RT700_0P77N

Mask Set Errata

Rev. 1.1 — 15 October 2025 Errata

1 Mask Set Errata for Mask 0P77N

1.1 Revision History

This report applies to mask 0P77N for these products:

- MIMXRT735SGAWAR
- MIMXRT758SGAWAR
- MIMXRT798SGAWAR
- MIMXRT735SGFOA
- MIMXRT758SGFOA
- MIMXRT798SGFOA

Table 1. Revision History

Revision	Release Date	Significant Changes
1.1	10/2025	The following errata were removed. • ERR052007 • ERR052408 • ERR051605 • ERR052005 • ERR052695 • ERR052006 The following errata were added. • ERR052942 • ERR052943 • ERR052985 • ERR051629 The following errata were revised. • ERR051734 • ERR052221 • ERR050135 • ERR052527
1.0	6/2025	Initial Revision

1.2 Errata and Information Summary

Table 2. Errata and Information Summary

Erratum ID	Erratum Title
ERR052221	ADC: After resynchronization occurs, the restart sequence runs twice
ERR052942	Boot ROM: NXP assets cannot be included in DICE calculation for Copy-to-Ram Boot Flow
ERR052943	Boot ROM: Revoked key usage interferes with authentication flow of XIP boot or debug authentication

Table 2. Errata and Information Summary*...continued*

Erratum ID	Erratum Title
ERR052222	CMX_PERFMON: Cache event counting not supported for User only or Privileged only count modes
ERR051051	Core: A partially completed VLLDM might leave Secure floating-point data unprotected
ERR050505	Core: Access permission faults are prioritized over unaligned Device memory faults
ERR050501	Core: DFSR.EXTERNAL is not set correctly when waking up from sleep
ERR051734	Core: DWT comparator match on cycle count is not reported to the ETM if there is no instruction executing on the processor
ERR050502	Core: Execution priority might be wrong for one cycle after AIRCR is changed
ERR050500	Core: Group priority of a Non-secure interrupt might be incorrect when AIRCR.PRIS is set
ERR050503	Core: Non-secure HardFault exception might preempt when disabled by AIRCR.BFHFNMINS
ERR050504	Core: Sorting of pending interrupts might be wrong when high latency IRQs are pending
ERR050875	CoreSight: AHB-AP can issue transactions where HADDR[1:0] is not aligned to HSIZE on the AHB
ERR050887	Coresight: CSTPIU fails to output sync after the pattern generator is disabled in Normal mode
ERR052985	HIFI4: The OSTIMER debug function is not operational on the HiFi4 DSP when running in debug mode.
ERR052122	I3C : Data size limitation in Message mode DDR transfer
ERR052344	I3C: Controller Clock stalling feature not available in I3C Controller
ERR052343	I3C: Target Early Termination Feature not available with DMA controller
ERR050135	JPEG DECODER: Multi-frame jpeg bitstream may not be correctly decoded when there is a small size frame inside
ERR051588	LPSPI:Reset transmit FIFO after FIFO underrun by LPSPI Slave.
ERR051629	LPUART:Transmit Complete bit (STAT[TC]) is not set.
ERR011439	MIPI DSI: Checksum is incorrect for DCS command long packet writes with zero-length data payload
ERR052402	PUF: PUF quality test may fail when VDD2 voltage is higher than 1.0V
ERR051405	SAI: Synchronous mode with BYP=1 not supported
ERR052198	uSDHC: eMMC CQE may timeout due to a HW logic issue
ERR052527	XSPI: Limitation in High-Priority scheme of AHB read access buffer

2 Known Errata

ERR052221: ADC: After resynchronization occurs, the restart sequence runs twice

Description

The RT700 ADC supports two triggers--trigger 0 and trigger 1. The ADC allows any software or hardware trigger to act as a resynchronization trigger. When trigger 1 is configured to resynchronize trigger 0 and trigger resume is enabled (CFG[TRES] = 1), asserting trigger 1 while trigger 0 is running will cause trigger 0 to stop and clear the FIFO, then the sequence restarts, but the sequence runs twice instead of one time as expected.

Workaround

Use trigger 0 to resynchronize trigger 1. Avoid using trigger 0 as a target trigger.

ERR052942: Boot ROM: NXP assets cannot be included in DICE calculation for Copy-to-Ram Boot Flow

Description

Description:

Device Identity Composition Engine (DICE), uses Immutable RoT during boot time to create a unique Device Identity, which considers Unique Device Secret (UDS), hardware state of the device and its firmware. Users can include NXP assets in addition to OEM boot configuration from fuses to be included as part of the calculation of the CDI.

Issue:

The copy-to-ram boot flow will fail to boot in the In-Field lifecycle, if NXP assets are included for the CDI calculation when DICE is enabled.

Workaround

Do not include NXP assets for DICE calculation (keep DICE_INC_NXP fuse = 0) or use Execute-in-Place boot flow instead.

ERR052943: Boot ROM: Revoked key usage interferes with authentication flow of XIP boot or debug authentication

Description

The Root of Trust Keys (RoTK) usage can be assigned to image authentication, SB3.1 authentication, debug authentication or a combination of the three. Each usage can be independently revoked, allowing fine-grained control over which functions a key is permitted to perform.

ROTKx_EX_IMG_AUTH – Excludes a key from being used during image authentication.

ROTKx_EX_SB3_AUTH– Excludes a key from being used during SB3.1 authentication.

ROTKx_EX_DBG_AUTH– Excludes a key from being used during debug authentication.

With that being said, the following issues occur:

When revoking the usage for SB3.1 authentication for a given key, the image fails to boot when signed by the same key despite it being valid for use in image authentication.

When revoking the usage for SB3.1 and debug authentication for a given key , the image fails to boot when signed by the same key despite it being valid for use in image authentication.

When revoking the usage for Image, SB3.1 and debug authentication for a given key, the debug authentication fails when signed by a different key despite it being valid for use in debug authentication.

Workaround

If a Root of Trust Key (RoTK) is used across multiple functions—such as image authentication, SB3.1 authentication, and debug authentication—revoking it due to loss of trust should ideally apply to all its usages. However, doing so may unintentionally break debug authentication, and retaining the key solely for debug access could weaken the overall security.

Workaround Options:

• Option A: Assign a dedicated RoTK to each function (e.g., ROT0 for image auth, ROT1 for SB3.1, ROT2 for debug). This allows revocation without affecting unrelated functions. Image signing keys can also be layered on top of RoTKs to enable revocation without immediately retiring the root key if not necessary.

• Option B: Use multiple RoTKs per function (e.g., two for image auth, two for SB3.1/debug). So if needed to revoke the keys for SB3.1/debug it will not affect image authentication as well as not breaking debug authentication on the other ROT keys.

• Option C: In In-Field Locked Lifecycle (LC), debug authentication is disabled, making debug auth failure irrelevant in that state. (This only would be a non-issue for the debug authentication part of it)

ERR052222: CMX_PERFMON: Cache event counting not supported for User only or Privileged only count modes

Description

CMX_PERFMON supports three modes: cache event counting in privileged mode only, cache event counting in user mode only, and cache event counting in both user and privileged modes. The PMCR[CMODE] bit field controls the operating mode. The CMX_PERFMON is unable to distinguish between user and privileged mod, hence, CMODE=10b and CMODE=11 are not supported for Cache event counting.

Workaround

There is no workaround.

ERR051051: Core: A partially completed VLLDM might leave Secure floating-point data unprotected

Description

Arm errata 2219175

Affects: Cortex-M33

Fault Type: Programmer Category B

Fault Status: Present in r0p0, r0p1, r0p2, r0p3, r0p4, r1p0. Open.

The VLLDM instruction allows Secure software to restore a floating-point context from memory. Due to this erratum, if this instruction is interrupted or it faults before it completes, then Secure data might be left unprotected in the floating point register file, including the FPSCR.

Configurations affected:

This erratum affects all configurations of the Cortex-M33 processor configured with the Armv8-M Security Extension and the Floating-point Extension.

Conditions:

This erratum occurs when all the following conditions are met:

• There is no active floating-point context, (CONTROL.FPCA==0)

• Secure lazy floating-point state preservation is not active, (FPCCR_S.LSPACT==0)

• The floating-point registers are treated as Secure (FPCCR_S.TS==1)

• Secure floating-point state needs to be restored, (CONTROL_S.SFPA == 1)

• Non-secure state is permitted to access to the floating-point registers, (NSACR.CP10 == 1)

• A VLLDM instruction has loaded at least one register from memory and does not complete due to an interrupt or fault

Implications:

If the floating-point registers contain Secure data, a VLSTM instruction is usually executed before calling a Non-secure function to protect the Secure data. This might cause the data to be transferred to memory (either directly by the VLSTM or indirectly by the triggering of a subsequent lazy state preservation operation). If the data has been transferred to memory, it is restored using VLLDM on return to Secure state. If the VLLDM is interrupted or it faults before it completes and enters a Non-secure handler, the partial register state which has been loaded will be accessible to Non-secure state.

Workaround

To avoid this erratum, software can ensure a floating-point context is active before executing the VLLDM instruction by performing the following sequence:

• Read CONTROL_S.SFPA

• If CONTROL_S.SFPA==1 then execute an instruction which has no functional effect apart from causing context creation (such as VMOV S0, S0)

ERR050505: Core: Access permission faults are prioritized over unaligned Device memory faults

Description

Cortex-M33 1080541-C :

A load or store which causes an unaligned access to Device memory will result in an UNALIGNED UsageFault exception. However, if the region is not accessible because of the MPU access permissions (as specified in MPU_RBAR.AP), then the resulting MemManage fault will be prioritized over the UsageFault.

Workaround

There is no workaround.

However, it is expected that no existing software is relying on this behavior since it was permitted in Armv7-M.

ERR050501: Core: DFSR.EXTERNAL is not set correctly when waking up from sleep

Description

Cortex-M33 1367266-C:

An external debug event which causes the processor to enter Debug state or the debug monitor should set DFSR.EXTERNAL. It has been found that this field is not set if the event occurs while the processor is asleep.

Workaround

There is no workaround.

ERR051734: Core: DWT comparator match on cycle count is not reported to the ETM if there is no instruction executing on the processor

Description

Cortex-M33 2435965-C

The Cortex-M33 Data Watchpoint and Trace (DWT) unit supports a Cycle count match event which can

be used to trigger the Embedded Trace Macrocell (ETM) to generate a trace packet from the processor.

Due to this erratum the event signal is only propagated when an instruction is executing in the pipeline

and so no event will be transferred to the ETM if the processor is idle.

Workaround

There is no workaround for this erratum, however, non-debug operation of the core is not affected.

ERR050502: Core: Execution priority might be wrong for one cycle after AIRCR is changed

Description

Cortex-M33 1435973-C:

AIRCR is used in the NVIC active tree to calculate the execution priority, which in turn is used to determine fault escalation, exception preemption, and other NVIC-related behaviors. When the active tree is pipelined and there are high latency IRQs active, there might be a glitch in the active tree output for one cycle after AIRCR is changed. The glitch results in NVIC producing wrong execution priority that is neither based on the old AIRCR value nor the new one.

Workaround

There is no workaround for this erratum.

ERR050500: Core: Group priority of a Non-secure interrupt might be incorrect when AIRCR.PRIS is set

Description

Cortex-M33 1113997-C:

When the processor is configured with Security extension and AIRCR.PRIS is 1, the Armv8-M architecture requires that the priorities of Non-secure interrupts are modified to ensure that Secure interrupts are prioritized over Non-secure interrupts. The Armv8-M architecture requires that lower priority numbers take precedence over higher priority numbers. Because of this erratum, a Non-secure interrupt with higher priority number might be handled in the wrong order compared to another Non-secure or Secure interrupt.

Workaround

There is no workaround for this erratum.

ERR050503: Core: Non-secure HardFault exception might preempt when disabled by AIRCR.BFHFNMINS

Description

Cortex-M33 1453380-C:

When the processor implements the Security Extension and AIRCR.BFHFNMINS is 1, the Non-secure banked version of SHCSR.HARDFAULTPENDED can be set to 1. This Non-secure pended HardFault might not preempt per architecture because it does not have enough priority (that is, the processor is in HardFault handler mode). If AIRCR.BFHFNMINS is subsequently changed to 0 with the Non-secure HardFault still pending, then the architecture requires that the Nonsecure HardFault should never preempt regardless of execution priority. Because of this erratum, the pended Non-secure HardFault exception preempts when AIRCR.BFHFNMINS is 0 and current execution priority is larger than -1 (Non-secure HardFault having higher priority).

Workaround

There is no workaround for this erratum.

ERR050504: Core: Sorting of pending interrupts might be wrong when high latency IRQs are pending

Description

Cortex-M33 1540599-C:

The NVIC contains a pending tree which sorts all pending and enabled interrupts based on priorities. If DHCSR.C_DEBUGEN and DHCSR.C_MASKINTS are 1, DHCSR.S_SDE is 0 and halting debug is allowed, then Nonsecure PendSV, Non-secure SysTick, and Non-secure IRQs should be masked off and they should not affect the sorting of pending and enabled secure interrupts. If multiple high latency IRQs are pending and enabled with different security targets and priorities, then Non-secure IRQs which should be masked off might cause the pending tree output to be a pending Secure nterrupt without highest priority. This is because of incorrect masking before doing priority comparisons in the tree.

Workaround

There is no workaround for this erratum.

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

Description

ARM errata 1624041

This erratum affects the following components:

• AHB Access Port.

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

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

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

Conditions:

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

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

Implications:

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

Workaround

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

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

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

Description

Arm errata 341182

Affects: CoreSight SoC-400 - Perpetual

Fault Type: Programmer Category C

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

This erratum affects the following components:

• Trace Port Interface Unit.

• CSTPIU and cxtpiu

- Component Revisions: r0p4, r0p5, r1p0

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

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

Conditions

The following conditions must all occur:

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

Implications

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

Workaround

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

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

2) Poll FFSR.FtStopped until 1 is read

3) Set FFCR.EnFTC==1

ERR052985: HIFI4: The OSTIMER debug function is not operational on the HiFi4 DSP when running in debug mode.

Description

OSTIMER provides dedicated interfaces to CPU0, HiFi4, CPU1, and HiFi1 cores. When these cores enter debug mode, OSTIMER can selectively enable or disable the corresponding counters. However, when HiFi4 enters debug mode, the OSTIMER_HIFI4 counter fails to disable as expected.

Workaround

OSTIMER on HiFi4 does not support halting its counter while the CPU0 timer continues running. To ensure synchronized debug behavior across all cores (CPU and HiFi), the Cross Trigger Interface (CTI) can be used to halt all cores simultaneously when any one of them enters debug mode

ERR052122: I3C : Data size limitation in Message mode DDR transfer

Description

The message length in DDR message (DMA) mode is defined in MWMSG_DDR_CONTROL2 [9:0].LEN field. Bits [9:8] of this field are ignored. Only bits [7:0] of this field are taken into account to define the transfer length in number of Half words. This limits the maximum amount of data transferred depending on the operation type. For Read operations the maximum amount of data is (255 - 2) = 253 half-words (506 bytes). For write operations it is (255 -1) = 254 halfwords (508 bytes)

Workaround

The application software needs to limit the data size for Write and Read operation in message (DMA) mode of DDR transfer to a maximum of 506 bytes for reads, and 508 bytes for writes.

ERR052344: I3C: Controller Clock stalling feature not available in I3C Controller

Description

The clock stalling feature as per section “5.1.2.5 : Controller Clock Stalling” in MIPI I3C Basic Specification(Improved Inter Integrated Circuit) Specification Version 1.1.1 is not implemented for I3C controller.

For target assuming the stalling of clock between C8 and C9 in I3C SDR write followed with repeated start such as read with register address use case, need to set the appropriate SCL frequency based on the Slave device performance.

Workaround

For target assuming the stalling of clock between C8 and C9 in I3C SDR write followed with repeated start such as read with register address use case, need to set the appropriate SCL frequency based on the Slave device performance.

ERR052343: I3C: Target Early Termination Feature not available with DMA controller

Description

The indication of Early read termination by I3C Target is reflected by register status and interrupt in I3C Controller. Due to custom interface of DMA controller

this pre-mature termination information is not propagated to it consequently the partial data is not transfer into memory in case of Early termination by Target.

So, Early termination cannot be used with this DMA controller for I3C.

Workaround

1) If DMA is used, do not use Target Early termination feature. Instead use smaller data size which is guaranteed to be arranged by Target always and early termination will never occur.

2) Do not use DMA , use Host to do all transfers and handle Early termination by ISR or by polling of status method.

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.

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

Description

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

Workaround

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

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

Description

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

Workaround

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

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.

ERR052402: PUF: PUF quality test may fail when VDD2 voltage is higher than 1.0V

Description

The Physical Unclonable Function (PUF) is used to create device unique Root of Trust (RoT) key. Its features consist of the following:

- Key strength of 256-bits
- Generation, storage, and reconstruction of keys
- Key sizes from 64-bits to 4096-bits
- Key output via dedicated hardware interface or through register interface
- PUF quality test with a score done during Enroll, Start and Reconstruct operations.

However, as the device has an operational range of up to 1.155V on VDD2, there is an issue with the PUF quality test where it may fail when VDD2 is higher than 1.0V.

Workaround

Restrict PUF Start/Reconstruct Operation up to 1.0V.

ERR051405: SAI: Synchronous mode with BYP=1 not supported

Description

If the transmitter or receiver is configured for synchronous mode ((SYNC=01) or Bit Clock Swap (BCS=1), and the receiver or transmitter that is the source of the BCLK is configured with BYP=1 and BCD=1, then the transmitter or receiver must set BCI=1.

Workaround

Set BCI=1 when configuring for synchronous mode (SYNC=01) or Bit Clock Swap (BCS=1) and the source of the BCLK has configured BCD=1 and BYP=1.

ERR052198: uSDHC: eMMC CQE may timeout due to a HW logic issue

Description

To facilitate command queuing in eMMC, uSDHC supports CQE (Command Queue) so that the host can queue data transfer tasks. In some cases where the CQDPT (Device Pending Task) bit is being set and cleared at the same time, the CQDPT bit is not cleared even after the task has completed execution, resulting in a CQE timeout.

Workaround

There are two possible workarounds:

1.Disable CQE feature using CQCFG register

2. Before sending a DCMD (Direct Command) request, SW must manually clear the CQDPT[bit n] by setting CQTCLR[bit n] if DPT[bit n] != CQTDBR[bit n], in order to clear the pending task.

ERR052527: XSPI: Limitation in High-Priority scheme of AHB read access buffer

Description

The system includes four AHB read access buffers: Buffer 0 through Buffer 3. Of these, Buffer 0 and Buffer 1 are configurable as high-priority buffers by setting the HP_EN bit in their respective control registers (BUF0CR[HP_EN] and BUF1CR[HP_EN]). Due to this hardware erratum, Buffer 0 and Buffer 1 cannot be independently configured as high-priority buffers. That is, enabling high priority for only one of these buffers (either Buffer 0 or Buffer 1) is not supported. Both buffers must be configured as high priority simultaneously. Attempting to set only one of them to high priority may result in the priority setting being ignored

Workaround

To enable high-priority access for these buffers Buffer 0 and Buffer 1, ensure that both BUF0CR[HP_EN] and BUF1CR[HP_EN] are set to 1 at the same time.

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.

Limiting values — Stress above one or more limiting values (as defined in the Absolute Maximum Ratings System of IEC 60134) will cause permanent damage to the device. Limiting values are stress ratings only and (proper) operation of the device at these or any other conditions above those given in the Recommended operating conditions section (if present) or the Characteristics sections of this document is not warranted. Constant or repeated exposure to limiting values will permanently and irreversibly affect the quality and reliability of the device.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at https://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.

No offer to sell or license — Nothing in this document may be interpreted or construed as an offer to sell products that is open for acceptance or the grant, conveyance or implication of any license under any copyrights, patents or other industrial or intellectual property rights.

Quick reference data — The Quick reference data is an extract of the product data given in the Limiting values and Characteristics sections of this document, and as such is not complete, exhaustive or legally binding.

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.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

HTML publications — An HTML version, if available, of this document is provided as a courtesy. Definitive information is contained in the applicable document in PDF format. If there is a discrepancy between the HTML document and the PDF document, the PDF document has priority.

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.

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

Trademarks

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

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

Contents

© October 2025 NXP B.V. All rights reserved.