the offset 0x0110 from PBRIDGE start address, corresponding to the unimplemented Peripheral Access Control Register E (AIPS_PACRE), result in a transfer error instead of returning the value 0x0 for read and ignoring write operations;
- Register space at offset 0x20--0x2C is marked as reserved but a transfer error will not be generated when accessed. Associated PACR registers 0x100+address is actually accessed;
- Accesses to the reserved offset ranges 0x0040--0x0080, 0x0120--0x012C and 0x0134--0x013C do not generate transfer error.

**Workaround:**

Do not access to the offset 0x0110 from PBRIDGE start address, corresponding to the unimplemented Peripheral Access Control Register E (AIPS_PACRE).

In addition no errors will be generated on accesses to the following reserved offset ranges:

- 0x20--0x2C;
- 0x0040--0x0080;
- 0x0120--0x012C;
- 0x0134--0x013C.

## 1.72 ERR007415: JTAG: PA[9] = JTAG TDO pad is not pull-up during reset

**Description:**

The GPIO PA[9] which is also used as JTAG TDO is configured as high impedance during power-up and while JTAG is under reset.

**Workaround:**

Use an external pull-up on pin PA[9] in case a high level is required during power-up and reset.

## 1.73 ERR007416: MEMU: Flexray data RAM and GTM RAMs Error not correctly registered within MEMU during online MBIST

**Description:**

During on-line Memory Built-in Self-Test (MBIST) and if the MBIST clock used by the GTM or the FlexRay is generated from a different clock source (RC-OSC, OSC, PLL0) than the system clock, the failing address and data information related to failures in FlexRay data RAM module or in any of the Generic Timer Module (GTM) RAMs, may not be correctly registered within the Memory Error Management Unit (MEMU). The test of FlexRay data RAM and GTM RAMs by MBIST in this case is performed correctly. Also the status of MBIST run corresponding to these memories is correctly reported by the STCU registers, namely, STCU Error status register (STCU_ERR_STAT), STCU2 On-Line MBIST Status Low/Medium Register (STCU_MBSI/mSW) and STCU2 On-Line MBIST End Flag Low/Medium Register (STCU_MBEI/mSW) register.

**Workaround:**

To guarantee correctness of error information in MEMU, resulting from online MBIST of FlexRay data RAM and GTM RAMs, it is recommended to use one of the two approaches:

- Perform off-line MBIST for FlexRay memories and Generic Timer Module (GTM) RAMs, using either RC-OSC or PLL0 as clock source or
- Perform on-line MBIST, using either RC-OSC or PLL0 as clock source, ensuring the same clock source is used for both the system clock and the GTM RAMs and FlexRay data RAM.

## 1.74 ERR007425: SENT: Unexpected NUM_EDGES_ERR error in certain conditions when message has a pause pulse

**Description:**

When the Single Edge Nibble Transmission (SENT) Receiver (SRX) is configured to receive a pause pulse (Channel ‘n' Configuration Register - CHn_CONFIG[PAUSE_EN] = 1) the NUM_EDGES error can get asserted spuriously (Channel ‘n' Status Register - CHn_STATUS(NUM_EDGES_ERR] = 1) when there is any diagnostic error (other than number of expected edges error) or overflow in the incoming messages from the sensor.

**Workaround:**

Software can distinguish a spurious NUM_EDGES_ERR error from a real one by monitoring other error bits. The following tables will help distinguish between a false and real assertion of NUM_EDGES_ERR error and other errors. Software should handle the first error detected as per application needs and other bits can be evaluated based on these tables. The additional error may appear in the very next SENT frame. *Table 2* contains information due to erratum behavior. *Table 3* contains clarification of normal NUM_EDGES_ERR behavior.

**Table 2. Erratum behavior of NUM_EDGES_ERR**

| First Error Detected | Other error bits asserted | Cause for extra error bits getting asserted | Action |
|---|---|---|---|
| NIB_VAL_ERR | NUM_EDGES_ERR asserted twice | Upon detection of the first error, the state machine goes into a state where it waits for a calibration pulse, the first NUM_EDGES_ERR error is for the current message as the state machine does not detect an end of message. The second error comes when both the Pause pulse and the Calibration pulse are seen as back to back calibration pulses and no edges in between. | Ignore both NUM_EDGES_ERR errors |
| FMSG_CRC_ERR | NUM_EDGES_ERR asserted twice | Same as NIB_VAL_ERR. | Ignore both NUM_EDGES_ERR errors |
| CAL_LEN_ERR | NUM_EDGES_ERR asserted once | Since the calibration pulse is not detected as a valid calibration pulse, the internal edges counter does not detect the end of one message and start of bad message (which has CAL_LEN_ERR); hence the NUM_EDGES_ERR gets asserted. | Ignore NUM_EDGES_ERR error |
| FMSG_OFLW | NUM_EDGES_ERR asserted once (random occurrence) | A message buffer overflow may lead the state machine to enter a state where it waits for a calibration pulse (behavior also seen in ERR007404). When in this state, the state machine can detect both a Pause pulse and a Calibration pulse as back to back calibration pulses and no edges in between. Then, the NUM_EDGES_ERR can get asserted. Since entry into this state is random, the error can be seen occasionally. | Ignore NUM_EDGES_ERR error |

**Table 3. Expected behavior, clarification of NUM_EDGES_ERR cases**

| First Error Detected | Other error bits asserted | Cause for extra error bits getting asserted | Action |
|---|---|---|---|
| NUM_EDGES_ERR (when edges are less than expected) | NIB_VAL_ERR is asserted | When the actual number of edges in the message are less than expected, then a pause pulse gets detected as a nibble since the state machine expects nibbles when actually there is a pause pulse present. This generates NIB_VAL_ERR. | Ignore the NIB_VAL_ERR |
| NUM_EDGES_ERR (when edges are more than expected) | NIB_VAL_ERR and PP_DIAG_ERR are asserted | When the actual number of edges in a message are more than expected, then after receiving the programmed number of data nibbles, the state machine expects a pause pulse.<br>However, the pause pulse comes later and gets detected as a nibble and hence NIB_VAL_ERR is asserted. Since the message length is not correct, PP_DIAG_ERR is also asserted. | Ignore NIB_VAL_ERR and PP_DIAG_ERR |

## 1.75 ERR007433: JTAGM: Nexus error bit is cleared by successful RWA

**Description:**

The JTAG Master module status register includes a Nexus error status bit (JTAGM_SR[Nexus_err]) that indicates the status of the last Nexus Read/Write Access (RWA) command. Once this information is latched, it can only be cleared by performing a successful RWA transaction via the same core that caused the error. In addition, if a RWA transaction is performed by a different core, the error bit will not be cleared and it is not possible to determine if the access by the second core RWA was successful or generated another error.

In general, this bit should only be set when the Nexus RWA accesses non-existent or protected memory spaces.

**Workaround:**

If the status information is required from a specific core, the user software or tool should read the error bit (ERR) of the e200zx core's Nexus Read/Write Access Control/Status register. To avoid setting the error bit, do not perform illegal memory accesses.

## 1.76 ERR007498: M_(TT)CAN: Transmitted bit in control field is falsified when using extreme bit time configurations

**Description:**

When the Modular CAN (M_CAN) or the Time Triggered Modular CAN (M_TTCAN) module transmits a frame, and when the values of both the Time segment after sample point (TSEG2) and the Baud Rate Prescaler (BRP) fields of the Bit Timing and Prescaler Register (BTP) are 0 (zero), one bit in the control field will be transmitted with an erroneous value. The effect is different for frames to be transmitted in Classic CAN format or CAN Flexible Data-rate (FD) format.

Case 1: Transmission of a classic CAN frame with the 2-bit wide CAN Mode Enable field (CME) of the CC Control Register (CCCR) is 0b00.

If the frame under transmission has a 29-bit identifier where the most significant bit (bit 28 of

identifier) is "1", then the reserved bit following the RTR bit will be transmitted recessive instead of dominant. As a consequence, this frame will be incorrectly interpreted as a CAN FD frame by any node with the CAN FD mode enabled that will therefore generate an error frame. Nodes supporting only the classic CAN mode will ignore this bit (reserved) and correctly receive the frame.

Case 2: For a transmission of a CAN FD Frame with the CME of the CCCR not equal to 0b00.

If the FD frame under transmission has a 29-bit identifier where the most significant bit is "0", or has an 11-bit identifier, then the FD Frame Format (FDF) bit of the frame is transmitted dominant instead of recessive and the rest of the frame is transmitted in Classic CAN format with an incorrect Data Length Code (DLC) field.

**Workaround:**

Do not use bit timing configurations where BTP.TSEG2 and BTP.BRP are both zero for CAN FD communication.

## 1.77 ERR007528: GTM: Action not always calculated immediately by DPLL

**Description:**

(GTM-IP-170)

If the Generic Timer Module (GTM) Digital PLL (DPLL) action calculation is interrupted by a new input event, the next TRIGGER/STATE input event action calculation (after the sub increment calculation is finished) starts at the previous internal action number. If new action data arrives during the sub increment calculation it is only used after the next input event. New Position Minus Time (PMT) data for an action with a higher action number is not recognized immediately.

Normally, the calculation of sub increments and PMT are not done in parallel because of resource sharing. When the DPLL is doing the action calculation it has exclusive access rights to RAM Region 1a which contains the PMT request values, so the DPLL cannot accept new PMT requests via the Advanced Routing Unit (ARU). Therefore requested actions are not calculated regularly with every tooth.

**Workaround:**

The GTM should only request actions which are not "past" with every new tooth. The synchronization of the Multi Channel Sequencer (MCS) task to Timer Input Module (TIM) input event can be done by routing the TIM edge capture event value via ARU to MCS. If new PMT data arrives after the action number has reached the value zero, the action is calculated immediately starting with the highest action number again.

You can request the action calculation tooth by tooth until an action runs in to the past. Additional PMT requests can be placed earlier while the DPLL is performing sub increment calculations because RAM Region 1a is exclusively used for PMT requests via ARU.

Send PMT requests at least 3 teeth before the action has to be executed. This ensures that the MCS and ARU Connected Timer Output Module (ATOM) get action results from a calculation an input event cycle before.

## 1.78 ERR007529: GTM: TIM overflow bit is not set and the signal level bit has inverse value when sent to ARU in some cases

**Description:**

(GTM-IP-172)

When the Generic Timer Module (GTM) Timer Input Module (TIM) is in Timer Input Event Mode (TIEM, TIMn_CHx_CTRL[TIM_MODE] = 2), with Advanced Routing Unit (ARU) enabled (TIMn_CHx_CTRL[ARU_EN] = 1), and Input Signal Level high (ISL, TIMn_CHx_CTRL[ISL] = 1, the Overflow Bit (ACB1) might not be set and the signal level bit (ACB0) will be incorrect.

This error occurs when two input signals change in close proximity (faster than the ARU routing time), for example, an edge initiates an ARU transfer and one system clock before the ARU request is serviced the second input signal changes. Note that the Interrupt Request bit associated with the Overflow is set correctly.

**Workaround:**

Workaround 1:

Use the TIM channel input filter to remove signal changes smaller than the ARU routing time by configuring the filter parameters for rising and falling edges (TIMn_CHx_FLT_FE/TIMn_CHx_FLT_RE) with a delay which is greater than the ARU routing time.

Workaround 2:

Select the Edge Counter (TIMn_CHx_ECNT or TIMn_CHx_CNT) to be transferred in the ARU data to the Multi Channel Sequencer (MCS) and use the MCS to reconstruct the correct TIM data as follows:

```
Last_CNT = -1
For each ARU_DATA
  If ARU_DATA(ACB1) ==0
    If Last_CNT != -1
      If Last_CNT+1 != ARU_DATA(CNT)
```

```
    Message (Hit on ERRATA: Detected overflow condition)
    ARU_DATA(ACB1) = 1
    ARU_DATA(ACB0) = not ARU_DATA(ACB0)
  else
  Message(No signal level present yet, cannot apply workaround)
Last_CNT = ARU_DATA(CNT)
```

## 1.79 ERR007530: GTM: New DPLL Position Minus Time data not received

**Description:**

(GTM-IP-173)

When the Generic Timer Module (GTM) Digital PLL (DPLL) receives Position Minus Time (PMT) requests after a TRIGGER/STATE event, only that request can be considered. The DPLL blocks new PMT requests for about 200 ns. New PMT requests are only accepted after the calculation of the pending action calculations are performed. This calculation starts in the state machine, about 10 us after the input event and is completed depending on the number of actions (A) to be calculated A*3.7 µs later. After this time the PMT request is accepted, but it is not possible to adjust the action calculation with updated data. The "old" value is always calculated. The PMT result is calculated based on older PMT input data because the pending data transfer with newer input data to the DPLL cannot be executed.

**Workaround:**

When the calculated action is transmitted to the Multi Channel Sequencer (MCS), check if there was an Advanced Routing Unit (ARU) transfer with new data for this action blocked by the ARU because the DPLL was not ready to receive new data within this time. If the ARU transfer was just completed, the corresponding action contains only the older PMT requirements. Ignore this action value and wait for the new value which appears about 3.7 µs after the PMT requirement update was transmitted.

## 1.80 ERR007531: GTM: DPLL Position Minus Time result is not sent to the ARU

**Description:**

(GTM-IP-174)

The Generic Timer Module (GTM) Digital PLL (DPLL) has a state where there is a delay between when the New Output Data Values Concerning To Action t bit (DPLL_ACT_STA[ACT_N(t)]) is reset and when the corresponding shadow bit (DPLL_ACT_STA_shadow[ACT_N(t)]) is set to "1", which starts the transfer of the output data via the Advance Routing Unit (ARU).

If during this delay a new input event occurs (DPLL_ACT_STA[ACT_N(t)]) and the internal state controller changes to process this new input event, DPLL_ACT_STA_shadow[ACT_N(t)] is not yet set, so there is no request to transmit the output data to the ARU. In this case, an action calculation is finished without transferring the data via the ARU. PMT calculations where the result is not "past" are not affected by this

issue. The time frame in which an incoming input signal causes the issue is about 25 system clock cycles.

**Workaround:**

Use a Multi Channel Sequencer (MCS) channel to read the PMT data from DPLL via non blocking ARU reads using the NARD or NARDI instructions. The data that is read should be tested to check whether the requested action is 'out of time' by reading the Time Base Unit Time Stamp Channel 0 (TBU_TS0), or 'out of angle' by reading the Time Base Unit Time Stamp Channel 1 or 2 (TBU_TS1/2). If the TBU_TSx values are not within an acceptable window of the PMT value, the MCS can request the old value again, or if the requested event is in the past, request a new value.

Additionally, the Timer Input Module 0 (TIM0) interrupt can be routed to the MCS to check if an active edge occurred. Each action which delivers a PMT result should be checked only once by the MCS. If the PMT result is transferred directly from the DPLL to the ARU connected Timer Output Module (ATOM), the MCS should be prepared to send a default value to the ATOM which is not in the past to ensure that even if the DPLL fails to send the PMT result to the ATOM, the ATOM does not miss the event completely.

## 1.81 ERR007532: M_TTCAN: Incorrect value of Reference Trigger Offset status for time slaves

**Description:**

When the Time Triggered Modular CAN (M_TTCAN) module is configured as time slave, read accesses to the Reference Trigger Offset (RTO) field of the TT Operation Status register (TTOST) always return the value 0x7F when the Error Level (EL) 2-bit field of TTOST is greater than 0b00, signalling the presence of error in the TTCAN operation.

The M_TTCAN should return the value configured in the Initial Reference Trigger Offset (IRTO) field of the TT Operation Configuration register (TTOCF).

**Workaround:**

Ignore the value of the RTO field when reading the TTOST register for time slaves.

## 1.82 ERR007538: M_(TT)CAN: Switch between CAN operating modes during transmission or reception may be ignored

**Description:**

When the Flexible Data rate (FD) mode is enabled in the Modular CAN (M_CAN) or in the Time Triggered Modular CAN (M_TTCAN), in other words, the 2-bit wide CAN Mode Enable field (CME) of the CC Control Register (CCCR) is not set to 0b00, request for change of CAN operation mode may be ignored if a frame reception or transmission is in process.

The M_(TT)CAN supports three modes of operation:

- A: CAN 2.0 mode
- B: CAN FD mode
- C: CAN FD mode with bit rate switching enabled

A change in the operation mode is done by writing the Change Mode Request field (CMR) of the CC Control Register (CCCR).

The affected transitions are between {A and B} or {B and C} modes.

The request is acknowledged (CCCR[CMR] reverts to be 0b00) but the M_(TT)CAN remains in its previous operation mode.

**Workaround:**

Verify the successful switch of operation mode by reading the CAN FD Bit Rate Switching (FDBS) and the CAN FD Operation (FDO) bits of the CCCR register. If the values do not match the intended operation mode, repeat the mode change request (write to CCCR[CMR]) and the check operation until the change succeeds.

## 1.83 ERR007587: SSCM: Multi-bit ECC error at RCHW locations will cause device to remain in reset as a security and safety precaution

**Description:**

The System Status and Control Module (SSCM) checks life cycle, Device Configuration Format (DCF) records, and all possible Reset Configuration Half-Word (RCHW) boot header locations in the Flash memory, and if any contain a multi-bit Error Correction Code (ECC) error, then the device will remain in reset to prevent improper code from being executed.

Once in this mode, it is not possible to exit reset to attempt to reprogram the Flash. Applications must avoid ECC errors at these locations. Interrupting flash program or erase can result in ECC errors at the programmed location, or in the block being erased. DCF records may only be programmed one-time, and must be programmed successfully without error.

The RCHW locations that are searched and must not have ECC errors are:

| Address |
|---|
| 0x60_C000 |
| 0x61_0000 |
| 0x62_0000 |
| 0xFC_0000 |
| 0xFC_4000 |
| 0xFC_8000 |
| 0xFC_C000 |
| 0x100_0000 |
| 0x104_0000 |
| 0x108_0000 |
| 0x10C_0000 |

All locations are checked, even if only one has a valid header.

**Workaround:**

When programming life cycle and DCF records, ensure that programming is not interrupted and can complete without error. When programming any of the locations in the RCHW list, ensure that programming is not interrupted and can complete without error. When erasing any block containing any of the locations in the RCHW list, ensure that the erase is not interrupted (especially by reset or loss of power).

## 1.84 ERR007589: LINFlexD: Spurious timeout error when switching from UART to LIN mode or when resetting LINTCSR[MODE] bit in LIN mode

**Description:**

If the LINFlexD module is enabled in Universal Asynchronous Receiver/Transmitter (UART) mode and the value of the MODE bit of the LIN Timeout Control Status register (LINTCSR) is 0 (default value after reset), any activity on the transmit or receive pins will cause an unwanted change in the value of the 8-bit field Output Compare Value 2 (OC2) of the LIN Output Compare register (LINOCR).

If the LINFlexD module is enabled in LIN mode and the value of the MODE bit of the LIN Timeout Control Status register (LINTCSR) is changed from ‘1’ to ‘0’, then the old value of the Output Compare Value 1 (OC1) and Output Compare Value 2 (OC2) of the LIN Output Compare register (LINOCR) is retained.

As a consequence, if the module is reconfigured from UART to Local Interconnect Network (LIN) mode, or LINTCSR MODE bit is changed from ‘1’ to ‘0’, an incorrect timeout exception is generated when the LIN communication starts.

**Workaround:**

If the LINFlexD module needs to be switched from UART mode to LIN mode, before writing UARTCR[UART] to 1, ensure that the LINTCSR[MODE] is first set to 1.

If the LINFlexD module is in LIN mode and LINTCSR[MODE] needs to be switched from 1 to 0 in between frames, the LINOCR must be set to 0xFFFF by software.

## 1.85 ERR007632: SENT: Incorrect rounding of the Status Nibble, Data Nibbles, and CRC Nibbles

**Description:**

The Single Edge Nibble Transmitter (SENT) Receiver does not properly round off incoming data to the nearest nibble. In some cases, the jitter specification in the SAE J2716 (SENT) specification dated January 2010 (revision 3) is not met. This leads to an incorrect rounding of the nibble data. As a result, the Channel n Fast Message Data Read Register (CHn_FMSG_DATA) or Channel n Fast Message Cyclic Redundancy Check (CRC) register (CHn_FMSG_CRC) values may not be correct, or the Message is not received at all due of a CRC mismatch.

**Workaround:**

The Nibble Length Variation Limit (NIB_LEN_VAR_LIMIT, bit 16 [Least Significant Bit is 31]) bit in the SENT Global Control (SRX_GBL_CTRL) register should be set to allow for additional tolerance to jitter and frequency variation in the received SENT signal. In addition, use a nominal micro-tick duration greater than or equal to 8us (with worst case u-tick duration of -25 % which is greater than or equal to 6us) if the High Frequency (protocol clock) is 80 MHz.

For a 40 MHz protocol clock, the nominal u-tick duration must be greater than or equal to 13 µs (with worst case u-tick duration of -25 % which is greater than or equal to 10 µs).

For a 100 MHz protocol clock, the nominal u-tick duration must be greater than or equal to 7 µs (with worst case u-tick duration of -25 % which is greater than or equal to 5 µs).

The protocol clock frequency (40, 80, or 100 MHz) should be set per the device capabilities.

## 1.86 ERR007652: M_CAN: TTCAN triggers protect part of memory

**Description:**

The section of the shared Controller Area Network (CAN) memory in the range 0xFFED_6D00 - 0xFFED_6EFF is protected and cannot be written by the microcontroller core processor unless the Configuration Change Enable (CCE) bit and the Initialization bit (INIT) in the CAN Core Control Register (CCCR) of the Time-Triggered (TT) Modular Controller Area Network module (M_TTCAN) are both set (1).

This protected area is targeted to be used for Triggers in TT mode but even when the TT functionality is not used, the protection is active when CCE and INIT bits of the M_TTCAN module are cleared.

**Workaround:**

The CCE and INIT bits must be set (1) in the M_TTCAN_CCCR to allow write access by the processor core to initialize this range (0xFFED_6D00 - 0xFFED_6EFF) of the CAN SRAM for use by the M_(TT)CAN modules. Once the CCE and INIT bits are cleared (0), read accesses are possible.

## 1.87 ERR007701: STCU2: Short Functional Reset reaction and Long Functional Reset reaction of the FCCU does not take effect upon PLL1 Loss-Of-Lock while MBIST ONLINE is running

**Description:**

Fault Collection and Control Unit (FCCU) implements Short Functional Reset reaction and Long Functional Reset reaction in case a fault is triggered by one of the FCCU channels. In particular, the FCCU channel 30, that monitors the loss of lock of PLL1, may implement Short Functional Reset reaction and Long Functional Reset reaction depending on the field RFSC30 of the Recoverable Fault State Configuration 1 register (FCCU_RFS_CFG1[RFSC30]).

On this device, PLL1 loss-of-lock will not trigger a reset reaction while MBIST ONLINE is running, whereas it correctly triggers a reset reaction in all other conditions.

The status provided by Self Test Control Unit (STCU2) error register (STCU_ERR_STAT) is

correct, i.e. the On-Line LOCK Error flag (LOCKESW) is set.

**Workaround:**

Do not rely on the FCCU to detect a PLL1 Loss of lock while MBIST ONLINE is ongoing. After its completion, check status of the STCU_ERR_STAT[LOCKESW] bit to detect if a loss of lock occurred.

Use an alternative method to dynamically check PLL1 Loss Of Lock, for example by using the CMU_1, CMU_2 or CMU_3 to monitor PLL1 clock output and trigger a fault to the FCCU.

## 1.88 ERR007750: PAD_RING: Incorrect control of internal pull-up and pull-down on GPIO PB[5] and PE[14]

**Description:**

The control of the internal pull-up and pull-down of the pins Port B[5] (PB[5], analog channel 35) and Port E[14] (PE[14], analog channel 7) is incorrectly implemented.

The bits Weak Pull-up Enable (WPUE) and Weak Pull-down Enable (WPDE) of the System

Integration Unit Lite (SIUL2) Multiplexed Signal Configuration Register 21 (SIUL2.MSCR21) for the pin PB[5] and SIUL2.MSCR78 for the pin PE[14] behaves as follows:

| WPDE | WPUE | Pull state and Resistance |
|---|---|---|
| 0 | 0 | Pull-down is active, pull-up is inactive, Resistance can be as low as 100 ohms |
| 1 | 1 | Pull-up and pull-down are inactive, The pad is high impedance |
| 1 | 0 | Pull-down is active, pull-up is inactive, Resistance can be as low as 100 ohms |
| 0 | 1 | Pull-down is active with a resistance as low as 100 ohms. Pull-up is active with a resistance as low as 25 kΩ. Small cross current can flow between the high voltage Input/Output (VDD_HV_IO) supply and VSS_HV_IO ground (200 µA) |

**Workaround:**

Set the configuration {WPDE=1 and WPUE=1} for analog conversions.

Do not depend on low impedance connection to ground for safety switch operation.

## 1.89 ERR007772: PMC: a SWT_2 destructive reset may be forced if the 5 V supply is over voltage for longer than 16 ms

**Description:**

When the High Voltage Detect, HVD600_C, that monitors the high voltage supply, asserts, setting the Voltage Detect 15 flag in the Power Management Controller (PMC) Reset Event Pending Register (PMCDIG_EPR_VD15[HVD15_C] due to an over voltage condition, a reset occurs.

If this reset is configured to be a functional reset and the voltage is over voltage for more than 16ms, the Software Watchdog Timer 2 (SWT_2) will time out while waiting for the voltage to return to a valid range.

The SWT_2 timeout will cause a destructive reset and indicate that there was an issue with Flash Initialization. Destructive resets are not affected.

**Workaround:**

If HVD600_C is configured to be a functional reset in the PMC Reset Event Select Register for Voltage Detect 15 (PMCDIG_RES_VD15[HVD15_C]), then software should expect that SWT_2 may time out and cause a destructive reset. Destructive resets are not affected.

## 1.90 ERR007788: SIUL2: A transfer error is not generated for 8-bit accesses to non-existent MSCRs

**Description:**

An 8-bit access attempt to non-existent MSCRs (Multiplexed Signal Configuration Registers) in the SIUL2 (System Integration Unit Light 2) address space does not generate a transfer error. 16-bit or 32-bit accesses to non-existent MSCRs will generate a transfer error.

**Workaround:**

Do not expect transfer errors on 8-bit accesses to non-existent MSCRs in the SIUL2 address space.

## 1.91 ERR007791: SIUL2: Transfer error not generated if reserved addresses within the range of SIUL BASE + 0x100 to 0x23F are accessed

**Description:**

If any reserved register within the System Integration Unit Lite 2 (SIUL2) register range from SIUL2 BASE + 0x100 to 0x23F is accessed then no transfer error will occur.

**Workaround:**

Software should not be dependent on the indication of a transfer error occurring from an access within the SIUL2 register range from SIUL2 BASE + 0x100 to 0x23F.

## 1.92 ERR007796: M_CAN: Message reception and transmission directly after detection of Protocol Exception Event

**Description:**

In the Modular CAN (M_CAN) module, if a Flexible Data rate (FD) frame is received and the reserved bit (res) following the FD frame format bit (FDF) is recessive, the protocol controller correctly detects a Protocol Exception Event. The reception of the message is not finished and the message is discarded but the first message after this Protocol Exception Event will generate the following wrong behaviors.

Case 1: Message reception directly after Protocol Exception Event

The Message RAM Access Failure flag (MRAF) of the M_CAN Interrupt Register (IR) is set even if there was no incorrect access to the Message RAM and the frame has been correctly received.

Case 2: Message transmission directly after Protocol Exception Event

The first frame transmitted after a Protocol Exception Event is transmitted with a faulty frame format. In this case, the MRAF bit is not set.

The other nodes on the CAN network will react to this faulty format and generate error frames causing the M_CAN cell to resend the frame, with a correct format.

**Workaround:**

For reception of messages: ignore the MRAF error. The MRAF signal is primarily intended to validate the correct integration of the M_CAN within a device.

For transmission of messages: the other node(s) will detect the error and trigger the resend of the frame.

## 1.93 ERR007801: WKPU: functional NMI filter enable trigger FCCU fault monitor channel #47

**Description:**

The Wake Up Unit (WKPU) supports a glitch filter for the Non-maskable Interrupt (NMI) pin. This filter can be enabled through the NMI Filter Enable bit in the NMI Configuration Register (WKPU.NCR[NFE0]) and is disabled by default.

When this glitch filter is enabled, the Fault Collection and Control Unit (FCCU) channel #47 will be permanently asserted. This prevents monitoring of the Test Circuitry Group 2 (DFT2) fault reporting associated with channel #47, indicating that circuitry has been put into a non-functional (test) mode.

**Workaround:**

Do not enable the NMI glitch filter (set WKPU.NCR[NFE0]=0).

## 1.94 ERR007824: DCI: Avoid asserting system reset when switching JTAG operating modes

**Description:**

Assertion of system reset during the transition of the debug pin operating mode, either from JTAG pin mode to the LVDS Fast Asynchronous Serial Transmission (LFAST) pin mode, or from LFAST pin mode to JTAG pin mode, could result in a loss of synchronization with the debugger or a reset of the debug system.

**Workaround:**

Tools should not assert system reset while the Debug and Calibration Interface (DCI) is switching the pin operating mode from JTAG to LFAST, or from LFAST to JTAG. If a system reset occurs due to any other conditions, the tool may lose communication with the microcontroller. If this occurs, the tool should reset the JTAG interface by toggling JCOMP (DEBUG_RXN) low (ground) while holding the TDO (DEBUG_RXP) pin either high or low. This forces the interface operation back to JTAG operating mode. This requires that the Enable Escape mode feature be enabled in the DCI Control Register (DCI_CR[EN_ESC_MODE] = 1).

## 1.95 ERR007840: M_CAN: Change of operation mode during start of transmission

**Description:**

In the Modular CAN (M_CAN) module, when the transmit Event FIFO is used and a change of CAN operation mode is requested (writing to the CAN Mode Request (CMR) field of the CAN Core Control Register (CCCR)) during the start of transmission, the following incorrect behaviors will occur.

Case 1: Change from classic CAN frame to Flexible Data rate (FD) frame with bit rate switching.

The Extended Data Length (EDL) and Bit Rate Switch (BRS) bits of the related Tx Event FIFO element do not match with the transmitted frame type. They signal a CAN FD frame with bit rate switching (EDL and BRS bits are set) while a classic CAN frame was transmitted.

Case 2: Change from classic CAN to CAN FD without bit rate switching.

The EDL bit of the related Tx Event FIFO element does not match with the transmitted frame type.

It signals a CAN FD frame without bit rate switching (EDL is set) while a classic CAN frame was transmitted.

Case 3: Change from CAN FD with bit rate switching to CAN FD without bit rate switching

The BRS bit of the related Tx Event FIFO element does not match with the transmitted frame type.

It signals a CAN FD frame without bit rate switching while a CAN FD frame with bit rate switching was transmitted.

Case 4: Change from CAN FD without bit rate switching to CAN FD with bit rate switching

The BRS bit of the related Tx Event FIFO element does not match with the transmitted frame type.

It signals a CAN FD frame with bit rate switching while a CAN FD frame without bit rate switching was transmitted.

Case 5: Change from CAN FD with/without bit rate switching to Classic.

The Message RAM Access Failure flag (MRAF) of the M_CAN module Interrupt Register (IR) is set (IR.MRAF = 1), the M_CAN switches to Restricted Operation Mode and the transmission is aborted.

**Workaround:**

Wait that all the Transmission Request Pending n flags (TRPn) of the Tx Buffer Request Pending register (TXBRP) are cleared (TXBRP.TRPn = '0') before changing the CAN operation mode.

## 1.96 ERR007841: M_CAN: Incorrect frame transmission after recovery from Restricted Operation Mode

**Description:**

When the Modular CAN (M_CAN) module detects a Message RAM Access Failure (MRAF) during a frame transmission, it sets the MRAF bit of the Interrupt Register (IR) and enters the restricted operation mode. The Restricted Operation Mode (ASM) field of the CAN Core Control Register [CCCR] is set. During the first transmission after leaving the Restricted Operation Mode by resetting the CCCR.ASM bit, a frame with an unexpected identifier and control field may be transmitted and could be accepted and acknowledged by a receiver.

**Workaround:**

Use the following procedure to exit from the Restricted Operation Mode:

Step 1: Cancel all pending transmission requests by writing 0xFFFF_FFFF to Transmit (TX) Buffer Cancellation Request (TXBCR) register

Step 2: Issue a clock stop request by setting the Clock Stop Request (CSR) bit of the CC Control Register (CCCR)

Step 3: Wait until the M_CAN sets the Initialization (INIT) and Clock Stop Acknowledge (CSA) bits of the CC Control Register (CCCR) to one

Step 4: First clear CSR bit

Step 5: Then clear INIT bit

Step 6: Wait until INIT is read as zero

Step 7: Issue a second clock stop request by setting CSR bit

Step 8: Wait until the M_CAN sets INIT and CSA bits to one

Step 9: Set the Configuration Change Enable (CCE) bit of the CC Control Register (CCCR), clear CSR and ASM bits in a single write operation.

Step 10: Restart M_CAN by clearing INIT bit.

Step 11: Configure the CAN operation mode by writing to the CAN Mode Request (CMR) field of the CC Control register.

Step 12: Request the transmissions canceled by step one

## 1.97 ERR007842: M_CAN: Erroneous Interrupt flag after setting / resetting INIT during frame reception

**Description:**

In the Modular Controller Area Network (M_CAN) module, when the Initialization bit

(INIT) is set in the CAN Core Control Register (CCCR) during the reception of a frame, the first reception of a frame after clearing the INIT bit will be correctly received but the Message RAM Access Failure flag (MRAF) of the Interrupt Register (IR) will be set.

**Workaround:**

If the Initialization (INIT) bit of the CC Control Register (CCCR) needs to be set during operation, proceed as follows:

Step 1: Issue a clock stop request by setting the Clock Stop Request (CSR) bit of CCCR register

Step 2: Wait until the M_CAN sets INIT and the Clock Stop Acknowledge (CSA) bits of the CCCR register to one

Before clearing INIT, first clear CSR.

## 1.98 ERR007847: GTM: MCS's CAT status may be incorrect

**Description:**

(GTM-IP-178)

The Generic Timer Module (GTM) Multi-channel Sequencer's (MCS) Advance Routing Unit (ARU) blocking read/write instructions, such as ARD, AWR, ARDI, and AWRI, describe the use of the Cancel ARU Transfer (CAT) status field to check whether the last ARU transfer was successful (CAT=0) or canceled (CAT=1). Because CAT can be written by software to cancel an ARU transfer at any time, CAT does not reliably reflect the last ARU transfer status.

**Workaround:**

Check data consistency of the ARU transfer by inspecting the transferred data (for example, check for the linear increment of the edge counter (ECNT) for data transfers from Timer Input Module (TIM) to the MCS) instead of relying on the CAT field.

## 1.99 ERR007848: GTM: Bit 0 of TIM edge counter register may not indicate the actual signal level

**Description:**

(GTM-IP-181)

When a Generic Timer Module (GTM) Timer Input Module (TIM) channel is enabled, bit 0 of the Edge Counter register (ECNT) may not reflect the current signal level of the filtered input

TIM[i]_CH[x]_FOUT until the next input edge occurs. This issue occurs when the ECNT register is not read before re-enabling the channel.

This erratum does not affect TIM Bit Compression Mode (TBCM).

**Workaround:**

After disabling the TIM channel, ensure that the ECNT register is read at least once before the TIM channel is re-enabled. Alternatively, before re-enabling a TIM channel, issue a TIM channel reset and reconfigure the TIM channel control registers.

## 1.100 ERR007855: SENT: Integer division during calibration pulse measurement causes reduced robustness

**Description:**

The calculation of the compensated microticks during the calibration pulse requires a division by 56. This constant integer division introduces quantization error that accumulates over each microtick while receiving the Status, Data and Cyclic Redundancy Check (CRC) nibbles.

This accumulated error leads to a reduced jitter tolerance window. As a result, incorrect values of nibbles may get sampled. Compensated tick period can be read from Channel ‘n' Clock Control Register (SRX_CHn_CLK_CTRL), Compensated Prescaler value (CM_PRSC) field.

**Workaround:**

The Nibble Length Variation Limit (NIB_LEN_VAR_LIMIT, bit 16 [Least Significant Bit is 31]) bit in the SENT Global Control (SRX_GBL_CTRL) register should be set to allow for additional tolerance to jitter and frequency variation in the received SENT signal.

In addition, use a nominal micro-tick duration greater than or equal to 8 µs (with worst case u-tick duration of -25 % which is greater than or equal to 6us) if the High Frequency (protocol clock) is 80 MHz.

For a 40 MHz protocol clock, the nominal u-tick duration must be greater than or equal to 13us (with worst case u-tick duration of -25 % which is greater than or equal to 10 µs).

For a 100 MHz protocol clock, the nominal u-tick duration must be greater than or equal to 7 µs (with worst case u-tick duration of -25 % which is greater than or equal to 5 µs).

The protocol clock frequency (40, 80, or 100 MHz) should be set per the device capabilities.

## 1.101 ERR007869: FCCU: FOSU monitoring of a fault is blocked for second or later occurrence of the same fault

**Description:**

The Fault Collection and Control Unit (FCCU) Output Supervision Unit (FOSU) will not monitor the FCCU for the second or later occurrence of a given fault in the following cases:

1. Reset is programmed as the only reaction for the fault.
2. Assertion of the fault coincides with the long/short functional reset reaction to a fault previously asserted.

**Workaround:**

There are two possible workarounds. Either one can be used with same effectiveness.

1. In addition to the reset reaction, enable either the interrupt (IRQ) or Non-maskable Interrupt (NMI) or error out signaling reaction for the faults that have a reset reaction enabled.
2. Apply the following procedure during the FCCU configuration after a reset and in the fault service routine while clearing the fault status inside the FCCU.
   a) Check for FCCU pending faults and clear them.
   b) Configure the FCCU as desired.
   c) Enable a fault as software recoverable by setting its corresponding bit in the NCF Configuration Register (FCCU_NCF_CFGn)
   d) Inject a fake fault to the fault set up in step “c” by writing the corresponding code into the NCF Fake Register (FCCU_NCFF)
   e) Check that there are no pending faults else clear the pending faults and repeat steps “d” and “e”
   f) Reconfigure the fault that was configured for software recovery mode.

## 1.102 ERR007873: SPC574KxB: Current injection causes leakage path across the DSPI and LFAST LVDS pins

**Description:**

The General Purpose Input/Output (GPIO) digital pins (including all digital CMOS input or output functions of the pin) connected to the differential LVDS drivers of the Deserial/Serial Peripheral Interface (DSPI) and LVDS Fast Asynchronous Serial Transmit Interface (LFAST) do not meet the current injection specification given in the operating conditions of the device electrical specification. When the LVDS transmitter or receiver is disabled and current is positively or negatively injected into one pin of the GPIO pins connected to the differential pair, a leakage path across the internal termination resistor of the receiver or through the output driver occurs potentially corrupting data on the complementary GPIO pin of the differential pair. All LFAST and DSPI LVDS receiver and transmitter GPIO pairs on the SPC574KxB exhibit the current injection issue.

There is an additional leakage path for the LFAST pins through the loop back test path when

current is negatively injected into a GPIO pin connected to an LFAST pair. In this case current will be injected into the same terminal of the GPIO pin connected through the loop back path (terminal to positive terminal, negative terminal to negative terminal). Two sets of pins are affected by the loop back path on the SPC574KxB:

- TXP/TXN = PD[6]/PA[14] and RXP/RXN = PF[13]/PD[7],
- TXP/TXN = PA[7]/PA[8] and RXP/RXN =PA[9]/PA[5].

There is no leakage issue when the pins are operating in normal LVDS mode (both LVDS pairs of the LFAST interface configured as LVDS).

**Workaround:**

As long as the GPIO pad pins are operated between ground (VSS_HV_IO) and the Input/Output supply (VDD_HV_IO) then no leakage current between the differential pins occurs. If the GPIO pad is configured as an input buffer then the input voltage cannot be above the supply, below ground, and no current injection is allowed. If the GPIO pad is configured as an output care should be taken to prevent undershoot/overshoot/ringing during transient switching of capacitive loads. This can be done by carefully configuring the output drive strength to the capacitive load and ensuring board traces match the

characteristic impedance of the output buffer to critically damp the rising and falling edges of the output signal.

## 1.103 ERR007904: PASS: Programming Group Lock bit (PGL) can be de-asserted by multiple masters writing the correct password sections to the CINn registers.

**Description:**

The eight Challenge Input Registers (CINn) in the Password and Device Security

Module (PASS) where the 256-bit unlock lock password (8 x 32-bit registers) is provided, can be written by multiple masters. If the written password is correct even though it has been provided from different masters, the password Group Lock (PASS PGL) in the Password Group n Lock 3 Status register (PASS_LOCK3_PGn) is de-asserted and UnLockMaster (MSTR) is set to 0xF.

Therefore, internal registers would not be writable by any of the master other than master whose ID is 0xF if the Master Only (MO) bit is set PASS_LOCK3_PGn.

If a Master wants to update internal registers, it needs to unlock the PASS by writing into all the 8 Password registers.

**Workaround:**

Set the master only bit inside the PASS (LOCK3_PGn.MSTR) to block other master accesses to the unlocked registers. If the written password has been provided from different

masters, a single master should perform the unlock operation again by writing into all the 8 password registers.

## 1.104 ERR007905: PIT: Accessing the PIT by peripheral interface may fail immediately after enabling the PIT peripheral clock

**Description:**

If a write to the Periodic Interrupt Timer (PIT) module enable bit (PIT_MCR[MDIS])

occurs within two bus clock cycles of enabling the PIT clock gate in the MC_CGM (Clock Generation Module) register, the write will be ignored and the PIT will not be enabled.

**Workaround:**

After enabling the PIT clock in the MC_CGM, insert a read of the PIT_MCR register before writing to the PIT_MCR register. This guarantees a minimum delay of two bus clocks to guarantee the write is not ignored.

## 1.105 ERR007911: MC_ME: Decorated accesses not supported to core local memories during reset

**Description:**

Resetting a core through a Mode Entry module (ME) mode change, while a decorated access is being performed to that core's local Instruction (I-MEM) or data memory (D-MEM) by a different core may cause the decorated access to end prematurely (without completion). A reset of a core is performed by setting the Reset on Mode Change bit of the Core Address register (ME_CADDR1[RMC] or ME_CADDR3[RMC] = 1) and then performing a mode change by writing the proper sequence to the Mode Entry Mode Control register.

**Workaround:**

There are 3 possible options for preventing loss of a decorated access:

1. Do not perform decorated accesses of either the safety (core 0) or the computational (core 1) core by a different core. (Allow decorated accesses to be performed only by a core to its own local memory. Do not allow decorated accesses to a different cores local memory.)
2. Do not use the mode entry module to reset or reboot the safety or computational core.
3. Implement a software controlled reset lock prior to performing any decorated access to the computational or safety cores' local memory that prevents the mode change from occurring until after the decorated access is completed. This requires that a flag be set and cleared before and after the decorated access occurs. Cores that perform mode change resets of other cores would need to check this flag prior performing a mode change controlled reset of cores.

## 1.106 ERR007932: NAR/SIPI: Part ID for NAR and Debug SIPI does not match the MIDR MINOR_MASK

**Description:**

The revision portion of Device Part Identification (ID) register in the Nexus Aurora Router (NAR) and the Serial Interprocessor Interface (SIPI) modules reads 0x0. However, it should match the mask minor revision (MINOR_MASK) in the System Integration Unit Lite (SIUL) Microcontroller Identification Register (MIDR), which is 0x1.

**Workaround:**

Write software such that it does not depend on the minor revision portion of SIPI Part

ID matching between the SIPI ID (value = 0x0) and MIDR MINOR_MASK (value = 0x1). Tools should use the JTAG Identification for the minor revision value instead of the NAR ID.

## 1.107 ERR007934: FEC: MDC and MDIO timing requirements and configuration

**Description:**

The timing specifications for the Fast Ethernet Controller Management Data Clock Output (MDC) and Management Data IO (MDIO) should indicate that the MDC and MDIO pins must be configured for the same drive strength (strong or medium). In addition, specification M10 "MDC falling edge to MDIO output invalid (minimum propagation delay)" Minimum value should be -10 ns.

**Workaround:**

Configure the FEC MDC and MDIO signals pin drive strength to the same strength (strong or medium) in the System Integration Unit Lite (SIUL2_MSCR_IO_*). Timing should be verified with specification M10 having a value of -10 ns.

## 1.108 ERR007935: PMC: Accessing the HVD_FLASH divider tap point through the ADC test channel may cause a device reset

**Description:**

If the Power Management Controller (PMC) Analog to Digital Converter select (PMC_ADC_CS[ADC_CHSE]) register is set to select the high voltage flash divider tap point (HVD_VFLASH, value 6'b000110), or the select hvd600 divider tap point (HVD_HV, value 6'b001010), and then the selected channel is read using the Successive Approximation Register Analog to Digital Converter 0 (SAR_ADC0) channel 34, a power on reset event (specifically (POR260_C) may occur which causes the microcontroller to reset.

**Workaround:**

Do not access the PMC signals HVD_VFLASH or hvd600 divider point using SAR_ADC0 channel 34.

## 1.109 ERR007947: XOSC: Incorrect external oscillator status flag after CMU event clear

**Description:**

If an external oscillator (XOSC) is enabled and it becomes unstable (or the crystal fails), the Oscillator Lost Reference status flag in the Clock Monitor Unit Interrupt Status register (CMU0.CMU_ISR[OLRI]) will be set. In addition, the Crystal Oscillator Status flag in the Mode Entry module Global Status Register (MC_ME_GS.S_XOSC) will be cleared (1 = stable clock, 0 = no valid clock). However, if the CMU_ISR[OLRI] is cleared while the oscillator is still in a failing condition, the MC_ME_GS.S_XOSC will incorrectly be set, indicating a valid crystal oscillator.

**Workaround:**

Monitor the XOSC external oscillator status using the MC_ME_GS.S_XOSC before

the CMU0.CMU_ISR.OLRI flag is set. After the CMU0.CMU_ISR.OLRI flag has been set, the MC_ME_GS.S_XOSC flag is valid only after a functional reset. Alternately, the response

to the OLRI flag after loss of XOSC clock, can be set in the FCCU to cause a functional reset to clear the MC_ME_GS.S_XOSC flag.

## 1.110 ERR007960: FCCU: Channels associated to MEMU cannot be activated through the fake fault register (FCCU_RFF)

**Description:**

The Fault Collection and Control Unit (FCCU) implements the Recoverable Fault Fake register (FCCU_RFF) that allows software application to trigger the so-called fake faults on the FCCU's channels in order to verify the entire path and reaction.

Channels from 16 to 24, associated to the Memory Error Management Unit (MEMU), cannot be triggered using fake fault mechanism:

- Channel 16: reporting occurrence of correctable fault in system RAMs;
- Channel 17: reporting occurrence of uncorrectable fault in system RAMs;
- Channel 18: reporting occurrence of overflow of system RAM ECC buffer;
- Channel 19: reporting occurrence of correctable fault in peripheral RAMs;
- Channel 20: reporting occurrence of uncorrectable fault in peripheral RAMs;
- Channel 21: reporting occurrence of overflow of peripheral RAM ECC buffer;
- Channel 22: reporting occurrence of correctable fault in flash memory;
- Channel 23: reporting occurrence of uncorrectable fault in flash memory;
- Channel 24: reporting occurrence of overflow of flash memory ECC buffer.

**Workaround:**

In order to verify the entire path and reaction associated to these channels, the associated fault can be simulated by writing the dedicated register in the MEMU module:

- channel 16: MEMU.DEBUG[FR_SR_CE] to '1';
- channel 17: MEMU.DEBUG[FR_SR_UCE] to '1';
- channel 18: MEMU.DEBUG[FR_SR_EBO] to '1';
- channel 19: MEMU.DEBUG[FR_PR_CE] to '1';
- channel 20: MEMU.DEBUG[FR_PR_UCE] to '1';
- channel 21: MEMU.DEBUG[FR_PR_EBO] to '1';
- channel 22: MEMU.DEBUG[FR_F_CE] to '1';
- channel 23: MEMU.DEBUG[FR_F_UCE] to '1';
- channel 24: MEMU.DEBUG[FR_F_EBO] to '1'.

## 1.111 ERR007981: LBIST: LBIST of the flash may leave flash in an unknown state and stress flash bit cells

**Description:**

Performing a Logical Built-In Self-test (LBIST) on the flash leaves the flash in an unknown state prior to reset. During this time (after performing the LBIST and the microcontroller [MCU] being reset), the flash array may be disturbed.

**Workaround:**

Do not perform LBIST on the partition which contains the flash.

## 1.112 ERR007996: PSI5: Incorrect SMC message decoding and timestamp generation in case of late last sensor message overlapping with next SYNC period pulse

**Description:**

As stated in section 6.6 of Peripheral Sensor Interface (PSI5) Standard v2.0 and v2.1, PSI5 sensor frames should not overlap with the SYNC pulse. This overlap is considered to be an error condition. In extension to the standard requirement, the PSI5 module implemented in this device allow the possibility to manage this overlap error condition.

In case of such overlap condition:

- PSI5 message extraction happens correctly
- Timing bit error [T] and CRC error [C] flags are correctly set within the PSI5 message
- The serial messaging channel (SMC) frame counter gets reset on Sync pulse and the extraction of SMC message does not happen correctly, resulting in loss of SMC message

The PSI5 module correctly handles this error condition for the PSI5 message, but it is not able to handle the SMC message correctly.

Also during this overlap condition, the timestamp appended with the overlapped slot is the new sync pulse timestamp. Whenever a sync pulse comes, the internal sync pulse timestamp capture registers are updated with the timestamp of the new sync pulse. Hence if any message overlaps with the sync pulse and that message slot is configured to capture the timestamp of the SYNC pulse (PSI5_SnFCR[TS_CAPT] = 1), then the timestamp appended for that slot is the timestamp corresponding to the new sync pulse and not of the previous sync pulse belonging to the PSI5 frame.

**Workaround:**

The PSI5 message is properly received and analyzed with the appropriate error flags set. Application software can identify from the properly received PSI5 message that an error on the bus occurred. If an SMC message is configured to be present in the slot, application software can request an immediate halt and restart of the SMC or it can wait until the SMC reception has finished and check the CRC of the SMC message:

- if incorrect, a resend of the SMC message can be re-started at that point in time. Further, application software can check the timestamp of the previous message in the

final slot from the previous SYNC period to identify if the wrong timestamp has been captured

- if the difference between the two messages is equivalent to two SYNC periods, application software can correctly identify that an overlap of the final message with the following SYNC pulse has occurred assuming the slot is configured to capture the SYNC pulse timestamp (PSI5_SnFCR[TS_CAPT] = 1) rather than the message timestamp (PSI5_SnFCR[TS_CAPT] = 0)

## 1.113 ERR008005: DSMC: Software reset of core0/core0s may trigger FCCU channel #11 event

**Description:**

Two separate Decorated Storage Memory Controller (DSMC) modules are implemented for the access to core0 (safety core) local memories:

- a reference DSMC module
- a checker DSMC module

The operations of the DSMC modules are compared via a Redundancy Control and Checker Unit (RCCU).

The reference and checker DSMC modules use separated reset signals.

The reference DSMC reset signal is incorrectly managed during a core0 software reset. When the last master accessing the DSMC before core0 reset is different than the default master after reset, a mismatch is detected between the Reference and checker DSMC and signaled to the Fault Collection and Control Unit (FCCU) Channel #11.

Note that only executing a software reset of the core0 is impacting the reference DSMC. Software reset of any other core(s), plus functional and destructive reset of the device will not generate a mismatch.

**Workaround:**

Before triggering a software reset of core0, the crossbar (XBAR) configuration should ensure that the default master for slaves accessing local memory is core0 (reset configuration).

The Core0 local memories are accessed via the Slave2 port of the XBAR_0. The Parking Control (PCTL) and Park (PARK) field of the XBAR_0 Control Register 2 must be set to the default value:

- XBAR_0.CHANNEL[2].CRS.PCTL = 0;
- XBAR_0.CHANNEL[2].CRS.PARK = 0;

## 1.114 ERR008019: PMC: Escalation of LVD and HVD functional resets should be expected

**Description:**

There are two conditions where Low Voltage Detect (LVD) and/or High Voltage Detects (HVD) functional resets may be escalated to a destructive reset.

First, as expected, if the ramp rate of the supply slowly exceeds a HVD trip point, the HVD may assert multiple times. These multiple assertions can cause a functional reset escalation to a destructive reset.

The LVDs and HVDs themselves cannot cause a destructive reset escalation.

Second, some LVD/HVDs can be configured so that a functional reset escalation is possible independently of the supply ramp rate.

The list of these LVD/HVDs is:

| | |
|---|---|
| LVD108_P | Low voltage supply PLL medium range low voltage detector |
| LVD112_C | Low voltage supply core medium range low voltage detector |
| HVD145_C | Low Voltage supply core high voltage detector |
| HVD145_F | Flash LV supply high voltage detector (cold point) |
| LVD270_EBI | High voltage supply I/O Oscillator low range low voltage detector |
| LVD270_IF | FlexRay I/O Supply low voltage detector |
| LVD270_IF2 | Second FlexRay I/O low voltage detector |
| LVD360_IM | HV IO main supply low voltage detector |
| LVD400_A | HV ADC supply low voltage detector |
| LVD400_IM | HV IO main supply low voltage detector |
| HVD360_F | HV flash supply high voltage detector |
| HVD600_A | HV ADC supply high voltage detector |

The configuration to cause this issue is if the Device Configuration Format record (DCF) client assigned to the Power Management Controller digital Reset Event Enable register (PMC_DIG REE/PMC_REE_BUS) must disable the LVD/HVDs.

Also, the software must enable the LVD/HVD as a functional reset by writing to the PMC_DIG REE (to enable) and RES (to select functional resets) registers.

*Note: The following LVD/HVDs do not allow an exit from reset until the supply has returned to a valid range:*

| | |
|---|---|
| LVD108_C | LVD on Low voltage internal supply (hot point) |
| LVD108_F | LVD on Flash Low voltage supply (hot point) |
| LVD270_C | HV PMC supply low voltage detector |
| LVD270_F | HV flash supply low voltage detector |
| LVD270_IJ | JTAG I/O supply low voltage detector |
| LVD270_IM | HV IO main supply low voltage detector |
| LVD270_O | Oscillator supply low voltage detector |
| LVD295_F | HV flash supply low voltage detector |
| LVD295_A | HV ADC supply low voltage detector |
| HVD600_C | HV PMC supply high voltage detector |

These cannot cause a functional reset escalation even if configured via the PMC_REE_BUS DCF client and software enabling as described.

**Workaround:**

Use LVD/HVD resets as destructive resets only.

Alternatively, if functional resets for the LVD/HVDs are used, the Reset Generation Module (RGM) functional escalation register (MC_RGM_FRET) should be cleared before enabling the LVD/HVD.

Also, if functional resets for the LVD/HVDs are used, the resets should not be enabled until the current status of the LVD/HVD (shown in the PMC_DIG GR_S register) shows that the voltage is currently within a valid range.

## 1.115 ERR008034: FCCU: Channel 40 fault (COMP_XBIC_DSMC_Monitor) may be reported if a CPU is reset upon a MC_ME mode change

**Description:**

The Fault Collection and Control Unit (FCCU) channel 40 may report a fault during or after completion of a Mode Entry module (MC_ME) mode change where a CPU is reset in the mode change (setting MC_ME_CADDRn[RMC]=1). FCCU channel 40 (COMP_XBIC_DSMC_Monitor) indicates an addressing or control fault resulting in corrupted transaction through the computational XBAR or interference by system RAM Decorated Storage Memory Controller (DSMC).

**Workaround:**

Disable the computational XBAR slave port 4 parking control (XBAR_0_CRS4[PCTL]=10) prior to a mode change where a CPU will be reset, or expect and clear a FCCU channel 40 fault following the mode change if it occurs. Fault is cleared by clearing the FCCU channel status, FCCU_RF_Sn[RFSm].

## 1.116 ERR008039: SDADC: digital filter and FIFO not disabled when MCR[EN] is cleared

**Description:**

When the Enable bit (EN) of the Sigma-Delta Analog to Digital Converter (SDADC) Module Configuration Register (MCR) is cleared (MCR[EN]=0), the digital part of the SDADC continues operating and does not go to low power mode if the module is disabled while a valid conversion is already in process and the application software continues to initiate conversions. As a consequence, the digital block of the SDADC still produces new conversion results in the Channel Data Register (CDR) and dummy data are transferred to the result First-In, First-Out (FIFO) buffers. In addition, interrupt and/or Direct Memory Access (DMA) events are still generated.

*Note: The analog part does enter the power-down mode, reducing the consumption on the ADC high voltage supply domain (VDD_HV_ADV).*

**Workaround:**

Do not initiate a conversion prior to enabling the SDADC (MCR[EN]=1). In addition, once the SDADC has been enabled (MCR[EN]=1), if the SDADC needs to be disabled (MCR[EN]=0), prior to clearing the EN bit, either turn off the clock to the SDADC module in the Clock Generation Module (CGM) or Select the External Modulator Mode (EMSEL) by setting the MCR[EMSEL] bit along with the clearing the MCR[EN].

## 1.117 ERR008042: FCCU: EOUT signals are active, even when error out signaling is disabled

**Description:**

Every time the Fault Collection and Control Unit (FCCU) moves into fault state caused by an input fault for which the error out reaction is disabled (FCCU_EOUT_SIG_ENn[EOUTENx]=0), the Error Out 1 and 2 (EOUT[0] and EOUT[1]) will

become active for a duration of 250 us plus the value programmed into the FCCU Delta Time register (FCCU_DELTA_T[DELTA_T]). EOUT is not affected if the FCCU moves into the alarm state that generates an interrupt (IRQ), if the Fault is cleared before the alarm timeout.

This erratum does not affect the outputs of other pins (for example, for communication modules like CAN/Flexray). Only the EOUT signal is impacted.

**Workaround:**

There are three possible workarounds:

1. Enable EOUT signaling for all enabled error sources.
2. In case external device (which evaluates EOUT) can communicate with the MCU, the following procedure could be used:
   a) Program any duration of EOUT as per application needs (FCCU_DELTA_T[DELTA_T])
   b) For faults requiring error out reaction, the software shall validate EOUT via separate communication channel (like I2C) while EOUT is asserted.
   c) External device shall implement a timeout mechanism to monitor EOUT validation by separate channel.
   d) Following scenarios shall be considered as valid EOUT reactions:

      Validation is performed while EOUT is asserted

      Timeout occurs but no validation and EOUT is still asserted.
3. In case external device (which evaluates EOUT) cannot communicate with the MCU, following procedure could be used:
   a) Program the error out duration to a duration x (FCCU_DELTA_T[DELTA_T]).
   b) For faults requiring error out reaction, clear the fault after the pin has continued to be asserted for a longer duration (for example 2*duration x). This will artificially create a long pulse on EOUT.
   c) For faults which do not require error out reaction, clear the fault within duration x. This will artificially create a short pulse on EOUT.
   d) External device should ignore short pulse of duration x while recognizing longer pulses as valid reaction.
   e) While clearing the fault, the associated software shall check the pending faults.

## 1.118 ERR008054: PIT: DMA request stays asserted when initiated by PIT trigger, until PIT is reset

**Description:**

When a Periodic Interrupt Timer 0 (PIT0) channel trigger is used to initiate a Direct Memory Access (DMA) transfer, the DMA request does not negate at the end of the DMA transfer.

The result is that if that DMA channel is re-enabled, a subsequent PIT-triggered DMA transfer will be initiated.

**Workaround:**

Either do not use the PIT0 to initiate DMA transfers, or write software such that

anytime a PIT0 channel trigger is used to initiate a DMA transfer, the PIT0 module is then reset after that DMA transfer is completed, prior to re-enabling the DMA channel that was used for that transfer. The PIT0 module should be reset by setting the PIT_RTC_0 reset bit in the Peripheral Reset Register 0 in the Reset Generation Module.

Other timer systems, such as the System Timer Module (STM), can be used instead of the PIT to trigger the DMA.

## 1.119 ERR008056: LBIST: Flash must be idle during LBIST

**Description:**

Logic Built-In Self-Test (LBIST) should only be performed on the flash partition while the Flash is in the idle state. LBIST operations must not be initiated for the LBIST partition including Flash while flash program or erase operations are in progress. This information was not included in the user documentation.

**Workaround:**

Flash must be in an idle state for LBIST. Ensure that flash is not performing program, erase, or flash user accessible test mode operations when LBIST is initiated.

## 1.120 ERR008062: M_CAN: Frame transmission in DAR mode

**Description:**

In the Modular CAN (M_CAN) module, when the Disable Automatic Re-transmission bit is set in the CAN Core Control Register CCCR (CCCR[DAR]), the two following incorrect behaviors occur.

1. Transmission of a frame will cause the Event Type (ET) field of the Transmit Event FIFO Element to be incorrectly set to '0b01' (transmit event) instead of '0b10' (transmission in spite of cancellation).
2. When multiple messages are transmitted sequentially using the same Transmit buffer, after a successful transmission, the next transmission will not start if it is requested before the CAN bus becomes idle. This message is then treated as if it had lost arbitration.

**Workaround:**

Do not use the same transmit buffer for consecutive transmissions in DAR mode.

Or wait at least for 4 CAN bit times after successful transmission before requesting the next transmission from the same transmit buffer.

## 1.121 ERR008082: SENT: A message overflow can lead to a loss of frames combined with NUM_EDGES_ERR being set

In the case of a Single Edge Nibble Transfer (SENT) receiver (Rx) message overflow

(CHn_STATUS[FMSG_OFLW] = 1) and if the following registers are continuously being read without clearing the FMSG_RDY[F_RDYn] bit, there is a possibility that one message will be lost.

Additionally, if the pause pulse feature is enabled, the module assert up to two NUM_EDGES_ERR in the status register (CHn_STATUS). In this case up to two frames can be lost.

*Note: Some debuggers perform a continuous read of memory which can cause this issue to occur.*

| Register | Register Name |
|---|---|
| CHn_FMSG_DATA | Channel Fast Message Data Read Register |
| CHn_FMSG_CRC | Channel Cyclic Redundancy Check Register |
| CHn_FMSG_TS | Channel Fast Message Time-stamp Register |

**Workaround:**

1. Software should ensure that SENT message overflow does not occur.

If interrupts are used (when the Enable FDMA (FDMA_EN) bit of Fast Message DMA Control Register (SRX_FDMA_CTRL) is set to 0) to read the SENT messages, the interrupt for data reception should be enabled by setting the Enable for Fast Message Ready Interrupt (FRDY_IE[n]) bit of Fast Message Ready Interrupt Control Register (SRX_FRDY_IE) for every channel n and the interrupt priority should be such that the software is able to read the message before the next message arrives.

When using Direct Memory Accesses (eDMA) to access the SENT (when the Enable FDMA

(FDMA_EN) bit of Fast Message DMA Control Register (SRX_FDMA_CTRL) is set to 1), the DMA request from the SENT module should be serviced before the next message arrives. The minimum duration between the reception of two consecutive messages in one channel is 92 times the utick length (time).

2. Ensure that the following registers are not read continuously either in the software code or as a result of a debugger being connected. The following registers should be read once per message and the FMSG_RDY[F_RDYn] bit should be cleared after the reads.

| Register | Register Name |
|---|---|
| CHn_FMSG_DATA | Channel Fast Message Data Read Register |
| CHn_FMSG_CRC | Channel Cyclic Redundancy Check Register |
| CHn_FMSG_TS | Channel Fast Message Time-stamp Register |

## 1.122 ERR008089: TDM: Diary updates require PECIE bit of MCR register of flash to be set

**Description:**

In this device, when writing a record to the Tamper Detection Module (TDM) diary to

allow erase of a flash block, the Program Erase Complete Interrupt Enable bit in the Flash Module Configuration Register (Flash_MCR[PECIE]) must be enabled, or further erase of the flash block may be allowed until the next destructive reset.

This behavior is different from other devices in the family.

**Workaround:**

Set the Program/Erase Complete Interrupt Enable (PECIE) bit in the Flash Module Configuration Register (MCR) before doing any flash block erase operation which includes TDM diary update operation. The PECIE interrupt does not need to be processed by the

Interrupt controller. To prevent the interrupt from being processed its priority should be left at the default of 0 within the Interrupt controller.

Alternately, If PECIE is not set, perform a destructive reset after erasing the flash block, prior to programming it.

## 1.123 ERR008117: MC_ME: Restrictions on enabling FlexRay in low power modes

**Description:**

The FlexRay module is dependent on the Auxiliary (AUX) Clock 2, for which the only source clock is Phase Lock Loop 0 (PLL0). This dependency between FlexRay and PLL0 results in the following restrictions during entry to the low power modes, STOP0 and HALT0:

- Entry into STOP0 with FlexRay enabled (by setting MC_ME_PCTLx[LP_CFG=n] and MC_ME_LP_PCn[STOP0]=1) is not possible, even if the Crystal Oscillator (XOSC) is selected as the FlexRay protocol clock source (FR_MCR[CLKSEL]=0).
- Entry into HALT0 with FlexRay enabled (by setting MC_ME_PCTLx[LP_CFG=n] and MC_ME_LP_PCn[HALT0]=1) is possible only if the PLL0 is enabled during HALT0 (by setting MC_ME_HALT0_MC[PLL0ON]=1).

(For MC_ME_PCTLx, x=107 for FlexRay0 and x=235 for FlexRay1.)

**Workaround:**

To enter STOP0, the FlexRay must be disabled (MC_ME_PCTLx[LP_CFG=n] and MC_ME_LP_PCn[STOP0]=0). To enter HALT0 with FlexRay enabled, enable the PLL0 in HALT0 (by setting MC_ME_HALT0_MC[PLL0ON]=1) prior to entering HALT0.

(For MC_ME_PCTLx, x=107 for FlexRay0 and x=235 for FlexRay1.)

## 1.124 ERR008122: GTM: (A)TOM's CCU1 event interrupt is not generated when CM1=0 or 1 and RST_CCU0=1

**Description:**

(GTM-IP-202)

If a Generic Timer Module (GTM) Timer Output Module (TOM) or Advanced Router Unit (ARU) connected TOM (ATOM) channel is configured with the reset source of the channel as the previous channels trigger (CHn_CTRL[RST_CCU0]=1) and the counter 0 (CN0) counts from 0 to MAX, the Counter Compare Unit 1 (CCU1) event interrupt is not generated if the Compare Register 1 (CM1) is 0 and Compare Register 0 (CM0) is greater than 0. If the Compare Register (CM1) is 1 and Compare Register 0 (CM0) is MAX+1 only one CCU1 interrupt will be generated.

**Workaround:**

To trigger channel x+1 which is the channel that triggers when channel x counter CN0 is reset use the configuration of CM0=MAX and CM1=1.

- When the duty cycle configuration is CM1=0 and CM0>0 on channel x use the CCU0 interrupt of triggering channel x+1 instead of CCU1 interrupt.
- When the duty cycle configuration is CM1=1 and CM0=MAX+1 on channel x use the CCU1 interrupt of triggering channel x+1 instead of CCU1 interrupt on channel x.

## 1.125 ERR008131: SPC57BD1: Boundary scan of the interconnect between PD and BD is not available

**Description:**

Boundary Scan of the interconnect between the Emulation Device "Buddy Die (BD)" to the main production MCU die is not available. There are no boundary scan cells for the interconnect to the BD and the BSDL file for the device does not include the interconnect in the boundary scan chain.

**Workaround:**

Do not attempt to perform Boundary Scan operations to test the interconnect between the BD and main production MCU. Boundary Scan of the main production MCU is possible, but the interconnect between the BD and main production MCU is not included in the boundary scan chain.

## 1.126 ERR008132: I2C: debug session request may not be correctly synchronized with I2C clock

**Description:**

The Inter-Integrated Circuit (I2C) module features an IPG_DEBUG mode that allows to freeze all ongoing activities (such as an ongoing transaction, counter values, and register status) for debugging purposes. This mode is configured by setting the Debug Enable bit (IPG_DEBUG_EN) of the I2C Bus Debug register (IBDBG) and it is automatically triggered when entering the debug session. There is a low probability (<1%) that I2C module is corrupted when a debug session is requested and IBDBG[IPG_DEBUG_EN] is asserted high.

**Workaround:**

Consider the possibility to have I2C corrupted in case a debug session is ongoing and IBDBG[IPG_DEBUG_EN] was set to enforce entry into DEBUG mode.

In case of such corruption cannot be managed by the application, the following workaround can be applied:

1. ensure that the IBDBG[IPG_DEBUG_EN] bit is cleared during normal mode;
2. when entering the debug session, write the IBDBG[IPG_DEBUG_EN] bit to 1 before any debug access to the I2C module;
3. run the debug session normally, accessing I2C registers if needed;
4. before leaving the session, write the IBDBG[IPG_DEBUG_EN] to 0, releasing the I2C debug mode.

## 1.127 ERR008133: SIUL2: PC[2] reset configuration selects alternate function 5 (FCCU EOUT1 with strong push-pull buffer)

**Description:**

The PC[2] pad is expected to behave like all other GPIOs during and after reset: an input buffer with weak pull-up enabled.

The Source Signal Select (SSS) field of the respective Multiplexed Signal Configuration Register (MSCR[34]) is configured, instead, to 0x0000_0101 corresponding to Fault Collection and Control Unit (FCCU) Error output #1 (SAFE - REDUNDANCY) functionality.

The actual behaviour will also depend on the FCCU configuration. In case the Error out (EOUT) functionality is enabled, PC[2] output buffer will also be configured to push-pull and the pad strength to strong.

**Workaround:**

Do not select the PC[2] pin in case the application requires to have an input buffer with pull-up configuration during and after reset.

## 1.128 ERR008145: MEMU: address registers in the uncorrectable error reporting tables can be written when the corresponding valid bit is not asserted

**Description:**

The Memory Error Management Unit (MEMU) allows error reporting tables to be written by the CPU to simulate a memory error condition or to disable memory errors from a known faulty location. This requires that the valid bit for the System RAM, Peripheral RAM, or Flash memory in the error reporting table (SYS_RAM_UNCERR_STS[VLD], PERIPH_RAM_UNCERR_STS[VLD], FLASH_UNCERR_STS[VLD]) corresponding to the address register (SYS_RAM_UNCERR_ADDR, PERIPH_RAM_UNCERR_ADDR, FLASH_UNCERR_ADDR) is asserted when the address is written to the register. The uncorrectable error reporting tables allow these address registers to be written when the valid bit is not asserted, when they should be read-only.

**Workaround:**

Before writing to any of the uncorrectable error reporting tables address registers (SYS_RAM_UNCERR_ADDR, PERIPH_RAM_UNCERR_ADDR, FLASH_UNCERR_ADDR), the corresponding valid bit (SYS_RAM_UNCERR_STS[VLD], PERIPH_RAM_UNCERR_STS[VLD], FLASH_UNCERR_STS[VLD]) must be asserted.

## 1.129 ERR008146: MEMU: ECC error syndrome is not transmitted to MEMU for system and core0 RAMs

**Description:**

During code execution, the Error Correction Code (ECC) syndrome is not reported by the Memory Error and Management (MEMU) for single-bit errors found in system RAM, and core0 (e200z2) RAMs (Data RAM, instruction RAM and cache RAM). Only the address information is reported by the MEMU.

The Error Correction code (ECC) syndrome is instead reported within MEMU for the accesses to the Flash memory module.

The Error Correction Code (ECC) syndrome is correctly reported when MBIST is run on these memories

**Workaround:**

Do not expect the ECC syndrome will be reported to the MEMU for system RAM and core0 RAM for single bit ECC errors. If considered safety-relevant, in case single errors are detected by the MEMU, the user must execute a software test to detect ECC error changes periodically within the Fault Tolerant Time Interval (FTTI).

## 1.130 ERR008225: SDADC: FIFO Flush Reset command requires clearing the Data FIFO Full Flag

**Description:**

When the Sigma-Delta Analog-to-Digital Converter (SDADC) FIFO is flushed by writing '1' to the FIFO Control Register FIFO Flush Reset bit (SDADC_FCR[FRST]), the FIFO is correctly flushed, but the Status Flag Register Data FIFO Full Flag (SDADC_SFR[DFFF]) may be incorrectly asserted, indicating the FIFO is full when it is empty.

**Workaround:**

Clear SDADC_SFR[DFFF] by writing a '1' to this field after performing a FIFO Flush Reset command or after the FIFO is disabled.

## 1.131 ERR008229: FCCU: Enabling the programmable glitch filter on EIN may cause a destructive reset

**Description:**

The Fault Collection and Control Unit (FCCU) external error input (EIN) can be filtered by the FCCU on-chip programmable glitch filter. However, the EIN is routed directly to the FCCU Output Supervision Unit (FOSU) without passing through the glitch filter. Additionally, when the glitch filter is programmed using the FCCU Control Register (FCCU_CTRL) FILTER_WIDTH field, the effective duration may vary depending on time at which the EIN signal arrives, which can cause a missed or false EIN signal. As a result, it is possible for the FOSU to recognize an event on EIN that is ignored by the FCCU resulting in a destructive reset when the FOSU timeout period expires.

**Workaround:**

Bypass the FCCU on-chip glitch filter by writing a 1 to the FCCU Control Register FILTER_BYPASS field (FCCU_CTRL[FILTER_BYPASS]) and use an external glitch filter to ensure that only valid external error events are recognized by the FCCU. The impact on safety can be avoided by using a feedback based signaling on Error In (EIN), wherein the EIN signal is kept asserted until the MCU acknowledges the event to the source.

## 1.132 ERR008310: XBIC: Crossbar Integrity Checker may miss recording information from an initial fault event in the case of back-to-back faults

**Description:**

When the Crossbar Integrity Checker (XBIC) detects back-to-back faults on a system bus path through the crossbar switch (AXBS), the fault information captured in the XBIC Error Status Register (XBIC_ESR) and the XBIC Error Address Register (XBIC_EAR) does not correspond to the initial fault event, but rather the subsequent fault event. While the fault event is properly detected, diagnostic status information in the XBIC_ESR and XBIC_EAR registers describing the initial fault event is lost. This defect can only occur in the event of a series of bus transactions targeting the same crossbar slave target, where the series of bus transactions are not separated by idle or stall cycles.

**Workaround:**

Expect that the XBIC_EAR and XBIC_ESR registers may not contain the initial fault information, but will contain the latest fault information.

## 1.133 ERR008314: SDADC: Double trigger is generated for conversion when SW trigger is connected to SDADC own HW trigger input

**Description:**

In the Sigma-Delta Analog-to-Digital Converter (SDADC), when the wraparound mechanism is enabled by setting the Wrap-Around Mode bit (WRMODE) of the Module Configuration Register (MCR) register, the number of the channels to be sampled gets incremented based on software (SW) and hardware (HW) triggers.

Each SDADC module is allowed to generate a SW trigger using the Software Trigger Key Register (STKR). The SW trigger generates a pulse which is used to synchronize other SDADCs. To do this, the SW trigger of a SDADC module is connected to the HW trigger input of other SDADCs.

In this device implementation, the SW trigger pulse is also connected to a HW trigger input of the same SDADC module.

Since there is a one cycle delay between the SW and the HW trigger pulses (for synchronization), the SDADC using the SW triggered pulse will be triggered a second time by its own HW trigger input. This causes the channel to be incremented twice with just one trigger.

**Workaround:**

Do not program the SDADC_MCR[TRIGSEL] register of the master SDADC channel to select the HW input from itself.

## 1.134 ERR008325: MC_ME: Invalid clock configuration not detected for PSI5 peripherals during mode change

**Description:**

The Invalid Mode Configuration (Clock Usage) Interrupt bit (I_ICONF_CU) in the Mode Entry (MC_ME) Interrupt Status register should be asserted if the clock for a peripheral is not enabled in the mode change TARGET_MODE configuration that enables a peripheral module.

However, the I_ICONF_CU bit will not be properly asserted if the Peripheral Sensor Interface 0, 1, or S (PSI5_0, PSI5_1, and PSI5_S) is enabled in a mode change and the clock to the module is not enabled. Therefore, the mode change will not be completed and there is no indication of the cause for the failure.

**Workaround:**

Software should not rely on the MC_ME_IS[I_ICONF_CU] to be set during mode transitions with an incorrect clocking configuration for the PSI5_0, PSI5_1, and PSI5_S blocks. If a mode change does not be completed, the Mode Transition Status bit in the Mode Entry Global Status register (MC_ME_GS[S_MTRANS]) will not get cleared and the Mode transition complete interrupt will not occur or set the status bit in the interrupt Status register (MC_ME_IS[I_MTC] = 0b1).

## 1.135 ERR008343: DCI: EVTO[1:0] outputs remain stuck low if asserted while the system clock source is the IRC

**Description:**

While the system clock source is the Internal Resistor Capacitor oscillator (IRC), if the Event Output pins (EVTO[1:0]) are asserted by the Debug and Calibration Interface (DCI)), they will never negate. This applies to all functions on EVTO[1:0] except the timer functions controlled by the EVTO Output selection (EOS0/EOS1) fields of the Generic Timer Module debug interface (GTMDI) Development Control (DC) register. This occurs when the EVTO pins are being used by the Development Trigger Semaphore (DTS) module.

**Workaround:**

For proper operation of EVTO[1:0] outputs, program clocks to select a system clock source other than the IRC before enabling and using EVTO[1:0].

## 1.136 ERR008429: GTM: Unexpected TIM CNTS register reset in TPWM OSM mode

**Description:**

(GTM-IP-205)

When a Generic Timer Module (GTM) Timer Input Module (TIM) channel is configured for One Shot PWM Measurement Mode (TIM[i]_CH[x]_CTRL[TIM_MODE]=0, TIM[i]_CH[x]_CTRL[OSM]=1) an active edge will stop the measurement. If this active edge is followed by an inactive edge 1 GTM system clock later, the Counter Shadow register (TIM[i]_CH[x]_CNTS) is unexpectedly reset.

**Workaround:**

Configure the TIM channel to use a Clock Management Unit (CMU) clock source slower than the GTM system clock so that any subsequent active edge is captured at a time after 1 GTM system clock, and/or enable the channel filter with parameters set so that two consecutive edges will be filtered out.

## 1.137 ERR008430: GTM: DPLL sub-inc generation and action calculations are delayed

**Description:**

(GTM-IP-208)

The Generic Timer Module (GTM) Digital Phase Lock Loop (DPLL) delays the start of sub increment (SUB_INC) generation and the action calculation (PMT) by one input event cycle if the DPLL starts after activation (DPLL_CTRL1[DEN]= 0 ->1) while the first TRIGGER detected flag (DPLL_STATUS[FTD]) is not raised and Pulse Correction Mode (PCM) is enabled (DPLL_CTRL1[PCMn]= 1). This results in an incorrect angle clock (TBU_TS1).

**Workaround:**

Set DPLL_CTRL1[PCMn] bits to '0' before the DPLL is activated.

## 1.138 ERR008438: GTM: Wrong signal level when TIM mode is changed from TBCM to any other mode

**Description:**

(GTM-IP-204)

When a Generic Timer Module (GTM) Timer Input Module (TIM) channel is disabled

(TIM[i]_CH[x]_CTRL[TIM_EN]=0) and in Bit Compression Mode (TBCM, TIM[i]_CH[x]_CTRL[TIM_MODE]=0x4) while the corresponding channel input is high, a mode change will not update the input signal level indication bit (Least Significant Bit of TIM[i]_CH[x]_GPR0[ECNT]).

**Workaround:**

If the input signal level information previously captured in TIM[i]_CH[x]_GPR0[ECNT] is sent to another submodule by the ARU, and is then used by the other submodule to make a decision on what action to take next, do not move from TBCM to any other TIM mode while the channel is disabled.

In general it would be unusual to use one TIM channel for a TBCM function and then reuse the channel for another function, therefore this should not have any implications for most use cases.

## 1.139 ERR008439: GTM: TOM and ATOM CM0, CM1 and CLK_SRC register updates may not be triggered

**Description:**

(GTM-IP-209)

The trigger signal between the Generic Timer Module (GTM) Timer Output Module (TOM) or ARU Connected TOM (ATOM) submodules (e.g. signal TOM_TRIG_[i]) can be stored in a register at the module output to break long combinational paths. When this store register is in place, it results in a delay of one system clock period of the trigger signal.

Between module instances TOM[i] / ATOM[i] and TOM[i+1] / ATOM[i+1], when there is a store register in the trigger path, this trigger is only recognized by the channel of TOM[i+1] / ATOM[i+1] if the channel is running from a source identical to the system clock (i.e. the selected Clock Management Unit Fixed Frequency Clock (CMU_FXCLKx) or Clock Management Unit Clock (CMU_CLKx) period is the system clock (SYS_CLK) ÷ 1). If another frequency is chosen to clock the TOM[i+1] / ATOM[i+1] channel, the trigger is not recognized by the Compare Registers (CM0/CM1) or the Clock Source (CLK_SRC) register.

**Workaround:**

TOM Workaround 1:

When there is a register in the trigger path between TOM[i] and TOM[i+1], the channel of

TOM[i+1] that should be triggered has to use a clock of period identical to SYS_CLK period. The configuration of the TOM outputs differs between devices, in some cases each TOM has the save trigger register, in some devices every second TOM module has the register. Check the GTM specification for the configuration applicable to the device in use.

TOM Workaround 2:

On TOM[i+1] configure a redundant channel to trigger another channel of TOM[i+1] as it was configured on TOM[i] to trigger the other channel. Then start TOM[i] and TOM[i+1] synchronously by using the Time Base Unit (TBU) comparator of the TOM Global Control (TGCx) unit (TOM[i]_TGC[y]_ACT_TB register).

ATOM Workaround 1:

When there is a register in the trigger path between ATOM[i] and ATOM[i+1], the channel of

ATOM[i+1] that should be triggered has to use a clock of period identical to SYS_CLK period. The configuration of the ATOM outputs differs between devices, in some cases each ATOM has the save trigger register, in some devices every second ATOM module has the register. Check the GTM specification for the configuration applicable to the device in use.
ATOM Workaround 2:

On ATOM[i+1] configure a redundant channel to trigger another channel of ATOM[i+1] as it was configured on ATOM[i] to trigger the other channel. Then start ATOM[i] and ATOM[i+1] synchronously by using the Time Base Unit (TBU) comparator of the ATOM Global Control (AGC) unit (ATOM[i]_AGC_ATC_TB register).

## 1.140 ERR008526: LINFlexD: LIN or UART state may be incorrectly indicated by LINSR[LINS] bitfield

**Description:**

The Local Interconnect Network (LIN) or Universal Asynchronous Receiver/Transmitter (UART) state is shown in the read only LIN state bits in the LIN Status Register (LINSR[LINS]). Whenever the LINFlexD (in either LIN or UART mode) updates these state bits, there is a possibility that the wrong LIN or UART state is indicated for one Peripheral Bridge Clock (PBRIDGEx_CLK) cycle. If software is asynchronously polling the LIN or UART state bits, the state read by the software may be incorrect.

**Workaround:**

The incorrect state in the LINSR[LINS] bit-field is auto-corrected in the next

PBRIDGEx_CLK cycle. Therefore, any software polling the LINSR[LINS] bit-field should read the bit-field twice and confirm that the back to back reads have the same value before taking further action based on the state.

## 1.141 ERR008561: LINFlexD: Corruption of Tx data in LIN mode with DMA feature enabled

**Description:**

The LINFlexD module is driven by two different clocks. The transmit/reception logic is controlled by the module clock (LIN_CLK) and register accesses are controlled by the peripheral bus clock (PBRIDGEx_CLK). In LIN mode, the re-synchronization of the "Idle on bit error" between the two clocks may cause the Direct Memory Access (DMA) Finite State Machine inside the LINFlexD module to move to the idle state while a transmission is in process. This unwanted idle state transition could lead trigger a new DMA request, potentially overwriting the Buffer Identifier Register (BIDR) and the Buffer Data Registers (BDRL and BDRM).

**Workaround:**

Do not enable the "Idle on bit error" of LIN Control Register 2 (lLINCR2[IOBE] = 0).

Instead of using the “Idle on bit error", use the bit error interrupt of LIN Interrupt Enable Register (LINIER[BEIE] = 1) to trigger an Interrupt service routine and force the LIN into idle mode through software if needed.

## 1.142 ERR008573: LINFlexD: Pre-mature header/response timeout in LIN mode

**Description:**

The LINFlexD instance is driven by two different clocks. The transmit/reception logic is controlled by the module clock (LIN_CLK) and register accesses are controlled by the peripheral bus clock (PBRIDGEx_CLK). The PBRIDGEx_CLK is used to control timer logic during header/response reception, while the LIN_CLK is used to control the header/response reception.

In LIN mode, the resynchronization between the two clocks may cause the pre-mature setting of the OCF bit (Output Compare Flag) of LIN Error Status Register (LINESR).

Depending on LIN Timeout, Control and Status Register, IOT = Idle On TimeOut (LINTCSR[IOT]) settings, effects of the pre-mature setting may be:

- if LINTCSR[IOT] is set, termination of the data reception
- if LINTCSR[IOT] is not set, generation of spurious timeout event, though the reception will continue.

**Workaround:**

Always configure LINTCSR[IOT] as ‘0'. In case of a time out event, the spurious event can be detected by comparing the LINOCR[OC1] and LINOCR[OC2] values with the timer

value in LINTCSR[CNT]. The difference should not be more than one bit time period.

## 1.143 ERR008602: LINFlexD: Tx through DMA can be re-triggered after abort in LIN/UART modes or can prematurely end on the event of bit error with LINCR2[IOBE] bit being set in LIN mode

**Description:**

The LINFlexD module is driven by two different clocks. The transmit/reception logic is controlled by the module clock (LIN_CLK) and register accesses are controlled by the peripheral bus clock (PBRIDGEx_CLK). Due to possible synchronization issue between the two clock domains, there is a possibility that DMA transmission get stuck due to DMA Finite State Machine doesn’t go into idle. This may occur in one of the following conditions:

- if an abort request is triggered (LINCR2[ABRQ]=1) in LIN or UART modes
- if idle on bit error feature is enabled (LINCR2[IOBE]=1) in LIN mode, and a bit error occurs.

DMA state machine will not generate any transaction, waiting for data transmission flag LINSR[DTF] to be set which will never occur.

**Workaround:**

If DMA is used:

- Bit error interrupt should be enabled through LINEIER[BEIE]. When a bit error interrupt is triggered, the interrupt service routine must either reset the DMA Tx channel enable (DMATXE) and the DMA Rx channel enable (DMARXE) registers
- if an abort is requested (LINCR2[ABRQ]=1) in LIN/UART mode, either reset DMATXE/DMARXE of LINFlexD after writing LINCR2 [ABRQ]

## 1.144 ERR008631: SDADC: low threshold watchdog cannot be used with signed data

**Description:**

Each Sigma Delta Analog to Digital Converter (SDADC) provides a watchdog (WDG) to monitor the converted data range. This watchdog should trigger when a converted value is either higher than the value configured in the WDG Threshold Register Upper Threshold Value bitfield (SDADC_WTHHLR[THRH]), or lower than the value configured in the Lower Threshold Value bitfield (SDADC_WTHHLR[THRL]). Instead, the low WDG threshold acts as a high WDG threshold, triggering when a converted value is greater than the value configured in SDADC_WTHHLR[THRL].

**Workaround:**

There are two workarounds available:

1. Do not use the WDG function by clearing the SDADC Module Control Register Watchdog Enable Bit (SDADC_MCR[WDGEN]).
2. Configure the WDG low threshold SDADC_WTHHLR[THRL] to the value 0x7FFF. This guarantees that a low threshold trigger will not be generated. The WDG high threshold (SDADC_WTHHLR[THRH]) can be used without restriction.

## 1.145 ERR008640: TSENSE: Temperature sensor ADC readings are inaccurate during over temperature condition

**Description:**

The temperature sensor ADC readings have a small inaccuracy when an overtemperature condition has been detected. Over-temperature detection is flagged in the Temperature Event Status register (PMC_ESR_TD) Temperature sensor 1 status flag 2 (TEMP1_2) or the Temperature sensor 0 status flag 2 (TEMP0_2) fields.

**Workaround:**

Use the temperature sensor ADC output to monitor temperature after the 150C flag is set (TEMP0_2 or TEMP1_2). After the temperature flag is set, the user software must monitor the ADC temperature sensor until the temperature is below 130 C (hysteresis window plus additional inaccuracy factor).

## 1.146 ERR008655: M_CAN: Setting the Configuration Change Enable (CCE) bit during a transmission scan can halt CAN transmissions

**Description:**

The Modular CAN (M_CAN) Transmission Handler normally scans for Transmission Buffers with pending transmission requests. Pending transmissions are indicated by Transmit (Tx)

Buffer Request Pending (TXBRP) register bits being set. If the user decides to reconfigure the M_CAN module by setting the Configuration Change Enable (CCE) bit of the CAN Core Control Register (CCCR) while the Transmission Handler is scanning for pending transmission requests, the TXBRP register can be cleared and the Transmission Handler FSM is halted. This has the effect of halting any M_CAN message transmission.

After the Initialization (INIT) and CCE bits have been cleared by the Host, the M_CAN is unable to transmit messages. When the Host requests a transmission by writing to the Tx Buffer Add Request (TXBAR) register, the respective Tx Buffer Request Pending bit in register TXBRP is set, but the Transmission Handler will not start the requested transmission.

**Workaround:**

If the M_CAN configuration needs to be changed while in use, place the M_CAN module in a halted state (similar to preparing a Power Down (Sleep Mode) as described in the Reference Manual). The following steps must be used:

Step 1: Cancel all pending transmission requests by writing 0xFFFF_FFFF to Tx Buffer Cancellation Request (TXBCR) register

Step 2: Issue a clock stop request by setting the Clock Stop Request (CSR) bit of the CC Control Register (CCCR)

Step 3: Wait until the M_CAN sets the Initialization (INIT) and Clock Stop Acknowledge (CSA) bits of the CC Control Register (CCCR) to one

Step 4: First clear the CSR bit of the CC Control Register (CCCR)

Step 5: Then, performing a separate write operation, clear the INIT bit of the CC Control Register (CCCR)

Step 6: Wait until INIT bit is read as zero

Step 7: Issue a second clock stop request by setting CSR bit

Step 8: Wait until the M_CAN sets INIT and CSA bits to one

Step 9: Set the Configuration Change Enable (CCE) bit of the CC Control Register (CCCR) and clear CSR bit in a single write operation

Now the M_CAN configuration can be modified.

## 1.147 ERR008673: LINFlexD: Module protocol clock must be greater than the eDMA clock to use eDMA transmit functionality

**Description:**

The Local Interconnect Network Flex (LINFlex) controller can operate in two modes:

LIN Mode and UART (Universal Asynchronous Receiver Transmitter) mode. It provides an Enhanced Direct Memory Access (eDMA) transmit (TX) and receive (RX) interface and is clocked by an asynchronous clock domain between the Peripheral Bridge Clock (PBRIDGE_CLK) and the Local Interconnect Network Clock protocol clock (LINCLK). When the DMA TX functionality of the LINFLEX module is enabled by setting the DMA Tx channel enable (DTE) bit of the DMA Tx Enable Register (DMATXE) and if the eDMA module is running at a frequency higher than (or equal to) the LINFlex's protocol clock, a possible clock glitch can occur when the data is crossing these clock domains, and the requested data transfer can be missed. The issue affects the LINFlex module(s) only and it can occur in either UART or LINFlex modes.

The DMA RX functionality of the LINFLEX module is not affected.

**Workaround:**

Do not use the LINFlex eDMA TX functionality or set the LINCLK frequency to at least 1.25 times the SXBAR_CLK frequency.

The eDMA module clock, Slow Crossbar Clock (SXBAR_CLK), is selected via System Clock Selector/Divider 1 (MC_CGM_SC_DC1) and the LINCLK is configured by the Auxiliary Clock Select Control 0 register (MC_CGM_AC0_SC). The clock configuration for both clocks is done in the Clock Generation Module (MC_CGM).

## 1.148 ERR008688: GTM: Data lost in ATOM when CMU_CLKx is slower than ARU

**Description:**

(GTM-IP-210)

The Generic Timer Module (GTM) Advanced Routing Unit (connected Timer Output Module

(ATOM) in Signal Output Mode Serial (SOMS) one-shot mode (ATOM[i]_CHn_CTRL[OSM=1]) starts to request new data from the Advanced Routing Unit (ARU) when the channel control (ATOM[i]_CHn_CTRL[ARU_EN]) is set to 1. When new data is delivered by the ARU and stored into the Shadow Registers (ATOM[i]_CHn_SR0 and ATOM[i]_CHn_SR1), the data is immediately transferred to the Compare Match Registers (ATOM[i]_CHn_CM0 and ATOM[i]_CHn_CM1) and the ATOM starts to shift with the next clock tick of the selected CMU_CLKx. In parallel, the ATOM immediately requests new data from the ARU. If the ARU delivers new data before the first bit of the previous data is shifted out, which means before the CMU_CLKx clock ticks, the data is stored into Shadow Registers but it is not marked as valid (ATOM[i]_CHn_STAT[DV] bit is not set) and therefore the data is ignored.

**Workaround:**

Software programming must ensure that the time between two consecutive data being delivered from ARU is greater than two periods of the selected CMU_CLKx clock. This can be accomplished by sourcing the data from the Multi-Channel Sequencer (MCS) instead of the First-In-First-Out (FIFO) buffer.

Ensure that the CMU_CLKx is twice the speed of the ARU round trip time. The data loss cannot occur if the ARU round trip time is greater than two CMU_CLKx periods.

## 1.149 ERR008689: GTM: F2A stream data are not deleted after stream disabling

**Description:**

(GTM-IP-212)

When the Generic Timer Module (GTM) First-In-First-Out to Advanced Routing Unit (FIFO-to- ARU, F2A) data stream is disabled in the F2A enable register (F2A[i]_ENABLE[STRn_EN] = 0b01), the existing valid data inside the stream is not deleted. After re-enabling this stream (F2A[i]_ENABLE[STRn_EN] = 0b10), the F2A delivers the old data, independently of the configured data transfer direction (F2A[i]_CHn_STR_CFG[DIR]).

**Workaround:**

Before re-enabling an F2A data stream, the old data must be removed from the FIFO. This can be done after disabling the stream by performing the following steps:

1. Set the ARU Read Address Register (F2A[i]_CHn_ARU_RD_FIFO) to the reset value 0x1FE.
2. 2. Configure the F2A stream direction into ARU to FIFO (F2A[i]_CHn_STR_CFG[DIR] = 0).
3. Enable the stream (F2A[i]_ENABLE[STRn_EN] = 0b10), so that the old data are transported into FIFO.
4. Finally, flush the FIFO channel (FIFO[i]_CHn_CTRL[FLUSH] = 1).

## 1.150 ERR008730: XBIC: XBIC may store incorrect fault information when a fault occurs

**Description:**

The Crossbar Integrity Checker (XBIC) may incorrectly identify a fault's diagnostic information when the slave response signals encounter an unexpected fault when crossing the crossbar switch (XBAR) during the data phase. While the fault event is detected, the diagnostic status information stored in the XBIC's Error Status Register (XBIC_ESR) and Error Address Register (XBIC_EAR) does not reflect the proper master and slave involved in the fault.

Instead, the preceding master or slave ID may be recorded.

**Workaround:**

Expect that when a fault is reported in the XBIC_EAR and XBIC_ESR registers the actual fault information may be from the preceding transition.

## 1.151 ERR008731: LINFlexD: Corruption of Received Rx data in UART mode

**Description:**

The LINFlexD module is driven by two different clocks. The transmit/reception logic is

controlled by the module clock (LIN_CLK) and register accesses are controlled by the peripheral bus clock (PBRIDGEx_CLK).

In the Universal Asynchronous Receive/Transmit (UART) Mode, the resynchronization of the access of the Buffer Data (BDRL and BDRM) registers may reset the internal counter which generates the baud sampling signal to receive the incoming bits. This will occur only if LIN Integer Baud Rate (LINIBRR) register is greater than 1.The data being received may not be correctly sampled:

- Sampling point will be anticipated by maximum of 1 bit period
- The final data may be shifted by one bit in the data BDRM register.

**Workaround:**

When using the LINFlex module in UART mode:

- If possible, LINIBRR should be configured with value ‘1’. This is possible in cases where the baud rate has ratio of 4,5,6,8,16 with respect to clock LIN_CLK. For example with LIN_CLK at 80MHz this allows 20 M baud down to 5 M baud reception rate.
- If the baud rate is not compatible with a LINIBRR value of ‘1’,
  - In UART full duplex mode, application software must manage systematic detection of the failure and on detection, request that the receive data be resent. See possible implementation below
  - In UART half-duplex mode (receive mode only), systematic detection can be implemented as above. Alternatively, software or DMA should ensure that BDRM is read before the start of the next frame reception. Reception information is available through interrupt, DMA request, or the UART status register (UARTSR). This information is available 5 times the LINIBRR[IBR] period of the LIN_CLK before the completion of the STOP bit reception. Further delay may be available depending on delay in between frame reception.

Systematic detection can be performed by using a '0' logic parity bit and enable the generation of the Framing Error Interrupt. This is done by programming UARTCR[PCE] = ‘1’, UARTCR[PC1] = ‘1’, UARTCR[PC0] = ‘0’, and LINIER[FEIE] = ‘1’

Depending on the timing of the possible glitch occurrence, detection will be indicated by

- Either the setting of UARTSR[FE] bit, in other words, a Framing Error which generates a Framing Error Interrupt,
- Or the setting of the noise flag (UARTSR[NF]). Note: No interrupt is generated in this case.

## 1.152 ERR008770: 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 mini slots (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 mini slot: 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.

## 1.153 ERR008884: XBAR: Masters on peripheral shell bus concentrator may stall or fetch data incorrectly

**Description:**

The default programming of bus traffic optimization for a subset of masters on the peripheral shell crossbar (SXBAR/XBAR_1) can cause those masters to stall, receive wrong read data, or get a spurious read access when uncorrectable Error Correction Code (ECC) errors are received from slave modules. For a read transaction following an uncorrectable ECC error from a slave on the computational shell crossbar (FXBAR/XBAR_0), a subset of masters can stall or receive the wrong data. For high-latency situations, a spurious read could occur after the mentioned read transaction. If a master of a bus concentrator on the SXBAR receives an uncorrectable ECC error from a slave on the computational shell crossbar, any master on that concentrator may stall, receive wrong data, or have a spurious read access. The master on concentrator 0 is the DMA controller(s). The masters on concentrator 1 are the Ethernet controller, FlexRay controller(s), SIPI_0 (Zipwire/Interprocessor Interface), and SIPI_1 (debug Zipwire).

**Workaround:**

Software should disable the Pending Read Enable for Slave 0 of the peripheral shell crossbar in the platform configuration module by clearing the Pending Read Enable S0 bit (PCM.IAHB_BE1.PRE_S0). This bit is initialized to 1 during reset. The PRE_S0 bit should be cleared prior to enabling additional masters, and prior to performing accesses by any peripheral shell crossbar master (Peripheral core 2 load/store, DMAx, Ethernet, FlexRayx, Zipwire, debug Zipwire) which might encounter an uncorrectable ECC error. After PRE_S0 is disabled, pipe-lined accesses from any peripheral shell crossbar master to a slave on the computational FXBAR (system ram, overlay ram, flash controller, core 0/1 local memory [IMEM/DMEM]) may have increased latency of up to 3 FXBAR clocks.

## 1.154 ERR008904: M_CAN: Incorrect activation of MRAF interrupt

**Description:**

In the Modular CAN (M_CAN) module, the Message RAM Access Failure flag (MRAF) of the Interrupt register (IR) may be set although there was no Message RAM access failure.

This behavior occurs when the module is receiving a frame and:

- the module is in the Error Passive state, and
- the Receive Error counter (REC), field of the Error Counter register (ECR), has the value 127.

**Workaround:**

In processing the interrupt from the M_CAN module, if the Error Passive flag of the Protocol Status register (ECR[RP]) is set (ECR[RP] == 1) and the Receiver Error counter equals 127 (ECR[REC] == 127), clear the IR.MRAF flag and exit the interrupt handler.

## 1.155 ERR008915: SARADC: wrong behavior when aborting the conversion of a chain

**Description:**

An ongoing conversion of the Successive Approximation Analog to Digital Converter (SARADC) can be aborted by setting the Abort Conversion (ABORT) bit of the Main Configuration Register (MCR).

During a conversion of chain, if the ABORT bit is set during the last cycles of the conversion evaluation phase or first cycles of the sampling phase, the Internal Channel Data Register (ICDRn) of the next channel may be corrupted and the corresponding end of conversion interrupt pending bit (EOC_CHn) of the Internal Channel Interrupt Pending Register (ICIPR) might be wrongly set.

For instance, assuming a normal chain conversion of two channels: channel x (CHx) and channel y (CHy), if the ABORT bit is set during the last cycle of the evaluation phase of CHx, then data of aborted conversion CHx may be written in the ICDR register corresponding to CHy and ICIPR[EOC_CH] bit corresponding to CHy might be wrongly set.

**Workaround:**

When an abort of a SARADC conversion is required, the software must:

1. poll the SARADC status (ADCSTATUS) bit field of the Main Status Register (MSR), until it is in sample phase (0b100);
2. wait for SARADC to start the conversion phase (ADCSTATUS=0b110);
3. issue an abort command by setting the MCR[ABORT] bit.

## 1.156 ERR008923: M_CAN: FD frame format not compliant to the new ISO/CD 11898-1: 2014-12-11

**Description:**

This version of the device implements a Modular Controller Area Network (M_CAN) module version that implements a Flexible Data (CAN-FD) frame format according to ISO/WD 11898-1: 2013-12-13. However, it is not compliant with the new ISO/CD 11898-1: 2014-12-11 format. The frame format was updated during the ISO standardization process.

The limitations are the following:

- the FD frame format is incompatible, the Cyclic Redundancy Check [CRC] does not include the added stuff bit count field
- the FD CRC computation is incompatible, a different seed value is used.

As a consequence this device is not suitable for use in CAN-FD networks that use the new FD frame format according to ISO/CD 11898-1: 2014-12-11.

**Workaround:**

Use CAN-FD mode in networks that only includes devices that conform to the ISO/WD 11898-1: 2013-12-13 frame format.

The Classic CAN mode is unaffected and can be used without restrictions.

## 1.157 ERR008933: LINFlexD: Inconsistent sync field may cause an incorrect baud rate and Sync Field Error Flag may not be set

**Description:**

When the LINFlexD module is configured as follows:

- LIN (Local interconnect network) slave mode is enabled by clearing the Master Mode Enable (MME) bit in the LIN Control Register 1 (LINCR1)
- Auto synchronization is enabled by setting the LIN Auto Synchronization Enable bit (LASE) in the LINCR1 register
- Sync Field value is not equal to 0x55 the LINFlexD module may automatically synchronize to an incorrect baud rate without setting the Sync Field Error Flag (SFEF) in the LIN Error Status register (LINESR).

The auto synchronization is only required when the baud-rate in the slave node cannot 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) [in other words, LINTCSR[MODE]= 0x0]:

1. Set the LIN state Interrupt enable bit (LSIE) in the LIN Interrupt Enable register (LINIER) [LINIER[LSIE] = 0x1]
2. When the Data Reception Completed Flag (DRF) get set in the LIN Status Register (LINSR), read the LIN State field (LINS) in LINSR
3. If LINSR[LINS]= 0b0101, read the Counter Value field (CNT) of the LINTCSR register, otherwise repeat step 2
4. If LINTCSR[CNT] 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 LSIE bit in the LINIER register
2. When the LINSR[DRF] bit get set in the LIN Status Register (LINSR), 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 get set 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 (by clearing LINCR1[LASE]=0) in LIN slave mode.

## 1.158 ERR008935: JTAGM: write accesses to registers must be 32-bit wide

**Description:**

The JTAG Master module (JTAGM) supports only 32-bit write accesses to its registers. A byte write access will be converted into a 32-bit write with the other bytes values at 0x0.

**Workaround:**

Perform only 32-bit write accesses on JTAGM registers. Do not use byte writes.

## 1.159 ERR008951: I2C: Attempting a start cycle while the bus is busy may generate a short clock pulse

**Description:**

When the I2C (Inter-Integrated Circuit) is operating in a multi-master network and a start cycle is attempted by the I2C device when the bus is busy, the attempting master will lose arbitration as expected but a short extra clock cycle is generated in the bus. After losing

arbitration, the master switches to slave mode but it does not detect the short clock pulse. The acknowledge signal is expected at the ninth clock by the current bus master but it is not sent as expected due to the undetected short clock pulse.

**Workaround:**

Software must ensure that the I2C BUS is idle by checking the bus busy bit in the I2C Bus Status Register (I2C_IBSR.IBB) before switching to master mode and attempting a Start cycle.

## 1.160 ERR008970: LINFlexD: Spurious bit error in extended frame mode may cause an incorrect Idle State

**Description:**

The LINFlexD module may set a spurious Bit Error Flag (BEF) in the LIN Error Status

Register (LINESR), when the LINFlexD module is configured as follows:

- Data Size greater than eight data bytes (extended frames) by configuring the Data Field Length (DFL) bitfield in the Buffer Identifier Register (BIDR) with a value greater than seven (eight data bytes)
- Bit error is able to reset the LIN state machine by setting Idle on Bit Error (IOBE) bit in the LIN Control Register 2 (LINCR2)

As a consequence, the state machine may go to the Idle State when the LINFlexD module tries the transmission of the next eight bytes, after the first ones have been successfully transmitted and Data Buffer Empty Flag (DBEF) was set in the LIN Status Register (LINSR).

**Workaround:**

Do not use the extended frame mode by configuring Data Field Length (DFL) bit-field with a value less than eight in the Buffer Identifier Register (BIDR) (BIDR[DFL] < 8)

## 1.161 ERR009003: FLASH: wrong address decoding error generation during Read-while-Write operation

**Description:**

During Flash Read while Write operations (RWW), it is possible for a Program or Erase operation to corrupt the Address Encode feature of the flash, and to falsely generate an Address Encode Error (AEE) event. The false AEE event may occur for RWW operations to any partitions.

This false error detection is reported within the Address Encode Error bit of the Flash Main Control Register (MCR[AEE]), and is triggering the Fault Collection and Control Unit (FCCU) input channels 38 and 39 event if enabled.

Read Data and ECC Parity bits returned for these Reads while writing are valid and not corrupted.

**Workaround:**

Disable the Fault Collection and Control Unit (FCCU) Failure input channels 38 and 39 by clearing the corresponding bits in the Recoverable Fault Enable Register 1 (FCCU_RF_E1).

FCCU input channel 38 is the Encoding Error Flash, and covers the MCR[AEE] event indication in addition to Flash Read Reference Error (MCR[RRE]) indications and is controlled by the FCCU_RF_E1[bit 25]. FCCU input channel 39 is the flash controller address feedback event indication and is controlled by the FCCU_RF_E1[bit 24].

Address Encode Error (AEE) fault recognition is a safety mechanism used to detect potential permanent and transient faults in the flash address decode logic. Even with AEE detection disabled, most faults that would be detected by AEE are still detected by other mechanisms as part of the device redundant safety concept. Consequently, disabling the AEE fault notifications in the FCCU has no impact to the overall functional safety integrity of the device, and the ISO26262 ASIL D target is achieved.

With the FCCU channels 38 and 39 disabled, the Read Reference error may be monitored by reading the fields from the Flash Module Configuration register (MCR[RRE]).

## 1.162 ERR009060: M_CAN: FD frame abort may cause Protocol exception event and extended Bus Integration state

**Description:**

In the Modular Controller Area Network module (M_CAN), when a transmission is aborted shortly before the transmission of the Flexible Data Frame (FDF) bit, a receiver will detect a recessive FDF bit followed by a recessive reserved (res) bit. In this case, the receiving M_CAN modules detect a protocol exception event and will enter the Bus Integration state. The receivers should leave the Bus Integration state after 11 consecutive recessive bits.

Instead of starting to count the 11 recessive bits immediately after entering the Bus Integration state, the M_CAN needs to see at least one dominant bit before it starts to count the sequence of 11 recessive bits.

**Workaround:**

Take into account that, in the described condition, the M_CAN module will take more time to recover as it will wait for 11 recessive bits after the first dominant bit on the network and not 11 recessive bits after entering the Bus Integration state.

## 1.163 ERR009157: MC_ME: system frequency limitation when resetting IOP core

**Description:**

Triggering a reset of the Peripheral Core_2 (Core_2 or IOP), when it is up and running and the system clock frequency is above 40 MHz, may corrupt the configuration of the computational crossbar switch.

The Core_2 will be reset when a mode transition is requested in the Mode Entry module (MC_ME) and one of the following condition occurs:

- The Core_2 is configured to remain under reset in the target mode via the Core Control 0 register of the ME module (ME_CCTL0)
- The reset of the Core_2 is requested by setting the Reset on Mode Change bit (RMC) in the Core Address 0 register (ME_CADDR0).

No corruption occurs in case the target mode requested is a functional reset: Target Mode field of the Mode Control register (ME_MCTL) is 0b0000.

**Workaround:**

If the Core_2 needs to be reset independently of the rest of the device, please ensure that the system clock is lower than 40 MHz by reducing it or by temporarily moving the system clock to the Internal RC oscillator (16 MHz) clock source.

## 1.164 ERR009158: MEMU: System RAM ECC errors on accesses by some masters report to MEMU Peripheral RAM table

**Description:**

As described in the reference manual, Error Correction Code (ECC) errors detected by Crossbar bus Concentrator 0 and Concentrator 1 are reported to the Memory Error Management Unit (MEMU) Peripheral RAM table. ECC errors detected by other system bus

masters are reported to the MEMU System RAM table. As a result, System RAM locations may appear in both the System RAM table and the Peripheral RAM table. The system bus masters which report to the MEMU Peripheral RAM table are the DMA controller(s), the Ethernet controller, and the Flexray controller(s), SIPI_0 (Zipwire/Interprocessor Interface).

**Workaround:**

Expect system RAM ECC errors encountered by masters on Concentrator 0 and Concentrator 1 to be reported to the MEMU peripheral RAM table. The system bus masters which report to the MEMU Peripheral RAM table are the DMA controller(s), the Ethernet controller, the FlexRay controller(s), and the SIPI_0 (Zipwire/Interprocessor Interface). System RAM ECC errors encountered by other system bus masters will result in errors reported to the MEMU System RAM table.

## 1.165 ERR009215: PAD_RING: Higher output impedance on PC[12] when Ethernet I/O segment is configured for 3.3 V supply

**Description:**

Port PC[12] belongs to the Ethernet segment that supports 3.3 V supply range. Its output buffer, however, may not guarantee full static and dynamic performances. When the 3.3 V supply range is selected on the segment by clearing the VSIO_IF bit (VSIO_IF control—Ethernet I/O segment supply) in the Power Management digital interface (PMC_dig) Voltage Supply for I/O Segments register (VSIO), PC[12] will have higher output impedance than all other pads in the same segment, with degradation up to ~40 %. PC[12] provides, instead, full output buffer performance when 5.0V supply range is selected by setting the VSIO_IF bit. The input buffer is not affected and it is fully functional in both 3.3 V and 5 V range.

None of the other pins of the Ethernet segment is affected and provide full specification in both 3.3 V and 5 V range.

**Workaround:**

Do not use PC[12] in 3.3 V configuration in case full static and dynamic performances of output buffer are required.

Please notice that 3.3 V supply range support is mainly aimed to support Ethernet protocol where PC[12] pin is used as an input pin, thus not affected by this limitation.

## 1.166 ERR009409: PMC: low voltage core regulator sense monitoring through SARADC channel cannot be used

**Description:**

Device provides possibility to monitor internal signals through a dedicated indirect test

channel of the Successive Approximation Register Analog-to-Digital Converter (SARADC) On this device, the monitoring of the low voltage core regulator sense (indirect test channel 37 = 6b100101) cannot be activated. In case it has been activated by mistake, a reset can be triggered depending on samples.

**Workaround:**

Do not use monitoring of the LV core regulator sense. It is possible instead to monitor either VDD_LV directly, through dedicated SARADC_B channel #104 or monitor the LV core supply sense through the indirect ADC test channel (indirect test channel 42 = 6b101010)

## 1.167 ERR009413: M_CAN: SPC574KxB, Data loss when the storage of a received frame is not complete before EOF

**Description:**

In the Modular Controller Area Network (M_CAN) module, during a frame reception, the receive handler accesses the Message RAM for acceptance filtering (read access) and storage of accepted messages (write access). The time needed for acceptance filtering and storage of a received message depends on the Host clock frequency, the number of M_CAN modules connected to a single Message RAM, the Message RAM arbitration scheme, and the number of configured filter elements.

In case the storage of a receive message has not been completed before the End Of Frame field (EOF) is reached, the following faulty behavior occurs:

- The last write to the Message RAM to complete the storage of the received message is omitted: this data is lost. This applies for data frames with a Data Length Code (DLC) different from 0, the worst case being for a DLC equal to 1
- The FIFO put index field (FnPI) of the Receive FIFO n Status register (RXFnS) is updated although the last FIFO element holds corrupted data
- The corresponding New Data flag (NDxx) of the New Data y Register (NDATy) is set although the receive buffer holds corrupted data
- The Message RAM Access Failure flag (MRAF) of the M_CAN Interrupt Register (IR) is not set

**Workaround:**

In this device the M_CAN Host Clock is the Peripheral Bridge Clock (PBRIDGE_CLK).

For Classic CAN operation up to 1 Mbit/s, limit the number of filter elements to:

- 30 elements for 20 MHz if the PBRIDGE_CLK is less than 40 MHz, but it is greater than or equal to 20 MHz
- 63 elements for PBRIDGE_CLK greater than or equal to 40 MHz

For non-ISO CAN FD operation up to 1 Mbit/second for arbitration and 4 Mbit/s for data phases:

- Limit the number of filter elements to 15 elements and the number of transmit buffer to 30 for if the PBRIDGE_CLK < 40 MHz, and it is greater than or equal to 20 MHz
- Limit the number of filter elements to 33 elements and the number of transmit buffer to 30 for PBRIDGE_CLK greater than or equal to 40 MHz

## 1.168 ERR009415: M_CAN: Message RAM / RAM arbiter not responding in time

**Description:**

If the Modular Controller Area Network (M_CAN) module needs to store a received frame, and the Message RAM / RAM Arbiter does not respond in time, this message cannot be stored completely and it is discarded with the reception of the next message. The Message RAM Access Failure flag (MRAF) of the M_CAN Interrupt Register (IR) gets correctly set (IR[MRAF]=1) but the next received message may be incompletely stored. In this case, the respective receive buffer or receive FIFO element contains inconsistent data.

**Workaround:**

Configure the RAM Watchdog register (RWD) to the maximum expected Message RAM access delay. In case the Message RAM / RAM Arbiter does not respond within this time, the Watchdog Interrupt flag (WDI) of the Interrupt Register (IR) will be set. The frame received after IR[MRAF] has been set and with IR[WDI] set must be discarded.

## 1.169 ERR009420: FCCU: FOSU may give destructive reset when a hardware recoverable fault of width less than one safe clock occurs

**Description:**

The Fault Collection and Control Unit Output Supervision Unit (FOSU) may issue a destructive reset in the following conditions:

- An input fault is programmed as hardware recoverable (FCCU_RF_CFG0, FCCU_RF_CFG1)
- The reaction programmed is only Error Output pin (EOUT) signaling (FCCU_EOUT_SIG_EN0,FCCU_EOUT_SIG_EN1)
- The source fault coming on the Recoverable Fault (RF) line is asserted only for less than one safe clock duration

**Workaround:**

1. If a fault must be configured as hardware recoverable, may last less than one safe clock cycle and requires EOUT signalling, program long/short functional reset for that HW recoverable fault besides EOUT signalling.
2. Alternatively if a fault assertion may last less than one safe clock cycle and only EOUT signaling is preferred as reaction, the said fault shall be configured as software recoverable

## 1.170 ERR009436: DSPI: AC timing limitation on DSPI4/DSPI5 for some pins associated to the SCK and SOUT functionality

**Description:**

AC timing limitation applies to some of the alternate function pins related to the Serial Clock (SCK) and Serial Data Out (SOUT) functionality of the Deserial Serial Peripheral Interface (DSPI) modules 4 and 5 (DSPI4 and DSPI5).

DSPI4 and DSPI5 SCK and SOUT functionality can be mapped on several pins across the package. The device datasheet provides the timing for the case where SCK and SOUT are using the same configuration of the output drive strength. This is required in order to reduce to minimum the mismatch on the propagation time between SCK and SOUT signal.

Datasheet parameter values for maximum tSUO (SOUT data valid time from SCK) are as below:

- Very strong pad, 25 pF load: 7 ns
- Strong pad, 50 pF load: 8 ns
- Medium pad, 50 pF load: 16 ns

On the device, following pins configured as SOUT are not matching DSPI tSUO datasheet value:

- DSPI4:
  PAD[45], a.k.a. PC[13], configured as SOUT requires +1.5ns to settle
- DSPI5:
  PAD[37], a.k.a. PC[5], configured as SOUT requires +1.5ns to settle
  PAD[38], a.k.a. PC[6], configured as SOUT requires +1.5ns to settle
  PAD[62], a.k.a. PD[14], configured as SOUT requires +1.5ns to settle

On the device, the following pin configured as SCK is not matching DSPI tSUO datasheet value:

- DSPI5
  PAD[36], a.k.a. PC[4], configured as SCK requires +3.0ns for SOUT to settle

**Workaround:**

When possible, do not use pins not matching datasheet parameter.

In case application is using the pin, the extra settle time described in this erratum must be added with respect to the datasheet value.

## 1.171 ERR009599: DSPI: AC timing limitation on DSPI2/DSPI4/DSPI5 in LVDS mode

**Description:**

AC timing limitation applies to the Serial Clock (SCK) and Serial Data Out (SOUT) functionality of the Deserial Serial Peripheral Interface (DSPI) modules 2, 4, and 5 (DSPI2, DSPI4 and DSPI5) when pads are configured as differential mode.

Datasheet parameter values for maximum tSUO (SOUT data valid time from SCK) and tHO (SOUT data hold time after SCK) are as below:

- tSUO: LVDS configuration for SOUT and SCK, max = 3.5 ns
- tHO: LVDS configuration for SOUT and SCK, min = -3.5 ns

On the device, the following pins configured as SOUT are not matching DSPI tSUO datasheet value:

- DSPI2: PAD[50] (a.k.a. PD[2]) configured as SOUT requires +1.0ns to settle PAD[51] (a.k.a. PD[3]) configured as SOUT requires +1.0ns to settle
- DSPI4: PAD[50] (a.k.a. PD[2]) configured as SOUT requires +1.5ns to settle PAD[51] (a.k.a. PD[3]) configured as SOUT requires +1.5ns to settle
- DSPI5: PAD[89] (a.k.a. PF[9]) configured as SOUT requires +1.0ns to settle PAD[90] (a.k.a. PF[10]) configured as SOUT requires +1.0ns to settle

On the device, the following pins configured as SOUT are not matching DSPI tHO datasheet value:

- DSPI5: PAD[89] (a.k.a. PF[9]) configured as SOUT requires -0.5ns for hold time PAD[90] (a.k.a. PF[10]) configured as SOUT requires -0.5ns for hold time

**Workaround:**

In case the application is using the pin, the extra settle time described in this erratum must be added with respect to yhe datasheet value.

## 1.172 ERR009656: DSPI: Frame transfer does not restart in case of SPI parity error in master mode

**Description:**

In the Deserial Serial Peripheral Interface (DSPI) module, in the scenario when:

1. Master/slave mode select bit (MTSR) of Module Configuration register (MCR) is set (MCR[MSTR]=0b1) to configure the module in master mode
2. SPI communication is selected via DSPI Configuration field (DCONF) in MCR (MCR[DCONF] = 0b00)
3. Parity reception check on received frame is enabled by setting the Parity Enable or Mask tASC delay (PE_MASC) bit of DSPI PUSH FIFO Register In Master Mode (PUSHR), i.e. PUSHR[PE]=0b1.
4. Parity Error Stop bit (PES) of MCR is set (MCR[PES]=0b1) which stops SPI frame transfer in case of parity error
5. Parity error is detected on received frame

Then the next frame transfer is stopped, the SPI Parity Error Flag bit (SPEF) of the DSPI Status Register (DSPI_SR) is set (SR[SPEF] =0b1) and the corresponding SPI parity error

interrupt is asserted. Even after the interrupt is serviced and SR[SPEF] is reset, the frame transfer does not restart.

**Workaround:**

Do not use SPI frame transfer stop in case of parity error detection for SPI transmission in master mode. For this, keep the Parity Error Stop bit of Module Configuration Register de-asserted (MCR[PES] = 0b0).

## 1.173 ERR009658: SPI: Inconsistent loading of shift register data into the receive FIFO following an overflow event

**Description:**

In the Serial Peripheral Interface (SPI) module, when both the receive FIFO and shift register are full (Receive FIFO Overflow Flag bit in Status Register is set (SR [RFOF] = 0b1)) and then the Clear Rx FIFO bit in Module Configuration Register (MCR [CLR_RXF]) is asserted to clear the receive FIFO, shift register data is sometimes loaded into the receive FIFO after the clear operation is completed.

**Workaround:**

1. Avoid a receive FIFO overflow condition (SR[RFOF] should never be 0b1). To do this, monitor the RX FIFO Counter field of the Status Register (SR[RXCTR]) which indicates the number of entries in receive FIFO and clear before the counter equals the FIFO depth.
2. Alternatively, after every receive FIFO clear operation (MCR[CLR_RXF] = 0b1) following a receive FIFO overflow (SR[RFOF] = 0b1) scenario, perform a single read from receive FIFO and discard the read data.

## 1.174 ERR009664: DSPI: Frame transfer does not restart in case of DSI parity error in master mode

**Description:**

In the Serial Peripheral Interface module, in the scenario when:

1. Master/slave mode select bit of module configuration register is set (MCR[MSTR]=0b1) to configure the module in master mode
2. Deserial Serial Interface (DSI) communication is selected via DSPI Configuration field (DCONF) in MCR (MCR[DCONF] = 0b01)
3. Parity reception check on received DSI frame is enabled by setting Parity Enable bit (PE) of DSI configuration register 0 (DSICR0[PE]=0b1)
4. Parity Error Stop (PES) bit of DSI configuration register0 is set (DSICR0[PES]=0b1) which stops DSI frame transfer in case of parity error
5. Parity error is detected on received frame

Then the next frame transfer is stopped, DSI parity error flag bit of status register is set (SR[DPEF] =0b1) and the corresponding DSI parity error interrupt is asserted. Even after the interrupt is serviced and SR [DPEF] is reset, the frame transfer does not restart.

**Workaround:**

DSI frame transfer stop in case of parity error detection is disabled. For this, keep the parity error stop bit of DSI configuration register0 de-asserted (DSICR0 [PES]=0b0).

## 1.175 ERR009764: SARADC: DMA interface limitation depending on PBRIDGE/SARADC clock ratio

**Description:**

The Successive Approximation Register Analog-to-Digital Converter (SARADC) modules can trigger Direct Memory Access (DMA) request through the DMA Enable (DMAE) register interface.

When the SARADC clock (SAR_CLK) frequency is slower than half of the peripheral bridge (PBRIDGEx_CLK) clock frequency, the SARADC may trigger a spurious transfer request to the DMA module after the completion of a first valid transfer.

**Workaround:**

Setting the DMA clear sequence enable (DCLR) bit in the DMAE register (DMAE[DCLR] = 1) forces the clearing of the DMA request on read access to the data register and therefore prevents the spurious DMA transfer request.

In case the Internal Channel Data Registers (ICDRn) are only accessed through DMA module (i.e. there are no bus accesses to ICDRn registers triggered by other than DMA bus master when the DMAE[DMAEN] bit is set), it is possible to configure DMAE[DCLR] bit to '1'. This will clear DMA transfer request on the first DMA read access, ensuring both that DMA triggered transfer will be completed successfully and that no other spurious DMA request will be triggered.

## 1.176 ERR009799: SIUL2: Incorrect selection of alternate functions for some pins

**Description:**

The SPC574KxB IO Signal Description and Input Multiplexing Tables with revision earlier than rev 2.6 wrongly describe the System Integration Unit Lite2 (SIUL2) alternate functions reported below:

IO Signal Table sheet:

1. PD[8]: SIUL2.MSCR886[SSS] register field value to select the Deserial Serial Peripheral Interface 2 (DSPI2) input (SIN) as alternate function for port PD[8] is 0000_1010 and not 0000_0110 as reported in xls sheet
2. PE[10]: SIUL2.MSCR895[SSS] register field value to select Deserial Serial Peripheral Interface 5 (DSPI5) SIN as alternate function for port PE[10] is 0000_1111 and not 0000_0011 as reported in xls sheet

Input Muxing sheet:

3. GTM TIM0_2: SIUL2.MSCR514[SSS] register field value 0010_0001 select the WATCHDOG_LIMIT_TRIGGER of the Sigma Delta Analog Digital Converter 3 and not Sigma Delta Analog Digital Converter 2 which is not present on this device.
4. GTM TIM1_2: SIUL2.MSCR522[SSS] register field value 0010_0001 select the WATCHDOG_LIMIT_TRIGGER of the Sigma Delta Analog Digital Converter 3 and not Sigma Delta Analog Digital Converter 2 which is not present on this device.

**Workaround:**

Use values described in this erratum.

*Note: The SPC574Kxx I/O Signal Description and Input Multiplexing Tables are contained in a Microsoft Excel® workbook file attached to this document. Locate the paperclip symbol on the left side of the PDF window, and click it. Double-click on the Excel file to open it and select the I/O Signal Description Table tab.*

# Appendix A Further information

## A.1 Reference document

1. *SPC574Kxx - 32-bit Power Architecture® based MCU for automotive applications* (RM0334, Doc ID 023671).
2. *32-bit Power Architecture® based MCU for automotive applications* (SPC574Kxx datasheet, Doc ID 023601).

## A.2 Acronyms

**Table 4. Acronyms**

| Acronym | Name |
|---|---|
| HSM | Hardware Security Module |
| SPR | Special Purpose Register |
| SWT | System Watchdog Timers |
| MEMU | Memory Error Management Unit |
| MBIST | Memory Built-In Self Test unit |
| FEC | Fast Ethernet Controller |
| RIF | Receive Interface FIFO |
| MIB | Message Information Block |
| ECC | Error Correction Code |
| MCSR | Multiplexed Signal Configuration Register |
| ADC | Analog to Digital Converter |
| DPLL | Digital Phase Lock Loop |
| FCCU | Fault Collection Control Unit |
| MC_ME | Mode Entry Module |
| MC_CGM | Clock Generation Module |
| BAF | Boot Assist Flash |
| DSPI | Deserial Serial Peripheral Interface |
| FOSU | Fault Collection and Control Unit Output Supervision Unit |
| PSI5 | Peripheral Sensor Interface |
| LFAST | LVDS Fast Asynchronous Serial Transmission |
| RGM | Reset Generation Module |
| IMEM | Instruction Memory |
| SARADC | Successive Approximation Register Analog to Digital Converter |
| SENT | Single Edge Nibble Transmission |
| OS | Operating System |

**Table 4. Acronyms (continued)**

| Acronym | Name |
|---|---|
| LINSR | LIN Status Register |
| SRC | Source field |
| TDM | Tamper Detection Module |
| CDR | Converted Data Register |
| ISR | Interrupt Service Routine |
| GTM | Generic Timer Module |
| TIM | Timer Input Module |
| LVDS | Low Voltage Differential Signaling |
| SDADC | Sigma Delta Analog to Digital Converter |
| XBIC | Crossbar Integrity Checker |
| PBRIDGE | Peripheral Bridge |
| GPIO | General Purpose Input/Output |
| MDC | Management Data Clock Output |
| MDIO | Management Data IO |
| DSMC | Decorated Storage Memory Controller |
| RCCU | Redundancy Control and Checker Unit |
| I2C | Inter-Integrated Circuit |
| SSS | Source Signal Select |
| EOUT | Error Out |
| EIN | External Error Input |
| FOSU | FCCU Output Supervision Unit |
| MCR | Module Configuration Register |
| STKR | Software Trigger Key Register |
| IRC | Internal Resistor Capacitor oscillator |
| DCI | Debug and Calibration Interface |
| GTMDI | Generic Timer Module Debug interface |
| CMU | Clock Management Unit |
| DPLL | Digital Phase Lock Loop |
| LIN | Local Interconnect Network |
| UART | Universal Asynchronous Receiver/Transmitter |
| DMA | Direct Memory Access |
| BIDR | Buffer Identifier Register |
| WDG | Watchdog |
| TXBRP | Transmit Buffer Request Pending |
| CCCR | CAN Core Control Register |

**Table 4. Acronyms (continued)**

| Acronym | Name |
|---|---|
| LINFlex | Local Interconnect Network Flex |
| eDMA | Enhanced Direct Memory Access |
| LINCLK | Local Interconnect Network Clock |
| SOMS | Signal Output Mode Serial |
| ARU | Advanced Routing Unit |
| SCM | Single Channel Device Mode bit |
| REC | Receive Error Counter |
| MME | Master Mode Enable |
| SFEF | Sync Field Error Flag |
| LINESR | LIN Error Status Register |
| JTAGM | JTAG Master module |
| DBEF | Data Buffer Empty Flag |
| AEE | Address Encode Error |
| M_CAN | Modular Controller Area Network |
| SOUT | Serial Data Out |
| SCK | Serial Clock |

# Appendix B Defect across silicon version

**Table 5. Defects across silicon version**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR003881 | MC_CGM: Core 2 must be used to change the clock ratio between the cores and the Cross Bar (XBAR) interfaces | x | x | x |
| ERR003970 | NAR: Trace messages include a 6-bit Source Identification field instead of 4-bits | x | x | x |
| ERR004136 | XOSC and IRCOSC: Bus access errors are generated in only half of non-implemented address space of XOSC and IRCOSC, and the other half of address space is mirrored | x | x | x |
| ERR004697 | TDM: Diary base address must be 16 KB aligned | x | x | x |
| ERR005749 | SDADC: New conversion data is discarded if the overflow (DFORF) status bit is set | x | x | x |
| ERR005947 | SARADC: ADC may miss a GTM trigger pulse if width of pulse is less than 1 AD Clk cycle | x | x | x |
| ERR006350 | LINFlexD: WLS feature cannot be used in buffered mode. | x | x | x |
| ERR006409 | GTM: ATOM Force Update does not activate a comparison when in SOMC mode | x | x | x |
| ERR006410 | GTM: Write to ATOM_CH_CTRL sets WRF if CCU0 compare match has already occurred, but CCU1 compare match is pending, in ATOM SOMC mode | x | x | x |
| ERR006411 | GTM: Incorrect Input Signal Characteristics when the TIM channel is in TIEM, TPWM, TIPM, TPIM or TGPS mode, when ECNT is selected as the captured GPRi value. | x | x | x |
| ERR006412 | GTM: Incorrect Input Signal Characteristics when the TIM channel is in TBCM mode and ECNT is selected as the captured GPR0 value. | x | x | x |
| ERR006538 | LINFlexD: Stop mode request may be ignored if requested before the end of a frame | x | x | x |
| ERR006639 | GTM: A compare match event does not clear WR_REQ when ATOM is in SOMC mode | x | x | x |
| ERR006640 | GTM: Valid edge after Timeout event ignored by TIM | x | x | x |
| ERR006642 | GTM: THVAL not available immediately after inactive trigger in DPLL | x | x | x |
| ERR006643 | GTM: Incorrect timestamp captured in CNTS when TIM operates in TPWM or TPIM modes if CMU_CLK is not equal to system clock | x | x | x |
| ERR006644 | GTM: Incorrect duty cycle in TOM PCM mode | x | x | x |
| ERR006645 | GTM: Clearing of DPLL PCM1/2 bits after the Missing Pulse Correction Values calculations delayed | x | x | x |
| ERR006720 | SIUL2: Logic state of LVDS input pads cannot be read via GPDI registers. | x | x | x |
| ERR006828 | e200zx: Local Instruction and Data Memories are not accessible during corresponding e200z4/e200z7 core reset. | x | x | x |
| ERR006835 | LFAST: Full 320Mbps may give bit errors | x | x | x |
| ERR006850 | SSCM: Nexus enable required for mode changes when a debugger is attached | x | x | x |
| ERR006906 | SDADC: Invalid conversion data when output settling delay value is less than 23 | x | x | x |

**Table 5. Defects across silicon version (continued)**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR006932 | NAR: Nexus timestamps are not implemented for the Nexus clients in the e200zx cores | x | x | x |
| ERR006967 | eDMA: Possible misbehavior of a preempted channel when using continuous link mode | x | x | x |
| ERR006994 | XBIC: XBIC may trigger false FCCU alarm | x | x | x |
| ERR007058 | STCU2: Nexus interface of Peripheral Core_2 is not reset after an LBIST execution | x | x | x |
| ERR007083 | GTM: The DPLL's SORI, TORI, MTI, and MSI interrupts may not be asserted | x | x | x |
| ERR007084 | GTM: An active edge input, that is rejected by the DPLL trigger plausibility check, does not assert a Missing Trigger Interrupt | x | x | x |
| ERR007085 | GTM: A TIM timeout occurs when the TDU is re-enabled | x | x | x |
| ERR007086 | GTM: TIM PWM and PIM modes may capture the wrong timestamp | x | x | x |
| ERR007087 | GTM: The DPLL's Address Pointer Extension value is added to the Address Pointer when the Address Pointer Status bit is 0 | x | x | x |
| ERR007088 | GTM: When ATOM is in SOMP mode the SR0/SR1 registers could be updated twice in one PWM period | x | x | x |
| ERR007099 | FCCU: Error pin signal length is not extended when the next enabled fault, with its alarm timeout disabled, occurs | x | x | x |
| ERR007103 | MC_CGM: Incorrect cause for the latest clock source switch may be reported by the CGM if a safe mode request arrives when the system clock is the IRC | x | x | x |
| ERR007109 | I2C: In master receive mode, data remains latched in I2C data I/O register (IBDR) until new data is received | x | x | x |
| ERR007115 | DSPI: Mixing 16 and 32 bits frame size in XSPI Mode can cause incorrect data to be transmitted | x | x | x |
| ERR007116 | CRC: AutoSAR 4.0 8-bit CRC8 0x2F is not supported in hardware | x | x | x |
| ERR007126 | MEMU: Instead of Byte 1 of MEMU CTRL Register, Byte 3 is currently protected | x | x | x |
| ERR007134 | RCCU: If any accesses to the I-MEM or D-MEM of the safety and checker core are performed while the cores are disabled, the cores will get out of lockstep when enabled | x | x | x |
| ERR007137 | MEMU: incorrect indication when a correctable error is signaled by the e200zx core cache | x | x | x |
| ERR007138 | SARADC: Missed conversion after ABORT of the last channel of an injected chain | x | x | x |
| ERR007185 | SDADC: Watchdog Crossover event missed if PBRIDGEx_CLK less than SD_CLK | x | x | x |
| ERR007190 | GTM: Simultaneous Core and DPLL accesses to RAM Region 2 may lead to the DPLL reading erroneous data | x | x | x |
| ERR007191 | GTM: The DPLL's SORI and TORI interrupts are not asserted | x | x | x |
| ERR007203 | SENT: In debug mode SENT message data registers appear to lose contents | x | x | x |
| ERR007204 | SENT: Number of Expected Edges Error status flag spuriously set when operating with Option 1 of the Successive Calibration Check method | x | x | x |

**Table 5. Defects across silicon version (continued)**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR007211 | MC_ME: Core register IAC8 is cleared during a mode change when the core is reset | x | x | x |
| ERR007222 | SARADC: Minimum value of precharge must be greater than or equal to 2 ADC clock cycles | x | x | x |
| ERR007223 | FCCU: FCCU_IRQ_EN register is writeable in all operating modes | x | x | x |
| ERR007226 | FCCU: the error-out signalling cannot be disabled in non Bi-stable protocols | x | x | x |
| ERR007227 | FCCU: FCCU Output Supervision Unit (FOSU) will not monitor faults enabled while already pending | x | x | x |
| ERR007228 | LINFlexD: Erroneous transmission in LIN master mode for payload greater than eight bytes | x | x | x |
| ERR007234 | PSI5: No transfer error generated for accesses within the unused range of the PSI5 peripheral window | x | x | x |
| ERR007246 | SARADC: First conversion after exit from stop mode may be corrupted | x | x | x |
| ERR007259 | e200zx: ICNT and branch history information may be incorrect following a nexus overflow | x | x | x |
| ERR007269 | LINFlexD: Erroneous timeout interrupt could be generated by LIN in master mode | x | x | x |
| ERR007274 | LINFlexD: Consecutive headers received by LIN Slave triggers error interrupt | x | x | x |
| ERR007297 | LINFlexD: Response timeout values is loaded in LINOCR[OC2] field instead of LINOCR[OC1] | x | x | x |
| ERR007305 | e200zx: JTAG reads of the Performance Monitor Counter registers are not reliable | x | x | x |
| ERR007325 | FCCU: Unsuccessful decorated storage access may cause erroneous signaling of FCCU Channel 40 | x | x | x |
| ERR007335 | MEMU: ECC errors may be double reported when initiated by the Safety core to local memory of other cores | x | x | x |
| ERR007339 | STCU2: STCU2 fault injected by FCCU is self clearing | x | x | x |
| ERR007352 | DSPI: reserved bits in slave CTAR are writable | x | x | x |
| ERR007356 | SDADC: The SDADC FIFO does not function correctly when FIFO overwrite option is used | x | x | x |
| ERR007360 | FEC: Minimum VDD is 3.15 V instead of 3.0 V | x | x | x |
| ERR007362 | SDADC: Additional DMA request generated after single read access | x | x | x |
| ERR007404 | SENT: Message overflow in SENT Receiver can lead to stall condition in the MCU | x | | |
| ERR007411 | PBRIDGE: Incorrect transfer error information on accesses to reserved locations | x | x | x |
| ERR007412 | PBRIDGE: Incorrect transfer error information for accesses to TDM and FEC reserved locations | x | x | x |
| ERR007414 | PBRIDGE: Incorrect transfer error when accessing reserved locations of the Peripheral Bridge | x | x | x |
| ERR007415 | JTAG: PA[9] = JTAG TDO pad is not pull-up during reset | x | x | x |
| ERR007416 | MEMU: Flexray data RAM and GTM RAMs Error not correctly registered within MEMU during online MBIST | x | x | x |

**Table 5. Defects across silicon version (continued)**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR007425 | SENT: Unexpected NUM_EDGES_ERR error in certain conditions when message has a pause pulse | x | x | x |
| ERR007433 | JTAGM: Nexus error bit is cleared by successful RWA | x | x | x |
| ERR007498 | M_(TT)CAN: Transmitted bit in control field is falsified when using extreme bit time configurations | x | x | x |
| ERR007528 | GTM: Action not always calculated immediately by DPLL | x | x | x |
| ERR007529 | GTM: TIM overflow bit is not set and the signal level bit has inverse value when sent to ARU in some cases | x | x | x |
| ERR007530 | GTM: New DPLL Position Minus Time data not received | x | x | x |
| ERR007531 | GTM: DPLL Position Minus Time result is not sent to the ARU | x | x | x |
| ERR007532 | M_TTCAN: Incorrect value of Reference Trigger Offset status for time slaves | x | x | x |
| ERR007538 | M_(TT)CAN: Switch between CAN operating modes during transmission or reception may be ignored | x | x | x |
| ERR007587 | SSCM: Multi-bit ECC error at RCHW locations will cause device to remain in reset as a security and safety precaution | x | | |
| ERR007589 | LINFlexD: Erroneous timeout error when switching from UART to LIN mode | x | x | x |
| ERR007632 | SENT: Incorrect rounding of the Status Nibble, Data Nibbles, and CRC Nibbles | x | | |
| ERR007652 | M_CAN: TTCAN triggers protect part of memory | x | x | x |
| ERR007701 | STCU2: Short Functional Reset reaction and Long Functional Reset reaction of the FCCU does not take effect upon PLL1 Loss-Of-Lock while MBIST ONLINE is running | x | x | x |
| ERR007750 | PAD_RING: Incorrect control of internal pull up and pull down on GPIO PB[5] and PE[14] | x | | |
| ERR007772 | PMC: a SWT_2 destructive reset may be forced if the 5 V supply is over voltage for longer than 16ms | x | x | x |
| ERR007788 | SIUL2: A transfer error is not generated for 8-bit accesses to non-existent MSCRs | x | x | x |
| ERR007791 | SIUL2: Transfer error not generated if reserved addresses within the range of SIUL BASE + 0x100 to 0x23F are accessed | x | x | x |
| ERR007796 | M_CAN: Message reception and transmission directly after detection of Protocol Exception Event | x | x | x |
| ERR007801 | WKPU: functional NMI filter enable trigger FCCU fault monitor channel #47 | x | x | x |
| ERR007824 | DCI: Avoid asserting system reset when switching JTAG operating modes | x | x | x |
| ERR007840 | M_CAN: Change of operation mode during start of transmission | x | x | x |
| ERR007841 | M_CAN: Incorrect frame transmission after recovery from Restricted Operation Mode | x | x | x |
| ERR007842 | M_CAN: Setting / resetting CCCR.INIT during frame reception | x | x | x |
| ERR007847 | GTM: MCS's CAT status may be incorrect | x | x | x |
| ERR007848 | GTM: Bit 0 of TIM edge counter register may not indicate the actual signal level | x | x | x |

**Table 5. Defects across silicon version (continued)**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR007855 | SENT: Integer division during calibration pulse measurement causes reduced robustness | x | | |
| ERR007869 | FCCU: FOSU monitoring of a fault is blocked for second or later occurrence of the same fault | x | x | x |
| ERR007873 | SPC574Kxx: Current injection causes leakage path across the DSPI and LFAST LVDS pins | x | x | x |
| ERR007904 | PASS: Programming Group Lock bit (PGL) can be de-asserted by multiple masters writing the correct password sections to the CINn registers. | x | x | x |
| ERR007905 | PIT: Accessing the PIT by peripheral interface may fail immediately after enabling the PIT peripheral clock | x | x | x |
| ERR007911 | MC_ME: Decorated accesses not supported to core local memories during reset | x | x | x |
| ERR007932 | NAR/SIPI: Part ID for NAR and Debug SIPI does not match the MIDR MINOR_MASK | x | x | x |
| ERR007934 | FEC: MDC and MDIO timing requirements and configuration | x | x | x |
| ERR007935 | PMC: Accessing the HVD_FLASH divider tap point through the ADC test channel may cause a device reset | x | | |
| ERR007947 | XOSC: Incorrect external oscillator status flag after CMU event clear | x | x | x |
| ERR007960 | FCCU: Channels associated to MEMU cannot be activated through the fake fault register (FCCU_RFF) | x | x | x |
| ERR007981 | LBIST: LBIST of the flash may leave flash in an unknown state and stress flash bit cells | x | | |
| ERR007996 | PSI5: Incorrect SMC message decoding and timestamp generation in case of late last sensor message overlapping with next SYNC period pulse | x | x | x |
| ERR008005 | DSMC: Software reset of core0/core0s may trigger FCCU channel #11 event | x | x | x |
| ERR008019 | PMC: Escalation of LVD and HVD functional resets should be expected | x | x | x |
| ERR008034 | FCCU: Channel 40 fault (COMP_XBIC_DSMC_Monitor) may be reported if a CPU is reset upon a MC_ME mode change | x | x | x |
| ERR008039 | SDADC: digital filter and FIFO not disabled when MCR[EN] is cleared | x | x | x |
| ERR008042 | FCCU: EOUT signals are active, even when error out signaling is disabled | x | | |
| ERR008054 | PIT: DMA request stays asserted when initiated by PIT trigger, until PIT is reset | x | x | x |
| ERR008056 | LBIST: Flash must be idle during LBIST | x | | |
| ERR008062 | M_CAN: Frame transmission in DAR mode | x | x | x |
| ERR008082 | SENT: A message overflow can lead to a loss of frames combined with NUM_EDGES_ERR being set | x | x | x |
| ERR008089 | TDM: Diary updates require PECIE bit of MCR register of flash to be set | x | x | x |
| ERR008117 | MC_ME: Restrictions on enabling FlexRay in low power modes | x | x | x |
| ERR008122 | GTM: (A)TOM's CCU1 event interrupt is not generated when CM1=0 or 1 and RST_CCU0=1 | x | x | x |

**Table 5. Defects across silicon version (continued)**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR008131 | SPC57BD1: Boundary scan of the interconnect between PD and BD is not available | x | x | x |
| ERR008132 | I2C: debug session request may not be correctly synchronized with I2C clock | x | x | x |
| ERR008133 | SIUL2: PC[2] reset configuration selects alternate function 5 (FCCU EOUT1 with strong push-pull buffer) | x | | |
| ERR008145 | MEMU: address registers in the uncorrectable error reporting tables can be written when the corresponding valid bit is not asserted | x | x | x |
| ERR008146 | MEMU: ECC error syndrome is not transmitted to MEMU for system and core0 RAMs | x | x | x |
| ERR008225 | SDADC: FIFO Flush Reset command requires clearing the Data FIFO Full Flag | x | x | x |
| ERR008229 | FCCU: Enabling the programmable glitch filter on EIN may cause a destructive reset | x | | |
| ERR008310 | XBIC: Crossbar Integrity Checker may miss recording information from an initial fault event in the case of back-to-back faults | x | x | x |
| ERR008314 | SDADC: Double trigger is generated for conversion when SW trigger is connected to SDADC own HW trigger input | x | x | x |
| ERR008325 | MC_ME: Invalid clock configuration not detected for PSI5 peripherals during mode change | x | x | x |
| ERR008343 | DCI: EVTO[1:0] outputs remain stuck low if asserted while the system clock source is the IRC | x | x | x |
| ERR008429 | GTM: Unexpected TIM CNTS register reset in TPWM OSM mode | x | x | x |
| ERR008430 | GTM: DPLL sub-inc generation and action calculations are delayed | x | x | x |
| ERR008438 | GTM: Wrong signal level when TIM mode is changed from TBCM to any other mode | x | x | x |
| ERR008439 | GTM: TOM and ATOM CM0, CM1 and CLK_SRC register updates may not be triggered | x | x | x |
| ERR008526 | LINFlexD: LIN or UART state may be incorrectly indicated by LINSR[LINS] bitfield | x | x | x |
| ERR008561 | LINFlexD: Corruption of Tx data in LIN mode with DMA feature enabled | x | | |
| ERR008573 | LINFlexD: Pre-mature header/response timeout in LIN mode | x | | |
| ERR008602 | LINFlexD: Tx through DMA can be re-triggered after abort in LIN/UART modes or can prematurely end on the event of bit error with LINCR2[IOBE] bit being set in LIN mode | x | x | x |
| ERR008631 | SDADC: low threshold watchdog cannot be used with signed data | x | x | x |
| ERR008640 | TSENSE: Temperature sensor ADC readings are inaccurate during over temperature condition | x | x | x |
| ERR008649 | TSENSE: Second threshold (165 °C) over- temperature flag is not implemented | | x | x |
| ERR008655 | M_CAN: Setting the Configuration Change Enable (CCE) bit during a transmission scan can halt CAN transmissions | x | x | x |
| ERR008673 | LINFlexD: Module protocol clock must be greater than the eDMA clock to use eDMA transmit functionality | x | x | x |

**Table 5. Defects across silicon version (continued)**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR008688 | GTM: Data lost in ATOM when CMU_CLKx is slower than ARU | x | x | x |
| ERR008689 | GTM: F2A stream data are not deleted after stream disabling | x | x | x |
| ERR008730 | XBIC: XBIC may store incorrect fault information when a fault occurs | x | x | x |
| ERR008731 | LINFlexD: Corruption of Received Rx data in UART mode | x | | |
| ERR008770 | FlexRay: Missing TX frames on Channel B when in dual channel mode and Channel A is disabled | x | x | x |
| ERR008884 | XBAR: Masters on peripheral shell bus concentrator may stall or fetch data incorrectly | x | x | x |
| ERR008904 | M_CAN: Incorrect activation of MRAF interrupt | x | x | x |
| ERR008915 | SARADC: wrong behavior when aborting the conversion of a chain | x | x | x |
| ERR008923 | M_CAN: FD frame format not compliant to the new ISO/CD 11898-1: 2014-12-11 | x | x | x |
| ERR008933 | LINFlexD: Inconsistent sync field may cause an incorrect baud rate and Sync Field Error Flag may not be set | x | x | x |
| ERR008935 | JTAGM: write accesses to registers must be 32- bit wide | x | x | x |
| ERR008951 | I2C: Attempting a start cycle while the bus is busy may generate a short clock pulse | x | x | x |
| ERR008970 | LINFlexD: Spurious bit error in extended frame mode may cause an incorrect Idle State | x | x | x |
| ERR009003 | FLASH: wrong address decoding error generation during Read-while-Write operation | x | x | x |
| ERR009060 | M_CAN: FD frame abort may cause Protocol exception event and extended Bus Integration state | x | x | x |
| ERR009157 | MC_ME: system frequency limitation when resetting IOP core | x | x | x |
| ERR009158 | MEMU: System RAM ECC errors on accesses by some masters report to MEMU Peripheral RAM table | x | x | x |
| ERR009215 | PAD_RING: Higher output impedance on PC[12] when Ethernet I/O segment is configured for 3.3 V supply | x | x | x |
| ERR009409 | PMC: low voltage core regulator sense monitoring through SARADC channel cannot be used | x | x | x |
| ERR009413 | M_CAN: SPC574Kxx, Data loss when the storage of a received frame is not complete before EOF | x | x | x |
| ERR009415 | M_CAN: Message RAM / RAM arbiter not responding in time | x | x | x |
| ERR009420 | FCCU: FOSU may give destructive reset when a hardware recoverable fault of width less than one safe clock occurs | x | x | x |
| ERR009436 | DSPI: AC timing limitation on DSPI4/DSPI5 for some pins associated to the SCK and SOUT functionality | x | x | x |
| ERR009599 | DSPI: AC timing limitation on DSPI2/DSPI4/DSPI5 in LVDS mode | x | x | x |
| ERR009656 | DSPI: Frame transfer does not restart in case of SPI parity error in master mode | x | x | x |
| ERR009658 | SPI: Inconsistent loading of shift register data into the receive FIFO following an overflow event | x | x | x |

**Table 5. Defects across silicon version (continued)**

| Defect ID | Title | Cut 2.0 | Cut 2.2 | Cut 2.3 |
|---|---|---|---|---|
| ERR009664 | DSPI: Frame transfer does not restart in case of DSI parity error in master mode | x | x | x |
| ERR009764 | SARADC: DMA interface limitation depending on PBRIDGE/SARADC clock ratio | x | x | x |
| ERR009799 | SIUL2: Incorrect selection of alternate functions for some pins | x | x | x |

# Revision history

**Table 6. Document revision history**

| Date | Revision | Changes |
|---|---|---|
| 01-Aug-2014 | 1 | Initial release. |
| 23-Oct-2015 | 2 | Added the following functional problems:<br>– ERR007414<br>– ERR007416<br>– ERR007701<br>– ERR007873<br>– ERR007934<br>– ERR007960<br>– ERR008005<br>– ERR008132<br>– ERR008133<br>– ERR008145<br>– ERR008146<br>– ERR008225<br>– ERR008229<br>– ERR008310<br>– ERR008314<br>– ERR008325<br>– ERR008343<br>– ERR008429<br>– ERR008430<br>– ERR008438<br>– ERR008439<br>– ERR008526<br>– ERR008561<br>– ERR008573<br>– ERR008602<br>– ERR008631<br>– ERR008640<br>– ERR008655<br>– ERR008673<br>– ERR008688<br>– ERR008689<br>– ERR008730<br>– ERR008731 |

**Table 6. Document revision history**

| Date | Revision | Changes |
|---|---|---|
| 23-Oct-2015 | 2<br>(cont.) | – ERR008770<br>– ERR008884<br>– ERR008904<br>– ERR008915<br>– ERR008923<br>– ERR008933<br>– ERR008935<br>– ERR008951<br>– ERR008970<br>– ERR009003<br>– ERR009060<br>– ERR009157<br>– ERR009158<br>– ERR009215<br>– ERR009409<br>– ERR009413<br>– ERR009415<br>– ERR009420<br>– ERR009436<br>– ERR009599<br>– ERR009656<br>– ERR009658<br>– ERR009664<br>– ERR009764<br>– ERR009799<br><br>Updated *Table 1: Device summary*.<br>Updated *Table 5: Defects across silicon version*. |

**IMPORTANT NOTICE – PLEASE READ CAREFULLY**

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2015 STMicroelectronics – All rights reserved