

Mask Set Errata for Mask 0N95W

This report applies to mask 0N95W for these products:

- Contact your NXP representative for orderable part number information.

Table 1. Errata and Information Summary

Erratum ID	Erratum Title
ERR050061	ANALOG: DPLL loses lock under corner conditions
ERR050055	ANALOG: DRC temperature sensor causes coupling errors
ERR050068	AUDIO: Incorrect 24 MHz clock source at Audio Clock Mux input
ERR050059	DC: Display write back function is not functional
ERR050060	DC: PRG on the fly bypass switch issue
ERR050125	DRAM: Controller automatic derating logic may not work as intended when the LPDDR4 memory temperature is above 85C at initialization
ERR010947	DRAM: DQS/DQSN glitch suppression resistors must be enabled during read-leveling
ERR050054	DRAM: Extra boot time is required with DDR3L ECC
ERR010944	DRAM: In LPDDR4 mode, tMPCWR timing violation in incremental DQS2DQ Training
ERR010946	DRAM: In LPDDR4 mode: Auto refresh must be disabled during DQS2DQ training
ERR010945	DRAM: PUB does not program LPDDR4 DRAM DDRPHY_MR22 prior to running DRAM ZQ calibration
ERR010948	DRAM: Timing Violation from Read/Write to MRW in LPDDR4 mode
ERR050395	ENET: Ethernet RX hang when receiving traffic through multiple queues
ERR011543	FlexCAN: Nominal Phase SJW incorrectly applied at CRC Delimiter
ERR050057	GPU: OpenCV and Vulkan conformance issue
ERR050067	ISI: Adjacent processing pipelines within the ISI sub-system can experience loss of data
ERR050066	ISI: Data overflows occur when input streams exceed AXI transaction frequency
ERR050058	ISI: Incomplete frames when using virtual channels 1, 2, 3
ERR050145	ISI: Memory overwrite occurring outside of allocated buffer space corrupting system memory
ERR050135	JPEG DECODER: multi-frame jpeg bitstream may not be correctly decoded when there is a small size frame inside
ERR010527	LPUART: Setting and immediately clearing SBK bit can result in transmission of two break characters

Table continues on the next page...



Table 1. Errata and Information Summary (continued)

Erratum ID	Erratum Title
ERR011418	MIPI DSI: Incorrect CRC and payload corruption reported with DCS long write command
ERR010930	PCIE: EOM single point sample error/valid result is not correct
ERR011370	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
ERR050108	ROM: eMMC/SD boot failure due to ROM code timeout under certain conditions
ERR050052	ROM: NAND boot fails when image header points to an unprogrammed block
ERR050053	ROM: USB HID device cannot be re-enumerated successfully after an unplug/plug USB cable operation
ERR050056	SNVS: LDO startup is too slow
ERR050141	USB2: Endpoint conflict issue in device mode
ERR050147	USB3: Multiple DMA write transfer complete interrupts are generated before final write access handshake to the AXI bus
ERR050115	USB3: Port Configuration Response is not compliant with the USB compliance TD 7.17 test case
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

Table 2. Revision History

Revision	Changes
0, 05/2019	Initial revision
1, 04/2020	<p>The following erratum was removed.</p> <ul style="list-style-type: none"> • ERR011193 <p>The following errata were added.</p> <ul style="list-style-type: none"> • ERR010527 • ERR050395 • ERR011370 • ERR050145 <p>The following errata were revised.</p> <ul style="list-style-type: none"> • ERR010944 • ERR050066 • ERR050068 • ERR010947 • ERR010945 • ERR010946 • ERR050125
1.1, 05/2020	<p>The following erratum was removed.</p> <ul style="list-style-type: none"> • ERR011439 <p>The following erratum was added.</p> <ul style="list-style-type: none"> • ERR011418

ERR050061: ANALOG: DPLL loses lock under corner conditions

Description: At cold temperatures, the DPLL does not lock for some parts at the extreme edge of the supply voltage tolerance.

Workaround: The DPLL is sensitive to noise on its VDD supply, in general this is VDD_MAIN, particularly at low temperature (< -20 degrees Celsius).

Numerous SCU related workarounds have been applied to reduce this sensitivity, however the issue remains.

The recommended workaround is close monitoring of the decoupling capacitors used on the power supply design, to ensure close proximity and a low impedance path to the regulator, as defined in the Hardware Development Guide (HDG).

Validation must include board testing at low temperature to check for any power supply sensitivity concerns.

ERR050055: ANALOG: DRC temperature sensor causes coupling errors

Description: DRC temperature sensor turning on and off in its normal operation couples and corrupts other analog blocks through top level metal coupling of the SoC. This causes symptoms such as variation in the DDR clock.

Workaround: Disable DRC temperature sensor and only use the SCU temperature sensor. Little variation is seen between the readings of these 2 temperature sensors. Margin for Overheat shutdown is still sufficient to avoid damage.

ERR050068: AUDIO: Incorrect 24 MHz clock source at Audio Clock Mux input

Description: The DSC of the Audio subsystem / Audio DMA provides two 24 MHz clock sources.

One comes directly from the 24 MHz oscillator (“24_MHz_functional”) and the other goes through SW gating used during the Reset sequence (“24_MHz_rst_clk”).

This second 24 MHz source is currently connected to the Audio Clock Mux (ACM) input and can be selected as the functional clock for the Audio GPT. The DSC reset clock is enabled and disabled during reset sequences, which would impact the modules using that clock through the ACM. Note that a reset sequence can take place (e.g. for the HIFI) while audio blocks are operational. The “24_MHz_functional” must be used.

Only the GPT are impacted by this incorrect connection.

Workaround: Do NOT select the 24 MHz clock source from the ACM’s (audio clock mux) GPT0...5 external clock generator. Instead, select the 24 MHz clock source available from the GPT’s internal clock multiplexer, which comes from the correct source. That is to say, within the GPT Control Register (CR), CLKSRC[8:6] must use “101b - Crystal oscillator as the Reference Clock (ipg_clk_24M)”, and avoid use of the “011b - External Clock”.

ERR050059: DC: Display write back function is not functional

Description: The mechanism to enable an output stream to a display and also to be copied into memory using one of the internal camera streams is not functional due to the pixel link receiver address being incorrectly tied-off.

Workaround: The display controller signature unit (CRC) can be used to check for display changes.

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.

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 85°C (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 85°C. The actual memory temperature threshold values may vary depending on memory vendors.

If the LPDDR4 memory temperature remains below 85°C 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 85°C 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))

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_DX8SLbDQCTL register.

ERR050054: DRAM: Extra boot time is required with DDR3L ECC

Description: For DDR3L with ECC, due to design limitations and ECC scrub requirements, the SCU must initialize the entire memory for ECC before beginning to load the M4 image. This requires up to 300 msec extra.

Workaround: The future planned fix will initialize only a small region of RAM, reduce the impact on boot time.

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.

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.

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.

ERR010948: DRAM: Timing Violation from Read/Write to MRW in LPDDR4 mode

Description: When software sends a MRW command in parallel with a Read/Write transaction, the Read/Write command can be followed by the MRW command, which can result in the following timing violations:

1. RD to MRW
2. RDA to MRW
3. WRA/MWRA to MRW

This can occur only in LPDDR4 mode. When the memory clock frequency is lower than 450 MHz, then one of above 3 violations may occur, when the memory clock frequency is NOT lower than 450 MHz, then above item 1 or item 2 violation may occur.

The above timing constraints were introduced in the LPDDR4 specification JESD209-4A.

Workaround: MRW commands sends in parallel with a Read/Write transaction must follow a specific sequence.

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

Description: M4-FreeRTOS and A35-Linux 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.

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.

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.

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.

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.

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.

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.

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.

ERR010527: LPUART: Setting and immediately clearing SBK bit can result in transmission of two break characters

Description: When the LPUART transmitter is idle (LPUART_STAT[TC]=1), two break characters may be sent when using LPUART_CTRL[SBK] to send one break character. Even when LPUART_CTRL[SBK] is set to 1 and cleared (set to 0) immediately.

Workaround: To queue a single break character via the transmit FIFO, set LPUART_DATA[FRETSC]=1 with data bits LPUART_DATA[T9:T0]=0.

ERR011418: MIPI DSI: Incorrect CRC and payload corruption reported with DCS long write command

Description: When the DSI packet payload[23:8] is equal to 0x0, either an incorrect CRC is generated or an extra two bytes of incorrect CRC values at 0xFF are sent erroneously corrupting the payload. The issue only happens under low-power mode transmit.

Workaround: Avoid software with a packet payload[23:8] equal to 0x0 or else use high-speed mode to send commands.

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.

ERR011370: PCIE: EP, PM_PME: L1 Exit Does Not Occur when PME Service Timeout Mechanism Expires

Description: Impacted Configuration(s): Upstream Port configurations:

device_type =4'b0000, located at PCIEX1_CTRL0, address 0x5f140000, bit[27:24]

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 8 "APPS_PM_XMT_PME" of HSIO GPR register "PCIEX1_CTRL2", address 0x5f140008.

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 Oppm 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.

ERR050108: ROM: eMMC/SD boot failure due to ROM code timeout under certain conditions

Description: This issue is related to boot from eMMC or SD devices.

On power-up, a boot monitor timer is initialized. On a successful boot, SECO firmware is loaded and run from the boot device—that is, eMMC or SD, which, after loading and verifying SCU firmware (SCFW), disables the timer. When a successful boot requires more than 300 msec, a timeout occurs that is considered a boot failure, and therefore generates a warm reset, which results in a looping boot failure.

Typically, the initialization time for most eMMC/SD devices is about 100 to 200 msec, however, the eMMC/SD specification allows up to 1 second for this initialization time.

Any eMMC/SD device that exceeds the timeout to initialization will fail to boot. Sudden power loss or power cycle stress testing to eMMC/SD devices can cause data corruption, which can force the eMMC/SD device to run an internal data check on the next power up, which results in a longer initialization time, forcing a timeout and a looping boot failure.

Workaround: Generally reducing eMMC/SD initialization time under 300 msec is the most effective way to avoid this looping boot failure.

For the data corruption case, change the boot mode to serial download mode then load and run an image via USB. This image can initialize the eMMC/SD device and exit out of the internal data check state.

In future silicon releases, ROM will consider this case and wait 1 second to avoid the boot failure.

ERR050052: ROM: NAND boot fails when image header points to an unprogrammed block

Description: ROM first reads the Image Container Set 0 header, and then the Image Container Set 1 header, unless the secondary boot has been disabled by a fuse, in which case the Set 1 header will not be read. ROM will then select the header with the newest software version for primary boot and the oldest one for secondary boot.

For NAND boot, the block where the Image Container Set 0 is programmed is specified by fuses. If the specified block has been erased or has random data on it, the NAND read API returns a failure. In this case, ROM will attempt to read from the block with the Image Container Set 1 header programming, which is also specified by fuses.

However, there is a bug in the ROM code. It will not read data from the block with the Image Container Set 1 header, and instead continues to read data from the block with the Image Container Set 0 header. However, because this block has not been programmed boot failure occurs, even if there is a valid Image Container Set 1 header available.

Workaround: Avoid upgrading Image Container Set 0. Only upgrade Image Container Set 1 if the boot image needs upgraded. Per detailed description, ROM always tries to read both image container sets and selects the one with the newest software version to boot.

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.

ERR050056: SNVS: LDO startup is too slow

Description: SNVS startup on LDO is too slow at low temperature with VDD_SNVS less than 2.8V. This issue may cause some parts to take longer to startup due to boot retry in SCU. The boot time to M4 operation may increase from <50ms to around 80ms because of the re-boot attempts.

Workaround: The workaround is to ensure that voltage is maintained at, or above, 2.8 volts at low temperature.

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.

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.

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.

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.

How to Reach Us:**Home Page:**nxp.com**Web Support:**nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer's applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, AltiVec, C-5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C-Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, and UMEMS are trademarks of NXP B.V. All other product or service names are the property of their respective owners. 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, μ Vision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) 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. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© 2020 NXP B.V.

