NXP Semiconductors

Errata

Document identifier: S32K396_1P40E

Rev. MAR2024, 3/2024

S32K396_1P40E

Mask Set Errata

Mask Set Errata for Mask 1P40E

Revision History

This report applies to mask 1P40E for these products:

- S32K396
- S32K39
- S32K37

Table 1. Mask Specific Information

major_mask_rev_num	0
minor_mask_rev_num	1
jtag_id	1995_601Dh

Table 2. Revision History

Revision	Date	Significant Changes
MAR2024	3/2024	The following errata were removed. • ERR051223 The following errata were added. • ERR052121 • ERR052204 • ERR051778 • ERR052066 The following errata were revised. • ERR011573
SEP2023	9/2023	The following errata were added. • ERR051988 • ERR051701 • ERR051648 The following errata were revised. • ERR051646
MAY2023	5/2023	Initial Revision

Errata and Information Summary

Table 3. Errata and Information Summary

Erratum ID	Erratum Title
ERR005642	ETPU2: Limitations of forced instructions executed via the debug interface

Table continues on the next page...

Table 3. Errata and Information Summary (continued)

Erratum ID	Erratum Title
ERR008252	eTPU: ETPU Angle Counter (EAC) Tooth Program Register (TPR) register write may fail
ERR009090	eTPU: Incorrect eTPU angle counter function under certain conditions
ERR009809	eTPU: MDU flags(Overflow/Carry) may be set incorrectly
ERR011573	Cortex-M7: Speculative accesses might be performed to memory unmapped in MPU.
ERR050595	GMAC/EMAC: Incorrect pps output generation on target time error
ERR050597	GMAC/EMAC: Transmit MAC management counters (MMC) updated incorrectly during frame preemption
ERR050705	GMAC/EMAC: Head-Of-Line blocking error due to incorrect packet size when gates of gate control list (GCL) are closed
ERR050706	GMAC/EMAC: MAC receiver incorrectly discards the received packets when preamble byte does not precede SFD or SMD
ERR050707	GMAC/EMAC: Incorrect Handling of Application Bus Error in Certain Boundary Conditions
ERR050727	Core: Data corruption for load following Store-Exclusive.
ERR050729	Core: ECC error causes data corruption when the data cache error bank registers are locked.
ERR050875	CoreSight: AHB-AP can issue transactions where HADDR[1:0] is not aligned to HSIZE on the AHB
ERR050887	Coresight: CSTPIU fails to output sync after the pattern generator is disabled in Normal mode
ERR051040	ITCM/DTCM: On the TCM backdoor accesses, burst termination (via MRC) due to entering protected region within the burst leads to an erroneous update of the protected region accessed by the burst.
ERR051046	Core: CTI might generate interrupts even when DBGENCTRL[CDBGEN] is low.
ERR051222	GMAC/EMAC: Un-Correctable FSM Timeout Safety Interrupt incorrectly getting generated due to long waiting FSM states.
ERR051256	FCCU: NCF[2]: eDMA TCD RAM Multibit ECC errors can be falsely asserted
ERR051588	LPSPI:Reset transmit FIFO after FIFO underrun by LPSPI Slave.
ERR051629	LPUART:Transmit Complete bit (STAT[TC]) is not set.
ERR051645	WKPU: Not all wakeup pins can support external wakeup events if standby mode is entered in Clocking Mode D
ERR051646	MCM: Interrupt from MCM module may not always be asserted
ERR051648	SPI: In Continuous Selection Format observed tASC timing differs from the expected one.
ERR051653	QSPI: Incorrect default state of INTA signal is causing interrupt request and false ECC error occurrence
ERR051665	eFlexPWM: Last one or two registers are not updated by automatic eDMA registers update.
ERR051686	SGEN: Status Error does not get cleared if HW trigger is out of acceptance window
ERR051701	SPI: De-asserting of CONT bit in the Continuous Selection Format is not mandatory for the last frame

Table continues on the next page...

Table 3. Errata and Information Summary (continued)

Erratum ID	Erratum Title
ERR051778	Embedded Flash Memory: Incorrect NCF[3] indication out of reset
ERR051988	LFAST: LVDS receiver pad fault is latched when receiver is enabled before the common voltage is settled on the line
ERR052066	SDADC: Parameters degradation caused by the signals on the GPIO/GPI pins
ERR052121	LPI2C: NACK Detect Flag can be set when IGNACK=1
ERR052204	Zipwire (SIPI): Performance impact for devices with outstanding frames configured to 2

Known Errata

ERR005642: ETPU2: Limitations of forced instructions executed via the debug interface

Description

The following limitations apply to forced instructions executed through the Nexus debug interface on the Enhanced Time Processing Unit (ETPU):

1- When a branch or dispatch call instruction with the pipeline flush enabled (field FLS=0) is forced (through the debug port), the Return Address Register (RAR) is updated with the current program counter (PC) value, instead of PC value + 1.

2- The Channel Interrupt and Data Transfer Requests (CIRC) instruction field is not operational.

Workaround

Workaround for limitation #1 (branch or dispatch call instruction):

Increment the PC value stored in the RAR by executing a forced Arithmetic Logic Unit (ALU) instruction after the execution of the branch or dispatch call instruction.

Workaround for limitation #2 (CIRC):

To force an interrupt or DMA request from the debugger:

1- Program a Shared Code Memory (SCM) location with an instruction that issues the interrupt and/or DMA request. Note: Save the original value at the SCM location.

2- Save the address of the next instruction to be executed.

3- Force a jump with flush to the instruction position.

4- Single-step the execution.

5- Restore the saved value to the SCM location (saved in step 1).

6- Force a jump with flush to the address of the next instruction to be executed (saved in step 2).

NOTE: This workaround cannot be executed when the eTPU is in HALT_IDLE state.

ERR008252: eTPU: ETPU Angle Counter (EAC) Tooth Program Register (TPR) register write may fail

Description

When the TPR is written with the Insert Physical Tooth (IPH) bit set to 1, and a physical tooth arrives at near the same time, the buffering of a second write to the TPR may fail, even if the required wait for one microcycle after the IPH write is observed.

Workaround

Wait at least two microcycles between consecutive writes to the TPR register, if the first write sets the IPH bit.

ERR009090: eTPU: Incorrect eTPU angle counter function under certain conditions

Description

The eTPU Angle Counter (EAC) can function incorrectly in some scenarios when all of the following conditions apply:

- EAC Tooth Program Register (TPR), Angle Ticks Number in the Current Tooth field (TICKS) = 0 [TPR.TICKS = 0]

and

- Tick Rate Register (TRR) and the eTPU Engine Time Base Configuration Register prescaler field [eTPU_TBR_TBCR_ENGn.TCRnP] satisfy the following condition:

(TRR - 1)*(TCRnP + 1) < 3, where TRR is the non-zero 15-bit integer part (the 15 most significant bits).

When the above conditions are met, three possible scenarios can cause the EAC to function incorrectly:

Scenario 1:

1. The EAC is in High Rate Mode, TRR = 1, and TPR Missing Tooth Counter field = 0 [TPR.MISSCNT = 0]

2. On an EAC transition from High Rate Mode to Normal mode, a positive value is written to TPR.MISSCNT

3. The first microcycle in Normal Mode coincides with a tick timing and either

a. A tooth does not arrive

or

b. A tooth arrives

Expected EAC behavior:

a. Nothing happens

or

b. The EAC transitions back to High Rate Mode

Actual (incorrect) EAC behavior:

a. The EAC transitions to Halt Mode, even though TPR.MISSCNT > 0

or

b. The EAC stays in Normal Mode, even though a tooth arrived before expected and TPR.MISSCNT > 0. The values of TPR.MISSCNT and TPR.LAST are reset, even though the EAC does not transition to High Rate Mode.

Scenario 2:

TCRnP = 0, TRR = 1 (integer part) and a new value is written to TPR.MISSCNT when the EAC transitions from High Rate Mode to Normal Mode. In this scenario, TPR.MISSCNT decrements on every microcycle, but the time the EAC takes to transition to Halt Mode is determined by the previous

TPR.MISSCNT value, so that one of the following unique situations is observed:

a. TPR.MISSCNT reaches zero, but the EAC transitions to Halt Mode only after a number of microcycles equal to the TPR.MISSCNT value before the write.

b. EAC transitions to Halt Mode with TPR.MISSCNT > 0 while, decrementing MISSCNT one more time. If TPR.MISSCNT > 1 during the mode transition, the EAC will stay in Halt mode with a non-zero value of TPR.MISSCNT.

Scenario 3:

1. The EAC transitions to Normal mode from High Rate or Halt Mode

2. The EAC enters Normal mode with TPR.LAST = 1

3. A tooth is received on the second or third microcycle after the EAC transitions to Normal mode. The tooth may be either a physical tooth or a dummy physical tooth generated by setting the Insert Physical Tooth (IPH) field of the TPR register (TPR.IPH = 1).

Observed result:

The EAC resets the values of TPR.LAST, TPR.IPH and the eTPU Engine Time Base2 (TCR2) register, but the EAC goes to Halt mode.

If a new TPR.TICKS value is written with the EAC in Normal mode, the value is effective after a new tooth is received in Halt mode, with TCR2 counting from 0.

Workaround

Limit the angle tick period to a minimum value that satisfies the condition (TRR - 1)* (TCRnP + 1) > 2, where TRR is the non-zero 15-bit integer part (the 15 most significant bits).

ERR009809: eTPU: MDU flags(Overflow/Carry) may be set incorrectly

Description

The MAC Carry (MC) & MAC Overflow (MV) flags can be incorrectly set on a MAC instruction if it is the first MDU operation in a thread and the last MDU operation in previous thread was aborted/terminated (thread ended before the operation finished).

Workaround

There are 2 workarounds:

(1) Do not abort/terminate a MDU operation

or

(2) Do not use a MAC instruction as the first MDU operation in a thread

ERR011573: Cortex-M7: Speculative accesses might be performed to memory unmapped in MPU.

Description

Arm errata 1013783-B

Fault Type: Programmer Cat B

Cortex-M7 can perform speculative memory accesses to Normal memory for various reasons. All other types of memory should never be subject to speculative accesses.

The memory attributes for a given address are defined by the settings of the MPU when it is enabled. Regions that are not mapped in the MPU do not have any explicit attributes and should not be subject to any speculative accesses.

Because of this erratum, Cortex-M7 can incorrectly perform speculative accesses to such unmapped regions.

Conditions:

To trigger this erratum, the data cache must be enabled and the MPU must be enabled with the default memory map disabled. That is:

- CCR.DC = 1; data cache is enabled.
- MPU_CTRL.ENABLE = 1; MPU is enabled.
- If MPU_CTRL.PRIVDEFNA = 1, then this erratum cannot occur from privileged mode.
- If MPU_CTRL.HFNMIENA = 1, then this erratum cannot occur from the NMI or HF handlers or exception handlers when FAULTMASK = 1.

In these situations, a PLD instruction targeting an unmapped region might result in an incorrect speculative access. The PLD instruction itself could be speculative because of branch prediction. Even a literal data value that corresponds to a PLD encoding could theoretically cause this issue. This makes it difficult to scan code to check if these conditions apply.

Therefore, Arm recommends that any software with the MPU and data cache configured as mentioned in the conditions above uses the workaround below.

Implications:

Processor execution is not directly affected by this erratum. The data returned from the speculative access is never used and if the access is inferred by the program, then an abort will be taken as required.

The only implications of this erratum are the access itself which should not have been performed. This might have an impact on memory regions with side-effects on reads or on memory which never returns a response on the bus.

Workaround

Instead of leaving memory unmapped, software should use MPU region 0 to cover all unmapped memory and make this region execute-never and inaccessible. That is, MPU_RASR0 should be programmed with:

- MPU_RASR0.ENABLE = 1; MPU region 0 enable.
- MPU_RASR0.SIZE = b11111; MPU region 0 size = 2^32 bytes to cover entire memory.
- MPU_RASR0.SRD = b00000000; All sub-regions enabled.
- MPU_RASR0.XN = 1; Execute-never to prevent instruction fetch.
- MPU_RASR0.AP = b000; No read or write access for any privilege level.
- MPU_RASR0.TEX = b000; Attributes = Strongly-ordered.
- MPU_RASR0.C = b0; Attributes = Strongly-ordered.
- MPU_RASR0.B = b0; Attributes = Strongly-ordered.

Note that the MPU supports addressing hitting in multiple regions with the highest numbered region taking priority.

Therefore, use of MPU region 0 in this way does not affect the existing organization and use of MPU regions.

ERR050595: GMAC/EMAC: Incorrect pps output generation on target time error

Description

There are two scenarios.

1. When programmed target time is lesser than system time, then pulse per second (pps) output signal gets generated incorrectly despite target error.

2. loss of sub-nanosecond accuracy if time correction of nanosecond field enabled in binary rollover mode.

Workaround

Workaround for scenario#1:. Software should not program target time register with already past system time.
Software must read current system time seconds register (MAC_System_Time_Seconds) and Nanoseconds register (MAC_System_Time_Nanoseconds) before programming target time seconds register (MAC_PPS(i)_Target_Time_Seconds) and Nanosecond registers (MAC_PPS(i)_Target_Time_Nanoseconds) where i can range from 0 to 3.

Workaround for scenario#2: Do not use Binary roll over mode for Nanosecond time update.

ERR050597: GMAC/EMAC: Transmit MAC management counters (MMC) updated incorrectly during frame preemption

Description

Accumulated byte counters would overflow when sum of bytes of all fragments of Frame Preempted packet exceeds the Jabber limit. This would happen only when frame preemption feature is enabled. This would result in incorrect value of transmit MAC management counters (MMC).

Workaround

The transmit MAC management counters (MMC) cannot be use when the frame pre-emption feature is enabled

ERR050705: GMAC/EMAC: Head-Of-Line blocking error due to incorrect packet size when gates of gate control list (GCL) are closed

Description

Incorrect head of line blocking (HLBF) error is getting detected because a packet from a Transmit Queue is available for scheduling but the GCL gates for that Transmit Queue are closed for two complete iterations of GCL. Due to this incorrect detection of HLBF error, Packet is getting incorrectly dropped if the DDBF field of the MTL_EST_Control register is set to 0 and hence there is data loss.

Workaround

The software must set DDBF field of the MTL_EST_Control register to 1 and provide a new gate control list (GCL).

ERR050706: GMAC/EMAC: MAC receiver incorrectly discards the received packets when preamble byte does not precede SFD or SMD

Description

Received packet would be discarded if preamble byte doesn't precede the SFD,SMD-S or SMD-C byte when frame preemption is enabled. This happens because start of packet detection logic of MAC receiver incorrectly checks the preamble byte. The packets without Preamble or with corrupted Preamble byte before SFD is mostly an error case which has very low probability of occurrence.

Workaround

If remote MAC transmitter doesn't transmit preamble byte, then it should be configured to send at least one preamble byte preceding the SFD,SMD-S or SMD-C byte. In case of corruption of preamble byte during transmission, there is no software workaround. This means the applications that support packet re-transmission can re-transmit the packet.

ERR050707: GMAC/EMAC: Incorrect Handling of Application Bus Error in Certain Boundary Conditions

Description

Transmit or Receive DMA channel is not able to handle bus error gracefully under following scenario

1. When OSP (Operating on second packet) mode is enabled and the bus error occurs on the non-last descriptor of current packet because the transmit status of previous packet is still pending to be written in the descriptor memory. Transmit DMA doesn't read and discard the pending transmit status of these two packets.

2. When the bus error occurs on the first data word of transmit packet of the pre-emption queue then MAC transmitter would not transmit that single byte pre-emption fragment packet which encountered the bus error, and subsequent pre-emption and express packet.

3. When the Bus Error occurs on the first data word of the transmit packet, the start of packet is not indicated in the single dummy word, as expected. In this scenario, the start and end of packet indication is pushed to the Transmit Queue. This will lead to a situation where MTL Transmit Read Controller would not transmit subsequent packets from any transmit Queue.

Workaround

System must apply software reset to GMAC/EMAC using the soft reset configuration in the block and reconfigure the GMAC/EMAC

ERR050727: Core: Data corruption for load following Store-Exclusive.

Description

ARM errata 1315869

Affects: Cortex-M7, Cortex-M7 with FPU Fault Type: Programmer Category C

Fault Status: Present in r0p1, r0p2, r1p0, r1p1 and r1p2. Open.

A load that follows a Store-Exclusive to the same address might forward data from an earlier store, situated between the Load-Exclusive and the Store-Exclusive, and not the data from the Store-Exclusive.

Conditions:

The following sequence is required for this erratum to occur:

1. A load exclusive sets the local monitor.

2. A store to the wanted address

3. Any of the following instructions to the wanted address. This instruction must not fail either the local or global monitor check.

• STREXB.

• STREXH.

• STREX.

4. A load to the wanted address.

There must be at most one instruction between the Store-Exclusive and the load. All accesses must be to Shareable memory.

Implications:

Data corruption occurs when the load returns data from the older store instead of the newer Store-Exclusive.

Stores between a Load-Exclusive and Store-Exclusive are not expected in real code because such stores can always clear the local monitor in some implementations.

This impacts Cortex-M7 and Cortex-M7 with FPU.

Configurations Affected

All configurations are affected.

Workaround

No workaround is necessary.

ERR050729: Core: ECC error causes data corruption when the data cache error bank registers are locked.

Description

ARM errata 1267980

Affects: Cortex-M7, Cortex-M7 with FPU

Fault Type: Programmer Category C

Fault Status: Present in r0p1, r0p2, r1p0, r1p1 and r1p2. Open.

The data cache contains two error bank registers, DEBR0 and DEBR1. These registers store the locations in the cache that Error Correcting Code (ECC) errors affect and prevent future allocations to those locations.

Software can lock each DEBR and this prevents the DEBR from being automatically updated when a data cache ECC error is detected.

Because of this erratum, if both DEBR0 and DEBR1 are locked and an ECC error is detected on a cacheable store, then the store data is written onto the bus but not written into the data cache. This might result in the data cache containing stale data.

Conditions:

• DEBR0 and DEBR1 are locked.

• The wanted address has been allocated to the cache.

• A cacheable store to the wanted address looks up in the cache, and an ECC error is found in the cache set that the store addresses.

Implications:

This erratum can cause data corruption in the data cache.

Configurations Affected

All configurations with a data cache and ECC are affected.

Workaround

Software must avoid locking both error bank registers.

ERR050875: CoreSight: AHB-AP can issue transactions where HADDR[1:0] is not aligned to HSIZE on the AHB

Description

ARM errata 1624041

This erratum affects the following components:

• AHB Access Port.

The ARM Debug Interface v5 Architecture Specification specifies a TAR (Transfer Address Register) in the MEM-AP that holds the memory address to be accessed.

TAR[1:0] is used to drive HADDR[1:0] when accesses are made using the Data Read/Write register DRW.

When the AHB-AP is programmed to perform a word or half-word sized transaction the AHB-AP does not force HADDR[1:0] to be aligned to the access size. This can result in illegal AHB transactions that are not correctly aligned according to HSIZE if HADDR[1:0] is programmed with an unaligned value.

Conditions:

1) TAR[1:0] programmed with a value that is not aligned with the size programmed in the CSW register of the AHB-AP.

2) An access is initiated by an access to the Data Read/Write Register (DRW) in the AHB-AP.

Implications:

As a result of the programming conditions listed above, AHB-AP erroneously initiates an access on the AHB with HADDR[1:0] not aligned to the size on HSIZE. This might initiate an illegal AHB access.

Workaround

TAR[1:0] must be b00 for word accesses, TAR[0] must be b0 for half-word accesses.

Software program should program TAR with an address value that is aligned to transaction size being made.

ERR050887: Coresight: CSTPIU fails to output sync after the pattern generator is disabled in Normal mode

Description

Arm errata 341182

Affects: CoreSight SoC-400 - Perpetual

Fault Type: Programmer Category C

Fault Status: Present in: r0p0, r1p0, r2p0, r2p1, r3p0, r3p1, r3p2, Fixed in Open

This erratum affects the following components:

- Trace Port Interface Unit.
- CSTPIU and cxtpiu
- Component Revisions: r0p4, r0p5, r1p0

The TPIU includes a pattern generator that can be used to determine the operating behavior of the trace port and timing characteristics. This pattern generator includes a mode that transmits the test pattern for a specified number of cycles, and then reverts to transmitting normal trace data.

As a result of this erratum, when the TPIU is configured to operate in Normal Mode (FFCR.EnFCont==0), the synchronization sequence that is required between the test pattern and the trace data is not generated. Synchronization will be generated at later times as determined by the synchronization counter.

Conditions

The following conditions must all occur:

- The TPIU is configured in normal mode, FFCR.EnFCont==0
- The TPIU is configured with the formatter enabled, FFCR.EnFTC==1
- The pattern generator is enabled in timed mode, Current_test_pattern_mode.PTIMEEN==1

Implications

The timed mode of the TPIU is intended to permit the TPIU to transition between an initial synchronization sequence using the pattern generator and functional mode without any further programming intervention. If the synchronization sequence is not generated at the end of the test pattern, the trace port analyzer is unlikely to be able to capture the start of the trace stream correctly. Synchronization will be correctly inserted based on the value configured in the FSCR, once the specified number of frames of trace data have been output.

Workaround

This workaround requires software interaction to detect the completion of the test pattern sequence. In addition, any trace data present at the input to the TPIU is lost whilst the pattern generator is active. Any trace data present in the input to the TPIU before the formatter is re-enabled (and synchronization generated) will not be decompressible.

1) After enabling the pattern generator, set FFCR.StopOnFl==1 and FFCR.FOnMan==1.

2) Poll FFSR.FtStopped until 1 is read

3) Set FFCR.EnFTC==1

ERR051040: ITCM/DTCM: On the TCM backdoor accesses, burst termination (via MRC) due to entering protected region within the burst leads to an erroneous update of the protected region accessed by the burst.

Description

XRDC’s MRC is used to setup R/W protection to system memory including cores’ D-TCM and I-TCM through backdoor access. If core that doesn’t have access into the protected D-TCM/I-TCM region performs an access to that region, using a burst sequence with start address outside of the protected region and end address within the protected region, then D-TCM/I-TCM content within the protected region and overlaid by the burst transfer will be modified. The written values will be random.

The burst termination via the MRC is notified to the master via the bus error. However the un-intended region post the termination gets modified instead of being aborted.

Workaround

There are two possible workarounds:

1. The application should align the start and end addresses of the burst transfer within a region (protected or unprotected).

2. Disable the burst optimization (with impact to performance) by configuring IAHBCFGREG[TCM_DIS_WR_OPT] as 1 to avoid this issue.

ERR051046: Core: CTI might generate interrupts even when DBGENCTRL[CDBGEN] is low.

Description

ARM errata 585224

The Cortex-M7 integration level interrupt request outputs (CTIIRQ) are connected to the CTI outputs TRIGOUT[2:1]. These triggers should not be generated when the DBGENCTRL[CDBGEN] is low. This behavior should be guaranteed by tying off respective bits of the TODBGENSEL mask to 0. This mask has been tied off in the processor integration level to the incorrect value, which can result in the output being triggered regardless of the DBGENCTRL[CDBGEN] value.

Conditions:

• The CTIIRQ outputs are used by the system and enabled by respective bit in IRSPRCn (n refers to the irq number) (Refer to IRQ number from the CTI IRQ number provided in Interrupt map).

• Programming CTI such that it can generate triggers on these outputs (for more information on configuring the trigger outputs, see the ARM CoreSight™ SoC-400 Integration Manual).

• The processor interrupts are enabled and DBGENCTRL[CDBGEN] has to be driven low.

Implications:

Because of this erratum, debug events configured to trigger interrupts on the CTIIRQ output might generate them even if the DBGENCTRL[CDBGEN] is low.

Workaround

If the CTI interrupts are not used and debug is disabled, it is recommended to disable the respected CTI IRQ by software by disabling it in the appropriate bit in IRSPRCn for the respective core.

ERR051222: GMAC/EMAC: Un-Correctable FSM Timeout Safety Interrupt incorrectly getting generated due to long waiting FSM states.

Description

If the DMA transmission is enabled by programming DMA_CH0_Tx_Control[ST] bit to '1', but the MAC transmission is disabled (MAC_Configuration[TE] bit set to 0), the FSM timeout is incorrectly generated after a duration configured via the MAC_FSM_ACT_Timer register field, leading to an FSM timeout interrupt.

Workaround

There are two workarounds, the software must implement either of the below:

1. Software must enable transmit DMA after enabling the MAC.

2. Software must enable both transmit DMA and MAC together.

ERR051256: FCCU: NCF[2]: eDMA TCD RAM Multibit ECC errors can be falsely asserted

Description

eDMA0 TCD RAM Multibit ECC Error and eDMA1 TCD RAM Multibit ECC Error are properly generated inside the module but their notification to FCCU are spuriously triggered (false failures) on NCF2.

Workaround

The eDMAs TCD RAM Multibit ECC Error needs to be handled over the ERM module interrupt instead of the FCCU handling.

1) Disable the routing of the eDMAs TCD RAM Multibit ECC errors to FCCU in the DCM (DCMRWD4[DMA_TCD_RAM_ECC_ERR_EN] = 0x0, DCMRWD14[DMA1_TCD_RAM_ECC_ERR_EN] = 0x0)

2) Enable the ERM interrupt for handling the eDMAs TCD RAM Multibit ECC error (ERM0_CR2[ENCIE] = 0x1, ERM1_CR2[ENCIE] = 0x1).

3) The interrupt service routine for the eDMAs TCD RAM Multibit ECC error must be executed within the FTTI.

ERR051588: LPSPI:Reset transmit FIFO after FIFO underrun by LPSPI Slave.

Description

Transmit FIFO pointers are corrupted when a transmit FIFO underrun occurs (SR[TEF]) in slave mode.

Workaround

When clearing the transmit error flag (SR[TEF] = 0b1) following a transmit FIFO underrun, reset the transmit FIFO (CR[RTF] = 0b1) before writing any new data to the transmit FIFO.

ERR051629: LPUART:Transmit Complete bit (STAT[TC]) is not set.

Description

When the CTS pin is negated and the CTS feature is enabled (MODIR[TXCTSE] = 0b1) and the TX FIFO is flushed by software then, the Transmit Complete (STAT[TC]) flag is not set.

Workaround

Clear (MODIR[TXCTSE]) bit and reset the transmit FIFO (FIFO[TXFLUSH] = 0b1) when flushing the FIFO with CTS enabled(MODIR[TXCTSE] = 0b1).

ERR051645: WKPU: Not all wakeup pins can support external wakeup events if standby mode is entered in Clocking Mode D

Description

Wakeup Unit (WKPU) pins can be used as External wakeup sources to generate wakeup event for MCU to exit Standby mode. Prior to entry to the Standby mode software must enter either Clocking Mode C (running from Fast Internal RC oscillator FIRC divided by two) or Clocking Mode D (running from FIRC). Before entering to standby mode software must configure which wake up pin which can support External wakeup source to exits from Standby mode. If following pins are used as WKPU External wake-up sources to exit from Standby mode and Standby mode entry is performed form Clocking Mode D the Standby mode exit is not possible:

PTC11,PTB0,PTC9,PTB16,PTB15,PTB13,PTD4,PTD2,PTB9,PTB8,PTA1,PTC6,PTA16,PTA15,PTE6,PTE2,PTB29,PTC23,PTC24,PTC25,PTC26,PTC29,PTC31,,PTD20,PTD23,PTD27,PTD29,PTD31,PTE18,PTF13,PTF19,PTF21,PTF24,PTF26,PTG25,PTH7,PTH9 and PTH11.

Workaround

For above mentioned pins entry to the Standby mode must be performed from Clocking Mode C in case those pins are intended to be use as External wakeup source from Standby mode.

ERR051646: MCM: Interrupt from MCM module may not always be asserted

Description

Miscellaneous Control Module (MCM) is supposed to generate an interrupt (Vector 54, NVIC 9) if either of the listed statements are true:

- FPU input denormal interrupt is enabled (ISCR[FIDCE]) and an input is denormalized (FIDC)
- FPU inexact interrupt is enabled (ISCR[FIXCE]) and a number is inexact (FIXC)
- FPU underflow interrupt is enabled (ISCR[FUFCE]) and an underflow occurs (FUFC)
- FPU overflow interrupt is enabled (ISCR[FOFCE]) and an overflow occurs (FOFC)
- FPU divide-by-zero interrupt is enabled (ISCR[FDZCE]) and a divide-by-zero occurs (FDZC)
- FPU invalid operation interrupt is enabled (ISCR[FIOCE]) and an invalid operation occurs (FIOC)
- TCM Write abort interrupt is enabled (ISCR[WABE]) and a write abort occurs (CM7 WABORTS INDICATOR)

The interrupt on Tightly-Coupled Memory (TCM) write abort may not get always generated on the TCM write abort occurrence even if the interrupt is enabled (MCM.ISCR[WABE] = 1) . TCM write abort occurs on any master access to TCM through the AHBS interface of the core that results to uncorrectable ECC error during write.

The FPU interrupts are working fine.

Workaround

There are two possible workarounds for this issue depending on whether the FCCU is required to handle TCM ECC errors and if more detailed information about the cause of NCF[2] is required.

Case 1: TCM ECC Error handling is needed at FCCU level.

The cause of interrupt is also signaled as Non-Critical Fault 2 (NCF[2]) in Fault Collection and Control Unit (FCCU) which is set on TCM Write abort event and also in respective bit in Device Configuration Module General-Purpose Registers (DCM_GPR) which retain the values over the functional reset. The TCM ECC source of the NCF[2] error (whether the ECC happened on the data or instruction TCM and which core related) is latched in one of the "Read-Only GPR On Destructive Reset register" DCMROD4 or DCMROD5. Following bits are latching the TCM ECC error source:

- DCM_GPR.DCMROD4[CM7_1_DTCM1_ECC_ERR] for data TCM 1 of core 1

- DCM_GPR.DCMROD4[CM7_1_DTCM0_ECC_ERR] for data TCM 0 of core 1
- DCM_GPR.DCMROD4[CM7_1_ITCM_ECC_ERR] for instruction TCM of core 1
- DCM_GPR.DCMROD4[CM7_0_DTCM1_ECC_ERR] for data TCM 1 of core 0
- DCM_GPR.DCMROD4[CM7_0_DTCM0_ECC_ERR] for data TCM 0 of core 0
- DCM_GPR.DCMROD4[CM7_0_ITCM_ECC_ERR] for instruction TCM of core 0
- DCM_GPR.DCMROD5[CM7_2_DTCM1_ECC_ERR] for data TCM 1 of core 2
- DCM_GPR.DCMROD5[CM7_2_DTCM0_ECC_ERR] for data TCM 0 of core 2
- DCM_GPR.DCMROD5[CM7_2_ITCM_ECC_ERR] for instruction TCM of core 2

Since NCF[2] recommended reaction is to perform functional reset of the MCU, the device would undergo functional reset on the NCF[2] occurrence and out of the functional reset the software will have to check the DCMROD4 and DCMROD5 register values to determine if TCM ECC error was cause of the device functional reset. In this case the information if the write abort was the cause of NCF[2] is lost since MCM.ISCR[WABSO] and MCM.ISCR[WABS] status is not retained across the functional reset. The content of DCM_GPR.DCMROD4 and DCM_GPR.DCMROD5 is retained across functional reset.

Case 2 : TCM ECC Error handling is not required at FCCU level.

Alternatively the cause of interrupt is signaled to Error Reporting Module (ERM) in ACP_ERM_0 channel 10,11,12,13,14, and 15 and ACP_ERM_1 channel 10,11 and 12. If the NCF[2] reporting is not enabled for ECC errors on the TCMs for all the cores then corresponding NCF[2] will not occur on the TCM ECC error including the TCM Write abort and device will not undergo functional reset. In this case the ERM module can be used to generate interrupt on the TCM ECC error occurrence and software can determine if the cause of the ECC error was the TCM write abort by checking if MCM.ISCR[WABSO] and MCM.ISCR[WABS] flags are set. In this case following bits must be disabled to avoid functional reset on TCM ECC error detection:

- DCM_GPR.DCMRWD4[CM7_1_DTCM1_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD4[CM7_1_DTCM0_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD4[CM7_1_ITCM_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD4[CM7_0_DTCM1_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD4[CM7_0_DTCM0_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD4[CM7_0_ITCM_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD5[CM7_2_DTCM1_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD5[CM7_2_DTCM0_ECC_ERR_EN] = 0
- DCM_GPR.DCMRWD5[CM7_2_ITCM_ECC_ERR_EN] = 0

Also, at the same time the interrupt on TCM write abort in MCM module must be disabled to avoid potential double interrupt for the same error source (MCM.ISCR[WABE] = 0).

ERR051648: SPI: In Continuous Selection Format observed tASC timing differs from the expected one.

Description

In Continuous Selection Format (CONT) mode, the expected timing for After SCK Delay (tASC) is by the description of the Clock and Transfer Attributes Register in the ASC field (SPI.CTARn.[ASC]). The actual observed tASC time differs from expected time as follows, depending on Clock Phase (SPI.CTARn.CPHA) settings:

For CPHA=1:

- If expected tASC < 8 cycles, then observed tASC = 10 cycles of protocol clock .

- If expected tASC >= 8 cycles, then observed tASC = expected tASC + 2 cycles of protocol clock.

For CPHA=0:

- If expected tASC <8 cycles, then observed tASC = 9 cycles of protocol clock.

- If expected tASC >= 8 cycles, then observed tASC = expected tASC + 2 cycles of protocol clock

SPI module register description is also applicable to DSPI module.

Workaround

To properly match the requirements of the target SPI device, adjust configuration of SPI.CTARn fields PASC and ASC as needed to account for the difference in observed timing as given in errata description.

ERR051653: QSPI: Incorrect default state of INTA signal is causing interrupt request and false ECC error occurrence

Description

If the Error Code Correction (ECC) signal for Flash Memory A (INTA) is not driven from the connected memory device the Uncorrectable ECC error status from Quad Serial Peripheral Interface (QSPI) A is detected right after the interface is enabled and causes the interrupt request and false ECC error will be signaled by QSPI module.

Workaround

Before enabling the QSPI interface configure the INTA functionality on pad in respective Input Multiplexed Signal Configuration register (IMCR) and configure respective Multiplexed Signal Configuration register (MSCR) of System Integration Unit Lite2 (SIUL) as input with internal pull-up resistor enabled:

SIUL2.MSCR[y].[IBE] = 1

SIUL2.MSCR[y].[PUE] = 1

SIUL2.MSCR[y].[PUS] = 1

The INTA pin must be a dedicated functionality and cannot be used for any other purpose.

ERR051665: eFlexPWM: Last one or two registers are not updated by automatic eDMA registers update.

Description

When automatic enhanced Direct Memory Access (eDMA) write request for update of the timing double buffered registers (SMmVALn and SMmFRACVALn) is used (SMmDMAEN[VALDE] = 1), the last two registers are not updated.

Workaround

Configure the eDMA to transfer two more values after the registers that are requested to be updated. Ensure that the additional two transferred values do not change the existing module configuration.

ERR051686: SGEN: Status Error does not get cleared if HW trigger is out of acceptance window

Description

If the Hardware trigger mode is used to generate output of the Sine Wave Generator module (SGEN) and the hardware trigger arrives outside the Acceptance window the Phase Error Interrupt bit is set (SGEN.STAT[PHERR]=1) and subsequently Error Interrupt Status gets set (SGEN.STAT[SERR]=1). It is expected to get both error bits cleared by software. However, Error Interrupt status bit (SGEN.STAT[SERR]) remains set even after writing 1 to this bit by SW and remains set until the valid Hardware trigger within acceptance window arrives. This causes multiple interrupt occurrence since the SGEN.STAT[SERR] bit remains pending.

Workaround

Do not enable interrupt from SGEN module if the Hardware trigger mode is configured. Use polling of the status bits to get the information about the SGEN module state.

ERR051701: SPI: De-asserting of CONT bit in the Continuous Selection Format is not mandatory for the last frame

Description

In Continuous Selection Format it is requested the Continuous Peripheral Chip Select Enable bit (SPI.PUSHR[CONT]) is de-asserted for the last frame to be transmitted. However when SPI.PUSHR[CONT] is not de-asserted for the last frame, data transfer happens correctly according to corresponding SPI.CTARE[DTCP] configuration.

SPI module register description is also applicable to DSPI module.

Workaround

De-assertion of the SPI.PUSHR[CONT] bit in the last frame to be transmitted is not mandatory. Data transfer works correctly as programmed in SPI.CTARE[DTCP] regardless the SPI.PUSHR[CONT] configuration in the last frame to be transmitted.

ERR051778: Embedded Flash Memory: Incorrect NCF[3] indication out of reset

Description

When the device comes out of reset, an incorrect Non-critical failure (NCF[3]) indication may be seen.

The DCMROD4[FLASH_EDC_ERR] bit is set out of reset but the MCRS[EEE] bit is not set.

The DCMROD4[FLASH_ADDR_ENC_ERR] is set out of reset but the MCRS[AEE] is not set.

Apart from these, no other sources of NCF[3] are asserted.

Workaround

Clear DCMROD4[FLASH_EDC_ERR], DCMROD4[FLASH_ADDR_ENC_ERR] and Non-critical failure NCF[3] out of reset when MCRS[EEE] and MCRS[AEE] are not set. The other sources of NCF[3] in DCMROD register (mentioned below) should not be set in this case.

DCMROD3[PF3_CODE_ECC_ERR]

DCMROD3[PF3_DATA_ECC_ERR]

DCMROD3[LC_ERR]

DCMROD4[PF0_CODE_ECC_ERR]

DCMROD4[PF0_DATA_ECC_ERR]

DCMROD4[PF1_CODE_ECC_ERR]

DCMROD4[PF1_DATA_ECC_ERR]

DCMROD4[FLASH_REF_ERR]

DCMROD4[FLASH_RST_ERR]

DCMROD4[FLASH_SCAN_ERR]

DCMROD4[FLASH_ACCESS_ERR]

DCMROD4[FLASH_ECC_ERR]

DCMROD5[SW_NCF_3]

ERR051988: LFAST: LVDS receiver pad fault is latched when receiver is enabled before the common voltage is settled on the line

Description

During the LVDS Fast Asynchronous Serial Transmission (LFAST) startup procedure when LVDS Receiver pad (Rx) is enabled and the common voltage is not yet settled on the receiver line the LVDS pad fault status is latched in the Read-Only GPR On Functional Reset 7 register (DCM_GPR.DCMROD7[LVDS_3P3V_RX_FAULT] = 1). This fault cannot be cleared until the common voltage for LVDS pad is settled on the line connected to Rx pad. If the fault is not cleared and the LFAST startup procedure continues without getting the flag cleared the communication between the Master and Slave might not be established correctly at the high speed.

Workaround

In this case, the recommended LFAST startup procedure cannot be followed entirely. The Tx pad on the Slave node should be enabled by software before the Rx pad is enabled instead of being enabled from Master using ICLC frame with payload 0x31. Also, make sure LVDS receiver pad fault is cleared after enabling Rx pads on both LFAST Master and Slave node before continuing in the LFAST startup procedure. Do-while loop can be utilized for checking and clearing the fault.

ERR052066: SDADC: Parameters degradation caused by the signals on the GPIO/GPI pins

Description

If Sigma Delta Analog To Digital Converter (SDADC) channel is used in single ended mode and at the same time signal of 2 MHz (valid only for 20MHz sampling frequency) 4 MHz, 8 MHz,16 MHz, 24 MHz, 32 MHz and 40 MHz frequency is present on any pin from the VDD_HV_A power domain, SDADC parameters are out of the specification.

Workaround

1) Use differential mode.

2) When using single ended mode GPI/GPIO pins cannot be used for signals functionality of exact 2MHz (valid only for 20MHz sampling frequency) 4 MHz, 8 MHz, 16 MHz, 24 MHz, 32 MHz and 40 MHz frequency.

ERR052121: LPI2C: NACK Detect Flag can be set when IGNACK=1

Description

The NACK detect flag (MSR[NDF]) can be set even when the Controller Configuration 1 (MCFGR1[IGNACK]=0b1).

The LPI2C will not automatically generate a STOP or repeated START if the NACK detect flag (MSR[NDF]=0b1) and the ignore NACK are set (MCFGR1[IGNACK]=0b1). Thus, the transfer will continue as if the (MSR[NDF]) had not been set.

The LPI2C will continue to block a new START condition if (MSR[NDF]=0b1).

Workaround

When (MCFGR1[IGNACK]=0b1), the (MSR[NDF]) must be cleared by software, writing (MSR[NDF]=0b1) to allow new I2C transfers to start.

ERR052204: Zipwire (SIPI): Performance impact for devices with outstanding frames configured to 2

Description

SIPI module supports only the configuration of 4 outstanding frames. When a device with outstanding frames configured to 2 is connected the performance of the connected device is reduced because the ACK from this device is delayed. There are no frames lost but frames coming from the connected device are delayed because the SIPI module is overburdened. The throughput performance impact to the connected device can be in the range upto 15% - 30% depending on full transfer size versus communicating with a device with 4 outstanding frames.

The performance is impacted only when:

1) DMA buffered writes is enabled (DMA_TCDn.CHn_CSR.EBW = 1).

2) Zipwire communications operate in full duplex mode.

3) Outstanding frame is configured to 2 on connected device.

Workaround

1) Use DMA channel linking. Each transfer to the SIPI buffers must be followed by a dummy DMA transaction (The throughput performance impact is around 5%).

2) Do not use full duplex mode.

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

Suitability for use in automotive applications (functional safety) — This NXP product has been qualified for use in automotive applications. It has been developed in accordance with ISO 26262, and has been ASIL classified accordingly. If this product is used by customer in the development of, or for incorporation into, products or services (a) used in safety critical applications or (b) in which failure could lead to death, personal injury, or severe physical or environmental damage (such products and services hereinafter referred to as "Critical Applications"), then customer makes the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, safety, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. As such, customer assumes all risk related to use of any products in Critical Applications and NXP and its suppliers shall not be liable for any such use by customer. Accordingly, customer will indemnify and hold NXP harmless from any claims, liabilities, damages and associated costs and expenses (including attorneys' fees) that NXP may incur related to customer's incorporation of any product in a Critical Application.

NXP B.V. — NXP B.V. is not an operating company and it does not distribute or sell products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, μVision, Versatile — are trademarks and/or registered trademarks of Arm Limited (or its subsidiaries or affiliates) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

CoolFlux — is a trademark of NXP B.V.

CoolFlux DSP — is a trademark of NXP B.V.

EdgeLock — is a trademark of NXP B.V.

EdgeScale — is a trademark of NXP B.V.

eIQ — is a trademark of NXP B.V.

I2C-bus — logo is a trademark of NXP B.V.

Kinetis — is a trademark of NXP B.V.

MagniV — is a trademark of NXP B.V.

Mantis — is a trademark of NXP B.V.

NXP SECURE CONNECTIONS FOR A SMARTER WORLD — is a trademark of NXP B.V.

SafeAssure — is a trademark of NXP B.V.

SafeAssure — logo is a trademark of NXP B.V.

SuperFlash — This product uses SuperFlash® technology. SuperFlash® is a registered trademark of Silicon Storage Technology, Inc.

Synopsys & Designware — are registered trademarks of Synopsys, Inc.

Synopsys — Portions Copyright © 2018-2022 Synopsys, Inc. Used with permission. All rights reserved.

Please be aware that important notices concerning this document and the product(s) described herein, have been included in section 'Legal information'.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 3/2024
Document identifier: S32K396_1P40E