GigaDevice Semiconductor Inc.

Device limitations of GD32L233

Errata Sheet

Revision 1.5

(Apr. 2026)

Table of Contents

List of Figures

List of Tables

1. Introduction

This document applies to GD32L233 product series, as shown in ***Table 1-1. Applicable products***. It provides the technical details that need to be paid attention to in the process of using GD32 MCU, as well as solutions to related problems.

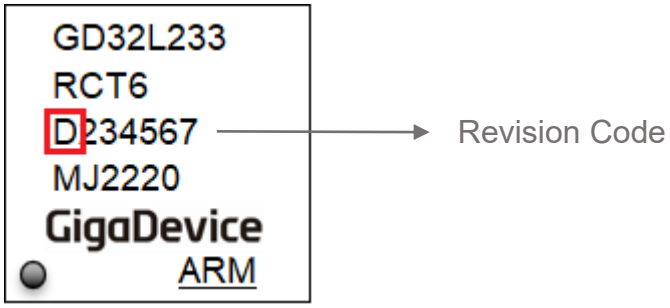
Table 1-1. Applicable products

Type	Part Numbers
MCU	GD32L233xx series

1.1. Revision identification

The device revision can be determined by the mark on the top of the package. The 1st code on the line 3 of the mark represents product revision code. As the picture shown in ***Figure 1-1. Device revision code of GD32L233***.

Figure 1-1. Device revision code of GD32L233



1.2. Summary of device limitations

The device limitations of GD32L233 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	Rev. Code C	Rev. Code D	Rev. Code E
PMU	*FWDGTRSTF flag cannot be set in Deep-sleep mode*	Y	Y	Y	Y
RCU	*The LXTALSTB bit cannot be cleared by disabling LXTAL when LXTAL stops unexpectedly*	Y	Y	Y	Y
ADC	*ADC data acquisition error occurs when the ADC clock is equal to or less than 1/4 of its APB bus clock*	Y	Y	Y	Y

Module	Limitations	Workaround			
		Rev. Code B	Rev. Code C	Rev. Code D	Rev. Code E
DAC	*In the DAC noise mode, when the configuration (DH + DWBW) value exceeds 4095, the DAC output voltage will have an abnormal break point*	Y	Y	Y	Y
USART	*When USART is woken up from mute mode by an idle frame, it will not be woken up when it enters mute mode again*	Y	Y	Y	Y
	IDLEF is set when waking up USART from mute mode via an idle frame	Y	Y	Y	Y
	When DENR = 1, DDRE = 0, and HCM = 1, RTS remains asserted high	N	N	N	N
I2C	*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	Y	Y	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	N	N	N	N
	When the I2C is configured as a master in 10-bit address mode and fails to send a STOP signal after transmitting a data frame, subsequent data frame transmissions will encounter anomalies	Y	Y	Y	Y
	Transmission timing abnormality when the I2C master is configured in 10-bit address reception mode with HEAD10R=1	Y	Y	Y	Y
	When I2C works in 7/10 address slave mode, receiving an abnormal timing will cause the SDA line to be stuck	Y	Y	Y	Y
	When I2C is operating as a master transmitter, if the slave responds with NACK to the last byte, a START condition cannot be correctly issued in the transfer complete interrupt	Y	Y	Y	Y
I2S	*When I2S pins are configured as open-drain with an external 1.8V pull-up, using MSB-aligned mode causes abnormal data reception on the slave*	Y	Y	Y	Y
SLCD	*Do not support the use of internal voltage source*	Y	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. FWDGTRSTF flag cannot be set in Deep-sleep mode

Description & impact

FWDGTRSTF bit cannot be set by hardware when MCU is in Deep-sleep / Deep-sleep 1 / Deep-sleep 2 mode and FWDGT reset is occurred.

Workarounds

The application programme can determine whether a FWDGT reset has occurred. For example, by marking whether the system has experienced a reset, and then excluding the cause of the reset, it can be determined if it was due to a FWDGT reset.

2.2. RCU

2.2.1. The LXTALSTB bit cannot be cleared by disabling LXTAL when LXTAL stops unexpectedly

Description & impact

When LXTAL stops unexpectedly, the LXTALSTB bit cannot be cleared by disabling the LXTAL, which prevents the LXTAL from restarting.

Workarounds

By repeatedly setting and resetting the LXTALBPS more than ten times to clear the LXTALSTB bit, and then reconfiguring the LXTAL. The reference code for clearing LXTALSTB bits is as follows:

```
void lxtal_stb_clear(void)
{
    volatile  uint32_t i = 0U;
    /* close LXTAL clock */
    rcu_osci_off(RCU_LXTAL);
    /* set PC14 */
    rcu_periph_clock_enable(RCU_GPIOC);
    gpio_mode_set(GPIOC, GPIO_MODE_OUTPUT, GPIO_PUPD_NONE,
GPIO_PIN_14);
    gpio_output_options_set(GPIOC, GPIO_OTYPE_PP, GPIO_OSPEED_50MHZ,
GPIO_PIN_14);
```

```
    GPIO_BOP(GPIOC) = GPIO_PIN_14;
    for(i = 0; i < 10; i++) {
        /* enable the LXTAL bypass mode */
        rcu_osci_bypass_mode_enable(RCU_LXTAL);
        /* disable the LXTAL bypass mode */
        rcu_osci_bypass_mode_disable(RCU_LXTAL);
    }
}
```

2.3. ADC

2.3.1. ADC data acquisition error occurs when the ADC clock is equal to or less than 1/4 of its APB bus clock

Description & impact

when the ADC clock is equal to or less than 1/4 of its APB bus 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. DAC

2.4.1. In the DAC noise mode, when the configuration (DH + DWBW) value exceeds 4095, the DAC output voltage will have an abnormal break point

Description & impact

When the DAC is configured in noise mode and the DAC output value is set to a large value, the superimposed value (DH+DWBW) will exceed the maximum value of 4095, resulting in an abnormal break point of zero voltage in the DAC output signal.

Workarounds

When the DAC output value and the noise wave peak value are configured, avoid the superimposed value (DH+DWBW) overflow.

2.5. USART

2.5.1. When USART is woken up from mute mode by an idle frame, it will not be woken up when it enters mute mode again

Description & impact

When USART works in multiprocessor communication mode and the USART is woken from mute mode by an idle frame, it will cause the USART will not be woken up when the bus is in idle mode and the USART enters mute mode.

Workarounds

When an idle frame is used to wake the USART mute mode, it is not allowed to enter mute mode while the bus is idle.

2.5.2. IDLEF is set when waking up USART from mute mode via an idle frame

Description & impact

When waking up USART from mute mode via an idle frame, IDLEF is set. If the IDLE interrupt is enabled at this time, the system will enter the IDLE interrupt handler after waking up via the idle frame.

Workarounds

Disable the IDLE interrupt before entering mute mode, and enable the IDLE interrupt when needed.

2.5.3. When DENR = 1, DDRE = 0, and HCM = 1, RTS remains asserted high

Description & impact

When DENR = 1, DDRE = 0, and HCM = 1, the RTS signal remains asserted high, causing hardware flow control to fail.

Workarounds

Ensure that the above three conditions are not all true at the same time during operation.

2.6. I2C

2.6.1. 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.

2.6.2. 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.

2.6.3. When the I2C is configured as a master in 10-bit address mode and fails to send a STOP signal after transmitting a data frame, subsequent data frame transmissions will encounter anomalies

Description & impact

When the I2C master fails to send a STOP signal after transmitting a data frame and the

software subsequently configures it to master receive mode, regardless of whether HEAD10R is set to 0 or 1, the waveform of the master receive part will always be RESTART + 10-bit address head + Master Receive. This means the HEAD10R configuration becomes ineffective.

If the I2C master fails to send a STOP signal after transmitting a data frame, when HEAD10R = 0, the subsequent RESTART will directly enter the waveform of master receive mode. when HEAD10R = 1, the subsequent RESTART will repeatedly send the first part of the address sequence in a loop.

Workarounds

When the I2C master is configured in 10-bit address mode, a corresponding STOP signal must be sent at the end of each frame transmission.

2.6.4. Transmission timing abnormality when the I2C master is configured in 10-bit address reception mode with HEAD10R=1

Description & impact

When the I2C master is configured in 10-bit address reception mode with HEAD10R=1, the I2C master timing sequence is START + 10-bit address head + Master Receive, which causes the slave device to not ACK and fails to address the slave device. Under this configuration, a normal master transmission sequence should be START + 10-bit address head (write) + second address byte + RESTART + 10-bit address head (read).

Workarounds

When the master needs to send the sequence START + 10-bit address head (write) + second address byte + RESTART + 10-bit address head (read), configure HEAD10R to 0.

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

2.6.6. When I2C is operating as a master transmitter, if the slave responds with NACK to the last byte, a START condition cannot be correctly issued in the transfer complete interrupt

Description & impact

When I2C is operating as a master and has finished transmitting the last byte of data, if the slave responds with a NACK signal, the master cannot correctly issue a START condition within the transfer complete (TC) interrupt, meaning the next transfer cannot be initiated.

Workarounds

Send a STOP condition in the NACK interrupt handler first, then initiate the next transfer.

2.7. I2S

2.7.1. When I2S pins are configured as open-drain with an external 1.8V pull-up, using MSB-aligned mode causes abnormal data reception on the slave

Description & impact

When the I2S pins are configured as open-drain with an external 1.8V pull-up, and I2S is operating as a master transmitter with the audio standard set to MSB-aligned mode, there is approximately one clock cycle of phase delay between SD and SCK after the WS pin is pulled high. This causes a mismatch between the data transmission and the clock, resulting in the slave device failing to receive data correctly.

Workarounds

Use the I2S Philips standard as the audio standard for both the I2S master and slave to avoid the issue described above.

2.8. SLCD

2.8.1. Do not support the use of internal voltage source

Description & impact

SLCD only supports the use of external voltage source but internal voltage source.

Workarounds

Use the external voltage source.

2.9. Core

About Cortex-M23 limitations, please refer to “Cortex-M23 Software Developer Errata Notice”. This document can be downloaded on ARM official website.

3. Revision history

Table 3-1. Revision history

Revision No.	Description	Date
1.0	Initial Release	Nov.11 2022
1.1	Add limitations of Rev. Code D	Mar.28 2023
1.2	Update note of chapter 1.2	Apr.4 2023
1.3	1. Update the description of PMU limitation, refer to ***FWDGTRSTF flag cannot be set in Deep-sleep mode*** 2. Add the RCU limitation, refer to ***The LXTALSTB bit cannot be cleared by disabling LXTAL when LXTAL stops unexpectedly*** 3. Add the I2C limitation, refer to ***When SDA line interference causes garbled data on the I2C bus, it can lead to a stuck in the I2C slave device*** 4. Add limitations of Rev. Code E	Sep.23 2024
1.4	1. Update the ***Summary of device limitations*** 2. Update the RCU limitation, refer to ***The LXTALSTB bit cannot be cleared by disabling LXTAL when LXTAL stops unexpectedly***	Feb.21 2025
1.5	1. Update the description of I2C limitation, refer to ***When I2C works in 7/10 address slave mode, receiving an abnormal timing will cause the SDA line to be stuck*** 2. Update the Workarounds of RCU limitation, refer to ***The LXTALSTB bit cannot be cleared by disabling LXTAL when LXTAL stops unexpectedly*** 3. Add the ADC limitation, refer to ***ADC data acquisition error occurs when the ADC clock is equal to or less than 1/4 of its APB bus clock*** 4. Add the DAC limitation, refer to ***In the DAC noise mode, when the configuration (DH + DWBW) value exceeds 4095, the DAC output voltage will have an abnormal break point*** 5. Add the USART limitation, refer to ***When***	Apr.30 2026

		USART is woken up from mute mode by an idle frame, it will not be woken up when it enters mute mode again and ***IDLEF is set when waking up USART from mute mode via an idle frame*** and ***When DENR = 1, DDRE = 0, and HCM = 1, RTS remains asserted high*** 6. Add the I2C limitation, refer to ***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*** and ***SMBUS master timeout caused by the slave pulling down the SCL line may result in the SMBUS master failing to issue a STOP signal*** and ***When the I2C is configured as a master in 10-bit address mode*** and ***fails to send a STOP signal after transmitting a data frame, subsequent data frame transmissions will encounter anomalies*** and ***Transmission timing abnormality when the I2C master is configured in 10-bit address reception mode with HEAD10R=1*** and ***When I2C is operating as a master transmitter, if the slave responds with NACK to the last byte, a START condition cannot be correctly issued in the transfer complete interrupt*** 7. Add the I2S limitation, refer to ***When I2S pins are configured as open-drain with an external 1.8V pull-up, using MSB-aligned mode causes abnormal data reception on the slave***	

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.