# MCXN23x_0P21K

## Mask Set Errata

Rev. 2.0 — 14 April 2025 Errata

# 1 Mask Set Errata for Mask 0P21K

## 1.1 Revision History

This report applies to mask 0P21K for these products:

- MCXN236VDFT
- MCXN235VDFT
- MCXN236VKLT
- MCXN235VKLT
- MCXN236VPBT
- MCXN235VPBT

**Table 1. Revision History**

| Revision | Release Date | Significant Changes |
|---|---|---|
| 2.0 | 4/2025 | The following errata were added.<br>• ERR052651<br>• ERR052650<br>• ERR052558<br>• ERR052344<br>• ERR052728<br>• ERR052406<br>The following errata were revised.<br>• ERR051617<br>• ERR052241<br>• ERR052122<br>• ERR051874 |
| 1.0 | 5/2024 | Initial Revision |

## 1.2 Errata and Information Summary

**Table 2. Errata and Information Summary**

| Erratum ID | Erratum Title |
|---|---|
| ERR050875 | CoreSight: AHB-AP can issue transactions where HADDR[1:0] is not aligned to HSIZE on the AHB |
| ERR051421 | SAI: Synchronous mode with bypass is not supported |
| ERR051588 | LPSPI:Reset transmit FIFO after FIFO underrun by LPSPI Slave. |
| ERR051617 | [I3C] In I2C controller mode generates unintended repeated START before sending STOP |
| ERR051629 | LPUART:Transmit Complete bit (STAT[TC]) is not set. |
| ERR051704 | DCDC: Failure changing to Low drive-strength mode |
| ERR051713 | ADC: Extra conversion can occur when moving to low power mode |

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

| Erratum ID | Erratum Title |
|---|---|
| ERR051874 | I3C: I2C clock stretching mode is not supported |
| ERR051989 | PWM: output may be abnormal when the value of phase delay register is reduced from a non-zero value to 0. |
| ERR051998 | ROM: Command "get-property 12" not supported when using USB interface |
| ERR052108 | ROM: LDO_SYS VDD level not returned to Normal voltage range after programming fuses |
| ERR052122 | I3C : Data size limitation in Message mode DDR transfer |
| ERR052147 | USB: ISO schedule issue in FS Host mode |
| ERR052241 | CDOG: Restart command cannot let timer count down |
| ERR052344 | I3C: Controller Clock stalling feature not available in I3C Controller |
| ERR052406 | ROM: chip boots in ISP mode if Boot_IFR0_Dis or Boot_flash_Dis field is burned in eFuse |
| ERR052558 | FlexCAN: Message buffer (MB) overrun status is cleared when reading Enhanced RX FIFO (ERF) |
| ERR052650 | FlexCAN: Frames dropped from Enhanced RX FIFO when message buffer is locked for more than one CAN frame time |
| ERR052651 | FlexCAN: CAN frames dropped when using Enhanced RX FIFO |
| ERR052728 | ELS: Some devices can have intermittent DTRNG errors |

# 2 Known Errata

## 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.

## ERR051421: SAI: Synchronous mode with bypass is not supported

### Description

The SAI does not receive or transmit when:

Scenario 1. The transmitter is configured for synchronous mode (TCR2[SYNC] = 0b1), in the Transmit Configuration 2 register, and the receiver is in bypass (RCR2[BYP]=0b1), in the Receiver Configuration 2 register, then there will not be a bit clock as it is the source of the BCLK.

Scenario 2. The receiver is configured for synchronous mode (RCR2[SYNC] = 0b1) in the Receiver Configuration 2 register and the transmitter is in bypass (TCR2[BYP]=0b1), in the Transmit Configuration 2 register, then there will not be a bit clock as it is the source of the BCLK.

### Workaround

If scenario 1, then set the TCR2[BCI] = 0b1, in the Transmit Configuration 2 register.

If scenario 2, then set the RCR2[BCI] = 0b1, in the Receiver Configuration 2 register.

## 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.

## ERR051617: [I3C] In I2C controller mode generates unintended repeated START before sending STOP

### Description

When I3C module is used as an I2C controller, repeated START may be randomly generated before STOP. That is, when MCTRL.REQUEST = STOP and MCTRL.TYPE = I2C, a STOP signal may or may not be preceded by a repeated START signal.

### Workaround

In I2C compatibility mode, set to MCONFIG[SKEW] = 1

## 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).

## ERR051704: DCDC: Failure changing to Low drive-strength mode

### Description

The DCDC output may fail when transitioning from Normal to Low drive-strength, resulting in the DCDC output voltage dropping to the point it is not able to adequately power the VDD_CORE supply, or causes temporary brown-out conditions. This failure may occur when both of these conditions occur:

1) The transition from Normal drive strength (DCDC_VDD_DS = 10b) to Low drive-strength (DCDC_VDD_DS = 01b) occurs when the DCDC is actively switching the output.

2) The voltage level set in the bitfield SPC->LP_CFG[DCDC_VDD_LVL] is greater than or equal to the current output voltage of the DCDC.

Because this failure requires a specific timing to manifest, it may fail very infrequently in an application. The greater the load current of the DCDC, the more likely the failure will occur because the DCDC will spend more

time in the active switching period. A higher rate of transitioning to Low drive-strength will also see a higher failure rate.

There are two scenarios when the DCDC drive-strength can transition from Normal to Low drive-strength, and this failure may occur:

1) While the MCU is in Active power mode, and the application changes the drive-strength setting by writing 01b to the bitfield SPC->ACTIVE_CFG[DCDC_VDD_DS]. Writing this bitfield will start the transition to Low drive-strength.

2) When the MCU enters a low-power mode (Deep Sleep, Power Down, or Deep Power Down), and Active mode uses Normal drive-strength with ACTIVE_CFG[DCDC_VDD_DS] = 10b, while the low-power mode uses Low drive-strength with LP_CFG[DCDC_VDD_DS] = 01b.

### Workaround

This issue will always be avoided when the voltage level at the low-power low drive-strength is lower than the current output voltage of the DCDC. Before transitioning to Low drive-strength, ensure the voltage level in LP_CFG[DCDC_VDD_LVL] is lower than the voltage level in Normal drive-strength configured by ACTIVE_CFG[DCDC_VDD_LVL]. As part of this workaround, the voltage level used in Low drive-strength configured by LP_CFG[DCDC_VDD_LVL] must not be set to the maximum value 11b for 1.2 V at any time in an application.

If the desired voltage level in LP_CFG[DCDC_VDD_LVL] is the same as the level currently set in ACTIVE_CFG[DCDC_VDD_LVL], a workaround is to temporarily increase the voltage level in ACTIVE_CFG[DCDC_VDD_LVL], and then transition to Low drive-strength with the lower level in LP_CFG[DCDC_VDD_LVL]. Here is the sequence for this workaround:

1) Ensure LP_CFG is configured for Low drive-strength and the desired voltage level in Low drive-strength mode

2) Wait for the SPC bit SC[BUSY] to be clear.

3) Write ACTIVE_CFG[DCDC_VDD_LVL] with the value for the voltage level one step higher than the desired level in LP_CFG[DCDC_VDD_LVL].

4) Start the transition to Low drive-strength

If the workaround sequence above is used when the MCU enters a low-power mode, then when the MCU wakes the DCDC will return to Normal drive-strength with the output voltage level configured in SPC->ACTIVE_CFG[DCDC_VDD_LVL]. If a lower voltage level is preferred, the application can lower DCDC voltage by waiting for the bit SC[BUSY] to be clear and writing the new voltage level to SPC->ACTIVE_CFG[DCDC_VDD_LVL].

## ERR051713: ADC: Extra conversion can occur when moving to low power mode

### Description

When high-priority trigger exceptions are enabled (ADCx->CFG[HPT_EXDI] = 0x1) and the ADC command uses the "Repeat until true" compare option (ADCx->CMDHa[CMPEN] = 0x3), an extra conversion occurs at the end of the conversion cycle if a higher priority trigger is asserted when a low power request is also made. This can result in erroneous extra data in the result FIFO and/or prevent the ADC module from being disabled in the low power mode (even if the Doze enable bit is set - ADCx->CTRL[DOZEN] = 0x1).

### Workaround

The ADC workaround is to do ONE of the following:

- Disable the ADC before entering low power mode (ADCx->CTRL[ADCEN] = 0)

- Disable high priority exceptions (ADCx->CFG[HPT_EXDI] = 0x1)

- If high priority exceptions are enabled (ADCx->CFG[HPT_EXDI] = 0x1) and "Repeat until true" compare option is used (ADCx->CMDHa[CMPEN] = 0x3), then the trigger command select (ADCx->TCTRLa[TCMD]) pointing to that command must be the highest priority (ADCx->TCTRLa[TPRI] = 0).

- User software waits for final conversion to be completed before entering low power mode.

## ERR051874: I3C: I2C clock stretching mode is not supported

### Description

When working as I2C master mode the I3C controller does not support Clock stretching feature.

### Workaround

The application must not use I2C clock stretching feature. Failure to do so will create clock contention on the I2C bus.

I2C Targets on the bus must keep Clock stretching feature disabled.

## ERR051989: PWM: output may be abnormal when the value of phase delay register is reduced from a non-zero value to 0.

### Description

When the value of the SMxPHASEDLY register is reduced from a non-zero value to 0 and the SMxCTRL2[RELOAD_SEL]=1, the submodule x may output an unexpected wide PWM pulse (x=1,2,3).

### Workaround

The minimum value of the SMxPHASEDLY register should be set as 1 in this process. To realize no phase delay between the submodule 0 and submodule x in this process, set the SMxPHASEDLY=1, SMxINIT=SM0INIT–1, SMxVALy=SM0VALy–1 (x=1,2,3, y=0,1,2,3,4,5).

## ERR051998: ROM: Command "get-property 12" not supported when using USB interface

### Description

When using the USB interface to access the device in ISP mode, command "get-property 12" returns a fail result. This applies to both Full-Speed and High-Speed USB interfaces.

### Workaround

There is no workaround for this issue. Customers should not use the "get-property 12" command when using USB as the ISP mode interface.

## ERR052108: ROM: LDO_SYS VDD level not returned to Normal voltage range after programming fuses

### Description

When programming any fuse using the ROM API, the voltage level of the LDO_SYS is not returned to Normal Voltage level (1.8V). That is, SPC0->ACTIVE_CFG[SYSLDO_VDD_LVL] = 1.

### Workaround

User software should return the LDO_SYS voltage level to normal level immediately after programming fuses (SPC0->ACTIVE_CFG &= ~SPC_ACTIVE_CFG_SYSLDO_VDD_LVL_MASK;).

Note that the SDK functions which program fuses already account for this errata.

## 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.

## ERR052147: USB: ISO schedule issue in FS Host mode

### Description

When working in FS Host mode, for ISO communication, if the first ISO package is equal or greater than about 238 bytes, then the second ISO transaction could not be scheduled in the same 1 ms frame.

For example, a typical application case affected is audio bi-directional communication with 48 KHz sample rate, 24 bit per sample and 2 channels for both direction. In this case, we need one ISO IN 288 bytes (>238 bytes) + one ISO OUT 288 bytes (>238 bytes) in one 1 ms frame. But actually, only one ISO OUT 288 bytes or only one ISO IN 288 bytes is available in one 1 ms frame due to the IP limitation.

### Workaround

No workaround.

## ERR052241: CDOG: Restart command cannot let timer count down

### Description

Due to this errata, once RESTART command is written to the RESTART register, the Instruction Timer is continually reloaded with the value in the RELOAD register and the counter will not count down (until subsequent accesses of CDOG registers).

### Workaround

Replace the RESTART register write instructions in all locations with a write to the STOP register immediately followed by a write to the RELOAD and START registers. Both the STOP and START registers should be written with the same value in this situation.

## 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.

## ERR052406: ROM: chip boots in ISP mode if Boot_IFR0_Dis or Boot_flash_Dis field is burned in eFuse

### Description

If either Boot_IFR0_Dis or Boot_flash_Dis field is burned in the eFuse, ROM boots into ISP mode instead of booting from the enabled boot source.

### Workaround

Use CMPA.BOOT_SRC to configure the boot source. Do not blow Boot_IFR0_Dis and Boot_flash_Dis fields in eFuse.

## ERR052558: FlexCAN: Message buffer (MB) overrun status is cleared when reading Enhanced RX FIFO (ERF)

### Description

Message buffer status becomes "full" when a frame arrives, and status becomes "overrun" when a second message arrives in the same message buffer, if first message has still not been read. If frame reception is happening in ERF and the frame is being read from ERF, these reads could incorrectly clear the MB overrun status. As a result, the overrun event can be missed by the application.

### Workaround

Use one of the following workarounds:

Workaround #1: Don’t use Enhanced RX FIFO (ERF).

Workaround #2: Don’t use any of the message buffers from MB0 to MB7 for reception if ERF is enabled. MB0 to MB7 can be used for transmission.

## ERR052650: FlexCAN: Frames dropped from Enhanced RX FIFO when message buffer is locked for more than one CAN frame time

### Description

If the message buffer and Enhanced RX FIFO are both configured for reception and FlexCAN message buffer is locked for more than one CAN frame time, FlexCAN will then start dropping received frames from the RX FIFO. This applies only to select message buffers (5 and 15)

### Workaround

There are two workarounds for this issue.

1) The message buffer must be read within one CAN frame time after locking the MB

2) Avoid using message buffers 5 and 15

## ERR052651: FlexCAN: CAN frames dropped when using Enhanced RX FIFO

### Description

When using message buffers 5 or 15, an incoming CAN frame will be lost (i.e., not latched into its expected Enhanced RX FIFO data element) without indication that the frame was lost if both of the following conditions are met simultaneously.

1) A write access is made to the Message Buffer Control and Status word (MB_CS) of the message buffer corresponding to the expected Enhanced RX FIFO data element.

2) The write access is made when receiving a frame at a specific Controller Host Interface (CHI) clock cycle. The specific clock cycle depends on the timestamp configuration as detailed below:

a) If the timestamp is disabled (CTRL2[TSTAMPCAP] = 00b) - Between the seventh bit of EOF and the second bit of IFS.

b) If the timestamp is enabled (CTRL2[TSTAMPCAP] = 01b or 10b or 11b) - Between the fifth bit of EOF and the seventh bit of EOF.

### Workaround

There are three workarounds for this errata:

1) Disable the Enhanced RX FIFO feature

2) Do not use message buffers 5 or 15

3) Avoid updated the Message Buffer Control and Status (MB_CS) word of message buffers 5 or 15 when any reception to the Enhanced RX FIFO could occur. This means, it would be safe to update the MB_CS of these message buffers when the FlexCAN is in Freeze mode or when it is otherwise not possible to receive frames from the CAN bus.

## ERR052728: ELS: Some devices can have intermittent DTRNG errors

### Description

Some devices with date code prior to 2511 can encounter intermittent DTRNG errors. A DTRNG error is caused when the TRNG is unable to gather sufficient entropy. The DTRNG errors can block ELS operations during application use. The ELS is also used by the ROM for secure operations and symptoms can include secure boot failures or issues loading SB3 files.

### Workaround

There are two workarounds for this issue:

- Devices from date code 2511 and onwards have an updated DTRNG configuration to address this erratum.

OR

- For devices prior to date code 2511, apply ROM field patch 1.0.7 available for earlier date codes at https://community.nxp.com/t5/MCX-Microcontrollers-Knowledge/MCX-N23x-Boot-ROM-update-to-vT1-0-7/ta-p/2074480

# 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