NXP Semiconductors
Errata

Document identifier: IMX8_1N94W
Rev. 5, 3/2023

IMX8_1N94W

Mask Set Errata

Mask Set Errata for Mask 1N94W

Revision History

This report applies to mask 1N94W for these products:

- IMX8
- i.MX 8QuadMax and i.MX 8QuadPlus

Table 1. Revision History

Revision	Date	Significant Changes
5	3/2023	The following errata were revised. • ERR051393
4	12/2022	The following errata were added. • ERR051393 • ERR050060 • ERR051407
3	5/2022	The following errata were removed. • ERR050060 The following errata were added. • ERR051041 • ERR051238 • ERR010910 • ERR051187 The following errata were revised. • ERR050341
2	5/2021	The following errata were removed. • ERR050340 • ERR010948 The following errata were added. • ERR050537 • ERR050395 • ERR050802 • ERR050058 • ERR050702 • ERR050660 • ERR050246 The following errata were revised.

Table continues on the next page...

Table 1. Revision History (continued)

Revision	Date	Significant Changes
		• ERR010946 • ERR010944 • ERR050341 • ERR010846 • ERR010945 • ERR010947
1	12/2019	The following errata were added. • ERR050421 The following errata were revised. • ERR050125
0	10/2019	Initial Revision

Errata and Information Summary

Table 2. Errata and Information Summary

Erratum ID	Erratum Title
ERR010846	SATA: Incorrect sized PRD entry used as part of the command descriptor can under some circumstances result in the controller becoming deadlocked
ERR010856	FTM: Safe state is not removed from channel outputs after fault condition ends if SWOCTRL is being used to control the pin
ERR010858	LPCG: IP clock gating register synchronization logic is sensitive to root clock gating
ERR010909	DB: Use of the Inline Encryption Engine (IEE) can cause a system lock-up under certain conditions
ERR010910	DC: Display Subsystem First Frame Programming Restriction
ERR010916	GPU: Texture border clamps to wrong maximum value
ERR010930	PCIE: EOM single point sample error/valid result is not correct
ERR010944	DRAM: In LPDDR4 mode, tMPCWR timing violation in incremental DQS2DQ Training
ERR010945	DRAM: PUB does not program LPDDR4 DRAM DDRPHY_MR22 prior to running DRAM ZQ calibration
ERR010946	DRAM: In LPDDR4 mode: Auto refresh must be disabled during DQS2DQ training
ERR010947	DRAM: DQS/DQSN glitch suppression resistors must be enabled during read-leveling
ERR011193	PCIE: EP, PM_PME: L1 Exit Does Not Occur when PME Service Timeout Mechanism Expires
ERR011194	PCIE: Plesiochronous loopback is not functional in PCIe Gen3
ERR011439	MIPI DSI: Checksum is incorrect for DCS command long packet writes with zero-length data payload

Table continues on the next page...

Table 2. Errata and Information Summary (continued)

Erratum ID	Erratum Title
ERR011543	FlexCAN: Nominal Phase SJW incorrectly applied at CRC Delimiter
ERR050053	ROM: USB HID device cannot be re-enumerated successfully after an unplug/plug USB cable operation
ERR050057	GPU: OpenCV and Vulkan conformance issue
ERR050058	ISI: Incomplete frames when using virtual channels 1, 2, 3
ERR050060	DC: PRG on the fly bypass switch issue
ERR050066	ISI: Data overflows occur when input streams exceed AXI transaction frequency
ERR050067	ISI: Adjacent processing pipelines within the ISI sub-system can experience loss of data
ERR050102	DRAM: Periodic hardware based DQS2DQ calibration is not supported
ERR050104	Arm/A53: Cache coherency issue
ERR050115	USB3: Port Configuration Response is not compliant with the USB compliance TD 7.17 test case
ERR050125	DRAM: Controller automatic derating logic may not work as intended when the LPDDR4 memory temperature is above 85C at initialization
ERR050135	JPEG DECODER: multi-frame jpeg bitstream may not be correctly decoded when there is a small size frame inside
ERR050141	USB2: Endpoint conflict issue in device mode
ERR050145	ISI: Memory overwrite occurring outside of allocated buffer space corrupting system memory
ERR050147	USB3: Multiple DMA write transfer complete interrupts are generated before final write access handshake to the AXI bus
ERR050148	USB3: Race condition possible during software update to TRB in the system memory and DMA reads of same TRB
ERR050149	USB3: TRB OUT endpoints transfer blockage and performance delays
ERR050171	Arm/A53: Hot plug issue
ERR050183	DC: 4Kp60 performance limitations
ERR050184	SNVS: ON_OFF_BUTTON input limitations in low-power modes
ERR050185	HDMI-TX/RX: No CEC TX/RX status interrupt
ERR050246	FlexCAN: Receive Message Buffers may have its Code Field corrupted if the Receive FIFO function is used
ERR050341	DRAM: LPDDR4 VREF training may result in a non-optimal value
ERR050368	SNVS: HDCP key install fails
ERR050395	ENET: Ethernet RX hang when receiving traffic through multiple queues
ERR050421	ADC: Missed triggers on ADC0 at low temperature
ERR050537	FlexSPI: Read timing sequence mismatches with several existing SPI NOR devices in dual, quad, and octal modes

Table continues on the next page...

Table 2. Errata and Information Summary (continued)

Erratum ID	Erratum Title
ERR050660	ISI: HDMI-RX legacy YUV420 8-bit format incorrectly captured in ISI
ERR050702	ISI: YUV444 video format not properly captured from HDMI-RX interface
ERR050802	DRAM: The LPDDR4 DRAM initialization may experience large training time variations or stall when Read Data Bus Inversion (DBI) bit deskew training is enabled
ERR051041	LPIT: CVAL cannot be read correctly during timer running
ERR051187	DisplayPort: Sampling Counters May Overflow at Controller Clock Frequency Above 170MHz
ERR051238	Hot plug-in/out on SATA Port Multiplier Ports (PMP) may not be detected
ERR051393	Arm/Cortex-A core memory corruption
ERR051407	USB3: USB full speed mode may fail to work

Known Errata

ERR010846: SATA: Incorrect sized PRD entry used as part of the command descriptor can under some circumstances result in the controller becoming deadlocked

Description

If an incorrectly sized PRD entry is received as part of the command descriptor, then the controller can under some circumstances become dead locked. This arises due to a size error being decoded which states the controller has data outstanding, and results in a miscalculation of the number of outstanding data transfers and the controller waiting indefinitely for these transfers.

Workaround

The controller requires to be reset to recover from this condition. The impact here is low as the controller is already in an error state due to the size of data mismatch. The SATA controller has its own soft reset control.

NXP software avoids the problem by always programming a correct-sized PRD entry. User should be aware of the issue when developing their own driver.

ERR010856: FTM: Safe state is not removed from channel outputs after fault condition ends if SWOCTRL is being used to control the pin

Description

If an FTM channel output is being controlled using the software output control register (FTM_SWOCTRL) and fault detection is also enabled for the channel, then when a fault is detected the output is forced to its safe value. However, when the fault condition has been cleared, the channel output will stay in the safe state instead of reverting to the value programmed by the FTM_SWOCTRL register.

Workaround

If fault control is enabled while the software output control register is also being used (FTM_SWOCTRL), then the FTM should be configured as follows:

-- FTM_MODE[FAULTM] configured for manual fault clearing (0b10)

-- For devices that include the FTM_CONF[NUMTOF] field, it must be cleared to 0b00000 (TOF set for each counter overflow). For FTM versions that don't include the FTM_CONF[NUMTOF] field this doesn't apply.

The procedure below must be used in the TOF interrupt handler when a fault is detected to ensure that the outputs return to the value configured by FTM_SWOCTRL.

1. Check the value of FTM_FMS[FAULTF].

-- If FTM_FMS[FAULTF] = 1 (fault occurred or is occurring), then set a variable to indicate that a fault was detected and continue to step 2.

-- If FTM_FMS[FAULTF] = 0 but the fault variable is set (fault is not active, but was previously detected), continue to step 6.

2. Write the FTM_OUTMASK register to set the bit or bits corresponding to any channels that are controlled by FTM_SWOCTRL to temporarily inactivate the channel output.

3. Clear fault conditions by reading the FTM_FMS register and then writing FTM_FMS with all zeroes.

4. Clear the FTM_SC[TOF] bit by reading the FTM_SC register, then writing a 0 to FTM_SC[TOF].

5. Exit the interrupt handler to skip following steps (they will execute the next time the TOF handler is called).

6. Clear the FTM_SWOCTRL by writing all zeroes to it.

7. Write FTM_SWOCTRL with the desired value again.

8. Clear the FTM_OUTMASK bits that were set in step 2.

9. Clear the fault variable that was set in step 1 when the fault condition was originally detected.

10. Clear the FTM_SC[TOF] bit by reading the FTM_SC register, then writing a 0 to FTM_SC[TOF].

ERR010858: LPCG: IP clock gating register synchronization logic is sensitive to root clock gating

Description

LPCG registers are accessed using a clock domain that is independent of the gated clock. LPCG registers require up to 4 cycles of the gated clock to ensure synchronization with the gating logic. Back-to-back writes to LPCG registers that occur within the 4-cycle window can be ignored by the LPCG logic and the LPCG register will not reflect the state of the gating logic.

Workaround

Software must ensure that at least 4 cycles of the gated clock elapse between writes to the LPCG registers. Writes to LPCG registers across the system interconnect will incur latency that avoids the LPCG synchronization window unless the gated clock is running at less than 24 MHz.

If software cannot ensure sufficient delay between LPCG writes to avoid the issue, LPCG must not be used for clock control. As an alternative, SCFW APIs can be used to gate clock root inputs to LPCG cells.

ERR010909: DB: Use of the Inline Encryption Engine (IEE) can cause a system lock-up under certain conditions

Description

A deadlock case can happen in the DRAM block (DB), when IEE_DET is enabled and a transaction loop occurs. If subsystem “A” (e.g. GPU0, HSIO, or DBLOG) has back-to-back accesses to the LSIO space (e.g. OCRAM), and at the same time, another subsystem “B” (e.g. AP, CM4 cores etc) is performing back-to-back burst access to the SCU/CM4 cores TCM space with IEE_DET enabled, then a deadlock can happen.

Two or more consecutive transactions (either read/write) going to any IO space outside DRAM, OCRAM or FlexSPI with IEE_DET enabled can have a deadlock, if two consecutive transactions are targeting this space, these transactions never finish.

Workaround

IEE operation is limited to DRAM, OCRAM and FlexSPI memory space only. Do not mark any space outside DRAM, OCRAM and FlexSPI as encrypted.

ERR010910: DC: Display Subsystem First Frame Programming Restriction

Description

The interaction between the display controller (DC) and the display prefetch resolve engine (DPR) has a software programming restriction which must be followed for proper operation of the display subsystem. Failure to adhere to the software restriction will result in a system hang.

At start-up a dummy frame is required to synchronise the DC, DPR and associated buffers. As a result the DPR prefetches data into its buffers, however as the DC considers this to be a dummy frame it does not request the stored pixel data to be used, but when it attempts to start the next, or first valid frame, the DPR will stall because an end of frame (EOF) was not received for the dummy frame.

Workaround

As the dummy frame required to synchronise the DPR and DC is not consumed by the DC, the timing controller should not be enabled until this dummy frame completes. Wait until the first frame completes (DisEngCfg_FrameCompleteX IRQ ; X is 0 or 1) before enabling the TCON.

ERR010916: GPU: Texture border clamps to wrong maximum value

Description

Out-of-bounds (OOB) data is discarded when accessed outside the texture map whenever the texture coordinates are greater than 1.0 (normalized value). The OOB data is discarded, however, the correct BorderColor value is returned. Therefore, the OOB access does not cause an immediate failure or data corruption.

There are two rare cases where an observable error may occur:

- If the MMU does not allow access, then an access violation may occur.

- If this OOB access lands in a different texture map with a different texel format, the polluted cache will return the wrong data for a subsequent access that hits the same location in the texture cache.

Workaround

For most popular 2D textures, software will append one line of superTile, each superTile 64*64*bpp, for most popular 32bpp formats such as RGBA8 texture one superTile is 16KB. Therefore, for a 1920 wide texture, software will add 16KB * (1920/64) = 480KB.

ERR010930: PCIE: EOM single point sample error/valid result is not correct

Description

There is an eye monitor in the SerDes analysis which can monitor the following:

a. Error and valid bits of a certain duration

b. Eye width

c. Eye height

d. Eye area

However, there is a design issue with item (a) causing incorrect error/valid bit results.

Workaround

Customers must not use the error/valid count results to check the eye quality. Instead, use the eye width, eye height, or eye area.

ERR010944: DRAM: In LPDDR4 mode, tMPCWR timing violation in incremental DQS2DQ Training

Description

In LPDDR4 mode with incremental DQS2DQ Training enabled and speed grade > 2133 Mbps, the hardware incremental dqs2dq training routine performs Power Down (PD) Entry-Exit Cycle to reset the MPC WR-RD FIFO pointers in the DRAM. The PUB sends the MPC WRFIFO command after waiting for tXP from PD Exit. However, JEDEC specification requires waiting an additional tMPCWR after tXP timing. This extra tMPCWR timing is not handled by the PUB Training algorithm, resulting in a violation of tMPCWR JEDEC parameter.

Workaround

Do not run incremental DQS2DQ Training of the PHY in LPDDR4 mode.

ERR010945: DRAM: PUB does not program LPDDR4 DRAM DDRPHY_MR22 prior to running DRAM ZQ calibration

Description

When the PHY Utility Block (PUB) initializes the DRAM, the DDRPHY_MR22 is programmed after ZQ Calibration. This may result in incorrect ZQ calibration results on the LPDDR4 DRAM side, because DDRPHY_MR22[CODT] works as the controller On Die Termination (ODT) replica during the Pull-Up calibration. Therefore the expected controller ODT must be programmed into DDRPHY_MR22 prior to the DRAM ZQ calibration.

Workaround

Run DRAM Initialization twice. Scripts provided by NXP’s DRAM RPA (Register Programming Aid) implement the required workaround.

ERR010946: DRAM: In LPDDR4 mode: Auto refresh must be disabled during DQS2DQ training

Description

If auto refresh is enabled during DQS2DQ training, a JEDEC Specification violation may occur. Auto refresh must be disabled during DQS2DQ training, which is performed during initial power-up (cold boot).

Workaround

For initial power-up (cold boot), disable auto refresh during DQS2DQ training. For self-refresh (warm boot), do not run DQS2DQ training. Restore the saved register values prior to self-refresh entry. Scripts provided by NXP’s DRAM RPA (Register Programming Aid) implement the required workaround.

ERR010947: DRAM: DQS/DQSN glitch suppression resistors must be enabled during read-leveling

Description

By default DQS/DQSN glitch suppression resistors are disabled. When external DQS/DQSn are not driven to valid differential states, the DQS cell's core-side outputs become unknown. This causes errors in the read-leveling gate training.

Workaround

Enable the strongest 355 ohm glitch suppression resistors during gate training. Scripts provided by NXP’s DRAM RPA (Register Programming Aid) implement the required workaround through DDRPHY_DX8SLbDQSCTL register.

ERR011193: PCIE: EP, PM_PME: L1 Exit Does Not Occur when PME Service Timeout Mechanism Expires

Description

Impacted Configuration(s): Upstream Port configurations:

CC_DEVICE_TYPE =DM and device_type =4'b0000

Defect Summary:

When a function issues a PM_PME Message, it sets the PME_Status bit. If the Downstream port has not cleared the PME_Status bit within 100ms, a PME Service timeout occurs. At this point, the Upstream port must resend the PM_PME message.

In the current implementation of the controller, the PME Service timeout does not trigger an exit from L1 to resend the PM_PME message.

System Usage Scenario:

Upstream ports using a wake-up mechanism followed by a power management event (PME) message.

Consequence(s):

The defect has the following effect:

The PME service routine cannot make forward progress until the PM_PME message is resent.

Workaround

Poll the PME_Status bit after sending the PME message to exit L1 state. If this bit remains 1 for 100ms or more, SW must re-toggle bit 10 "PCIE_CTRL_APPS_PME" of register "SRC_PCIEPHY_RCR" .

ERR011194: PCIE: Plesiochronous loopback is not functional in PCIe Gen3

Description

Customers should be using mesochronous loopback when sending arbitrary bit streams.

Plesiochronous loopback: Is loopback from Rx back to Tx after the PCIe elastic buffer function in the PCS.

The intent of this reverse loopback scheme is to send arbitrary bit-streams through the elastic FIFO on the Rx side of the PCS and back through the Tx side of the PCS into the PMA. However, this does not work at Gen3 speed. This mode is not practical because the entire PCS PCIe pipeline is designed for protocol-dependent data, and requires many bypass paths to enable arbitrary bit streams through it. Moreover, there is no way to support elasticity when the bit-stream is protocol-agnostic, rendering the elastic FIFO useless.

Mesochronous (meso) loopback: Is loopback from Rx back to Tx before any elastic buffer, hence requiring 0ppm frequency difference between TxClk and RxClk, and requires TxClk and RxClk to be phase-adjusted using an automatic CDR skip-bit routine (as described in the PUG). Meso loopback assumes that the intersection set of the setup+hold margin for all 20 bits in the Rx to Tx STA path has a large open window. The SDC constraints were originally intended to contain max_delay and min_delay constraints to ensure this, but customers may not have optimized the window. Historically, mesochronous mode rarely worked at the highest protocol speeds due to this dependency on customer’s timing optimization.

Workaround

Customers must use meso loopback when sending arbitrary bit streams.

ERR011439: MIPI DSI: Checksum is incorrect for DCS command long packet writes with zero-length data payload

Description

According to the MIPI DSI specification, long packets are comprised of a Packet Header and a payload of 0 to 2^16-1 bytes. For the special case of a zero-length payload, the specification requires the checksum must be set to 0xFFFF.

The MIPI DSI controller produces an incorrect checksum for DCS commands issued via long packets with zero-length payloads in DSI Low-Power mode (LP). There is no issue for similar commands issued in DSI High-Power mode (HP).

This issue should not affect normal application operation because packets with zero data length will normally be sent using the short packet format. However, because the MIPI DSI spec specifically states this behavior, MIPI DSI certification will fail with long packets of zero-length.

Workaround

Use short packet format to send DCS commands with zero length data payloads.

ERR011543: FlexCAN: Nominal Phase SJW incorrectly applied at CRC Delimiter

Description

During the reception of a CAN-FD frame when the Bit Rate Switch (BRS) is enabled, the Synchronization Jump Width (SJW) for the CRC Delimiter bit is incorrectly defined by the Nominal Phase SJW. The CAN specification stipulates that the CRC Delimiter bit should have a SJW set by the Data Phase SJW.

When a resynchronization event is triggered for the CRC delimiter bit (recessive in correct operation), the sample point will be adjusted by an amount as defined by the Nominal Phase SJW rather than the specified Data Phase SJW. This may result in the incorrect detection of a dominant bit leading to a CAN error frame. However, as the CRC delimiter bit position will only apply the SJW upon the detection of an unexpected dominant bit on the CAN bus, an error frame is already likely. For the case the SJW is applied at the CRC delimiter and a recessive bit is not detected, the receiving node will issue an error frame.

The CAN protocol is designed to handle resynchronization errors and hence the CAN bus will recover from the insertion of the incorrect SJW at the CRC delimiter. Upon detecting the error frame the transmitting node will re-transmit the frame.

The following FlexCAN configurations are not affected:

- Classical CAN frames (CAN 2.0B)
- CAN FD frames with bit rate switch disabled (BRS = 0)
- CAN FD frames with Nominal Phase SJW equal to Data Phase SJW
- CAN FD transmissions

Configuration for the FlexCAN:

- Nominal Phase SJW is configured by the Resync Jump Width bit in the CAN Control Register 1 (CAN_CTRL1[RJW]) or by the Extended Resync Jump Width bit in the CAN Bit Timing Register (CAN_CBT[ERJW])
- Data Phase SJW is configured by the Fast Resync Jump Width bit in the CAN FD Bit Timing Register (CAN_FDCBT[FRJW])

Workaround

The robustness of the CAN protocol ensures that the receiver automatically recovers from the application of the incorrect SJW. The CAN protocol is designed to recover from resynchronization errors and hence any frame that is not correctly received will be re-sent by the transmitting node.

ERR050053: ROM: USB HID device cannot be re-enumerated successfully after an unplug/plug USB cable operation

Description

The USB HID device enumerates successfully on the Host side when booting from serial download mode. However, after disconnecting the USB cable and re-connecting the cable again, the USB HID device will not re-enumerate on Host side because ROM incorrectly resets the USB.

Workaround

Reset the device, or power down and re-power on the device.

ERR050057: GPU: OpenCV and Vulkan conformance issue

Description

GPU may hang when running OpenCV or Vulkan conformance tests under corner conditions.

Workaround

Software workaround has been integrated into L4.14 BSP release and later release. This workaround has a small performance impact <1% during OpenCV or Vulkan tests.

ERR050058: ISI: Incomplete frames when using virtual channels 1, 2, 3

Description

Except for virtual channel 0, virtual channels 1, 2, and 3 do not have proper VSYNC timing from MIPI CSI2 when different cameras are multiplexed together. As a result, frames stored in memory can be corrupted due to missing last lines.

Workaround

Virtual channels 1, 2, and 3 do not work normally for multiplexed cameras. Only single camera operation is supported by the MIPI CSI2 interface.

ERR050060: DC: PRG on the fly bypass switch issue

Description

When the display controller switches the DPR/PRG from bypass to non-bypass on the fly, it causes a sync error. A screen artifact (3-4 lines of the overlay) can be seen at the top of the overlay. The bypass to non-bypass on the fly switch can occur when the overlay pixel format changes from DPR/PRG unsupported to supported.

Workaround

Careful timing of the overlay change can hide this problem. Following the sequence “overlay OFF – 1 frame – overlay ON” can also hide the problem. Because this workaround requires a deterministic handling of interrupts, a non-realtime OS, such as Linux, cannot guarantee the timing of the overlay change.

ERR050066: ISI: Data overflows occur when input streams exceed AXI transaction frequency

Description

The Image Sensing Interface (ISI) has a short elasticity buffer relative to the length of a line. The buffer can be as few as 85 pixels or as many as 512 pixels depending on the output format. Most RGB formats have 128 pixels. Because of the short buffer, if there is any delay in latency, then an overflow can occur. The possibility of overflow increases when the number of active channels increases.

In addition, memory reads and the last line of a scaling process consume data as fast as possible (instead of at the rate of the incoming pixel stream), therefore, the output buffer fills faster and requires even lower latency to process the data.

Workaround

The design target was intended to support up to a single 8 Mpixel (4K) stream at 30 fps, or multiple streams up to the equivalent data rate. However, combinations of sensors which add up to less than 2Mpixel are supported with current design. That’s to say, if 1 sensor is used, 2Mpixels stream can be supported; if 2 sensors are used, 1Mpixels of each stream can be supported; and so on.

In the case of scaling, the last line of each frame must be cropped and discarded.

To reduce overflow possibility, one possibly way is to lower ISI clock which help slow down the data to output buffer.

ERR050067: ISI: Adjacent processing pipelines within the ISI sub-system can experience loss of data

Description

Using adjacent channels where one channel’s line ends when the next channel’s line begins (common in virtual channel functionality) can cause the second channel to skip a line every 8 or 16 lines.

Using adjacent channels can also effect the width and format of the line by creating a final write that does not fill a 128 byte buffer.

Workaround

For virtual channel applications the pipeline order can be adjusted to avoid adjacent channel assignments, for example, VC 0, 1, 2, and 3 assigned to pipelines 0, 2, 1, 3.

ERR050102: DRAM: Periodic hardware based DQS2DQ calibration is not supported

Description

If periodic hardware based DQS2DQ calibration is enabled, the resultant latency introduced can cause underrun conditions, or worst case a lock-up, in some of the key sub-systems, such as the display and imaging interfaces, impacting their performance capabilities.

Workaround

Currently DQS2DQ calibration only takes place on power up and when resuming from low power modes. To date no failures or stability issues have been observed across the full process, voltage and temperature ranges.

ERR050104: Arm/A53: Cache coherency issue

Description

Some maintenance operations exchanged between the A53 and A72 core clusters, involving some Translation Look-aside Buffer Invalidate (TLBI) and Instruction Cache (IC) instructions can be corrupted.

The upper bits, above bit-35, of ARADDR and ACADDR buses within in Arm A53 sub-system have been incorrectly connected. Therefore ARADDR and ACADDR address bits above bit-35 should not be used.

Workaround

The following software instructions are required to be downgraded to TLBI VMALLE1IS:

TLBI ASIDE1

TLBI ASIDE1IS

TLBI VAAE1

TLBI VAAE1IS

TLBI VAALE1

TLBI VAALE1IS

TLBI VAE1

TLBI VAE1IS

TLBI VALE1

TLBI VALE1IS

The following software instructions are required to be downgraded to TLBI VMALLS12E1IS:

TLBI IPAS2E1IS

TLBI IPAS2LE1IS

The following software instructions are required to be downgraded to TLBI ALLE2IS:

TLBI VAE2IS

TLBI VALE2IS

The following software instructions are required to be downgraded to TLBI ALLE3IS:

TLBI VAE3IS

TLBI VALE3IS

The following software instructions are required to be downgraded to TLBI VMALLE1IS when the Force Broadcast (FB) bit [9] of the Hypervisor Configuration Register (HCR_EL2) is set:

TLBI ASIDE1

TLBI VAAE1

TLBI VAALE1

TLBI VAE1

TLBI VALE1

The following software instruction is required to be downgraded to IC IALLUIS:

IC IVAU, Xt

Specifically for the IC IVAU, Xt downgrade, setting SCTLR_EL1.UCI to 0 will disable EL0 access to this instruction. Any attempt to execute from EL0 will generate an EL1 trap, where the downgrade to IC ALLUIS can be implemented.

ERR050115: USB3: Port Configuration Response is not compliant with the USB compliance TD 7.17 test case

Description

USB 3.0 Compliance TD 7.17 test case is used to verify that a downstream PUT will go to SS.Inactive if tPortConfiguration expires, and an upstream PUT will go to SS.Disabled if tPortConfiguration expires. However, this test case fails because the port configuration response is not compliant with the TD 7.17 test case.

This requirement is not present in the USB 3.0 specification, however, the USB 3.0 compliance TD 7.17 test case requires it. This test does not affect user applications and it does not affect USB 3.0 function.

Workaround

To pass the USB compliance test waive the TD 7.17 tPortConfiguration test.

ERR050125: DRAM: Controller automatic derating logic may not work as intended when the LPDDR4 memory temperature is above 85C at initialization

Description

LPDDR4 memories require periodic refreshes to maintain memory contents. Per the JEDEC specification JESD209-4 the memory refresh rate needs to increase and timings de-rated as the memory operational temperature exceeds vendor-defined temperature thresholds. The LPDDR4 Mode Register 4 (MR4) contains temperature/refresh rate information and a Temperature Update Flag (TUF).

An issue exists with the automatic derating logic of the DDR controller that only samples the LPDDR4 MR4 register when the Temperature Update Flag (TUF) field (MR4[7]) is 1’b1. If the LPDDR4 memory is initialized and starts operation above 85ºC (MR4[2:0] > 3'b011), the MR4 Temperature Update Flag (TUF) will not set. The DDR Controller will therefore not automatically adjust the memory refresh rate or de-rate memory timings based on the LPDDR4 memory temperature. This may result in the controller incorrectly setting the refresh period, potentially causing the LPDDR4 memory losing data contents and leading to possible data integrity issues above 85ºC. The actual memory temperature threshold values may vary depending on memory vendors.

If the LPDDR4 memory temperature remains below 85ºC at initialization (Consumer-grade memory devices), then the derating logic works as intended, automatically adjusting the memory refresh period and memory timing during the entire system operation. The issue does not occur in this specific scenario since derating is not required.

This erratum does not impact other SoC supported DDR memory interfaces such as DDR4 or DDR3L.

Workaround

The software workaround for LPDDR4 based systems is to check the memory temperature via the MR4 and determine if it is above 85ºC at initialization. If the temperature is below 85 then automatic temperature derating logic is left enabled (default setting), otherwise the derating logic is disabled and software should manually adjust the memory refresh rate and memory timings. Once the memory temperature is below 85C (MR4[2:0] == 3'b011), the software should readjust the memory refresh rate and memory timings to nominal settings and then re-enable the automatic derating logic.

The software workaround has been integrated into the BSP GA release : (imx_4.14.98_2.0.0_p1 (SCFW v1.2.1))

ERR050135: JPEG DECODER: multi-frame jpeg bitstream may not be correctly decoded when there is a small size frame inside

Description

When the JPEG decoded frame with a resolution that is no larger than 64x 64 and it is followed by a next decoded frame with a larger resolution, then this next decoded frame may be corrupted.

Workaround

The decoded image resolution should be larger than 64x 64.

ERR050141: USB2: Endpoint conflict issue in device mode

Description

An endpoint conflict occurs when the USB is working in device mode and an isochronous IN endpoint exists.

When the endpointA IN direction is an isochronous IN endpoint, and the host sends an IN token to endpointA on another device, then the OUT transaction may be missed regardless the OUT endpoint number. Generally, this occurs when the device is connected to the host through a hub and other devices are connected to the same hub.

The affected OUT endpoint can be either control, bulk, isochronous, or an interrupt endpoint.

After the OUT endpoint is primed, if an IN token to the same endpoint number on another device is received, then the OUT endpoint may be unprimed (Cannot be detected by SW), which causes this endpoint to no longer respond to the host OUT token, and thus, no corresponding interrupt occurs.

Workaround

Do not connect to a hub in the case when ISO IN endpoint(s) is used. When the hub(s) must be connected in this scenario, the endpoint number(s) of the ISO IN endpoint(s) should be different from the endpoint number(s) of any type of IN endpoint(s) used in any other device(s) connected to the same host.

ERR050145: ISI: Memory overwrite occurring outside of allocated buffer space corrupting system memory

Description

Under marginal timing conditions, when an incomplete frame is received, resulting in an early or late VSYNCH error, it is possible for the ISI to overwrite system memory outside its allocated buffer space, resulting in unpredictable behavior.

Workaround

To prevent this, the xRDC can be programmed to grant write access to the ISI only within its allocated frame buffer space. User applications must ensure the SCFW creates an ISI domain containing the ISI itself and its frame buffers, which will prevent overwrites into system memory. The ISI can generate an interrupt to indicate an exception has occurred, if required.

ERR050147: USB3: Multiple DMA write transfer complete interrupts are generated before final write access handshake to the AXI bus

Description

In USB device mode and Multiple DMA transfers mode, the DMA write-transfer-complete-interrupt is generated multiple clock cycles after the final DMA write access on the AXI bus. The transfer does not wait for completion of the system memory write access handshake.

Delay between the last DMA write access and the DMA interrupt request is determined by an internal operation of the DMA and lasts longer than 50ns. The current DMA interrupt request delay is shorter after DMA write access. Within the interrupt handler, software checks the interrupt source to determine which source introduces the additional delay. During these checks, software has the opportunity to access the system memory data before the DMA write is complete.

This issue may be critical for AXI interconnects that use buffering for write accesses. For these systems, READ access to the system memory may be executed before the WRITE access is complete to the same location even if the WRITE access was requested much earlier than read access.

Workaround

Using Singular DMA transfer mode can avoid this issue, by setting DSING to 1 and set DMULT to 0 in register USB_CONF.

ERR050148: USB3: Race condition possible during software update to TRB in the system memory and DMA reads of same TRB

Description

Transfer Ring Block (TRB) data structure is larger than 64-bit and therefore requires two separate read accesses on a 64-bit data bus. Because of race conditions between software updates to TRB and DMA reads of the TRB, it is possible that DMA read access may be interleaved with the software write access to the same TRB. The race condition might cause TRB content read by DMA to be inconsistent leading to data corruption during the USB transfer.

This situation can occur in USB device mode.

Critical race condition scenario:

Initial assumption: TRB ownership (cycle bit) is set to software and software is expected to update TRB sequence of events.

1. DMA reads first part of TRB that stores pointer to the USB data buffer.

2. Software writes first part of the TRB and sets new value of the pointer.

3. Software writes second part of the TRB that stores TRB ownership bit (cycle bit) and sets ownership to DMA.

4. DMA reads second part of the TRB and determines that ownership is set to DMA and begins processing data buffer using incorrect pointer that has been fetched during step 1.

Workaround

Recommend software driver workaround:

Software checks DMA enqueue and dequeue pointers to determine status of the DMA ring. If the DMA is near the end of the TRB ring the software postpones the update of the ownership bit in the system memory. Software waits until DMA stops and reports the end of the transfer ring by indicating a "descriptor missing" interrupt. The ownership (cycle) bit is updated by software when the DMA is stopped.

Limitations of the Software workaround:

There is a potential performance impact although none observed in real applications.

ERR050149: USB3: TRB OUT endpoints transfer blockage and performance delays

Description

During USB device mode, the on-chip buffer for OUT endpoints is implemented as a FIFO queue for all USB OUT packets.

All configured and enabled Device OUT endpoints are ready to receive OUT data packets when the Device FIFO queue is available whether or not the TRB ring is prepared by software and whether the DMA is ready to read OUT packets.

When an OUT packet is received but the DMA is not prepared for transfer (TRB is missing) the DMA generates a "descriptor missing" interrupt to notify software that the transfer ring for DMA should be prepared.

Linux Class driver cannot guarantee creation of the TRB in response to "descriptor missing" interrupt.

Workaround

Recommend software driver workaround:

In response to the "descriptor missing" interrupt the software driver prepares the local buffer and enables DMA to receive data from the OUT FIFO to the local buffer in the system memory.

Limitations of the software workaround:

- The local buffer created by the software driver may overflow when a USB Class Application in Linux does not receive data for an extended time.

- The "Descriptor missing" interrupt service impacts application performance (particularly ISO transfers) especially when the "descriptor missing" interrupt is serviced with extended delays.

ERR050171: Arm/A53: Hot plug issue

Description

The A53 core cluster can hang during a power-down/up sequence of the L2 cache domain, as a result of an uninitialized FIFO.

Workaround

To prevent a potential hang of the A53 core cluster after every power cycle, including boot, the following code must be executed within the Arm Trusted Firmware (ATF) to initialize the problematic FIFO to a known state:

```
func bl31_entrypoint
#ifdef PLAT_imx8qm
ldr x1, stm
ldr w0, =0x80000000
str w0, [x1] /* 1 */
```

… a total of 32 single writes (str commands) are required

```
str w0, [x1] /* 32 */

#endif

.ltorg

stm:

.quad 0x5D1B0020
```

The code example shown is as it appears within the ATF whenever the A53 or A72 core comes out of reset. Where an operating system does not utilize ATF, the workaround needs to be implemented elsewhere.

ERR050183: DC: 4Kp60 performance limitations

Description

When processing 4K (3840 x 2160) images the Display Controller (DC) can suffer from data underruns preventing the frame rate from reaching the desired 60 frames per second (fps). Typically the image displayed is as a result of a composition engine combining the VPU decoded output together with a GPU graphic layer / overlay.

To prevent underruns the Display Prefetch and Resolve (DPR) module must output bursts of 512 bytes, however it can only read VPU tile and GPU super-tile formats in bursts of 64 bytes, this prevents the DPR from reading both the VPU tile and GPU super-tile formats directly without underruns occurring when attempting to display 4K images at 60 fps.

Workaround

The Wayland display server protocol used in the Linux BSP package utilises the GPU 3D engine as the compositor by default resulting in underruns when processing 4K images at 60 fps.

The GPU 2D engine can be used to convert the VPU tile format into a linear ARGB tile enabling the DPR to output 512 byte bursts. However, the GPU 2D processing power is not sufficient to convert a VPU tile and blend a graphic layer in a single run, or blend two 4K planes in a single run at 60 fps.

As a result of this, the graphics layer (e.g. tool bar) must be limited to a quarter of the display size to support 4K images at 60 fps.

To use the GPU 2D engine, it must be enabled in the Weston configuration file:

/etc/xdg/weston/weston.ini

```
[core]

..

use-g2d=1

..
```

For the Android BSP, which does not use Wayland, a different approach is required. Here the hardware composition capabilities of the DC are utilised to combine the VPU decoded output with the graphics layer to enable the DPR to output bursts of 512 bytes. The graphics layer can be linear or GPU tile format, the VPU output must be linear. The GPU 2D engine is used to convert the VPU tile format to linear.

Refer to the Android BSP documentation (User Guide) on how to accelerate high resolution video playback through the DC.

ERR050184: SNVS: ON_OFF_BUTTON input limitations in low-power modes

Description

The ON_OFF_BUTTON input cannot generate a wakeup event whilst in the KS1 low-power mode. A periodic wake-up can be used to monitor the ON_OFF_BUTTON input during KS1, however to ensure ON_OFF_BUTTON input event is recognized by this periodic check, the ON_OFF_BUTTON input must remain asserted longer than the length of the periodic wake-up timer.

Workaround

The SCU Firmware (SCFW) has implemented a periodic wakeup event using the SYSCTR timer that runs from the 32 KHz clock source. The wakeup interval is set based on the SCU WDOG timeout, configured to be one second in the SCFW startup code. This is not a configurable timeout period.

Shorter wakeups could be configured through the selected software BSP/SDK to reduce the fixed SCFW defined one second period, however, shortening the period will come at the expense of increased overall power consumption when using KS1.

ERR050185: HDMI-TX/RX: No CEC TX/RX status interrupt

Description

The interrupt steering logic does not support the CEC status interrupts within in the HDMI-TX and HDMI-RX sub-systems.

Workaround

When CEC is in use, poll CEC RX and TX status registers every 20 msecs for any status changes.

ERR050246: FlexCAN: Receive Message Buffers may have its Code Field corrupted if the Receive FIFO function is used

Description

If the Code Field of a Receive Message Buffer is corrupted it may deactivate the Message Buffer, so it is unable to receive new messages. It may also turn a Receive Message Buffer into any type of Message Buffer as defined in the Message buffer structure section in the device documentation.

The Code Field of the FlexCAN Receive Message Buffers (MB) may get corrupted if the following sequence occurs.

1- A message is received and transferred to an MB (i.e. MBx)

2- MBx is locked by software for more than 20 CAN bit times (time determines the probability of erratum to manifest).

3- SMB0 (Serial Message Buffer 0) receives a message (i.e. message1) intended for MBx, but destination is locked by the software (as depicted in point 2 above) and therefore NOT transferred to MBx.

4- A subsequent incoming message (i.e. message2) is being loaded into SMB1 (as SMB0 is full) and is evaluated by the FlexCAN hardware as being for the FIFO.

5- During the message2, the MBx is unlocked. Then, the content of SMB0 is transferred to MBx and the CODE field is updated with an incorrect value.

The problem does not occur in cases when only Rx FIFO or only a dedicated MB is used (i.e. either RX MB or Rx FIFO is used). The problem also does not occur when the Enhanced Rx FIFO and dedicated MB are used in the same application. The problem only occurs if the FlexCAN is programmed to receive in the Legacy FIFO and dedicated MB at the same application.

Workaround

This defect only applies if the Receive FIFO (Legacy Rx FIFO) is used. This feature is enabled by RFEN bit in the Module Control Register (MCR). If the Rx FIFO is not used, the Receive Message Buffer Code Field is not corrupted.

If available on the device, use the enhanced Rx FIFO feature instead of the Legacy Rx FIFO. The Enhanced Rx FIFO is enabled by the ERFEN bit in the Enhanced Rx FIFO Control Register (ERFCR).

The defect does not occur if the Receive Message Buffer lock time is less than or equal to the time equivalent to 20 x CAN bit time.

The recommended way for the CPU to service (read) the frame received in a mailbox is by the following procedure:

1. Read the Control and Status word of that mailbox.

2. Check if the BUSY bit is deasserted, indicating that the mailbox is not locked. Repeat step 1) while it is asserted.

3. Read the contents of the mailbox.

4. Clear the proper flag in the IFLAG register.

5. Read the Free Running Timer register (TIMER) to unlock the mailbox

In order to guarantee that this procedure occurs in less than 20 CAN bit times, the MB receive handling process in software (step 1 to step 5 above) should be performed as a 'critical code section' (interrupts disabled before execution) and should ensure that the MB receive handling occurs in a deterministic number of cycles.

ERR050341: DRAM: LPDDR4 VREF training may result in a non-optimal value

Description

During LPDDR4 initialization, when performing LPDDR4 VREF training (through DDRPHY_PIR[VREF]), a discrepancy may be observed between the training-generated DDRPHY_MR14 value and an actual signal-measured VREF_DQ value such that the “trained” DDRPHY_MR14 value may not result in the most optimal VREF_DQ setting. However, the discrepancy is minimal such that no DRAM data failures have occurred due to this.

Workaround

A software workaround has been included in the SCU firmware (SCFW) since version 1.4.0. The software workaround has been included in the MX8QXP DDR Register Programming Aid (RPA) since RPA version 14.

ERR050368: SNVS: HDCP key install fails

Description

In order to support High-bandwidth Digital Content Protection (HDCP), secret keys must be fused into the processor, which can not be read by the user. Currently the key installation process is not functioning as intended, resulting in non-functional HDCP operation. Without HDCP support, the supported HDMI-TX specification version is limited.

Workaround

Please contact NXP representative for additional details

ERR050395: ENET: Ethernet RX hang when receiving traffic through multiple queues

Description

Two or more applications are enabled to share the same Ethernet module by using different queues. At least 2 queues are configured to receive packets, with flushing enabled (RX_FLUSHx). When queues become full, packets are normally flushed, but under certain conditions of traffic, a lock-up of the Rx path can happen instead. When this occurs, the buffer descriptor for the last received packet contains an incorrect packet size (equal to the maximum buffer size). Packets cannot be received anymore, but the TX path remains unaffected. To recover the RX path, the ENET hardware block must be reset and re-configured.

Workaround

Unless the use case demands it, disable flushing to ensure the problem does not happen.

Or if reset is acceptable:

To recover the RX path, the ENET hardware block must be reset and re-configured

ERR050421: ADC: Missed triggers on ADC0 at low temperature

Description

There are eight ADC channels (ADC_IN[7..0]) split between two ADC (0 and 1) modules on the i.MX8 family of parts. ADC0 is connected to ADC_IN[3..0] and ADC1 is connected to ADC_IN[7..4]. This errata is only applicable to ADC0 (i.e. ADC_IN[3..0]).

An initial trigger takes the ADC Command Sequencer finite state machine (FSM) out of RESET state and sequences through several states during the execution of a conversion. After completion, the FSM transitions to the IDLE state. When a second trigger occurs, the state machine should transition back through a sequence of states while the conversion occurs. However, under certain conditions (-40C) a race condition with the trigger logic can result in resetting the conversion engine on the transition out of the IDLE state, putting the FSM back into its RESET state waiting for an "initial" trigger. The result is that the second trigger is not detected, however, the trigger detect logic is now in a position to receive the next trigger and execute a new conversion as expected successfully.

Workaround

Clear then set the CTRL[ADCEN] bit to cause a reset of the trigger detect and Command Sequencer FSM logic immediately before writing to the SWTRIG trigger register to initiate ADC conversions.

ERR050537: FlexSPI: Read timing sequence mismatches with several existing SPI NOR devices in dual, quad, and octal modes

Description

The FlexSPI controller expects every read command has at least one latency cycle between address phase and data phase to account for turnaround time on the IO bus. In multiple IO modes such as dual, quad, and octal modes, the FlexSPI controller inserts one additional clock cycle following the address (or command modifier) phase in order to prevent contention on bidirectional IO pins.

It will cause drive conflict if the SPI NOR device’s timing sequence does not contain dummy cycles after the command/address cycles. Such drive conflict might result in reading wrong data value. The problem usually happens when reading a SPI slave's register space.

Workaround

For FlexSPI memory device that supports multi IO Read command with zero latency cycle between address phase and data phase, use single line mode for read command, or use different data line to issue commands and read data.

The official NXP BSP release uses a signal line (1S-1S-1S) mode, but not multiple IO modes when access FlexSPI device registers.

ERR050660: ISI: HDMI-RX legacy YUV420 8-bit format incorrectly captured in ISI

Description

With Legacy YUV420 8 bit format showing up from MIPI-CSI subsystem to the Pixel Link input of the ISI within IMAGING subsystem, video data is received in U1Y1Y2U2Y3Y4…/V1Y1Y2V2Y3Y4… sequences in odd/even lines.

In ISI output Buffer, Chrominance data (U or V) is correct but for Luminance Y data, Y1 and Y2 bytes are swapped and so on i.e. Y2Y1Y4Y3...

Workaround

NXP software avoids this problem by swapping these luminance bytes in frame buffer. User should be aware of the issue when developing their own driver.

ERR050702: ISI: YUV444 video format not properly captured from HDMI-RX interface

Description

Colors out of the ISI are wrong when using an HDMI source with YUV444 format.

Instead of YUV, the output of the ISI is YVU. The U and V are swapped.

This inversion can be noticed in YUV or RGB color spaces.

Workaround

When configuring the HDMI RX in YUV444, the pixel link has to be programmed accordingly.

We can use this configuration to detect YUV444 format and enable the CSC of ISI to invert the YVU into YUY by using a matrix:

Table:

1 0 0
0 0 1
0 1 0

A similar matrix can be used when converting input YUV444 pixel stream into the RGB color space. Instead of inverting the U and V in the identity matrix, the U and V can be inverted in the CSC RGB->YUV conversion matrix.

ERR050802: DRAM: The LPDDR4 DRAM initialization may experience large training time variations or stall when Read Data Bus Inversion (DBI) bit deskew training is enabled

Description

Read DBI bit deskew training is an extension of the Read bit deskew training. When performing LPDDR4 Read bit deskew training (through the DDRPHY_PIR register), the following issues may be encountered:

• When Read DBI deskew training is enabled (DDRPHY_DTCR0.DTRDBITR = 2'bx1), there is a possibility to observe large training time variations or even a stall

• When Read DBI deskew training is disabled (DDRPHY_DTCR0.DTRDBITR = 2'bx0), incorrect values are programmed in DDRPHY_DXnBDLR5 (i.e. the DM Read BDL) after Read bit deskew training is completed

Workaround

A software workaround has been included in the SCU firmware (SCFW) since version 1.3.1. The software workaround has been included in the MX8QM DDR Register Programming Aid (RPA) since RPA version 19.

ERR051041: LPIT: CVAL cannot be read correctly during timer running

Description

The LPIT implements a functional clock domain for the counter and a bus clock domain for the register interface. The CVAL register increments on each clock cycle of the functional clock domain. Reading the register value happens on the bus clock domain. As these clock domains are not synchronous, there is a possibility that the register is read whilst the counter value is updating. This can lead to reading incorrect values as some bits of the counter may have settled whilst others are still transitioning to the new state.

Workaround

There should be no need to read the timer value since the timer is normally used to generate a periodic interrupt. However, if the timer value needs to be read, software can read the register more than once until the value matches the previous value. This ensures that the read operation did not coincide with the timer update and the value read is the actual timer value.

ERR051187: DisplayPort: Sampling Counters May Overflow at Controller Clock Frequency Above 170MHz

Description

Description: Display Port Aux uses Manchester encoding for transferring data. The controller is detects the rising and falling edges of the serial signal to extract bit period and synchronization. At the nominal operating frequency, 1 bit time is 1us, however the specification allows this to be between 0.8 and 1.2us (DP1.3 2.7.7).

During the AUX_SYNC_END and AUX_STOP phases, a particular encoding is used for synchronization which would not happen under any other circumstance, which results in the signal being high for 2-bit times followed by low for 2-bit times. As Manchester encoding is encoded as transitions within the bit times, this 2-bit level time is extended by half a cycle resulting in the maximum time between transitions being 2.5 bit times.

Periodically, an 11-bit counter is sampled and passed into a separate 9-bit register to indicate the last transition time. It is the comparison with this 9-bit register that various determinations are made on how to advance the state machines and sample data. Given that the maximum transition spacing of 2.5 bit times and longest bit time of 1.2us, this results in a maximum transition separation of 3us. When running with a core clock frequency of 200MHz, this means that the 11-bit counter exceeds the storage capacity of the 9-bit sampling register, essentially overflowing and causing the loss of the upper bits. End result is a wrong low value being stored.

This incorrect value is used for determining the next sampling point causing data corruption. The AUX_STOP pattern will not be detected properly and the transaction terminates with error. This then results in the failure to read the sink device EDID and to perform Link Training.

Workaround

Workaround: For full DP spec compliance and support of sink devices that may operate with a 1.2us bit time, the maximum core clock frequency must be limited to 170MHz.

The core clock frequency can be set from 100MHz to 170MHz without functionality impact.

ERR051238: Hot plug-in/out on SATA Port Multiplier Ports (PMP) may not be detected

Description

A hot plug on PMP is communicated through "Set Device Bits" (SDB) FIS. On receiving SDB FIS, the SATA host controller is expected to generate an interrupt.

In this SATA controller, the SNotification register latches the asynchronous notification event but interrupt is not generated. Hence software fails to detect the hot plug event. In case of a hot plug-out, the software tries to access the device which is no longer connected. This may result in multiple retries followed by multiple errors and port freeze.

Workaround

Poll SNotification register for asynchronous notification event.

ERR051393: Arm/Cortex-A core memory corruption

Description

A race condition in the Cortex-A CPU subsystem during initialization can cause memory setup values to be incorrectly applied to some Cortex-A cluster internal memories.

This may lead to memory corruption on some devices.

Workaround

An SCU firmware revision implementing SCF-838 is required to modify the Cortex-A CPU subsystem initialization sequence to avoid the race condition. Updates have been integrated to Linux BSP release starting from L5.10.35.

ERR051407: USB3: USB full speed mode may fail to work

Description

USB3 module supports USB3 .0 PHY and USB2.0 PHY. Very limited parts may fail to work on full speed mode (both host and device modes) for USB3 port due to higher threshold in full speed receiver of USB2.0 PHY. One example failure symptom is, the enumeration is failed when connecting full speed USB mouse to USB3 port, especially under high temperature.

Workaround

The recommended workaround is to configure threshold voltage value of single ended receiver by setting USB2.0 PHY register AFE_RX_REG5[2:0] to 3'b101 (Register Address is 0x5B198048). The workaround has been integrated to Linux BSP release starting from L5.10.9_1.0.0.

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.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

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.

Suitability for use in automotive applications — This NXP product has been qualified for use in automotive applications. 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.

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.

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.

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: 3/2023
Document identifier: IMX8_1N94W