

GigaDevice Semiconductor Inc.

GD32A50x_Firmware_Library

Release Note

(2025 年 2 月)

(Feb. 2025)

目录 Table of Contents

目录 Table of Contents	1
1. 发布内容 Publishing content	2
1.1. 代码 Code	2
1.2. 文档 Document	2
2. 版本信息 Version information	2
3. 发布版本功能描述 Release feature description	2
4. 发布版本变更列表 Release the changelist	5
5. 发布版本已知问题勘误 Known errata in the release version	8

1. 发布内容 Publishing content

1.1. 代码 Code

- GD32A50x Standard Peripheral Firmware
- GD32A50x Peripheral Examples
- GD32A50x Utilities
- GD32A50x Template

1.2. 文档 Document

- 《GD32A50x_固件库使用指南》
- 《GD32A50x_Firmware_Library_User_Guide》

2. 版本信息 Version information

名称Name	GD32A50x_Firmware_Library.7z
版本Version	V1.5.0
发布日期Release date	2025.02.06
支持的开发板Supported boards	GD32A503V-EVAL-V1.1
编译环境Compiling environment	<ul style="list-style-type: none">■ IAR Embedded Workbench 8.32.1■ ARM Keil 5.29

3. 发布版本功能描述 Release feature description

类型Type	例程名称Demo Name	备注Description
Examples- ADC	ADC0_ADC1_regular_parallel	-
	ADC0_analog_watchdog_0	-
	ADC0_analog_watchdog_1	-
	ADC0_oversample_shift	-
	ADC0_regular_channel_discontinuous_mode	-
	ADC0_regular_channel_with_DMA	-
	ADC0_resolution	-
	ADC0_software_trigger_regular_channel_polling	-

	ADC0_temperature_Vref	-
Examples-BKP	Tamper	-
Examples-CAN	communication_classical_CAN	-
	communication_FDmode	展示如何使用CAN FD帧。 Demonstrate how to use CAN FD frames.
	communication_Loopback	-
	Pretended_Networking_mode	-
Examples-CMP	Blanking_output	-
	Interrupt	
	Port_output	-
	Timer0_CH0IC	-
Examples-CRC	CRC_calculate	-
Examples-DAC	DAC_ADC_convert	-
	DAC_output_voltage	-
	DAC_DMA_convert	-
	DAC_LFSR_noise_mode	-
	DAC_TRGISEL_trigger	通过TRGISEL选配触发信号，触发DAC。 Trigger the DAC by selecting the trigger signal through TRGISEL.
	DAC_triangle_noise_mode	-
Examples-DBG	DBG_timer1_stop	-
Examples-DMA	DMA_RequestGen	-
	DMA_SYNC	-
	RAM_TO_RAM	-
	RAM_TO_USART	-
	Reload_exti	-
Examples-EXTI	Key_external_interrupt_mode	-
Examples-FMC	Data_flash_erase_program	展示对Data flash进行擦除和编程，包括正常编程和快速编程操作。 Demonstrate the erasing and programming of Data flash, including normal programming and fast programming operations.
	Erase_Program	-
	Write_protection	-
Examples-FWDGT	FWDGT_key	-
Examples-GPIO	Keyboard_polling_mode	-
	Running_led	-

Examples-I2C	I2C_EEPROM	-
	I2C_EEPROM_dma	-
	I2C_EEPROM_interrupt	-
	Master_receiver	-
	Master_receiver&slave_transmitter	-
	Master_receiver&slave_transmitter_interrupt	-
	Master_transmitter	-
	Master_transmitter&slave_receiver	-
	Master_transmitter&slave_receiver_dma	-
	Master_transmitter&slave_receiver_interrupt	-
	Slave_receiver	-
	Slave_transmitter	-
Examples-MFCOM	MFCOM_I2C	-
	MFCOM_I2S	-
	MFCOM_SPI	-
	MFCOM_UART	-
Examples-PMU	Deepsleep_wakeup_exti	-
	Deepsleep_wakeup_RTC	-
	Low_voltage_detector	-
	Over_voltage_detector	-
	Standby_wakeup_pin	-
	Standby_wakeup_RTC	-
Examples-RCU	Ckout_pin_clock_output	-
	Reset_source_detect	-
	System_clock_switch	-
Examples-RTC	Calendar_demo	-
Examples-SPI	I2S_master_transmit_dma	-
	I2S_slave_recieve_dma	-
	SPI_master_slave_fullduplex_dma	-
	SPI_master_slave_fullduplex_nssp_mode	-
	SPI_master_slave_fullduplex_polling	-
	SPI_master_slave_fullduplex_ti_mode	-
	SPI_master_slave_simplex_dma	-
	SPI_master_transmit_slave_receive_interrupt	-
Examples-TIMER	TIMER0_6-steps	展示如何配置TIMER0外设以生成带有死区时间的三个互补信号。 Demonstrate how to configure the TIMER0 peripheral to generate three complementary signals with dead time.
	TIMER0_deadtime_break	-
	TIMER0_dma	-

	TIMER0_dma_burst	-
	TIMER0_pwmout_complementarysignals	-
	TIMER0_pwmout_independtsignals	-
	TIMER1_extclock_count	-
	TIMER1_exttrigger	-
	TIMER1_inputcapture	-
	TIMER1_ocactive	-
	TIMER1_ocinactive	-
	TIMER1_octoggle	-
	TIMER1_pwminputcapture	-
	TIMER1_pwmout	-
	TIMER1_singlepulse	-
	TIMER1_timebase	-
	TIMERs_cascadesynchro	-
	TIMERs_parallelsynchro	-
Examples- TRIGSEL	cmp_trigger_extout	展示如何使用CMP和TRIGSEL输出CMP0_OUT。TRIGSEL选择CMP0_OUT作为触发源，然后CMP0_OUT输出到TRIGSEL EXTOUT0。 Demonstrate how to use CMP and TRIGSEL to output CMP0_OUT. TRIGSEL selects CMP0_OUT as the trigger source, and then CMP0_OUT is output to TRIGSEL EXTOUT0.
	extinput_trigger_timer1	-
	Timer0_pwm_trigger_adc	-
Examples- USART	DMA_transmitter&receiver	-
	DMA_transmitter&receiver_interrupt	-
	Half_duplex_transmitter&receiver	-
	IDLE_receiver_interrupt	-
	Printf	-
	Receiver_timeout	-
	Synchronous	-
	Transmitter&receiver_interrupt	-
Examples- WWDGT	WWDGT_delay_feed	-

4. 发布版本变更列表 Release the changelist

序号 Serial number	模块 Module	接口名/示例名 Interface name/Example name	变更原因 Reason of change	变更内容 Content of change	变更版本 Change version
------------------------	--------------	--	--------------------------	---------------------------	---------------------------

1	System	void SystemInit(void) / static void system_clock_24m_ pll_irc8m(void) / static void system_clock_48m_ pll_irc8m(void) / static void system_clock_72m_ pll_irc8m(void) / static void system_clock_100m_ pll_irc8m(void) / static void system_clock_24m_ pll_hxtal(void) / static void system_clock_48m_ pll_hxtal(void) / static void system_clock_72m_ pll_hxtal(void) / static void system_clock_100m_ pll_hxtal(void)	To enhance the robustness of the MCU operation, switch frequencies in a stepwise manner during frequency switching.	Add a three-step frequency switching function in the system_gd32a50x.c file for both increasing and decreasing frequency stages.	V1.4.0
2	FMC	-	Remove the EEPROM function in the GD32A50x series.	Remove all API interfaces and function macros related to EPROM functions from the FMC standard library.	V1.4.0
3	USART	void usart_baudrate_set(uint32_t usart_periph, uint32_t baudval)	Improve the robustness of the API interface.	The API interface should add a judgment for the baud rate parameter being 0 to prevent the application layer from setting the baud rate parameter to 0, which would cause a divide-by-zero error.	V1.4.0
4	SYSCFG	-	Remove the boot from SRAM feature for version B of MCU, only reflect it for version E.	In the syscfg_bootmode_get function for retrieving the boot status, remove the boot from SRAM status.	V1.4.0
6	CRC	uint32_t crc_block_data_calculate(void *array, uint32_t size, uint8_t data_format)	Resolve the issue of API interfaces violating the MISRA C 2004 rule 17.4.	Modify the internal implementation of the API interface to avoid using array indexing on pointers.	V1.4.0

7	CAN-Example	-	The use of the CAN module clock source is restricted and should be referred to the errata.of GD32A50x series.	The CAN clock source defaults to PCLK2	V1.4.0
8	FMC	fmc_state_enum ob1_parameter_config(uint32_t dflash_size)	The 64 KB flash capacity model is not supported.	Remove the macro for a data flash size of 16K (OB1CS_DF_16K).	V1.5.0
9	System	void SystemInit(void)	To improve the robustness of MCU during frequency switching.	Add a software delay when the MCU switches to the IRC8M clock source.	V1.5.0
10	System	static void system_clock_24m_pll_irc8m(void) / static void system_clock_48m_pll_irc8m(void) / static void system_clock_72m_pll_irc8m(void) / static void system_clock_100m_pll_irc8m(void) / static void system_clock_24m_pll_hxtal(void) / static void system_clock_48m_pll_hxtal(void) / static void system_clock_72m_pll_hxtal(void) / static void system_clock_100m_pll_hxtal(void)	To improve the robustness of MCU during frequency switching.	Use the new read-modify-write method to configure registers for clock source switching.	V1.5.0
11	misc	void nvic_irq_enable(IRQn_Type nvic_irq, uint8_t nvic_irq_pre_priority	-	1. Change the parameter type of nvic_irq to the IRQn_Type enumeration. 2. Add a check for the range of the nvic_irq parameter within the function.	V1.5.0

		, uint8_t nvic_irq_sub_priority)			
12	CAN- Example	communication_FD mode	1. The transmission delay compensation value exceeds the configurable range of the register. 2. Transmission delay compensation should be determined based on the CAN transceiver used and the baud rate of the CANFD frames.	In the example, disable the transmission delay compensation feature for CANFD frames.	V1.5.0
13	RCU- Example	System_clock_switch	To enhance the robustness of the MCU during frequency switching, demonstrate how to correctly switch the clock.	1. Use the new read-modify-write method to configure registers for clock source switching. 2. Add a three-step frequency switching function in both increasing and decreasing frequency stages.	V1.5.0

5. 发布版本已知问题勘误 Known errata in the release version

序号 Serial number	模块 Module	BUG 描述 BUG description	规避方式 Method of evasion

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 its 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 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 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.