GigaDevice Semiconductor Inc.

Device limitations of GD32C231

Errata Sheet

Revision 1.2

(Nov. 2025)

Table of Contents

Table of Contents **2**

List of Figures **4**

List of Tables **5**

1. Introduction **6**

1.1. Revision identification **6**

1.2. Summary of device limitations **6**

2. Descriptions of device limitations **8**

2.1. PMU **8**

2.1.1. When the chip is powered on, the PORRSTF flag probability cannot be set 8

2.1.2. Frequent wake-up signal before and after MCU enters the standby mode results in wake-up failure in the standby mode 8

2.1.3. When MCU enters Deep-sleep / Deep-sleep1 mode, if a system reset occurs, a very small number of chips may experience program runaway 8

2.2. RCU **9**

2.2.1. The clock monitor of the HXTAL is abnormal 9

2.2.2. When OBRLD is set, both EPRSTF and OBLRSTF flags are set 9

2.3. GPIO **10**

2.3.1. PA12 port LOCK function is abnormal 10

2.4. ADC **10**

2.4.1. ADC data acquisition error occurs when the ADC clock is much slower than the PCLK clock 10

2.4.2. When the ADC operates in routine sequence mode, setting the ADCON bit again after enabling the ADC will start an ADC conversion 10

2.4.3. When the ADC external trigger function is enabled and the trigger source switches from low level to high level, an ADC conversion will be triggered 11

2.5. SPI **11**

2.5.1. When the SPI slave works in non-TI mode and the data valid sampling edge is the first clock transition edge and CRC function is enabled, if the slave is not selected by the chip and there is still a clock on the SCK line, which will cause the slave CRC to continue working 11

2.6. I2S **12**

2.6.1. The I2S MCK clock output function is abnormal 12

2.7. I2C **12**

2.7.1. When I2C works in 7/10 address slave mode, receiving an abnormal timing will cause the SDA line to stuck 12

2.7.2. When the I2C slave is configured in 10-bit address mode, if the external master does not send a STOP signal after transmitting a frame of data, the I2C slave will be unable to match the slave address in subsequent operations 13

2.7.3. SMBUS master timeout caused by the slave pulling down the SCL line may result in the SMBUS master failing to issue a STOP signal 13

3. Revision history 14

List of Figures

Figure 1-1. Device revision code of GD32C231 6

List of Tables

Table 1-1. Applicable products 6
Table 1-2. Device limitations 6
Table 3-1. Revision history 14

1. Introduction

This document applies to GD32C231 product series, as shown in ***Table 1-1. Applicable products***. It offers technical guidance for using GD32MCU and provides workaround to current device limitations.

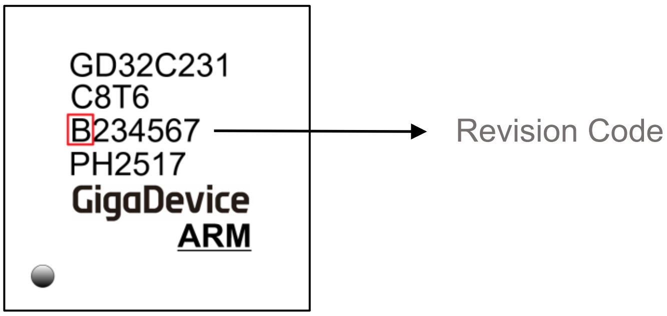
Table 1-1. Applicable products

Type	Part Numbers
MCU	GD32C231xx series

1.1. Revision identification

The device revision can be identified according to the mark on the top of the package. The 1st code on Line 3 of the mark is the product revision code, as shown in ***Figure 1-1. Device revision code of GD32C231***.

Figure 1-1. Device revision code of GD32C231



1.2. Summary of device limitations

The device limitations of GD32C231 are shown in ***Table 1-2. Device limitations***, please refer to Section 2 for more details.

Table 1-2. Device limitations

Module	Limitations	Workaround
		Rev. Code B
PMU	***When the chip is powered on, the PORRSTF flag probability cannot be set***	Y
	Frequent wake-up signal before and after MCU enters the standby mode results in wake-up failure in the standby mode	N
	When MCU enters Deep-sleep / Deep-sleep1 mode, if a system reset occurs, a very small number of chips may experience program runaway	Y
RCU	***The clock monitor of the HXTAL is abnormal***	Y

Module	Limitations	Workaround
		Rev. Code B
	When OBRLD is set, both EPRSTF and OBLRSTF flags are set	Y
GPIO	*PA12 port LOCK function is abnormal*	Y
ADC	*ADC data acquisition error occurs when the ADC clock is much slower than the PCLK clock*	Y
	When the ADC operates in routine sequence mode, setting the ADCON bit again after enabling the ADC will start an ADC conversion	Y
	When the ADC external trigger function is enabled and the trigger source switches from low level to high level, an ADC conversion will be triggered	Y
SPI	*When the SPI slave works in non-TI mode and the data valid sampling edge is the first clock transition edge and CRC function is enabled, if the slave is not selected by the chip and there is still a clock on the SCK line, which will cause the slave CRC to continue working*	Y
I2S	*The I2S MCK clock output function is abnormal*	Y
I2C	*When I2C works in 7/10 address slave mode, receiving an abnormal timing will cause the SDA line to stuck*	Y
	When the I2C slave is configured in 10-bit address mode, if the external master does not send a STOP signal after transmitting a frame of data, the I2C slave will be unable to match the slave address in subsequent operations	Y
	SMBUS master timeout caused by the slave pulling down the SCL line may result in the SMBUS master failing to issue a STOP signal	Y

Note:

Y = Limitation present, workaround available

N = Limitation present, no workaround available

'--' = Limitation fixed

2. Descriptions of device limitations

2.1. PMU

2.1.1. When the chip is powered on, the PORRSTF flag probability cannot be set

Description & impact

When the chip is powered on, the PORRSTF flag probability cannot be set, making it impossible to use the PORRSTF flag to detect whether a power-on reset has occurred.

Workarounds

During the application initialization phase, the code should first check whether a specified backup domain register (such as RTC_BKP0) contains a marked value (such as 0xA5). If it does not contain the marked value, it is considered that a power-on reset has occurred, and the software should then write the marked value to the backup domain register. Otherwise, it is considered that a power-on reset has not occurred. This method requires occupying one backup domain register.

2.1.2. Frequent wake-up signal before and after MCU enters the standby mode results in wake-up failure in the standby mode

Description & impact

When there are frequent wake-up signals on the wake-up pin (WKUPx), if the MCU exits quickly (within 20us) after entering the standby mode, the internal signal CORE_POR_H cannot be set to 1 after reset, which results in the CPU cannot run, and finally the MCU cannot be woken up.

Note: When the above problem occurs, the external NRST reset also fails to make the CPU run again, and the chip needs to be repowered.

Workarounds

Not available. For the above application scenarios, it is not recommended to use standby mode, and it is recommended to use Deep-sleep / Deep-sleep1 mode instead.

2.1.3. When MCU enters Deep-sleep / Deep-sleep1 mode, if a system reset occurs, a very small number of chips may experience program runaway

Description & impact

When the MCU enters Deep-sleep / Deep-sleep1 mode (at this time, the EFLASH enters low-power mode or power-down mode), if a reset occurs, the EFLASH power-on timing parameter settings approach the limit (this restriction does not apply to other modes). This may cause abnormal loading of the first few bytes of EFLASH data in a very small number of chips, resulting in MCU program runaway.

Workarounds

Use one of the following solutions:

1) Avoid resetting the MCU in deep sleep mode/deep sleep mode 1 in the application (such as, NRST or FWDGT reset). Normal wake-up does not have this issue.

2) Enable the hardware watchdog through option bytes. If the issue occurs, the hardware watchdog can reset and recover the system.

Note: If you have any questions regarding this erratum, it is recommended to contact the original manufacturer's technical support for more detailed information.

2.2. RCU

2.2.1. The clock monitor of the HXTAL is abnormal

Description & impact

After the HXTAL clock monitor is enabled, when the HXTAL clock is lost, there is a probability that the NMI interrupt will not be generated. That is, the HXTAL clock monitor cannot reliably detect the loss of the HXTAL clock.

Workarounds

After enabling the HXTAL clock monitor function, the software polls the current clock source of the system to check whether a clock switch has occurred.

2.2.2. When OBRLD is set, both EPRSTF and OBLRSTF flags are set

Description & impact

A system reset caused by OBRLD being set will result in both EPRSTF and OBLRSTF flags being set simultaneously.

Workarounds

In the system reset source processing code snippet, when it is detected that both EPRSTF and OBLRSTF are set simultaneously, the software determines it as a system reset caused by option byte loading, and clears all reset flags after processing all reset source decisions.

2.3. GPIO

2.3.1. PA12 port LOCK function is abnormal

Description & impact

After enabling the LOCK function of PA12 port immediately after reset, the CTL12 bit field in the GPIOA_CTL register can still be configured, but the configuration result does not match expectations.

Workarounds

To use the LOCK function on PA12 port, software needs to first read the entire GPIOA_CTL register, write the read value back, and then use the LOCK function.

2.4. ADC

2.4.1. ADC data acquisition error occurs when the ADC clock is much slower than the PCLK clock

Description & impact

When the ADC clock is much slower than the PCLK clock, the ADC_RDATA register is read immediately after the EOC is set and a data acquisition error occurs.

Workarounds

When the delay between reading EOC flag and reading ADC_RDATA is no more than two ADC clocks, after the EOC flag is set, software needs to delay two ADC clocks before reading the ADC_RDATA register.

2.4.2. When the ADC operates in routine sequence mode, setting the ADCON bit again after enabling the ADC will start an ADC conversion

Description & impact

When the ADC operates in routine sequence mode, after enabling the ADC (ADCON = 1), setting the ADCON bit again will start an ADC conversion.

Workarounds

The patch has been added in GD32C2x1_Firmware_Library_V1.0.0 and later versions. Before enabling the ADC, check the ADCON bit once. Refer to the code below:

```
void adc_enable(void)
{
    if(0U == (ADC_CTL1 & ADC_CTL1_ADCON)) {
        /* enable ADC */
        ADC_CTL1 |= (uint32_t)ADC_CTL1_ADCON;
    }
}
```

2.4.3. When the ADC external trigger function is enabled and the trigger source switches from low level to high level, an ADC conversion will be triggered

Description & impact

When the ADC external trigger function is enabled and the trigger source switches from low level to high level, an ADC sampling will be triggered. For example, when the external trigger source function of the routine sequence is enabled (ETERC = 1), and the external trigger source switches from software trigger (0b111) to TIMER0_CH1 (0b001), an ADC conversion will be triggered. This issue imposes limitations on applications that disable the external trigger function by switching the external trigger source.

Workarounds

When disabling the ADC external trigger, directly disable the ADC external trigger function instead of switching the external trigger source.

2.5. SPI

2.5.1. When the SPI slave works in non-TI mode and the data valid sampling edge is the first clock transition edge and CRC function is enabled, if the slave is not selected by the chip and there is still a clock on the SCK line, which will cause the slave CRC to continue working

Description & impact

When SPI works in the slave non-TI mode (TMOD = 0) and the data effective sampling edge is the first clock transition edge (CKPH = 0), and the CRC function is enabled, if the slave is not selected by the chip at this time, but there is still a clock on the SCK line, the slave CRC will continue to work, and then CRCERR will be set. This issue imposes limitations on multi-slave (one-master, multiple-slave) applications

Workarounds

Use one of the following solutions:

1) Use software chip selection. When the slave detects that it is not selected, it actively disables the CRC functionality.
2) The master and slave agree that the effective data sampling edge is the second clock transition edge (CKPH =1).

2.6. I2S

2.6.1. The I2S MCK clock output function is abnormal

Description & impact

When the I2S master clock output function (MCKOEN = 1) is enabled, the MCK clock cannot be output properly, which makes it impossible to provide clock input to devices that require an additional master clock.

Workarounds

Emulating MCK clock outputs by using timers to generate PWM signals. For specific implementation details, refer to the <11_I2S_Audio_Player> example in "GD32C2x1_Demo_Suites\GD32C231C_EVAL_Demo_Suites".

2.7. I2C

2.7.1. When I2C works in 7/10 address slave mode, receiving an abnormal timing will cause the SDA line to stuck

Description & impact

When the I2C is operating as a slave device in 7-bit address mode and the I2C master simulates I2C communication via IO. If the master sends the following sequence, the I2C slave will enter an error state, causing it to malfunction and the SDA line to remain low:

Start + 10-bit Match Head Address + Start + 7-bit Address Read + Wait ACK + Start

When the I2C is operating as a slave device in 10-bit address mode and the I2C master simulates I2C communication via IO. If the master sends the following sequence, the I2C slave will enter an error state, causing it to malfunction and the SDA line to remain low:

Start + 10-bit Mismatch Head Address + Start
or
Start + 10-bit Match Head Address + Wait ACK + 10-bit Mismatch 8-bit Address + Start

Workarounds

Software periodically checks the status of the SDA line. If SDA is detected to be stuck low, reinitialize the I2C module.

2.7.2. When the I2C slave is configured in 10-bit address mode, if the external master does not send a STOP signal after transmitting a frame of data, the I2C slave will be unable to match the slave address in subsequent operations

Description & impact

When the I2C slave is configured in 10-bit address mode, if the external master does not send a STOP signal after transmitting a frame of data and instead sends a START signal to initiate the transmission of a second frame, the I2C slave will misinterpret the second byte of the slave address (the lower 8 bits of the 10-bit address) as data, and the address match flag (ADDSEND) will not be set. For example, if the slave is in address polling mode, it will continuously wait for an address match and remain stuck in a loop. Similarly, if the slave is in interrupt or DMA mode, it will fail to process subsequent data due to the inability to match the slave address.

Workarounds

When the I2C slave is operating in 10-bit address mode, the external I2C master must send the corresponding STOP signal at the end of each frame transmission.

2.7.3. SMBUS master timeout caused by the slave pulling down the SCL line may result in the SMBUS master failing to issue a STOP signal

Description & impact

When I2C acts as an SMBUS master, the timeout caused by the slave pulling down the SCL line may result in the SMBUS master failing to issue a STOP signal, which does not comply with SMBUS protocol requirements.

Workarounds

Not available.

3. Revision history

Table 3-1. Revision history

Revision No.	Description	Date
1.0	Initial Release	Jun.7 2025
1.1	Add limitations of PMU, refer to ***Frequent wake-up signal before and after MCU enters the standby mode results in wake-up failure in the standby mode***	Sep.22 2025
1.2	1) Add limitations of PMU, refer to ***When MCU enters Deep-sleep/Deep-sleep1 mode, if a system reset occurs, a very small number of chips may experience program runaway*** 2) Update the workarounds description of PMU limitation, refer to ***Frequent wake-up signal before and after MCU enters the standby mode results in wake-up failure in the standby mode*** 3) Add limitations of RCU, refer to ***When OBRLD is set, both EPRSTF and OBLRSTF flags are set*** 4) Update the description of GPIO limitation, refer to ***PA12 port LOCK function is abnormal*** 5) Add limitations of ADC, refer to ***When the ADC operates in routine sequence mode, setting the ADCON bit again after enabling the ADC will start an ADC conversion*** 6) Add limitations of ADC, refer to ***When the ADC external trigger function is enabled and the trigger source switches from low level to high level, an ADC conversion will be triggered*** 7) Add limitations of SPI, refer to ***When the SPI slave works in non-TI mode and the data valid sampling edge is the first clock transition edge and CRC function is enabled, if the slave is not selected by the chip and there is still a clock on the SCK line, which will cause the slave CRC to continue working*** 8) Add limitations of I2C, refer to ***When the I2C***	Nov.3 2025

	slave is configured in 10-bit address mode, if the external master does not send a STOP signal after transmitting a frame of data, the I2C slave will be unable to match the slave address in subsequent operations 9) Add limitations of I2C, refer to ***SMBUS master timeout caused by the slave pulling down the SCL line may result in the SMBUS master failing to issue a STOP signal***	

Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the “Product”), is owned by the Company according to the laws of the People’s Republic of China and other applicable laws. The Company reserves all rights under such laws and no Intellectual Property Rights are transferred (either wholly or partially) or licensed by the Company (either expressly or impliedly) herein. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

To the maximum extent permitted by applicable law, the Company makes no representations or warranties of any kind, express or implied, with regard to the merchantability and the fitness for a particular purpose of the Product, nor does the Company assume any liability arising out of the application or use of any Product. Any information provided in this document is provided only for reference purposes. It is the sole responsibility of the user of this document to determine whether the Product is suitable and fit for its applications and products planned, and properly design, program, and test the functionality and safety of its applications and products planned using the Product. The Product is designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only, and the Product is not designed or intended for use in (i) safety critical applications such as weapons systems, nuclear facilities, atomic energy controller, combustion controller, aeronautic or aerospace applications, traffic signal instruments, pollution control or hazardous substance management; (ii) life-support systems, other medical equipment or systems (including life support equipment and surgical implants); (iii) automotive applications or environments, including but not limited to applications for active and passive safety of automobiles (regardless of front market or aftermarket), for example, EPS, braking, ADAS (camera/fusion), EMS, TCU, BMS, BSG, TPMS, Airbag, Suspension, DMS, ICMS, Domain, ESC, DCDC, e-clutch, advanced-lighting, etc.. Automobile herein means a vehicle propelled by a self-contained motor, engine or the like, such as, without limitation, cars, trucks, motorcycles, electric cars, and other transportation devices; and/or (iv) other uses where the failure of the device or the Product can reasonably be expected to result in personal injury, death, or severe property or environmental damage (collectively "Unintended Uses"). Customers shall take any and all actions to ensure the Product meets the applicable laws and regulations. The Company is not liable for, in whole or in part, and customers shall hereby release the Company as well as its suppliers and/or distributors from, any claim, damage, or other liability arising from or related to all Unintended Uses of the Product. Customers shall indemnify and hold the Company, and its officers, employees, subsidiaries, affiliates as well as its suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Product.

Information in this document is provided solely in connection with the Product. The Company reserves the right to make changes, corrections, modifications or improvements to this document and the Product described herein at any time without notice. The Company shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. Information in this document supersedes and replaces information previously supplied in any prior versions of this document.