Silicon Errata for the CYT2BL Series Rev. A

Document Revision: 2.0
Date: 2023-10-17

This document describes the errata for CYT2BL Rev. A Series of devices. Details include trigger conditions, scope of impact, available workarounds, and applicable silicon revisions. Contact your local Infineon Sales representative for further questions.

Part Numbers Affected

Part Number
All CYT2BL Rev. A parts

CYT2BL Rev. A Qualification Status

Product Status: Engineering Samples/Production Samples

CYT2BL Rev. A Errata Summary

This table defines the errata applicable to CYT2BL Rev. A Series of devices.

Items	Part Number	Silicon Revision	Fix Status
Errata ID 67 ConfigureFmInterrupt API assumes a parameter with 8 bytes boundary, but actual boundary is 4 bytes	CYT2BL3CAAES CYT2BL4CAAES CYT2BL5CAAES CYT2BL7CAAES CYT2BL8CAAES CYT2BL3BAAQ0AZSGS CYT2BL3BAAQ0AZEGS CYT2BL3CAAQ0AZSGS CYT2BL3CAAQ0AZEGS CYT2BL4BAAQ0AZSGS CYT2BL4BAAQ0AZEGS CYT2BL4CAAQ0AZSGS CYT2BL4CAAQ0AZEGS CYT2BL5BAAQ0AZSGS CYT2BL5BAAQ0AZEGS CYT2BL5CAAQ0AZSGS CYT2BL5CAAQ0AZEGS CYT2BL7BAAQ0AZSGS CYT2BL7BAAQ0AZEGS CYT2BL7CAAQ0AZSGS CYT2BL7CAAQ0AZEGS CYT2BL8BAAQ0AZSGS CYT2BL8BAAQ0AZEGS CYT2BL8CAAQ0AZSGS CYT2BL8CAAQ0AZEGS	Rev. A	No silicon fix planned. Use workaround.
Errata ID 68 SMPU/MPU/PPU protection region size is limited to 2 GB			No silicon fix planned. Use workaround.
Errata ID 69 DirectExecute API may return error if called with arguments placed in SRAM			No silicon fix planned. Use workaround.
Errata ID 96 CAN FD RX FIFO top pointer feature does not function as expected			No silicon fix planned. Use workaround.
Errata ID 97 CAN FD debug message handling state machine not get reset to Idle state when CANFD_CH_CCCR.INIT is set			No silicon fix planned. Use workaround.
Errata ID 98 TPIU peripheral ID mismatch			No silicon fix planned.
Errata ID 137 Limitation of Work flash reading			No silicon fix planned. Use workaround.

Items	Part Number	Silicon Revision	Fix Status
Errata ID 138 ROM boot code clears to zero last 2 KB of SRAM	CYT2BL3CAAES CYT2BL4CAAES CYT2BL5CAAES CYT2BL7CAAES CYT2BL8CAAES CYT2BL3BAAQ0AZSGS CYT2BL3BAAQ0AZEGS CYT2BL3CAAQ0AZSGS CYT2BL3CAAQ0AZEGS CYT2BL4BAAQ0AZSGS CYT2BL4BAAQ0AZEGS CYT2BL4CAAQ0AZSGS CYT2BL4CAAQ0AZEGS CYT2BL5BAAQ0AZSGS CYT2BL5BAAQ0AZEGS CYT2BL5CAAQ0AZSGS CYT2BL5CAAQ0AZEGS CYT2BL7BAAQ0AZSGS CYT2BL7BAAQ0AZEGS CYT2BL7CAAQ0AZSGS CYT2BL7CAAQ0AZEGS CYT2BL8BAAQ0AZSGS CYT2BL8BAAQ0AZEGS CYT2BL8CAAQ0AZSGS CYT2BL8CAAQ0AZEGS	Rev. A	No silicon fix planned. Use workaround.
Errata ID 139 Limitation of programming SFlash Normal Access Restrictions (row 13)			No silicon fix planned. Use workaround.
Errata ID 147 CAN FD controller message order inversion when transmitting from dedicated Tx Buffers configured with same Message ID			No silicon fix planned. Use workaround.
Errata ID 161 Boot time specifications are not correct			Datasheet (002-28876 Rev. *D) was updated.
Errata ID 162 Temperature sensor accuracy issue			No silicon fix planned. Use workaround. TRM (002-19314 Rev. *I) was updated.
Errata ID 164 Addition of the sampling time spec for the temperature sensor			No silicon fix planned. Use workaround. Datasheet (002-28876 Rev. *D) was updated.
Errata ID 167 CAN FD incomplete description of Dedicated Tx Buffers and Tx Queue related to transmission from multiple buffers configured with the same Message ID			No silicon fix planned. Use workaround. TRM (002-19314 Rev. *H) was updated.
Errata ID 168 DeepSleep wakeup time increases at cold temperature			No silicon fix planned. Datasheet (002-28876 Rev. *D) was updated.
Errata ID 169 IMO accuracy spec change			No silicon fix planned. Datasheet (002-28876 Rev. *F) was updated.
Errata ID 175 Misleading status is returned for Flash and eFuse system calls if there are pending NC ECC faults in SRAM controller #0			No silicon fix planned. TRM (002-19314 Rev. *I) was updated.
Errata ID 176 WDT reset causes loss of SRAM retention			No silicon fix planned. TRM (002-19314 Rev. *I) was updated.
Errata ID 185 Crypto ECC errors may be set after boot with application authentication			No silicon fix planned. TRM (002-19314 Rev. *I) was updated.
Errata ID 198 Incomplete erase of Code Flash cells could happen Erase Suspend / Erase Resume is used along with Erase Sector operation in Non-Blocking mode			Fixed to update the Flash settings from date code 312xxxxx

Items	Part Number	Silicon Revision	Fix Status
Errata ID 199 Limitation for keeping the port state from peripheral IP after wakeup from DeepSleep			No silicon fix planned. TRM (002-19314 Rev. *I) was updated.
Errata ID 201 A part of the PWR_CTL2.BGREF_LPMODE description is lacked in the existing register TRM			No silicon fix planned. Register TRM (002-29852 Rev. *C) was updated.
Errata ID 202 Limitation of clock configuration before entering DeepSleep mode			No silicon fix planned. TRM (002-19314 Rev. *I) was updated.
Errata ID 203 Several data retention information in Register TRM are incorrect			No silicon fix planned. Register TRM (002-29852 Rev. *C) was updated.
Errata ID 204 SCBx_INTR_TX.UNDERFLOW bit may be set unintentionally			No silicon fix planned. Register TRM (002-29852 Rev. *C) was updated.
Errata ID 206 Hardfault may occur when calling some SROM APIs while executing EraseSector or ProgramRow in non-blocking mode			No silicon fix planned. TRM (002-19314 Rev. *J) will be updated.
Errata ID 209 CAN FD sporadic data corruption (payload) in case acceptance filtering is not finished before reception of data R3 (DB7..DB4) is completed			No silicon fix planned. Use workaround.
Errata ID 212 Description for PASS SARx to TCPWMx direct connect triggers one-to-one is incorrect in datasheet			No silicon fix planned. Datasheet (002-28876 Rev. *H) will be updated.

67. ConfigureFmInterrupt API assumes a parameter with 8 bytes boundary, but actual boundary is 4 bytes

- **Problem Definition**
  STATUS_ADDR_PROTECTED will be returned if ConfigureFmInterrupt API is called with arguments stored in SRAM with 4 bytes boundary of available SRAM or protected boundary SRAM.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Call ConfigureFmInterrupt API with arguments stored in SRAM at 4 bytes boundary of available SRAM or protected boundary of SRAM.
- **Scope of Impact**
  ConfigureFmInterrupt API will fail by returning STATUS_ADDR_PROTECTED error status when called with argument having 4 bytes boundary of available SRAM or protected boundary of SRAM.
- **Workaround**
  Allow 4 bytes margin (i.e. assume that API parameter size as 8 and store the arguments) for API (ConfigureFmInterrupt) parameter.
- **Fix Status**
  No silicon fix planned. Use workaround.

68. SMPU/MPU/PPU protection region size is limited to 2 GB

- **Problem Definition**
  If SMPU/MPU/PPU protection block size is configured for 4 GB (PROT_SMPU_SMPU_STRUCT_ATT0.REGION.SIZE = 31), then during protection check in SROM, the value of the internal uint32 variable will overflow (4G = 0x1 0000 0000). Therefore, SROM assumes the protection size equals zero, and no protection will be applied.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Configure SMPU/MPU/PPU to protect with region size equal to 4 GB or the region size with value 31u.
- **Scope of Impact**
  If SMPU/MPU/PPU is configured to protect region size of 4 GB, then SROM software does not apply any protection as per the request.
- **Workaround**
  Use two protection blocks of region size equal to 2 GB if 4 GB region size protection is required.
- **Fix Status**
  No silicon fix planned. Use workaround.

69. DirectExecute API may return error if called with arguments placed in SRAM memory

- **Problem Definition**
  If DirectExecute API is called in master PC (other than PC0 or PC1) with arguments in SRAM_SCRATCH_ADDR, then the API will return STATUS_ADDR_PROTECTED status.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Call DirectExecute API with arguments in SRAM_SCRATCH_ADDR and master PC configured > 1.
- **Scope of Impact**
  DirectExecute API, if called with master PC configured > 1 and arguments in SRAM_SCRATCH_ADDR, the API will return STATUS_ADDR_PROTECTED.
- **Workaround**
  Call DirectExecute API with master PC0 or PC1, if arguments are stored in SRAM memory.
- **Fix Status**
  No silicon fix planned. Use workaround.

96. CAN FD RX FIFO top pointer feature does not function as expected

- **Problem Definition**
  RX FIFO top pointer function calculates the address for received messages in Message RAM by hardware. This address should be re-start back from the start address after reading all messages of RX FIFO n size (n: 0 or 1). However, the address does not re-start back from the start address when RX FIFO n size is set to 1 (CANFD_CH_RXFnC.FnS = 0x01). This results in CPU/DMA to read messages from the wrong address in Message RAM.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  RX FIFO top pointer function is used when RX FIFO n size set to 1 element (CANFD_CH_RXFnC.FnS = 0x01).
- **Scope of Impact**
  Received message cannot be correctly read by using RX FIFO top pointer function, when RX FIFO n size set to 1 element.
- **Workaround**
  Any of the following.
  1) Set RX FIFO n size to 2 or more when using RX FIFO top pointer function.

2) Do not use RX FIFO top pointer function when RX FIFO n size set to 1 element. Instead of RX FIFO top pointer, read received messages from the Message RAM directly.

- **Fix Status**
  No silicon fix planned. Use workaround.

97. CAN FD debug message handling state machine not get reset to Idle state when CANFD_CH_CCCR.INIT is set

- **Problem Definition**
  If either CANFD_CH_CCCR.INIT bit is set by the Host or when the M_TTCAN module enters BusOff state, the debug message handling state machine stays in its current state instead of being reset to Idle state. Configuring the bit CANFD_CH_CCCR.CCE does not change CANFD_CH_RXF1S.DMS.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Either CANFD_CH_CCCR.INIT bit is set by the Host or when the M_TTCAN module enters BusOff state.
- **Scope of Impact**
  The errata is limited to the use case when the Debug on CAN functionality is active. Normal operation of CAN module is not affected, in which case the debug message handling state machine always remains in Idle state. In the described use case, the debug message handling state machine is stopped and remains in the current state signaled by the bit CANFD_CH_RXF1S.DMS. In case CANFD_CH_RXF1S.DMS is set to 0b11, DMA request remains active.
  Bosch classifies this as non-critical error with low severity, there is no fix for the IP. Bosch recommends the workaround listed also here.
- **Workaround**
  In case the debug message handling state machine has stopped while CANFD_CH_RXF1S.DMS is 0b01 or 0b10, it can be reset to Idle state by hardware reset or by reception of debug messages after CANFD_CH_CCCR.INIT is reset to zero.
- **Fix Status**
  No silicon fix planned. Use workaround.

98. TPIU peripheral ID mismatch

- **Problem Definition**
  TPIU peripheral ID indicates that it is M3-TPIU instead of M4-TPIU.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  When debugger reads PID registers for component identification.
- **Scope of Impact**
  The debuggers read the TPIU as M3-TPIU and no other impact other than this.
- **Workaround**
  No specific workaround required. Debuggers can use trace features.
- **Fix Status**
  No silicon fix planned.

137. Limitation of work flash reading

- **Problem Definition**
  Work flash can be read via different CPU cores but only one CPU core is assigned for non-correctable ECC error handling.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Reading work flash via CPU core (CM0+, CM4) and ECC fault interrupt routed to two CPU cores.

- **Scope of Impact**
  Only one CPU core can be assigned for non-correctable ECC error handling.
- **Workaround**
  Any of the following:
  Option A (Recommended solution)
  Set each CPU core to use a separate DMA (M-DMA or P-DMA) channel to read work flash. If non-correctable ECC error occurs, the DMA transaction get aborted and respective CPU core gets the interrupt to manage the non-correctable ECC error.

  Option B[1]
  Set non-correctable ECC error handling to reset. This way no one CPU core needs to manage the non-correctable ECC error handling.

  Note [1]: Not recommended to use for EEPROM emulation. EEPROM emulation needs to cope with aborted write/erase. In such a scenario, option B leads to deadlock in endless resets. However, option B can be used if work flash update is not intended in the field.

  Option C[2]
  Assign one CPU core for non-correctable ECC error handling and that core informs the error to the other core which caused the error, but it takes time.

  Note [2]: Not recommended to use with MCAL, since the inter-core communication is too slow.

- **Fix Status**
  No silicon fix planned. Use workaround.
  Infineon FLS and FEE driver were updated with workaround A.
  TRM was updated for this limitation.

138. ROM boot code clears to zero last 2 KB of SRAM

- **Problem Definition**
  ROM boot code clears to zero last 2 KB of SRAM. This region is available to the user after boot. However, data retention across resets is not guaranteed in this area.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  After ROM boot
- **Scope of Impact**
  Data retention across resets is not guaranteed in last 2 KB of SRAM.
- **Workaround**
  Do not use last 2 KB of SRAM for data retention.
- **Fix Status**
  No silicon fix planned. Use workaround.
  TRM was updated for this limitation.

139. Limitation of programming SFlash Normal Access Restrictions (row 13)

- **Problem Definition**
  CM0+ cache is not disabled when programming SFlash Normal Access Restrictions (row 13) by WriteRow SROM API. Occasionally, writing to SFlash Normal Access Restrictions (row 13) may return error status “0xF00000A4” (ProgramRow is invoked on unerased cells or blank check fails).
- **Parameters Affected**
  N/A

- **Trigger Condition(s)**
  WriteRow SROM API is called on Normal Access Restrictions (row 13).
- **Scope of Impact**
  Error status – 0xF00000A4 “ProgramRow is invoked on unerased cells or blank check fails” is returned.
- **Workaround**
  Disable CM0+ cache before call to WriteRow (to SFlash row 13) and enabling the cache back after the SROM API execution. Following sequence could be a recommended sequence:
  1) FLASHC_CM0_CA_CTL0.CA_EN = 0; // Disable the CM0+ cache
  2) Trigger WriteRow SROM API on NAR (row 13)
  3) WriteRow successful
  4) FLASHC_CM0_CA_CTL0.CA_EN = 1; // Enable the CM0+ cache
- **Fix Status**
  No silicon fix planned. Use workaround.
  TRM was updated for this limitation.

147. CAN FD controller message order inversion when transmitting from dedicated Tx Buffers configured with same Message ID

- **Problem Definition**
  Configuration:
  Several Tx Buffers are configured with same Message ID. Transmission of these Tx Buffers is requested sequentially with a delay between the individual Tx requests.

  Expected behavior:
  When multiple Tx Buffers that are configured with the same Message ID have pending Tx requests, they shall be transmitted in ascending order of their Tx Buffer numbers. The Tx Buffer with lowest buffer number and pending Tx request is transmitted first.

  Observed behavior:
  It may happen, depending on the delay between the individual Tx requests, that in the case where multiple Tx Buffers are configured with the same Message ID the Tx Buffers are not transmitted in order of the Tx Buffer number (lowest number first).
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  When multiple Tx Buffers that are configured with the same Message ID have pending Tx requests.
- **Scope of Impact**
  In the case described it may happen, that Tx Buffers configured with the same Message ID and pending Tx request are not transmitted with lowest Tx Buffer number first (message order inversion).
- **Workaround**
  Any of the following:
  1) First write the group of Tx message with the same Message ID to the Message RAM and then afterwards request transmission of all these messages concurrently by a single write access to CANFDx_CHy_TXBAR. Before requesting a group of Tx messages with this Message ID ensure that no message with this Message ID has a pending Tx request.
  2) Use the Tx FIFO instead of dedicated Tx Buffers for the transmission of several messages with the same Message ID in a specific order.

  Applications not able to use workaround #1 or #2 can implement a counter within the data section of their messages sent with same ID in order to allow the recipients to determine the correct sending sequence.
- **Fix Status**
  No silicon fix planned. Use workaround.

161. Boot time specifications are not correct

- **Problem Definition**
  TRAVEO™ T2G silicon's shipped until now (flash boot version earlier than 3.1.0.556) have a higher boot time compared to what is mentioned in datasheet rev. B.
- **Parameters Affected**
  SID80A, SID80B, SID81A, SID81B
- **Trigger Condition(s)**
  Device boot
- **Scope of Impact**
  System startup time will be longer than expected.
- **Workaround**
  None
- **Fix Status**
  Boot time specifications was updated in datasheet (002-28876 Rev. *D) for flash boot version 3.1.0.556 and later:
  SID80A: 1800 µs
  SID80B: 2740 µs
  SID81A: 80 µs (No change)
  SID81B: 5000 µs

162. Temperature sensor accuracy issue

- **Problem Definition**
  TRAVEO™ T2G Body Entry devices have inconsistent references applied to temperature sensor calibration.
- **Parameters Affected**
  SID200, SID201, SID201A
- **Trigger Condition(s)**
  Using the temperature sensor
- **Scope of Impact**
  It results in inaccurate calculation of measured temperature.
- **Workaround**
  Set bit 9, 8 and 6 of PASS_TEST_CTL register (address: 0x409F0080) to 1 while keeping the other bits unchanged, after a reset or DeepSleep wakeup.
- **Fix Status**
  No silicon fix planned. Use workaround.
  TRM (002-19314 Rev. *H) was updated to add the PASS_TEST_CTL register description.

164. Addition of the sampling time spec for the temperature sensor

- **Problem Definition**
  Existing datasheet (rev. B) didn't have defined the sampling time spec for the temperature sensor. The settling time for the temperature sensor needs to be included in the sampling time of ADC.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Using the temperature sensor
- **Scope of Impact**
  For the temperature sensor, the sampling time needs to be longer than normal A/D conversion.
- **Workaround**
  Set the sampling time to 3 µs and more.
- **Fix Status**
  No silicon fix planned. Use workaround.
  Datasheet (002-28876 Rev. *D) was updated to add the sampling time spec for the temperature sensor.

167. CAN FD incomplete description of Dedicated Tx Buffers and Tx Queue related to transmission from multiple buffers configured with the same Message ID

- **Problem Definition**
  The following is the updated description in Section 3.5.2 Dedicated Tx Buffers and 3.5.4 Tx Queue of the Architecture TRM related to transmission from multiple buffers configured with the same Message ID.

  3.5.2 Dedicated Tx Buffers
  - Wording TRM:
  If multiple Tx Buffers are configured with the same Message ID, the Tx Buffer with the lowest buffer number is transmitted first.
  - Enhancement:
  These Tx buffers shall be requested in ascending order with lowest buffer number first. Alternatively all Tx buffers configured with the same Message ID can be requested simultaneously by a single write access to CANFDx_CHy_TXBAR.

  3.5.4 Tx Queue
  - Wording TRM:
  If multiple queue buffers are configured with the same Message ID, the queue buffer with the lowest buffer number is transmitted first.
  - Replacement:
  In case that multiple Tx Queue buffers are configured with the same Message ID, the transmission order depends on numbers of the buffers where the messages were stored for transmission. As these buffer numbers depend on the then current states of the PUT Index, a prediction of the transmission order is not possible.

  - Wording TRM:
  An Add Request cyclically increments the Put Index to the next free Tx Buffer.
  - Replacement:
  The PUT Index always points to that free buffer of the Tx Queue with the lowest number.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Using multiple dedicated Tx Buffers or Tx Queue Buffers configured with the same Message ID.
- **Scope of Impact**
  In the case the dedicated Tx Buffers with the same Message ID are not requested in ascending order or at the same time or in case of multiple Tx Queue Buffers with the same Message ID, it cannot be guaranteed, that these messages are transmitted in ascending order with lowest buffer number first.
- **Workaround**
  In case a defined order of transmission is required the Tx FIFO shall be used for transmission of messages with the same Message ID. Alternatively dedicated Tx Buffers with the same Message ID shall be requested in ascending order with lowest buffer number first or by a single write access to CANFDx_CHy_TXBAR. Alternatively a single Tx Buffer can be used to transmit those messages one after the other.
- **Fix Status**
  No silicon fix planned. Use workaround. TRM (002-19314 Rev. *H) was updated accordingly.

168. DeepSleep wakeup time increases at cold temperature

- **Problem Definition**
  DeepSleep wakeup time increases at cold temperature (-5 °C to -40 °C).
- **Parameters Affected**
  SID63/A/B/C/D

- **Trigger Condition(s)**
  DeepSleep wakeup at cold temperature (-5 °C to -40 °C)
- **Scope of Impact**
  DeepSleep wakeup time increases at cold temperature (-5 °C to -40 °C).
- **Workaround**
  None
- **Fix Status**
  No silicon fix planned. The following note was added to the affected SIDs in the datasheet (002-28876 Rev. *D).
  Note: At cold temperature -5 °C to -40 °C, the DeepSleep to Active transition time can be higher than the max time indicated by as much as 20 µs

169. IMO accuracy spec change

- **Problem Definition**
  IMO accuracy spec is changed from +/- 1% to +/- 4% because IMO is not trimmed after DeepSleep wakeup.
- **Parameters Affected**
  SID310
- **Trigger Condition(s)**
  Using IMO
- **Scope of Impact**
  IMO accuracy spec change
- **Workaround**
  None
- **Fix Status**
  No silicon fix planned. SID310A was updated in datasheet (002-28876 Rev. *D). Root and intermediate clocks table was updated as below in datasheet (002-28876 Rev. *F).

Clock	Maximum permitted clock frequency (MHz) [1]	Source	Maximum permitted clock frequency setting (MHz) [1]	
			Clock source of PLL/FLL: ECO [2]	Clock source of PLL/FLL: IMO [3]
CLK_HF0	160	PLL200#0	160	155
		FLL	100	96
	100	PLL200#0	100	98
		FLL	100	96
CLK_HF1	100	PLL200#0	100	98
		FLL	100	96
CLK_HF2	2	ILO	N/A	N/A
CLK_FAST	160	PLL200#0	160	155
		FLL	100	96
	100	PLL200#0	100	98
		FLL	100	96
CLK_SLOW	100	PLL200#0	100	98
		FLL	100	96
CLK_PERI	100	PLL200#0	100	98
		FLL	100	96

[1]: Maximum clock frequency after the corresponding clock source (PLL/FLL + dividers). All internal tolerances and affects are covered by these frequencies.
[2]: For ECO: up to ±150 ppm uncertainty of the external clock source are tolerated by design.

[3]: For IMO: the IMO operation frequency tolerance is included. When DeepSleep mode isn't used, maximum permitted clock frequency setting of clock source IMO case is equal to clock source ECO case.

175. Misleading status is returned for Flash and eFuse system calls if there are pending NC ECC faults in SRAM controller #0

- **Problem Definition**
  Flash and eFuse system calls will return misleading status of 0xf0000005 (“Page is write protected”) even for non-protected row or 0xf0000002 (“Invalid eFuse address”) for valid eFuse address in case of pending NC ECC faults in SRAM controller #0.
- **Parameters Affected**
  Return status of Flash and eFuse system calls
- **Trigger Condition(s)**
  NC ECC fault(s) pending in SRAM controller #0 and SWPUs are populated in the design.
- **Scope of Impact**
  Flash and eFuse system calls will not work until the NC ECC fault(s) pending in SRAM controller #0 is properly handled.
- **Workaround**
  If the NC ECC fault(s) are not due to HW malfunction (i.e., if the faults are due to usage of non-initialized SRAM or improper SRAM initialization), then clearing of these pending faults will resolve the issue.
- **Fix Status**
  No silicon fix planned. TRM (002-19314 Rev. *I) was updated.

176. WDT reset causes loss of SRAM retention

- **Problem Definition**
  Architecture TRM Table 19-1 shows WDT reset can retain SRAM if there is an orderly shutdown of the SRAM only during a warning interrupt. However, this is wrong. WDT reset causes loss of SRAM retention.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  WDT reset
- **Scope of Impact**
  WDT reset causes loss of SRAM retention.
- **Workaround**
  None
- **Fix Status**
  No silicon fix planned. TRM (002-19314 Rev. *I) was updated.

185. Crypto ECC errors may be set after boot with application authentication

- **Problem Definition**
  Due to the improper initialization of the Crypto memory buffer, Crypto ECC errors may be set after boot with application authentication. In spite of the Crypto ECC errors, the result of the authentication is reliable.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Boot device with application authentication.
- **Scope of Impact**
  Crypto ECC errors may be set after boot with application authentication.
- **Workaround**
  Clear or ignore Crypto ECC errors which generated during boot with application authentication.

- **Fix Status**
  No silicon fix planned. TRM (002-19314 Rev. *I) was updated.

198. Incomplete erase of Code Flash cells could happen Erase Suspend / Erase Resume is used along with Erase Sector operation in Non-Blocking mode

- **Problem Definition**
  Code Flash memory can be erased in “Non-Blocking” mode; a Non-Blocking mode supported option allow users to suspend an ongoing erase sector operation. When an ongoing erase operation is interrupted using “Erase Suspend” and “Erase Resume”, Flash cells may not have been erased completely, even after the erase operation complete is indicated by FLASHC_STATUS register. Only Code Flash is impacted by this issue, Work Flash and Supervisory Flash (SFlash) are not impacted.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Using EraseSector System Call in Non-Blocking mode for CM0+ to erase Code Flash and the ongoing erase operation is interrupted using EraseSuspend and EraseResume System calls.
- **Scope of Impact**
  When Code Flash sectors are erased in Non-Blocking mode and the ongoing erase operation is interrupted by Erase Suspend / Erase Resume, it cannot be guaranteed that the Code Flash cells are fully erased. Any read on the Code Flash area after the erase is complete or read on the programmed data after ProgramRow is complete can trigger ECC errors.
- **Workaround**
  Use any of the following:
  1) User can use Non-Blocking mode for EraseSector, but users must not interrupt the erase operation using Erase Suspend / Erase Resume.
  2) If a Code Flash sector erase operation is interrupted using Erase Suspend / Erase Resume, then erase the same sector again without Erase Suspend / Erase Resume before reading the sector or programming the sector.
- **Fix Status**
  Fixed to update the Flash settings from date code 312xxxxx, via Manufacturing Test Program Update for Code Flash setting; this fix is transferred to TRAVEO™ T2G devices during Infineon Factory Test Flow. Fixed devices will be identified by Device Date Code, which is marked on every TRAVEO™ T2G device.

199. Limitation for keeping the port state from peripheral IP after wakeup from DeepSleep

- **Problem Definition**
  The port state is not retained when the port selects peripheral IP (except for LIN or CAN FD) and MCU wakes up from DeepSleep.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  The port selects peripheral IP (except for LIN or CAN FD) and MCU wakes up from DeepSleep.
- **Scope of Impact**
  Unexpected port output change might affect user system.
- **Workaround**
  If the port selects peripherals IP (except for LIN or CAN FD) and the port output value need to keep after wakeup from DeepSleep, set HSIOM_PRTx_PORT_SEL.IOy_SEL = 0 (GPIO) before DeepSleep and set the required output value in GPIO configuration registers. After wakeup, change HSIOM_PRTx_PORT_SEL.IOy_SEL back to the peripheral IP.
- **Fix Status**
  No silicon fix planned. TRM (002-19314 Rev. *I) was updated to add above workaround.

201. A part of the PWR_CTL2.BGREF_LPMODE description is lacked in the existing register TRM

- **Problem Definition**
  The following is lacked from the PWR_CTL2.BGREF_LPMODE description in the existing register TRM.
  “This register will not set unless CLK_ILO0_CONFIG.ILO0_ENABLE==1. When changing back to continuous operation, keep ILO0 enabled for at least 5 ILO0 cycles after clearing this bit to allow for internal synchronization.”
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Using the PWR_CTL2.BGREF_LPMODE
- **Scope of Impact**
  PWR_CTL2.BGREF_LPMODE may not be set or cleared.
- **Workaround**
  Use the PWR_CTL2.BGREF_LPMODE according to the following description.
  “This register will not set unless CLK_ILO0_CONFIG.ILO0_ENABLE==1. When changing back to continuous operation, keep ILO0 enabled for at least 5 ILO0 cycles after clearing this bit to allow for internal synchronization.”
- **Fix Status**
  No silicon fix planned. Register TRM (002-29852 Rev. *C) was updated.

202. Limitation of clock configuration before entering DeepSleep mode

- **Problem Definition**
  DeepSleep should not be entered while any FLL/PLL is enabled and using ECO as its reference clock. Since the unstable ECO clock after wakeup is outside the allowed reference clock limits for FLL/PLL, there is possibility of failing the DeepSleep wakeup.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  DeepSleep transition while any FLL/PLL is enabled and using ECO as its reference clock.
- **Scope of Impact**
  There is possibility of failing the DeepSleep wakeup.
- **Workaround**
  If any FLL/PLL is operating with the ECO as its reference clock, change the clock to either ECO direct or IMO direct or IMO with FLL/PLL before entering DeepSleep.
- **Fix Status**
  No silicon fix planned. TRM (002-19314 Rev. *I) was updated to add above workaround.

203. Several data retention information in Register TRM are incorrect

- **Problem Definition**
  The following registers are described as ‘Retained’ in the Register TRM while it is not guaranteed that the value before entering DeepSleep mode is still readable from the register.
  - SARADC: PASSx_SARy_CHz_RESULT
  - SRSS: PWR_LVD_STATUS
  - SRSS: PWR_LVD_STATUS2
  - SRSS: CLK_CAL_CNT1
  - SRSS: CLK_CAL_CNT2
  - SRSS: CLK_FLL_STATUS
  - ~~SRSS: WDT_CNT~~ (It is retained during DeepSleep or Hibernate if DPSLP_PAUSE == 1 or HIB_PAUSE == 1)
  - SRSS: WDT_INTR
  - SRSS: WDT_INTR_MASKED
- **Parameters Affected**
  N/A

- **Trigger Condition(s)**
  Use of the related function and wakeup from DeepSleep mode.
- **Scope of Impact**
  The values before entering DeepSleep are not retained.
- **Workaround**
  For PASSx_SARy_CHz_RESULT, any of following:
  1) Store the conversion values at another memory location before entering DeepSleep mode
  2) Restart the conversion after wakeup from DeepSleep mode

  For the other registers:
  Rewrite the register value or read the status flags again after wakeup.
- **Fix Status**
  No silicon fix planned. Register TRM (002-29852 Rev. *C) was updated.

204. SCBx_INTR_TX.UNDERFLOW bit may be set unintentionally

- **Problem Definition**
  There is possibility of setting the SCBx_INTR_TX.UNDERFLOW bit even if the FIFO is not empty.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Using the TX FIFO for SCB when the AHB-Lite interface clock (CLK_GR6) frequency of the AHB bus is greater than 3x the SCB functionality clock (PCLK_SCBx_CLOCK).
- **Scope of Impact**
  SCBx_INTR_TX.UNDERFLOW bit may be set unintentionally.
- **Workaround**
  Ignore the SCBx_INTR_TX.UNDERFLOW bit if the FIFO is not empty.
- **Fix Status**
  No silicon fix planned. Register TRM (002-29852 Rev. *C) was updated.

206. Hardfault may occur when some SROM APIs while executing EraseSector or ProgramRow in non-blocking mode

- **Problem Definition**
  The following SROM APIs read data from bank#0 (or bank#1 if dual bank mode with mapping B is used) in SFlash. While doing that the check for active non-blocking erase or program of bank#0 (or bank#1 if dual bank mode with mapping B is used) is not performed. Therefore, reading bank#0 (or bank#1 if dual bank mode with mapping B is used) while there is an active erase/program operation will trigger a bus error which can result in a hardfault occurrence based on FLASHC_FLASH_CTL register settings.
  Affected SROM APIs:
  - ReadSWPU
  - WriteSWPU
  - GenerateHash
  - Checksum*
  - ComputeBasicHash*
  - CheckFactoryHash
  - ProgramWorkFlash**

  *: Note that it should not be called if you programming/erasing bank that you are going to calculate.
  **: Note that it is not possible to use it during non-blocking operation.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Calling the affected SROM APIs while executing EraseSector or ProgramRow in non-blocking mode on bank#0 (or bank#1 if dual bank mode with mapping B is used).

- **Scope of Impact**
  The affected SROM APIs cannot be used while executing EraseSector or ProgramRow in non-blocking mode on bank#0 (or bank#1 if dual bank mode with mapping B is used).
- **Workaround**
  Do not use the affected SROM APIs while executing EraseSector or ProgramRow in non-blocking mode on bank#0 (or bank#1 if dual bank mode with mapping B is used).
- **Fix Status**
  No silicon fix planned. TRM (002-19314 Rev. *J) will be updated.
- **Impact on Infineon Software**
  Impact: Limitation
  Related modules: S-LLD, HSM-Perf-Lib
  Comment: While executing EraseSector or ProgramRow in non-blocking mode on bank#0 (or bank#1 if dual bank mode with mapping B is used), users must not do anything of following:
  a) call CySldProt_GetSwpuFlashStructCfg
  b) call CySldProt_VerifySecureDomainFlashWriteProtection if CySldProt_SwpuFlashStructGroupConfigurations is non-empty.

209. CAN FD sporadic data corruption (payload) in case acceptance filtering is not finished before reception of data R3 (DB7..DB4) is completed

- **Problem Definition**
  During frame reception the Rx Handler accesses the external Message RAM for acceptance filtering (read accesses) and for storing of the accepted messages (write accesses).

  The time needed for acceptance filtering and for storing of a received message depends on
  · the Host clock frequency
  · the worst-case latency of the read and write accesses to the external Message RAM
  · the number of configured filter elements
  · the workload of the transmit message (Tx) handler in parallel to the receive message (Rx) handler

  Received data bytes (DB0..DBm) from the CAN Core are buffered in the cache of the Rx Handler before they are written to the Message RAM (in words of 4 byte). Data words inside the Message RAM are numbered from R2 to Rn (n ≤ 17).

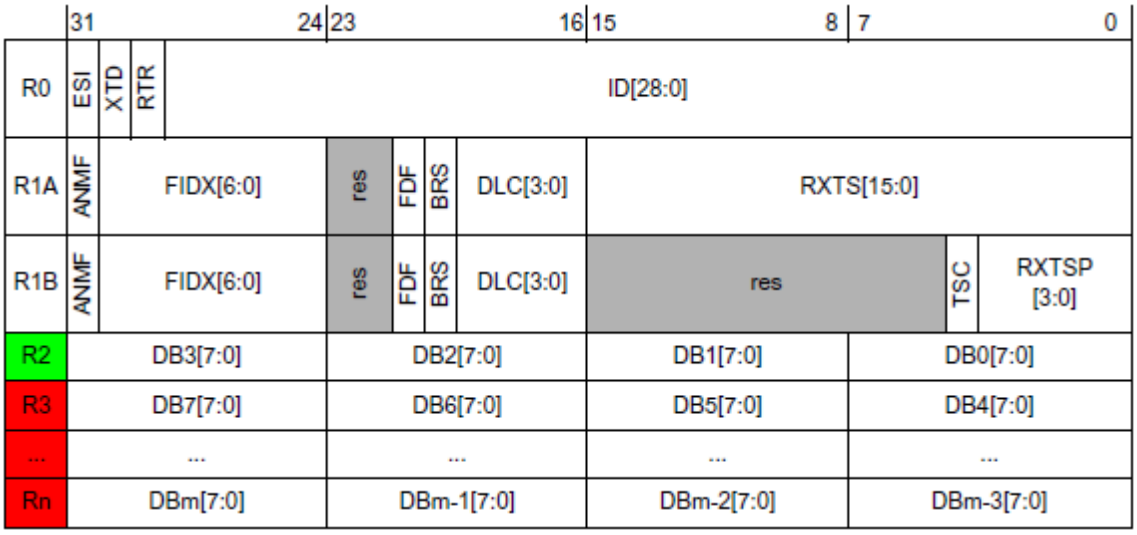

Figure 1. Rx Buffer and FIFO Element

Under the following conditions a received message will have corrupted data while the received message is signaled as valid to the host.

1) The data length code (DLC) of the received Message is greater than 4 (DLC > 4)
2) The storage of Ri of a received message into the Message RAM (after acceptance filtering is done) has not completed before R(i+1) is transferred from the CAN Core into the cache of the Rx Handler (where 2 ≤ i ≤ 5).
3) While condition 1) and 2) apply, a concurrent read of data word Ri from the cache and write of data word R(i+1) into the cache of the Rx handler happens.

The data will be corrupted in a way, that in the Message RAM R(i+1) has the same content as Ri.

Despite the corrupted data, the M_TTCAN signals the storage of a valid frame in the Message RAM:
· Rx FIFO: FIFO put index RXFnS.FnPI is updated.
· Dedicated Rx Buffer: New Data flag NDATn.NDxx is set.
· Interrupt flag IR.MRAF is not set.

The issue may occur in FD Frame Format as well as in Classic Frame Format.

Figure 2 shows how the available time for acceptance filtering and storage is reduced.

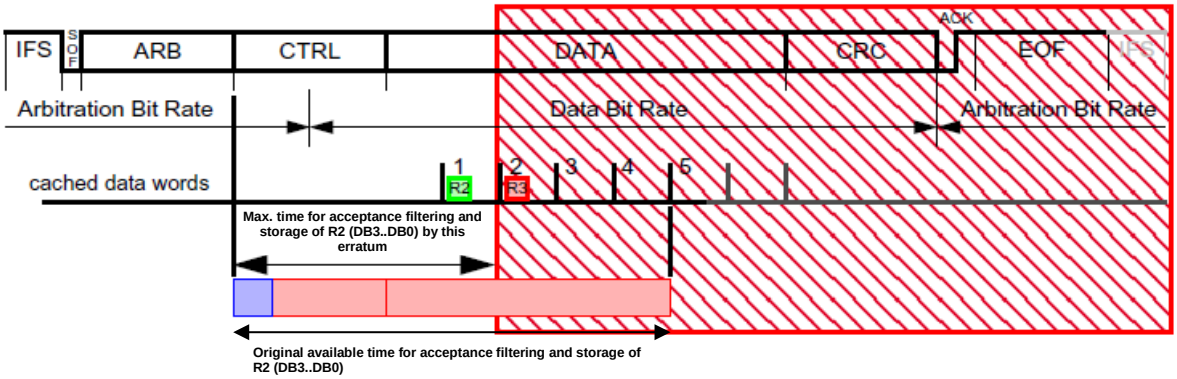

Figure 2. CAN Frame with DLC>4

Table 1. TRAVEO T2G: Minimum host clock frequency for CAN FD when DLC = 5

Number of configured active filter element 11-bit IDs / 29-bit IDs [1,2]	Number of active CAN channels in an instance	Arbitration bit rate = 0.5 Mbps				Arbitration bit rate = 1 Mbps			
		Data bit rate = 0.5 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 5 Mbps
32 / 16	2	3.9 MHz	7.1 MHz	13.1 MHz	22.8 MHz	7.7 MHz	14.1 MHz	26.1 MHz	31.5 MHz
	3	5.4 MHz	9.9 MHz	18.3 MHz	31.8 MHz	10.7 MHz	19.7 MHz	36.5 MHz	44.0 MHz
	4	6.9 MHz	12.7 MHz	23.5 MHz	40.8 MHz	13.8 MHz	25.3 MHz	46.9 MHz	56.5 MHz
64 / 32	2	7.4 MHz	13.5 MHz	24.9 MHz	43.4 MHz	14.7 MHz	26.9 MHz	49.8 MHz	60.0 MHz
	3	10.3 MHz	18.8 MHz	34.9 MHz	60.7 MHz	20.5 MHz	37.6 MHz	69.7 MHz	84.0 MHz
	4	13.2 MHz	24.2 MHz	44.8 MHz	78.0 MHz	26.3 MHz	48.4 MHz	89.5 MHz	107.9 MHz[3]

Number of configured active filter element 11-bit IDs / 29-bit IDs [1,2]	Number of active CAN channels in an instance	Arbitration bit rate = 0.5 Mbps				Arbitration bit rate = 1 Mbps			
		Data bit rate = 0.5 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 1 Mbps	Data bit rate = 2 Mbps	Data bit rate = 4 Mbps	Data bit rate = 5 Mbps
96 / 48	2	10. 8 MHz	19.9 MHz	36.8 MHz	64.0 MHz	21.6 MHz	39.7 MHz	73.5 MHz	88.6 MHz
	3	15.1 MHz	27.8 MHz	51.5 MHz	89.6 MHz	30.2 MHz	55.6 MHz	102.9 MHz[3]	124.0 MHz[3]
	4	19.4 MHz	35.7 MHz	66.1 MHz	115.1 MHz[3]	38.8 MHz	71.4 MHz	132.2 MHz[3]	159.3 MHz[3]
128 / 64	2	14.3 MHz	26.3 MHz	48.6 MHz	84.7 MHz	28.4 MHz	52.5 MHz	97.2 MHz	117.2 MHz[3]
	3	20.0 MHz	36.8 MHz	68.0 MHz	118.5 MHz[3]	40.0 MHz	73.5 MHz	136.0 MHz[3]	164.0 MHz[3]
	4	25.7 MHz	47.2 MHz	87.5 MHz	152.3 MHz[3]	51.4 MHz	94.4 MHz	174.9 MHz[3]	210.8 MHz[3]

1. M_TTCAN starts always at filter element #0 and proceeds through the filter list to find a matching element. Acceptance filtering stops at the first matching element and the following filter elements are not evaluated for this message. Therefore, the sequence of configured filter elements has a significant impact on the performance of the filtering process.
2. Acceptance filtering search for 11-bit IDs and 29-bit IDs filter element is running separately, only one configured filter setting should be considered. Searching for one 29-bit filter element requires approximately double cycles for one 11-bit filter element.
3. Frequency is not reachable since the maximum host clock frequency for M_TTCAN in TRAVEO™ T2G is 100 MHz.

- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Under the following conditions a received message will have corrupted data while the received message is signaled as valid to the host.
  1) The data length code (DLC) of the received Message is greater than 4 (DLC > 4)
  2) The storage of Ri of a received message into the Message RAM (after acceptance filtering is done) has not completed before R(i+1) is transferred from the CAN Core into the cache of the Rx Handler (where 2 ≤ i ≤ 5).
  3) While condition 1) and 2) apply, a concurrent read of data word Ri from the cache and write of data word R(i+1) into the cache of the Rx handler happens.
- **Scope of Impact**
  The erratum is limited to the case when the Host clock frequency used in the actual device is below the limit shown in Table 1.
  Corrupted data is written to the Rx FIFO element respective the dedicated Rx Buffer.
  The received frame is nevertheless signaled as valid.
- **Workaround**
  Check whether the minimum Host clock frequency, that is shown in Table 1, is below the Host clock frequency used in the actual device.
  If yes, there is no problem with the selected configuration.
  If no, use one of the following two workarounds.

  First workaround

Try different configuration by changing the following parameters until the actual host clock frequency (CLK_GR5) is above the minimum host frequency shown in Table 1.
· Increase the CLK_GR5 frequency in the actual device
· Reduce the CAN-FD Data Bit rate
· Reduce the number of configured filter elements
· Reduce the number of active CAN channels in an instance
Also, use DLC>=8 instead of DLCs 5, 6, and 7 in the CAN Environment/System, as they place higher demands on the minimum Host clock frequency (the worst case is DLC=5) or restrict your CAN Environment/System to DLC 4.

Note: While changing the actual host clock frequency, CLK_GR5 must always be equal or higher than PCLK_CANFD[x]_CLOCK_CAN[y] for all configurations.

Second workaround
Due to condition 3) the issue occurs only sporadically. Use an end-to-end (E2E) protection (for example, checksum or CRC covering the data field) and add it to all messages in the CAN system, to detect data corruption in received frames.

- **Fix Status**
  No silicon fix planned. Use workaround.

- **Impact on Infineon Software**
  Impact: Limitation
  Related modules: CAN, MCU
  Comment: The user must evaluate the impact of the erratum for each CAN instance separately. A CAN instance is the entirety of CanControllers with the same CanControllerInstance value.
  1) For the number of active CAN nodes: Use the maximum number of CanController configurations of a CAN instance that can be active (Autosar controller state STARTED or SLEEP) at a time.
  2) For the host clock frequency: In McuPeriGroupSettings locate the setting with McuPeriGroup=MCU_PERI_GROUP5_MMIO5 and take the value from McuPeriGroupClockFrequency.
  4) For the number of configured active filter element 11-bit IDs / 29-bit IDs": Use the corresponding values from the "Message RAM (…) linking table" in the generated Can_PBcfg.h file. Note that each CanController has its separate table. Take the maximum values.
  5) For the Arbitration bit rate: Use the maximum CanControllerBaudRate value of all the CanControllers.
  6) For the Data bit rate: Use the maximum CanControllerFdBaudRate value of all the CanControllers if configured. Otherwise use CanControllerBaudRate.

212. Description for PASS SARx to TCPWMx direct connect triggers one-to-one is incorrect in datasheet

- **Problem Definition**
  The existing datasheet shows 'trig=2' in the description for PASS SARx to TCPWMx direct connect triggers one-to-one, which is incorrect as TCPWM's input trigger selection (TR_IN_SEL) value. The correct value is '4' as shown in the architecture TRM chapter 25 descriptions and table 25-2.
- **Parameters Affected**
  N/A
- **Trigger Condition(s)**
  Using the triggers one-to-one for PASS SARx to TCPWMx direct connect
- **Scope of Impact**
  The triggers one-to-one for PASS SARx to TCPWMx direct connect cannot work if TCPWM's input trigger selection is not correct.
- **Workaround**
  Use '4' as TCPWM's input trigger selection (TR_IN_SEL) value for PASS SARx to TCPWMx direct connect

- **Fix Status**
  No silicon fix planned. Datasheet (002-28876 Rev. H) will be updated.
- **Impact on Infineon Software**
  Impact: No
  Related modules: PWM
  Comment: MCAL PWM module does not support one-to-one triggers.

Revision history

Document revision	Date of release	Description of Change
1.0	2020-08-28	Initial released revision
1.1	2021-02-18	Added errata ID 147.
1.2	2021-06-29	Added errata ID 161, 162, 164.
1.3	2021-09-16	Updated “Workaround” of errata ID 147. Added errata ID 167, 168, 169.
1.4	2022-01-21	Added errata ID 175, 176.
1.5	2022-07-04	Updated “Fixed status” of errata ID 169. Added errata ID 185.
1.6	2022-08-25	Updated “Problem Definition” of errata ID 185. Added errata ID 198, 199.
1.7	2022-12-02	Added errata ID 201, 202, 203, 204.
1.8	2023-02-08	Updated the description to 5 ILO0 cycles in errata ID 201. Updated “Problem Definition” of errata ID 203. Fixed the register name of errata ID 204. Added errata ID 206.
1.9	2023-06-15	Updated errata ID 206 to add (or bank#1 if dual bank mode with mapping B is used). Added errata ID 209.
2.0	2023-10-17	Updated errata ID 206 to add the affected SROM APIs. Added errata ID 212.

Trademarks
All referenced product or service names and trademarks are the property of their respective owners.

Edition 2023-10-17

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Go to www.infineon.com/support

IMPORTANT NOTICE

The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.