GigaDevice Semiconductor Inc.

Device limitations of GD32F3x0

Errata Sheet

Table of Contents

List of Figure

List of Tables

1. Introduction

This document applies to GD32F3x0 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.

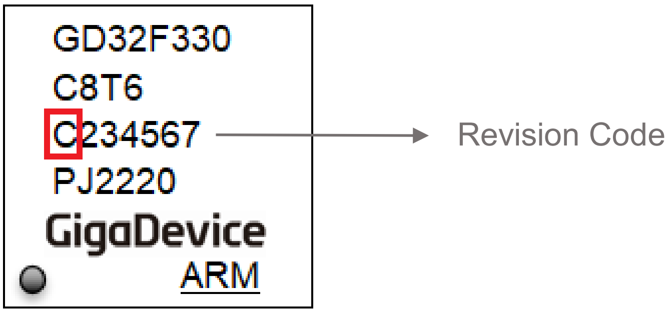
Table 1-1. Applicable products

Type	Part Numbers
MCU	GD32F310xx series
	GD32F330xx series
	GD32F350xx 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 GD32F3x0*** .

Figure 1-1. Device revision code of GD32F3x0



1.2. Summary of device limitations

The device limitations of GD32F3x0 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 C	Rev. Code F	Rev. Code F
FMC	***When writing a half-word program to a non-zero wait area, writing the high address first and then the low address will cause PGERR to be set***	Y	Y	Y
PMU	***MCU cannot be waked up when an interrupt occurs before entering deep-sleep***	Y	Y	Y
	Standby mode cannot be entered normally	Y	Y	Y

	Standby mode cannot be waked up due to frequent wakeup signals before or after entering standby mode	N	N	N
RCU	The LXTALSTB bit cannot be cleared by disabling LXTAL when LXTAL stops unexpectedly	Y	Y	Y
	MCU cannot be waked up after entering deep-sleep mode when DSLP_HOLD bit is set	Y	--	--
ADC	ADC data acquisition error occurs when the ADC clock is much slower than the PCLK clock	Y	Y	Y
DAC	In the DAC noise mode, when the DH value is configured too high, the DAC output voltage will have an abnormal break point	Y	Y	Y
TIMER	When the BRKP bit is set to 0, if the CMP output signal is used to trigger the braking of TIMER0, then the brake of TIMER0 will remain continuously active	Y	Y	Y
USART	The high baud rate of USART will cause data loss when using hardware flow control mode	N	--	--
	In deep-sleep mode, the parity error caused by wakeup frames will set PERR bit	N	--	--
	When the USART is configured in half-duplex single-wire mode and the TX / RX pin functions are swapped, the transmission and reception data polarity is inconsistent	N	--	--
	In mute mode, when noise errors, frame errors, and parity errors occur with the wake-up address, the corresponding error flag bits will be set to 1	N	--	--
	Frame errors are only detected on the first stop bit when the stop bit is configured as 2 bits	N	--	--
I2C	Read one more data because the BTC flag was not cleared	Y	--	--
	Due to the timing difference of START signal between I2C0 and I2C1, the master that wins the arbitration cannot receive ACK, and the bus is blocked	N	--	--
USBFS	When USBFS-OTG is communicating with low-speed devices through the HUB, the data transmission is abnormal	N	--	--
Core	VDIV or VSQRT instructions might not complete correctly when very short ISRs are used	Y	Y	Y

Note:

Y = Limitation present, workaround available

N = Limitation present, no workaround available

'--' = Limitation fixed

2. Descriptions of device limitations

2.1. FMC

2.1.1. When writing a half-word program to a non-zero wait area, writing the high address first and then the low address will cause PGERR to be set

Description & impact

When performing half-word programming in a non-zero wait state region, if the half-word data is first written to the lower half-word address of the higher word address, then subsequent half-word programming to the higher half-word address of the adjacent lower word address will cause the PGERR bit to be set. For example, if a half-word 0xA55A is first written to 0x08010008, then subsequent half-word writing to 0x08010006 will cause the PGERR bit to be set.

Workarounds

Use word programming instead of half-word programming in the non-zero wait area.

2.2. PMU

2.2.1. MCU cannot be waked up when an interrupt occurs before entering deep-sleep

Description & impact

When system application programme exists other interrupt code(such as systick / timer period interrupt) and needs to enter deep-sleep mode through WFE or WFI instruction, the system will exist the risk of not being wakedup after running for some time.

Workarounds

Application programme needs to mask all interrupts except EXTI for wakeup source before entering deep-sleep mode, then unmask the needed interrupts after wakeup.The latest firmware library has circumvented this problem.

2.2.2. Standby mode cannot be entered normally MCU

Description & impact

When system application programme exists other interrupt code(such as systick 1us period interrupt) and needs to enter standby mode through WFI instruction, the system cannot enter the standby mode normally.

Workarounds

Application programme needs to mask all interrupts except wakeup source before entering standby mode, then unmask the needed interrupts after wakeup.

2.2.3. Standby mode cannot be waked up due to frequent wakeup signals before or after entering standby mode

Description & impact

When reset the internal signal STBY_CTL to enter to standby mode, if the Tglitch is smaller than 100ns, which will cause the mcu cannot be waked up. The narrow glitch will result in incorrect Vcore voltage.

Note: The T_{glitch} is the time between STBY_CTL low level and the wakeup signal (PA0 high level).

Workarounds

Not available.

2.3. RCU

2.3.1. 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);
    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);
    }
}
```

2.3.2. MCU cannot be waked up after entering deep-sleep mode when DSLP_HOLD bit is set

Description & impact

When DSLP_HOLD bit is set and debug the mcu in deep-sleep mode, the mcu will not be waked up.

Workarounds

When the DSLP_HOLD bit is set to enable low power debugging, the application programme need switch the system clock to IRC8M before entering the deep-sleep mode.

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, reading the ADC_RDATA register immediately after the EOC is set will result in a data acquisition error.

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 need to delay two ADC clocks before reading the ADC_RDATA register.

2.5. DAC

2.5.1. In the DAC noise mode, when the DH value is configured too high, 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.

2.6. TIMER

2.6.1. When the BRKP bit is set to 0, if the CMP output signal is used to trigger the braking of TIMER0, then the brake of TIMER0 will remain continuously active

Description & impact

When BRKP is configured to be effective at a low level, if the CMP output signal is used to trigger the braking of TIMER0, then the brake of TIMER0 will remain continuously active.

Workarounds

Configure BRKP to 1 and use a high level instead of a low level to trigger the brake.

2.7. USART

2.7.1. The high baud rate of USART will cause data loss when using hardware flow control mode

Description & impact

In the case of high baud rate of USART, when using the hardware flow control mode, the transmitter does not detect the RTS level in time due to the timing lag of the receiver RTS pulling up, resulting in one more data sent by the USART transmitter.

Workarounds

Not available.

2.7.2. In deep-sleep mode, the parity error caused by wakeup frames will set PERR bit

Description & impact

In deep-sleep mode, a parity error caused by a wakeup frame will result in a parity error (the PERR bit in the USART_STAT register is set).

Workarounds

The software ignores the parity error flag generated in this case.

2.7.3. When the USART is configured in half-duplex single-wire mode and the TX / RX pin functions are swapped, the transmission and reception data polarity is inconsistent

Description & impact

When the USART is configured in half-duplex single-wire mode and the functions of the TX/RX pins are swapped, if the TINV and RINV configurations are inconsistent, which will cause the polarity of the transmitted and received data to be inconsistent.

Workarounds

Configure TINV and RINV with the same settings.

2.7.4. In mute mode, when noise errors, frame errors, and parity errors occur with the wake-up address, the corresponding error flag bits will be set to 1

Description & impact

In mute mode, when noise errors, frame errors, and parity errors occur with the wake-up address, the corresponding error flag bits will be set to 1.

Workarounds

The software ignores the error flag generated in this case.

2.7.5. Frame errors are only detected on the first stop bit when the stop bit is configured as 2 bits

Description & impact

Frame errors are only detected on the first stop bit when the stop bit is configured as 2 bits.

Workarounds

Do not set the stop bit length to 2 bits, or use software methods to verify the frame data.

2.8. I2C

2.8.1. Read one more data because the BTC flag was not cleared

Description & impact

If an interrupt occurs before reading I2C_DATA register when RBNE flag is set and BTC flag is reset, I2C will read an additional data if BTC flag is set during the interrupt processing because the read data operation cannot clear the BTC flag.

Workarounds

1) Using interrupt method to read the I2C_DATA register (need higher interrupt priority).
2) Using DMA method to read the I2C_DATA register (recommend).

2.8.2. Due to the timing difference of START signal between I2C0 and I2C1, the master that wins the arbitration cannot receive ACK, and the bus is blocked

Description & impact

The timing of the START signal of I2C0 and I2C1 is different, which leads to the misplaced SCL clock signal sent out, and the clock sent out in the address transmission phase is more than 8 clock signals. As a result, the master that wins the arbitration cannot receive ACK, so the SCL signal line is pulled down and the I2C bus is stuck.

Workarounds

Not available.

2.9. USBFS

2.9.1. When USBFS-OTG is communicating with low-speed devices through the HUB, the data transmission is abnormal

Description & impact

When USBFS-OTG is communicating with low-speed devices through the HUB, the data transmission is abnormal.

Workarounds

Not available.

2.10. Core

2.10.1. VDIV or VSQRT instructions might not complete correctly when very short ISRs are used

This limitation refers to Arm ID number 776924 in “Cortex-M4 & Cortex-M4 with FPU Software Developers Errata Notice”.

Description & impact

The VDIV and VSQRT instructions take 14 cycles to execute. When an interrupt is taken a VDIV or VSQRT instruction is not terminated, and completes its execution while the interrupt stacking occurs. If lazy context save of floating point state is enabled then the automatic stacking of the floating point context does not occur until a floating point instruction is executed inside the interrupt service routine.

Lazy context save is enabled by default. When it is enabled, the minimum time for the first instruction in the interrupt service routine to start executing is 12 cycles. In certain timing conditions, and if there is only one or two instructions inside the interrupt service routine, then the VDIV or VSQRT instruction might not write its result to the register bank or to the FPSCR.

The failure occurring conditions are as follows:

1) The floating point unit is enabled.
2) Lazy context saving is not disabled.
3) A VDIV or VSQRT is executed.
4) The destination register for the VDIV or VSQRT is one of s0 - s15.
5) An interrupt occurs and is taken.
6) The interrupt service routine being executed does not contain a floating point instruction.
7) Within 14 cycles after the VDIV or VSQRT is executed, an interrupt return is executed.

A minimum of 12 of these 14 cycles are utilized for the context state stacking, which leaves 2 cycles for instructions inside the interrupt service routine, or 2 wait states applied to the entire stacking sequence (which means that it is not a constant wait state for every access).

In general, this means that if the memory system inserts wait states for stack transactions then this erratum cannot be observed.

The implications of this limitation is that the VDIV or VQSRT instruction does not complete correctly and the register bank and FPSCR are not updated, which means that these registers hold incorrect, out of date, data.

Workarounds

A workaround is only required if the floating point unit is enabled. A workaround is not required if the stack is in external memory.

There are two possible workarounds:

1) Disable lazy context save of floating point state by clearing LSPEN to 0 (bit 30 of the FPCCR at address 0xE000EF34).
2) Ensure that every interrupt service routine contains more than 2 instructions in addition to the exception return instruction.

3. Revision history

Table 3-1. Revision history

Revision No.	Description	Date
1.0	Initial Release	Aug.30 2022
1.1	Update note of chapter 1.2	Apr.4 2023
1.2	1. Add PMU limitation, referring to chapter 2.1.3 2. Add core limitation, referring to chapter 2.12.1	Nov.2 2023
1.3	1. Add FMC limitations, refer to ***When writing a half-word program to a non-zero wait area, writing the high address first and then the low address will cause PGERR to be set***. 2. Add RCU limitations, refer to ***The LXTALSTB bit cannot be cleared by disabling LXTAL when LXTAL stops unexpectedly*** and ***MCU cannot be waked up after entering deep-sleep mode when DSLP_HOLD bit is set*** 3. Add ADC limitations, refert to ***ADC data acquisition error occurs when the ADC clock is much slower than the PCLK clock*** 4. Add DAC limitations, refer to ***In the DAC noise mode, when the DH value is configured too high, the DAC output voltage will have an abnormal break point*** 5. Add ***USART*** limitations 6. Add USBFS limitations, refer to ***When USBFS-OTG is communicating with low-speed devices through the HUB, the data transmission is abnormal*** 7. Add limitations of Rev. Code F	Aug.12 2024

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 under the intellectual property laws and treaties of the People’s Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. 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.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do release the Company as well as it’s suppliers and/or distributors from any claim, damage, or other liability arising from or related to all Unintended Uses of the Products. Customers shall indemnify and hold the Company as well as it’s 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 Products.

Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and Products and services described herein at any time, without notice.