User Manual

G32F031x8

G32M3122x8

G32M3114x8

G32M3115x8

Arm® Cortex® -M0+ core-based 32-bit MCU

Version: V1.3

Contents

1 Introduction and Document Description Rules

1.1 Introduction

This reference manual provides application developers with all the information about how to use MCU system architecture, memory and peripherals.

For information about Arm® Cortex®-M0+ core, please refer to Arm® Cortex®-M0+ technical reference manual; please refer to the corresponding datasheet for detailed data such as model information, dimensions and electrical characteristics of the device; for all MCU series models, please refer to the corresponding data manual for memory mapping, peripheral existence and their number.

Note that: Zhuhai Geehy Semiconductor Co., Ltd. is hereinafter referred to as "Geehy".

1.2 Document description rules

1.2.1 "Register functional description" rules

(1) Control (CTRL) registers are all "set to 1 and cleared to 0 by software", unless otherwise specified.

(2) The control registers are usually followed by verb abbreviations to make a distinction. The verbs can be: EN-Enable, CFG-Configure, D-Disable, SET-Setup and SEL-Select

(3) The status register abbreviation is usually followed by FLG to make a difference.

(4) The value and data registers usually include V, VALUE, D and DATA, which are not followed by verbs, such as xxPSC and CNT.

1.2.2 Full Name and Abbreviation Description of Terms

Table 1 R/W Abbreviation and Description

R/W	Description	Abbreviation
read/write	The software can read and write this bit.	R/W
read-only	The software can only read this bit.	R
write-only	The software can only write this bit, and after reading this bit, the reset value will be returned.	W
read/clear	The software can read this bit and clear it by writing 1. Writing 0 has no effect on this bit.	RC_W1
read/clear	The software can read this bit and clear it by writing 0. Writing 1 has no effect on this bit.	RC_W0

R/W	Description	Abbreviation
read/clear by read	The software can read this bit, reading this bit will automatically clear it to 0, and writing this bit is invalid.	RC_R
read/set	The software can read and set this bit, and writing 0 has no effect on this bit.	R/S
read-only write trigger	The software can read this bit and writing 0 or 1 can trigger an event but has no effect on the value of this bit.	RT_W
toggle	The software can reverse this bit only by writing 1, and writing 0 has no effect on this bit.	T

Table 2 Functional Description and Full Name and Abbreviation of Terms of Commonly Used Registers

Full name in English	English abbreviation
Enable	EN
Disable	D
Clear	CLR
Select	SEL
Configure	CFG
Contrl	CTRL
Controller	C
Reset	RST
Stop	STOP
Set	SET
Load	LD
Calibration	CAL
Initialize	INIT
Error	ERR
Status	STS
Ready	RDY
Software	SW
Hardware	HW
Source	SRC
System	SYS
Peripheral	PER
Address	ADDR
Direction	DIR

Full name in English	English abbreviation
Clock	CLK
Input	I
Output	O
Interrupt	INT
Data	DATA
Size	SIZE
Divider	DIV
Prescaler	PSC
Multiplier	MUL
Period	PRD

Table 3 Full Name and Abbreviation of Modules

Full name in English	English abbreviation
Reset and Clock Control	RCC
Clock Recovery System	CRS
Power Management Unit	PMU
Nested Vector Interrupt Controller	NVIC
External Interrupt /Event Controller	EINT
Direct Memory Access	DMA
Debug MCU	DBG MCU
General-Purpose Input Output Pin	GPIO
Alternate Function Input Output Pin	AFIO
Timer	TMR
Watchdog Timer	WDT
Independent Watchdog Timer	IWDT
Windows Watchdog Timer	WWDT
Real-Time Clock	RTC
Universal Synchronous Asynchronous Receiver Transmitter	USART
Inter-integrated Circuit Interface	I2C
Serial Peripheral Interface	SPI
Inter-IC Sound Interface	I2S
Controller Area Network	CAN
Universal Serial Bus Full-Speed Device	USBD

Full name in English	English abbreviation
HDMI-CEC Controller	HDMI-CEC
Analog-to-Digital Converter	ADC
Digital-to-Analog Converter	DAC
Touch Sensing Controller	TSC
Comparator	COMP
Cyclic Redundancy Check Calculation Unit	CRC
Operational Amplifier	OPAMP

2 System Architecture

2.1 Full Name and Abbreviation Description of Terms

Table 4 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Advanced High-Performance Bus	AHB
Advanced Peripheral Bus	APB

2.2 System architecture block diagram for F031

The main system mainly consists of two master modules and three slave modules. The main modules are Arm® Cortex®-M0+ core and general-purpose DMA (2 channels). The slave modules are internal SRAM, internal Flash, and AHB/APB bridge on the bus matrix, where the AHB/APB bridge connects all peripheral devices.

These are connected through a multi-level AHB bus architecture, as shown in the figure below:

Figure 1 AHB Bus Connection Relationship

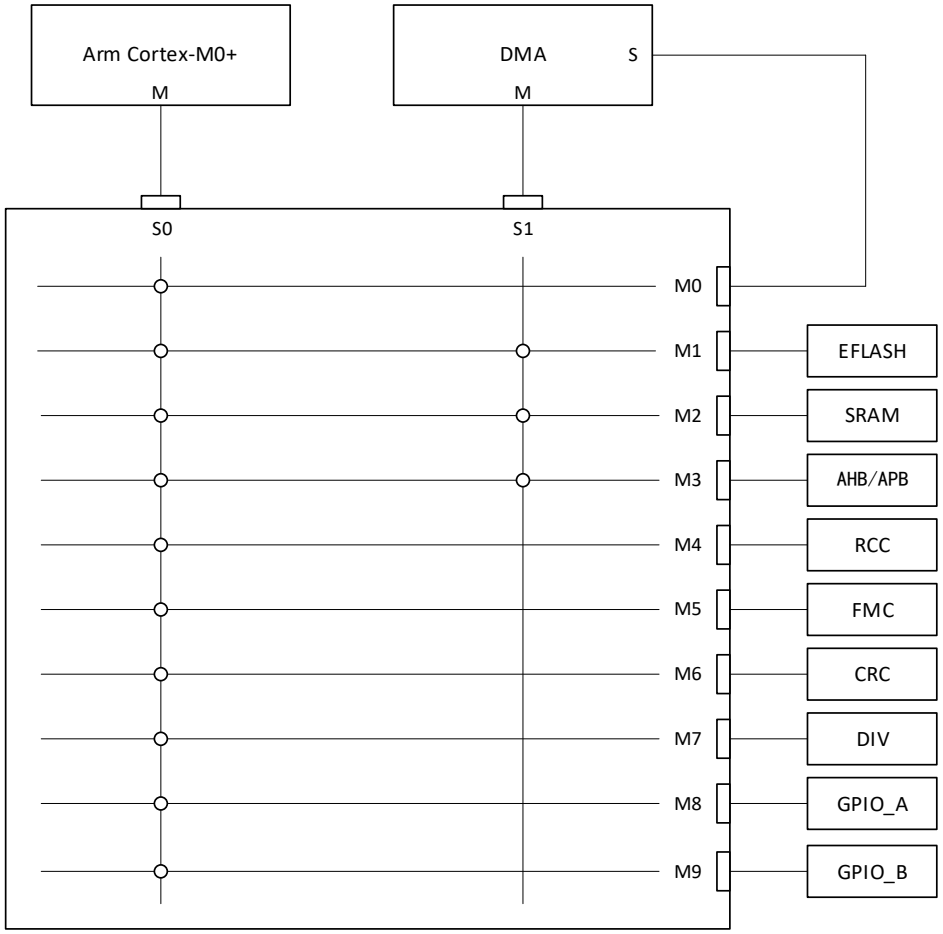
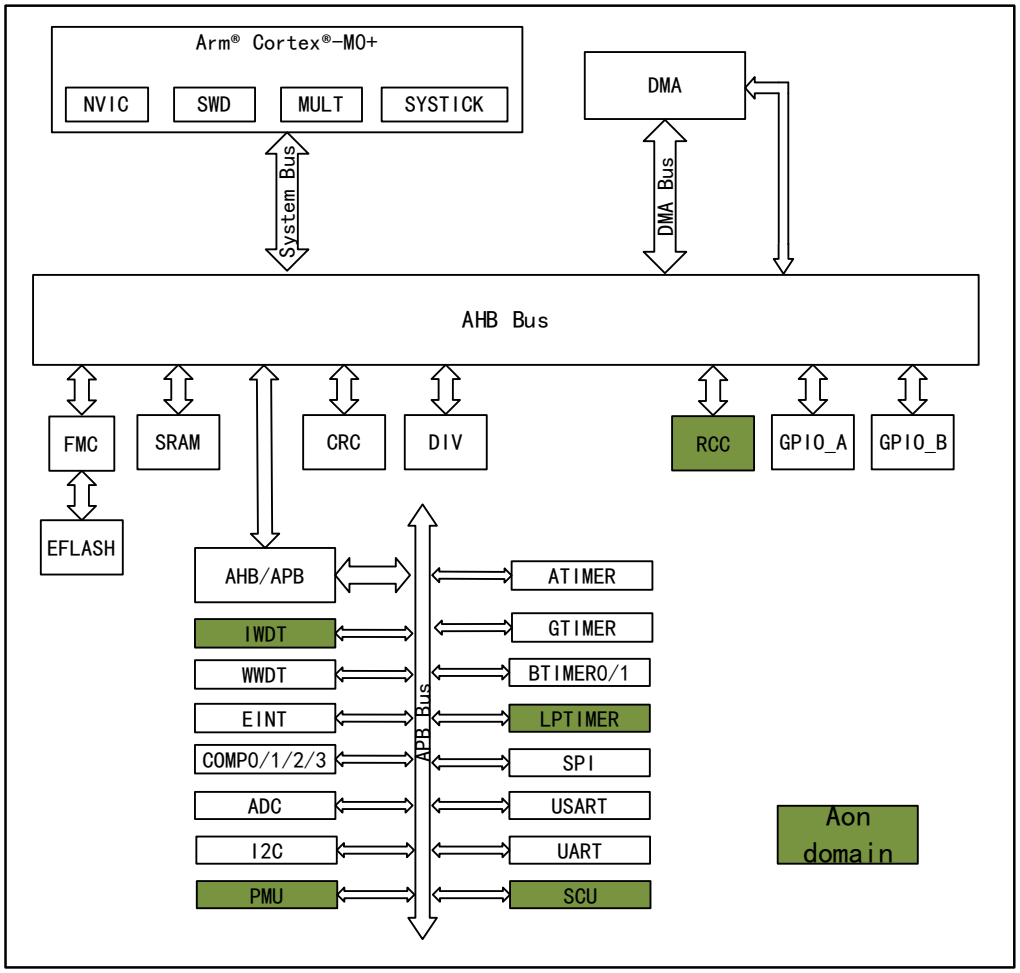
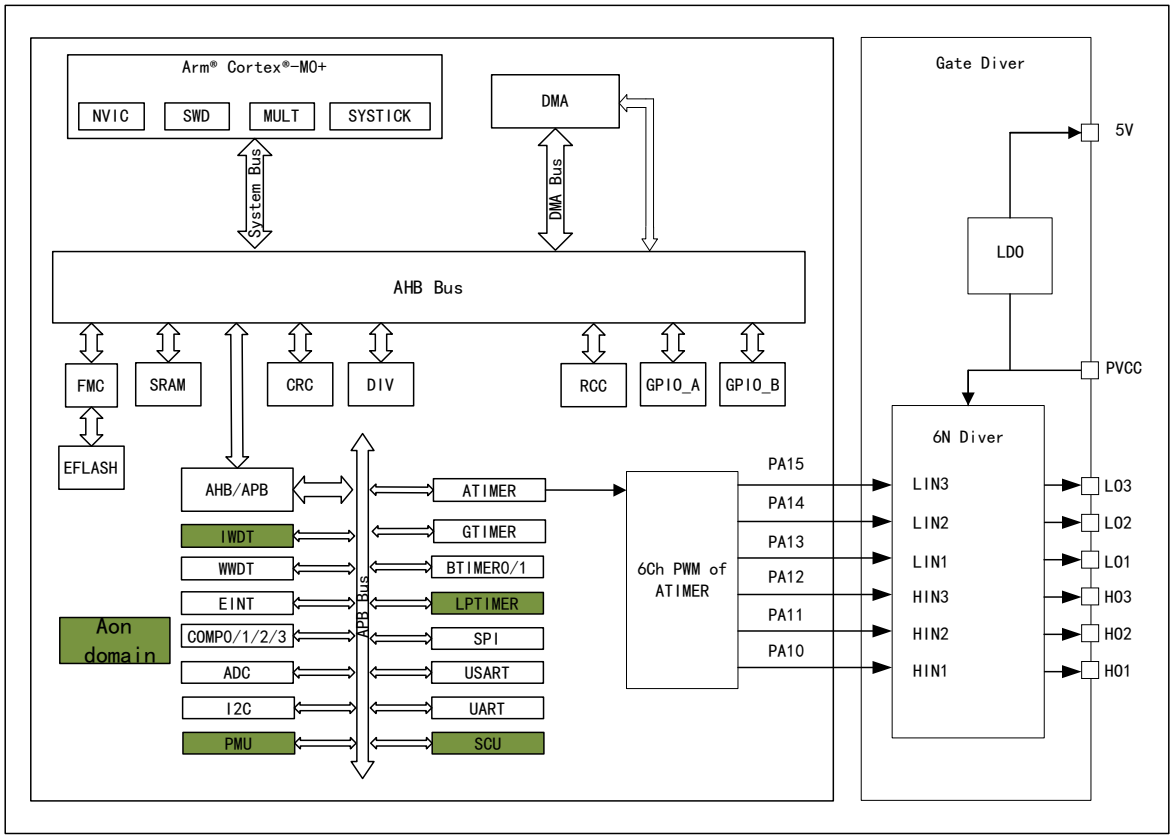


Figure 2 System Architecture Block Diagram



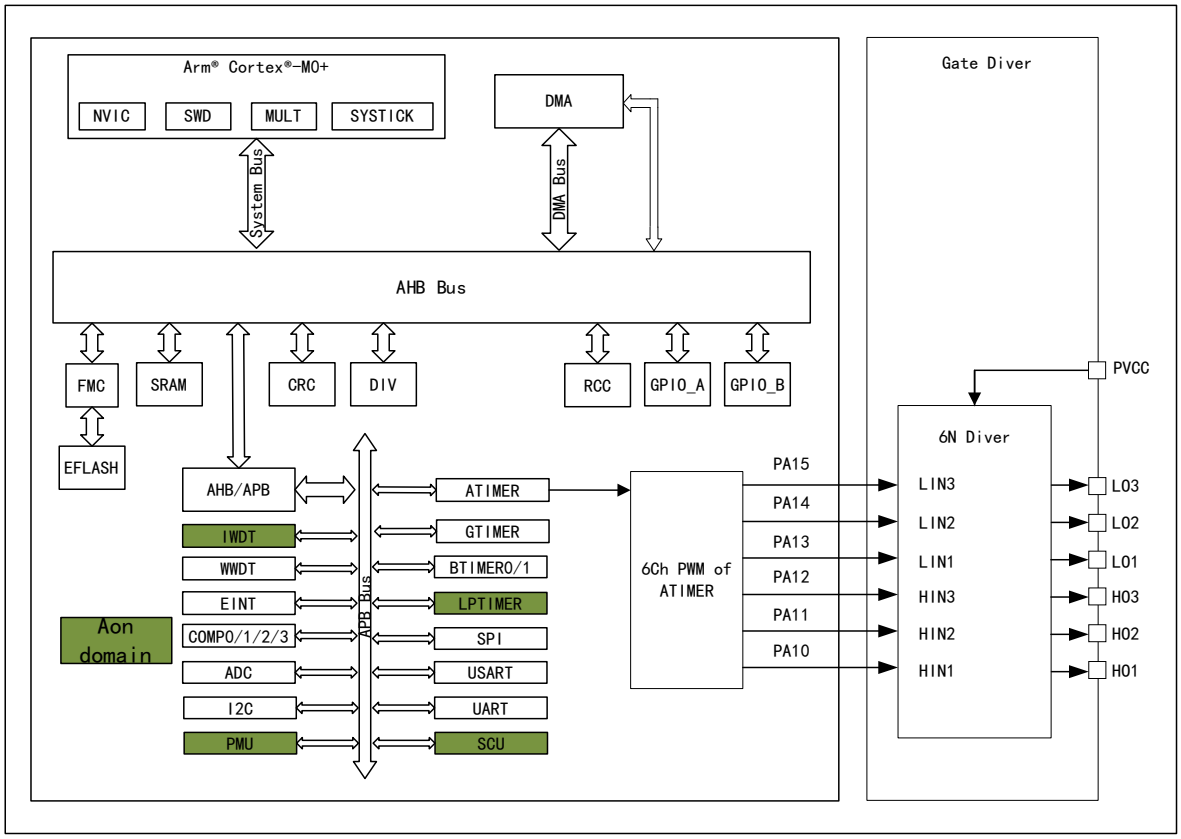
2.3 System architecture block diagram for M3114

Figure 3 System Architecture Block Diagram



2.4 System architecture block diagram for M3115

Figure 4 System Architecture Block Diagram



2.5 System architecture block diagram for M3122

Figure 5 System Architecture Block Diagram

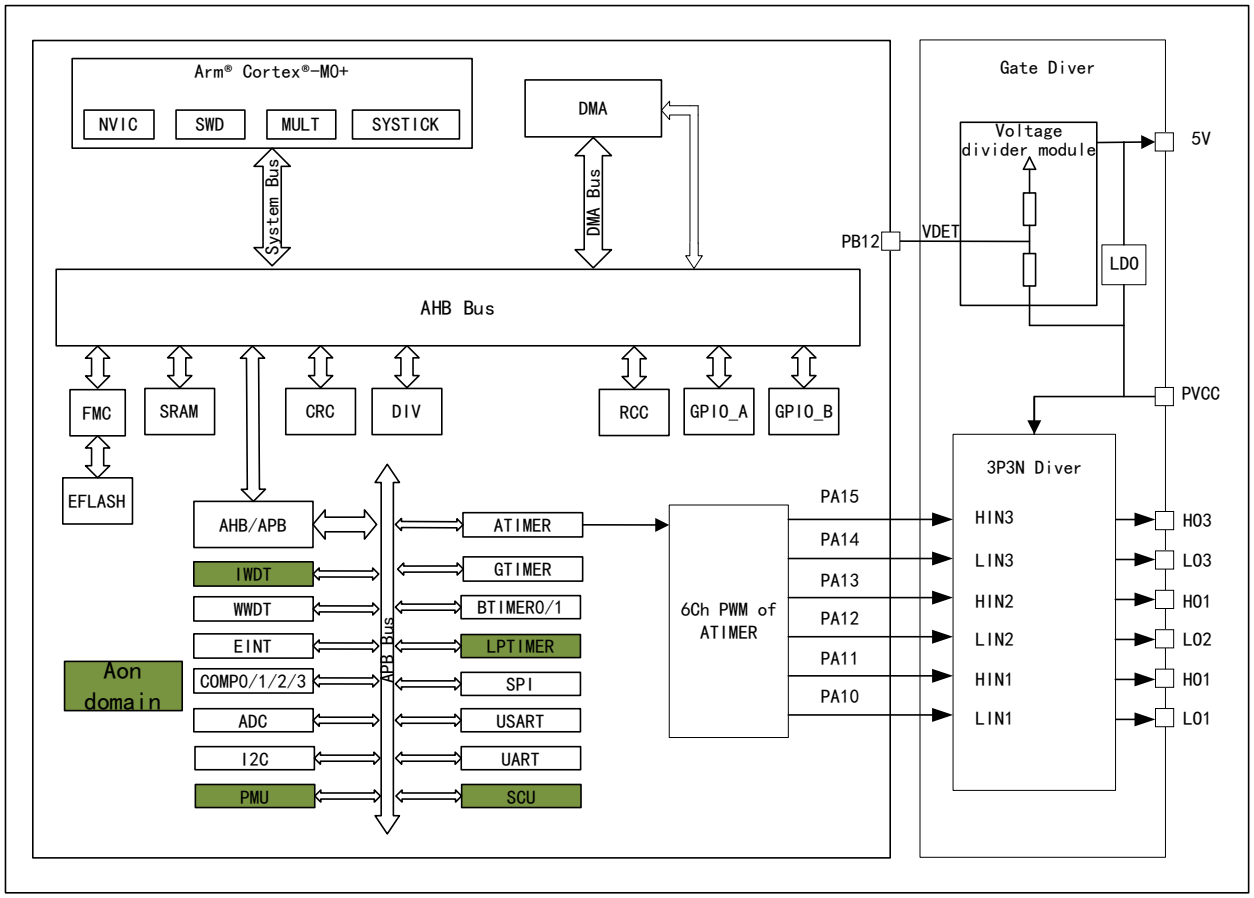


Table 5 Bus Name

Name	Description
System bus	Connect the system bus (peripheral bus) of Arm® Cortex®-M0+ core and the bus matrix.
DMA bus	Connect AHB master control interface of DMA and the bus matrix.
Bus matrix	Coordinate the access of the core and DMA; consist of CPU AHB, system bus, DMA bus and FMC, SRAM, and AHB1/APB bridges. AHB peripheral is connected with the system bus through the bus matrix and is allowed to access DMA.
AHB/APB bridge	The bridge provides synchronous connection between AHB and APB buses. The non-32-bit access to APB register will be converted into 32 bits automatically.

2.6 Memory mapping

The memory mapping address is totally 4GB address. The assigned addresses include the core (including core peripherals), on-chip Flash (including main memory area and user area), on-chip SRAM, and bus peripherals (including AHB and APB peripherals).

Figure 6 Address Mapping

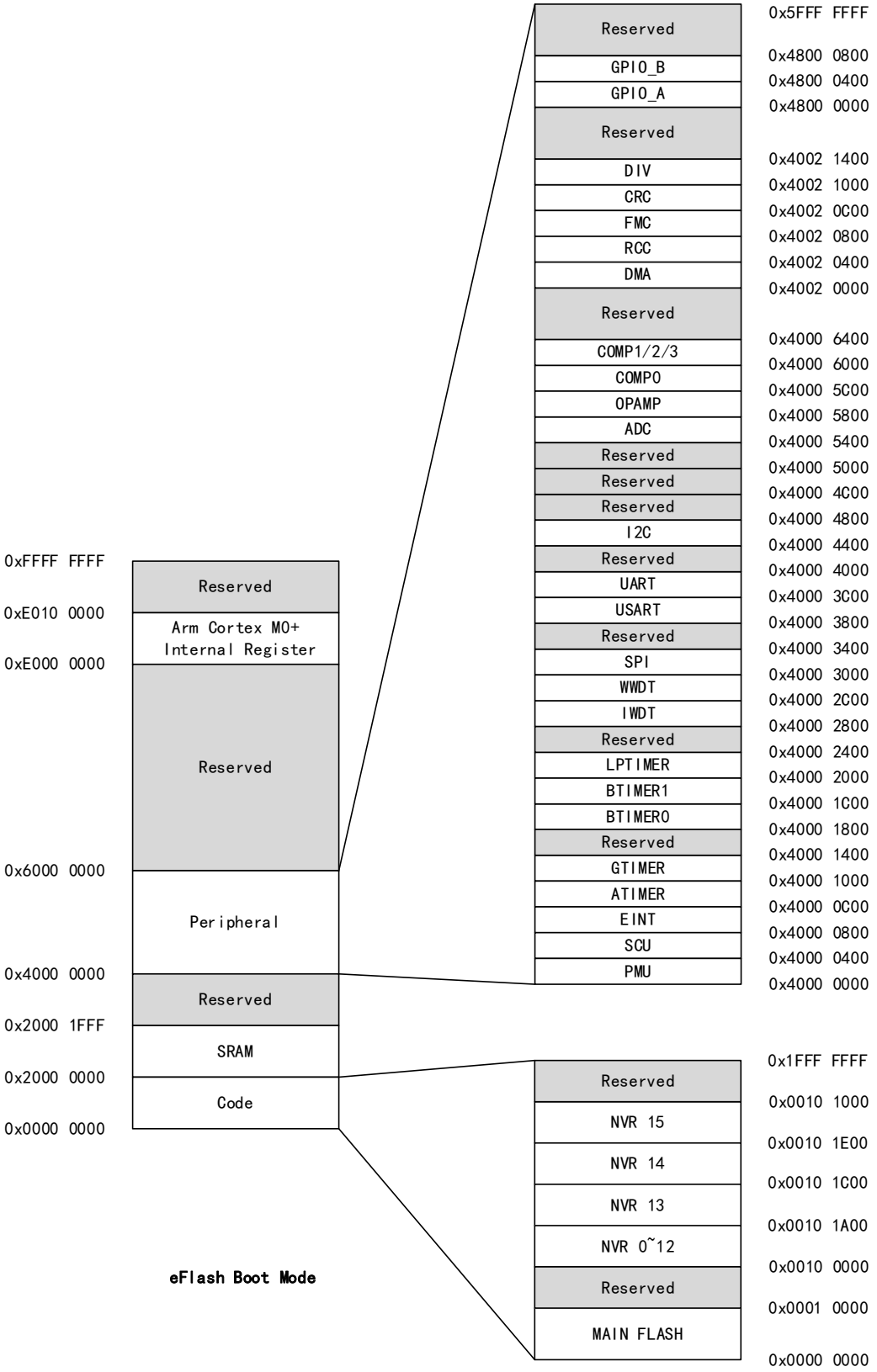


Table 6 Memory address allocation

Module	Name	Address range	Size (bytes)
Reserve	-	0x2000 2000 - 0x3FFF FFFF	~512 MB
SRAM	-	0x2000 0000 - 0x2000 1FFF	8 KB
Reserve	-	0x0010 2000 - 0x1FFF FFFF	~510MB
NVR 15 (Manufacturer Area)	-	0x0010 1E00 - 0x0010 1FFF	512 B
NVR 14 (Manufacturer Area)	-	0x0010 1C00 – 0x0010 1DFF	512 B
NVR 13(User Option Byte Area)	-	0x0010 1A00 – 0x0010 1BFF	512 B
NVR 0~12 （Bootloader Area）	-	0x0010 0000 – 0x0010 19FF	6.5 KB
MAIN FLASH	sector 127	0x0000FE00 - 0x0000FFFF	512 B
	sector 126	0x0000FC00-0x0000FDFF	512 B
	sector 125	0x0000FA00-0x0000FBFF	512 B
	sector 124	0x000F800-0x0000F9FF	512 B
	…	…	…
	sector 2	0x00000400-0x000005FF	512 B
	sector 1	0x00000200-0x000003FF	512 B
	sector 0	0x00000000-0x000001FF	512 B

Table 7 Peripheral register address allocation

Bus	Peripheral module	Address range	Size (bytes)
-	Cortex®-M0+ internal peripherals	0xE000 0000 - 0xE00F FFFF	1MB
-	Reserve	0x4800 0800 - 0x5FFF FFFF	383 MB
AHB	GPIO_B	0x4800 0400 - 0x4800 07FF	1 KB
	GPIO_A	0x4800 0000 - 0x4800 03FF	1 KB
-	Reserve	0x4002 1400 - 0x47FF FFFF	127MB
AHB	DIV	0x4002 1000 - 0x4002 13FF	1 KB
	CRC	0x4002 0C00 - 0x4002 0FFF	1 KB
	FMC	0x4002 0800 - 0x4002 0BFF	1 KB
	RCC	0x4002 0400 - 0x4002 07FF	1 KB
	DMA	0x4002 0000 - 0x4002 03FF	1 KB
-	Reserve	0x4000 8000 - 0x4001 FFFF	96KB
APB	Reserve	0x4000 6400 - 0x4000 7FFF	7 KB
	COMP1	0x4000 6000 - 0x4000 63FF	1 KB
	COMP0	0x4000 5C00 - 0x4000 5FFF	1 KB

Bus	Peripheral module	Address range	Size (bytes)
	OPAMP	0x4000 5800 - 0x4000 5BFF	1 KB
	ADC	0x4000 5400 - 0x4000 57FF	1 KB
	Reserve	0x4000 5000 - 0x4000 53FF	1 KB
	Reserve	0x4000 4C00 - 0x4000 4FFF	1 KB
	Reserve	0x4000 4800 - 0x4000 4BFF	1 KB
	I2C	0x4000 4400 - 0x4000 47FF	1 KB
	Reserve	0x4000 4000 - 0x4000 43FF	1 KB
	UART	0x4000 3C00 - 0x4000 3FFF	1 KB
	USART	0x4000 3800 - 0x4000 3BFF	1 KB
	Reserve	0x4000 3400 - 0x4000 37FF	1 KB
	SPI	0x4000 3000 - 0x4000 33FF	1 KB
	WWDT	0x4000 2C00 - 0x4000 2FFF	1 KB
	IWDT	0x4000 2800 - 0x4000 2BFF	1 KB
	Reserve	0x4000 2400 - 0x4000 27FF	1 KB
	LPTIMER	0x4000 2000 - 0x4000 23FF	1 KB
	BTIMER1	0x4000 1C00 - 0x4000 1FFF	1 KB
	BTIMER0	0x4000 1800 - 0x4000 1BFF	1 KB
	Reserved	0x4000 1400 - 0x4000 17FF	1 KB
	GTIMER	0x4000 1000 - 0x4000 13FF	1 KB
	ATIMER	0x4000 0C00 - 0x4000 0FFF	1 KB
	EINT	0x4000 0800 - 0x4000 0BFF	1 KB
	SCU	0x4000 0400 - 0x4000 07FF	1 KB
	PMU	0x4000 0000 - 0x4000 03FF	1 KB

2.6.1 Embedded SRAM

Built-in 8KB static SRAM. It allows access by byte, half word (16 bits) or full word (32 bits). The start address of SRAM is 0x2000 0000.

2.7 Start the boot mode

2.7.1 Boot mode

The chip is activated by the Main memory (Main Flash). The main memory is mapped to the boot space, but it can still be accessed at its original address, that is, the contents of the flash memory can be accessed in two address areas.

3 Flash Memory

This manual is only applicable to G32F031/M3122/M3114/M3115 series products, mainly introducing the Flash storage structure, read, erase, write, read/write protection, unlock/lock characteristics, and the function description of related registers.

3.1 Full Name and Abbreviation Description of Terms

Table 8 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Embedded Flash	EFLASH
Flash Memory Controller	FMC

3.2 Main characteristics

(1) Main area 64KB, NVR 4KB, bit width 64 bits

(2) Supports read, write, erase, read-write protection

(3) Supports unlock/lock functions

(4) The bus will be pulled low during erasing and writing

(5) Area access permissions can be configured, and registers can read the partition permission status

(6) Software triggers option byte reloading

(7) A hardware error interrupt is triggered when erase accessing out of bounds

(8) When read access is out of bounds, there will be read protection when configuring read_protect, and 0xA5A5A5A5 will be returned

(9) When configuring read_protect, SWD accessing any address in the MAIN FLASH area will have read protection and return 0xA5A5A5A5

(10) Under the configuration of read_protect in OPT, SWD accesses SRAM and can generate a full erase of FLASH

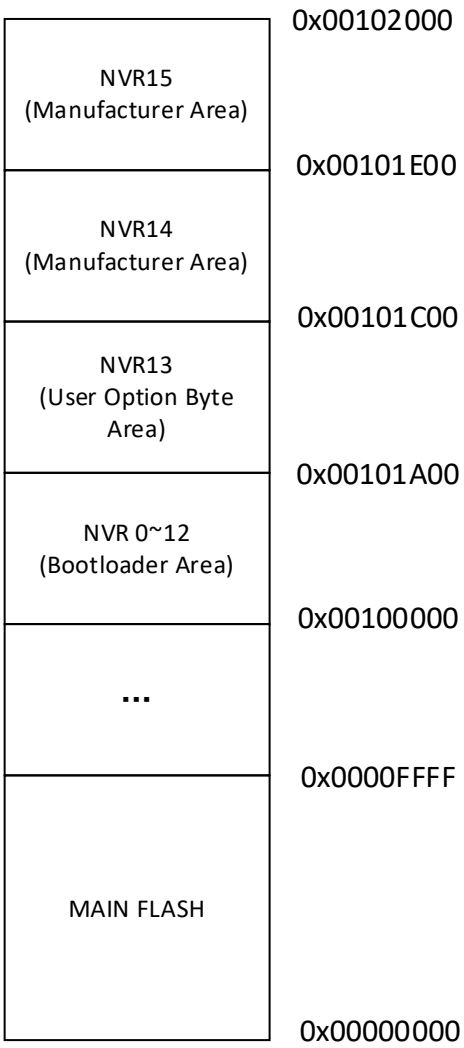
(11) Do not erase or write FLASH when the voltage is below 2V

(12) Only 32-bit erase and write operations are supported

3.3 Functional description

3.3.1 Storage area address mapping

Figure 7 Storage area address mapping



3.3.2 User Option Byte Area (NVR13)

The addresses 0x00101A00 to 0x00101BFF are the NVR13 user option byte area. Some of these addresses have special functions, as shown in the following table:

Table 9 User Option Byte Area (NVR13)

Address	Name	Bit width	R/W	Description
0x00101BE8	SYS_REMAP	32-bit	R/W	When the chip is powered on, choose to start from the BOOT area or the MAIN area 0x55AAAA55: Start from the MAIN area

Address	Name	Bit width	R/W	Description
				Other values: Boot from the BOOT area
0x00101BEC	SWD_DISABLE	32-bit	R/W	SWD accesses flash permission 0x89BC3F51: Disable SWD access function, that is, prohibit SWD from accessing FLASH, including reading, erasing and writing. 0x5389BCEA: The SWD read FLASH function is prohibited, but the SWD erase and write FLASH function is not prohibited. Other values: SWD permissions are unrestricted
0x00101BF0	READ_PROTECT	32-bit	R/W	Read protection in the MAIN FLASH area 0x89BC3F51: When executing code in user mode (bootstrap from user FLASH) : All operations can be performed on the MAIN FLASH area. 1.When the code is bootstrap or executed from the system memory: Read or write access to the Flash main area is prohibited, and a bus error hardfault interrupt is triggered. 2. When accessing SRAM through the JTAG interface, the MAIN area will be automatically erased. 3. When SWD is connected, FLASH reading is prohibited. Other values: Unprotected
0x00101BF4	FLASH_CRYPT_EN	32-bit	R/W	flash encryption and decryption switch 0x55AAAA55: Encryption and decryption enable Other values: No encryption or decryption mode
0x00101BF8	PRIVATE_KEY	32-bit	R/W	Encryption key 0xFFFFFFFF: The customer has not written it. The default key 0x1122AABB will be used Other values: Written by the customer
0x00101BFC	OTPC_EN	32-bit	R/W	OTP configuration enable 0x55AAAA55: Only read allowed Other values: Can be accessed freely

3.3.3 Manufacturer Area (NVR14/NVR15)

The addresses 0x00101C00 to 0x00101DFF are the NVR14 vendor area. Some of these addresses have special functions, as shown in the following table:

Table 10 Manufacturer Area (NVR14)

Address	Name	Bit width	R/W	Description
0x00101C08	Measured value of VBB1.5V	32-bit	R	VBB1.5V measured value (Unit : mV)
0x00101C0C	Measured value of HSI	32-bit	R	HSI 64MHz measured value (Unit : Hz)
0x00101C10	Measured value of LSI	32-bit	R	LSI 32.768K measured value (Unit : Hz)
0x00101C14	Vsensor_CAL	32-bit	R	The normal temperature calibration value of the Tsensor is the original data collected at a temperature of 25℃ and VDD5=5V
0x00101C1C	Measured value of VBG	32-bit	R	The original data collected with the built-in reference voltage VBG at a temperature of 25℃ (±5℃) and VDD5=5V(±10mV)
0x00101D80	Product model identification: PID	16-bit	R	The product model identification (PID) is used to uniquely identify a specific combination of information such as the chip model and version. For detailed information, please refer to the section "24.2 Product Model Identification"

Addresses 0x00101E00 to 0x00101FFF are the NVR15 vendor area. Some of these addresses have special functions, as shown in the following table:

Table 11 Manufacturer Area (NVR15)

Address	Name	Bit width	R/W	Description
0x00101E88	UID0	32-bit	R	For detailed information, please refer to the content of the "24.1 Product Identity Marking" section
0x00101E8C	UID1	32-bit	R	For detailed information, please refer to the content of the "24.1 Product Identity Marking" section
0x00101E90	UID2	32-bit	R	For detailed information, please refer to the content of the "24.1 Product Identity Marking" section

3.3.4 Main functions of FMC

- Read and write protection processing of EFLASH
- Realize the read and write data between the MCU and the EFLASH
- Support sector erase and full erase for EFLASH
- The Main array supports sector erasure and full erase
- The user options area only supports sector erase
- The initialization information is automatically loaded after power-on

3.3.5 Data encryption function

The data in the MAIN FLASH area has the encryption function. It is determined whether to encrypt based on the FLASH_CRYPT_EN enable signal loaded in the option byte area and the XOR encryption factor PRIVATE_KEY.

- FLASH_CRYPT_EN = 0x55AAAA55 enables encryption, other values have no encryption.
- PRIVATE_KEY = 0xFFFFFFFF indicates that the user has not written, and the default key 0x1122AABB will be used; other values are written by the customer.

3.3.6 Prefetching function

When the system is powered on, the prefetching function is automatically activated. The prefetch function can be disabled by configuring the register prefetch enable. When a reset occurs, the prefetch cache will be cleared.

3.3.7 Direction for use

3.3.7.1 EFLASH unlock

Table 12 unlock code

	MAIN FLASH	Trim user area（NVR13）
KEY0	0xABCD6789	0x33AADD55
KEY1	/	0xEDCC1F55

Before erasing or writing Flash, it must be unlocked. When an incorrect unlock code is written, an unlock error status will occur and a corresponding interrupt will be generated. After the Flash Key is unlocked incorrectly, erasure and writing of Flash will be prohibited. Erasure and writing can be unlocked by system reset or by clearing the error status after rewriting the correct unlock code. After the normal erase and write are completed, any value (not an unlock code) is written to the KEY register to return the initial off-unlock state. The specific description of the unlock code can be found in the contents of the registers FMC_MKEY and FMC_NVRCKEY.

3.3.7.2 EFLASH erasure

Table 13 The erasure methods for different areas

MAIN FLASH	User option byte area (NVR13)
Chip/Sector	Sector

MAIN FLASH area chip erased

This operation is only for the main area. The steps are as follows:

(1) Write the unlock sequence to FMC_MKEY to unlock the main area of flash

(2) Set FMC_CR->OPERATE to 0x10(Chip Erase)

(3) Write 0xA5A5 (any value is acceptable) to any address in the FLASH-Main area.

(4) Detect that the FMC_SR->BUSYFLG bit is reset to zero, or FMC_SR->OPENDFLG is set to 1

MAIN FLASH area sector erased

This operation is only for the main area. The steps are as follows:

(1) Write the unlock sequence to FMC_MKEY to unlock the main area of flash

(2) Set FMC_CR->OPERATE to 0x10(Sector Erase)

(3) Write 0xA5A5 (any value is acceptable) to any address in the FLASH-Main sector area.

(4) Detect that the FMC_SR->BUSYFLG bit is reset to zero, or FMC_SR->OPENDFLG is set to 1

User option byte area (NVR13) sector erased

This operation is only for the NVR13 area. The steps are as follows:

(1) Write the unlock sequence to FMC_NVRKEY to unlock the user option byte area (NVR13) of flash

(2) Set FMC_CR->OPERATE to 0x10(Sector Erase)

(3) Write 0xA5A5 (any value is acceptable) to any address in the FLASH-user option byte area (NVR13) sector area.

(4) Detect that the FMC_SR->BUSYFLG bit is reset to zero, or FMC_SR->OPENDFLG is set to 1

3.3.7.3 EFLASH programme

EFLASH is programmed as one 32-bit piece of data each time.

MAIN FLASH area write programe

This operation is only for the main area. The steps are as follows:

(1) Write the unlock sequence to FMC_KEY to unlock the main area of flash

(2) Set FMC_CR->OPERATE to 0x11(write)

(3) Write data to any address in the FLASH MAIN area.

(4) Detect that the FMC_SR->BUSYFLG bit is reset to zero, or FMC_SR->OPENDFLG is set to 1

User option byte area (NVR13) write programe

This operation is only for the user option byte area (NVR13). The steps are as follows:

(1) Write the unlock sequence to FMC_NVRKEY to unlock the user option byte area (NVR13) of flash

(2) Set FMC_CR->OPERATE to 0x11(write)

(3) Write 32 bit data to any address in the user option byte area (NVR13).

(4) Detect that the FMC_SR->BUSYFLG bit is reset to zero, or FMC_SR->OPENDFLG is set to 1

3.4 Register address mapping

Table 14 FLASH Register Address Mapping

Register name	Description	Offset address
FMC_WKEY	Write protect register	0x00
FMC_MKEY	Key register for unlocking the MAIN area	0x04
FMC_NVRCKEY	Key register for unlocking the NVRC area	0x0C
FMC_CR	Control register	0x14
FMC_IER	Interrupt enable register	0x18
FMC_SR	Status register	0x1C
FMC_CR1	Control register 1	0x20

3.5 Register functional description

3.5.1 Write protect register (FMC_WKEY)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	WKEY	R/W	Flash register write protect register Write a valid KEY sequence, writing 0x3399AA55 is the unlocked state, and other registers can be written. Write any value to disable unlock, and write 0 to lock. 0: Not unlocked 1: Unlock

3.5.2 Unlock the key register of MAIN area (FMC_MKEY)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	MKEY	R/W	MAIN FLASH area erase Key input register Write a valid KEY sequence, writing 0xABCD6789 is the unlocked state, and other registers can be written. Write any value to disable unlock, and write 0 to lock. 0: Not unlocked 1: Unlock

3.5.3 Key register for unlocking the NVRCKEY area (FMC_NVRCKEY)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	NVRCKEY	R/W	Option byte area (NVR13) Key input register Write KEY0 0x33AADD55 to unlock the option byte area (NVR13). If OPTC_EN encrypts this area, the unlock will be invalid and you need to write KEY1 0xEDCC1F55 again to unlock. Writing any value can disable the unlock. A readback reading of 1 indicates unlocking, and 0 indicates not unlocking. 0: Not unlocked 1: Unlock

3.5.4 Control register (FMC_CR)

Offset address: 0x14

Reset value: 0x0000 0400

Field	Name	R/W	Description
31:16	Reserved		
15	OPTLOAD	R/W	Option byte forced update This bit can only be written as 1 when the data written in the higher 16 bits is 0xA5A5. When written as 1, this bit will force the option byte to update, and this operation will trigger a system reset. 0: Invalid 1: Valid
14:11	Reserved		
10	PREEN	R/W	Prefetch enable 0: Disable 1: Enable
9:5	Reserved		
4	READONLY	R/W	Flash ip Read-only control Cannot be erased/written. 0: Invalid 1: Valid
3:2	Reserved		

Field	Name	R/W	Description
1:0	OPERATE	R/W	Flash operation types configure 00: Read operation 01: Sector erase 10: Chip erase 11: Write When the BUSYFLG bit of FMC_SR is set, this register cannot be written.

3.5.5 Interrupt enable register (FMC_IER)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:7	Reserved		
6	RPTIEN	R/W	FLASH read/write protection interrupt enable 0: Disable 1: Enable
5	KEYIEN	R/W	Flash KEY error interrupt enable 0: Disable 1: Enable
4	OPEIEN	R/W	FLASH erase end interrupt enable 0: Disable 1: Enable
3:0	Reserved		

3.5.6 Status register (FMC_SR)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:7	Reserved		
6	RPTERRFLG	R/W	FLASH read write protection interrupt flag SWD_DISABLE= 0x889BC3F51, disable swd access function (read and write), SWD_DISABLE= 0x5389BCEA, not disable the write function, disable the swd read function. At this point, the bus feeds back 0xA5A5A5A5 data. Other values of SWD are unrestricted. Set to 1 by hardware, and write 0 by software to clear to 0. 0: No interrupt 1: Interrupt occurs
5	KEYERRFLG	R/W	Flash KEY error interrupt flag Erase and write flash without unlocking. Operating illegal areas within 0x2000 0000 that are not flash will also generate an interrupt. Set to 1 by hardware, and write 0 by software to clear to 0. 0: No interrupt

Field	Name	R/W	Description
			1: Interrupt occurs
4	OPENDFLG	R/W	FLASH erase end interrupt flag Set to 1 by hardware, and write 0 by software to clear to 0. 0: No interrupt 1: Interrupt occurs
3:1	Reserved		
0	BUSYFLG	R	FLASH erase busy flag 0: Idle 1: Busy

3.5.7 Control register1 (FMC_CR1)

Offset address: 0x14

Reset value: 0x0000 0300

Field	Name	R/W	Description
31:2	Reserved		
1:0	LATENCY	R/W	Access delay cycle The default power-on value is 3 at 64MHz. 00: FLASH needs 0 wait cycle to read data 01: FLASH needs 1 wait cycle to read data 10: FLASH needs 2 wait cycles to read data 11: FLASH needs 3 wait cycles to read data The LATENCY of configurable registers below 32MHz is 00. For 32MHz, a register LATENCY of 01 needs to be configured. For 64MHz, a register LATENCY of 1x needs to be configured.

4 System configuration controller (SCU)

The SCU module is used to control the counting mode of the entire chip peripheral in DBG mode.

4.1 Register address mapping

Only 32-bit read and write operations are supported.

Table 15 SCU Register Address Mapping

Register name	Description	Offset address
SCU_KEY	SCU reads and writes protection register	0x00
SCU_DBG	System DBG control register	0x04

4.2 Register functional description

4.2.1 SCU reads and writes protection register (SCU_KEY)

Offset address: 0x00

Reset value: 0x0000 00000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:17	Reserved		
16	LOCKFLG	R	System register write protect flag 0: The system register is in the protected state 1: The system register is in writable state Note: Software writes 1 to clear, writing 0 is invalid. If LOCKFLG is written 1 at the same time, and LOCKKEY is written 0x87E4 to unlock, this bit is set 1, and the system register is in writable state.
15:0	LOCKKEY	W	Password protection configuration for system register write operations Writing 0x87E4 to unlock, and LOCKFLG is automatically set 1 at the same time. When writing any other value, lock it and reset LOCKFLG to zero at the same time.

4.2.2 System DBG control Register (SCU_DBG)

Offset address: 0x04

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:8	Reserved		
7	WWDTDBG	R/W	The WWDT counting method in DEBUG mode 0: Counting is not affected 1: Pause the count in DEBUG mode

Field	Name	R/W	Description
6	IWDTDBG	R/W	The IWDT counting method in DEBUG mode 0: Counting is not affected 1: Pause the count in DEBUG mode
5	LPTMRDBG	R/W	The LPTMR counting method in DEBUG mode 0: Counting is not affected 1: Pause the count in DEBUG mode
4	BTMR1DBG	R/W	The BTMR1 counting method in DEBUG mode 0: Counting is not affected 1: Pause the count in DEBUG mode
3	BTMR0DBG	R/W	The BTMR0 counting method in DEBUG mode 0: Counting is not affected 1: Pause the count in DEBUG mode
2	Reserved		
1	GTMRDBG	R/W	The GTMR counting method in DEBUG mode 0: Counting is not affected 1: Pause the count in DEBUG mode
0	ATMRDBG	R/W	The ATMR counting method in DEBUG mode 0: Counting is not affected 1: Pause the count in DEBUG mode

5 Reset and Clock Control (RCC)

5.1 Full Name and Abbreviation Description of Terms

Table 16 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Reset and Clock Control	RCC
Reset	RST
Power-On Reset	POR
Power-Down Reset	PDR
High Speed Internal Clock	HSICLK
Low Speed Internal Clock	LSICLK
Calibrate	CAL
Trim	TRM
Non Maskable Interrupt	NMI

5.2 Reset functional description

The supported reset is divided into five forms, namely, system reset, NRST reset , power reset ,standby reset and LPTMR reset.

5.2.1 System reset

5.2.1.1 "System reset" reset source

- Window watchdog termination count (WWDT reset)
- Independent watchdog termination count (IWDT reset)
- Software reset (SW reset)
- Abnormal system reset caused by CPU deadlock
- Load option byte reset
- Power reset

A system reset will occur when any of the above events occurs. Besides, the reset event source can be identified by viewing the reset flag bit in RCC_RSTCSR (reset control/status register).

When a system reset occurs, the chip will reset all registers except the following:

- RCC reset control register RCC_RSTCSR
- RCC Regarding the AON domain control register RCC_AONCSR
- PMU and STANDBY related registers: PMU_LPCR, PMU_WKCR, PMU_WKSR
- PVD control register PMU_PVDCSR

Note:

(1) When configured to STANDBY0 mode, LPTMR is affected by system reset and LPTMRRST. When configured in STANDBY1 mode, LPTMR is only affected by POR, NRST, and LPTMRRST.

(2) When configured to STANDBY0 mode, IWDT is affected by system reset and STANDBY processes; When configured to STANDBY1 mode, if the system is in non-Standby mode and is still affected by system reset, it will not be affected by the STANDBY mode entry process. It will continue to count and wake up STANDBY, and reset in the STANDBY wake-up process.

Software Reset

Software can be reset by setting SYSRESETREQ in Arm ® Cortex ® -M0+ interrupt application and reset control register to "1".

Load option byte reset

The load byte reset is triggered by OPTLOAD bit in FMC_CR register which is controlled by software.

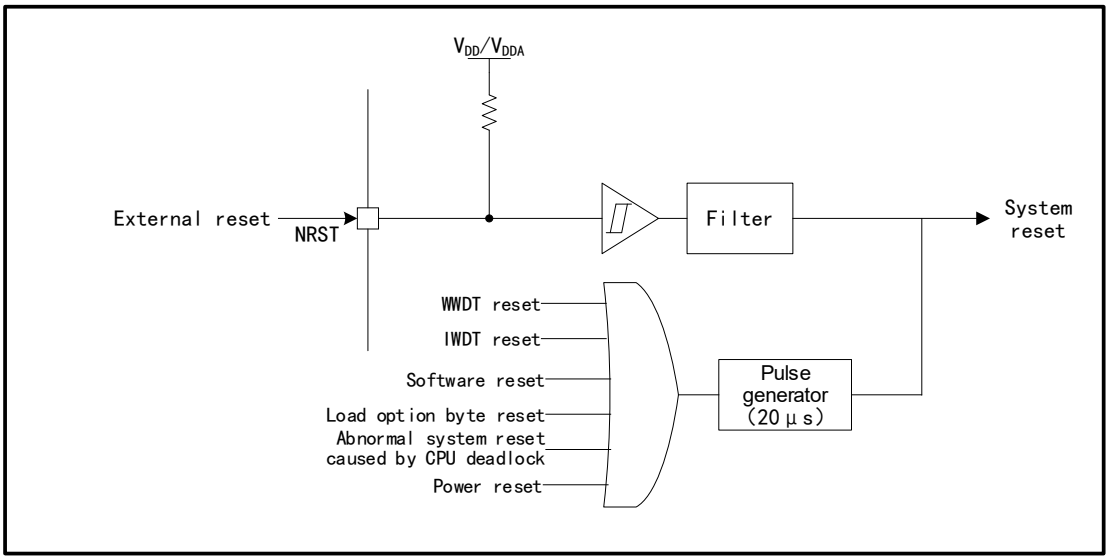
5.2.1.2 "System reset" reset circuit

When an external reset NRST generates a reset signal, it first passes through an analog filter of approximately 50ns, then through a digital filter of up to 10ms (whether the digital filter is enabled and the magnitude of the digital filter can be configured through the NRSTFLTSEL bit in the RCC_RSTCSR register), and then acts on the system.

After the internal reset source generates a reset signal, a low-level pulse of approximately 20μs is produced through the pulse generator to act on the system without causing a change in the level of the external NRST pin.

The "system reset" reset circuit is shown in the figure below.

Figure 8 "System Reset" Reset Circuit



5.2.2 NRST reset

When a low level occurs at the external NRST pin, the system NRST reset occurs.

When an NRST reset occurs, the chip will reset all registers except the following:

- RCC reset control register (RCC_RSTCSR)

5.2.3 Power reset

"Power reset" reset source is as follows:

- Power-on reset (POR reset)
- Power-down reset (PDR reset)

A power reset will occur when any of the above events occurs. Power reset will reset all registers.

5.2.4 STANDBY reset

When the system executes the STANDBY process, a STANDBY reset occurs in the system.

When STANDBY occurs, since all the digital CORE domain are powered down, the chip will reset all the registers under the CORE domain:

- In STANDBY0 mode, LPTMR and IWDT will be forcibly reset
- In STANDBY1 mode, LPTMR is not affected, and IWDT will be reset during the STANDBY wake-up process

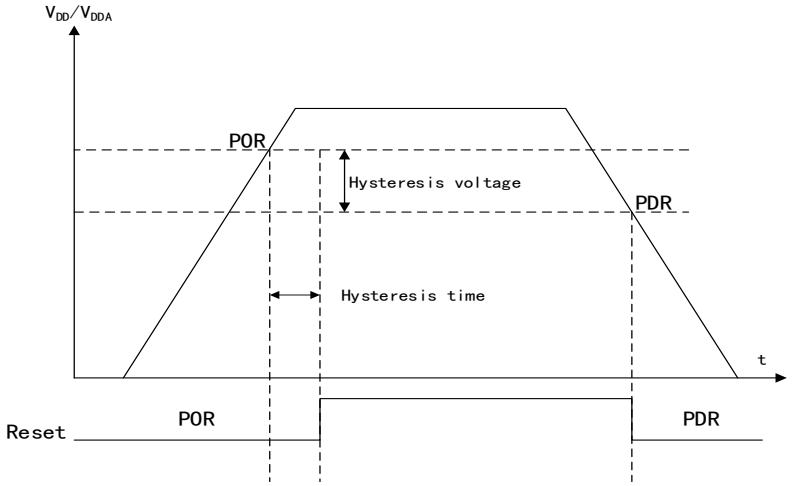
5.2.5 LPTMR reset

When LPTMRRST is set to 1 in RCC_AONCSR configuration, LPTMR reset occurs.

The LPTMR reset will reset LPTMREN in RCC_AONCSR and all the logic in the LPTMR module.

5.2.6 Power-on/power-off reset timing

Figure 9 Power-on Reset, Power-off Reset, Low-power Wake-up Timing Diagram



5.3 Functional description of clock management

The chip provides two different clock sources that can be used as system clock sources:

- Internal high-speed clock HSICLK: 64MHz high-speed RC oscillator
- Internal low-speed clock LSICLK: 32.768KHz low-speed RC oscillator

5.3.1 HSICLK internal high-speed clock signal

HSICLK clock signal is generated by internal 64MHz RC oscillator.

The HSI clock frequency of each chip has been calibrated by the manufacturer to ±0.5% before leaving the factory. The HSIRDY bit of the RCC_CR register indicates whether HSI is stable. HSI clock will be input to the system only when the hardware sets the HSIRDY bit to 1. If the HSIRDYIEN bit of the RCC _IER register is enabled, the corresponding interrupt will be generated. HSI can be enabled and disabled through the HSIEN bit of the RCC _CR register. When the HSIEN bit is configured to 0, the HSIRDY flag will be cleared after 4 HSI clock cycles.

Hardware logic protection:

- The system clock provides hardware logic protection for the HSI. When HSI is used as the system clock source, HSIEN cannot be turned off and writing 0 is invalid.
- The system has hardware logic protection for FLASH programming erasure. To ensure normal operation, when HSIEN is turned off, the flag bit HSIRDY will be reset along with the HSIEN hardware, but the

FLASH programming erase can continue to work, and the CLKOUT function still ensures that the clock can be output.

5.3.2 LSICLK low-speed internal clock signal

LSI clock signal is generated by internal 32.768KHz RC oscillator.

It can be used as the working clock for LPTMR and IWDT, and also as the system clock. It can continue to work as a low-power clock source in stop and standby 1 mode. The LSIRDY bit of the RCC_AONCSR register indicates whether LSI is stable. LSI clock will be input to the system only when the hardware sets the LSIRDY bit to 1. If the LSIRDYIEN bit of the RCC_IER register is enabled, the corresponding interrupt will be generated. LSI can be enabled and disabled through the LSIEN bit of the RCC_AONCSR register. When the LSIEN bit is configured to 0, the LSIRDY flag will be cleared after 2 LSI clock cycles.

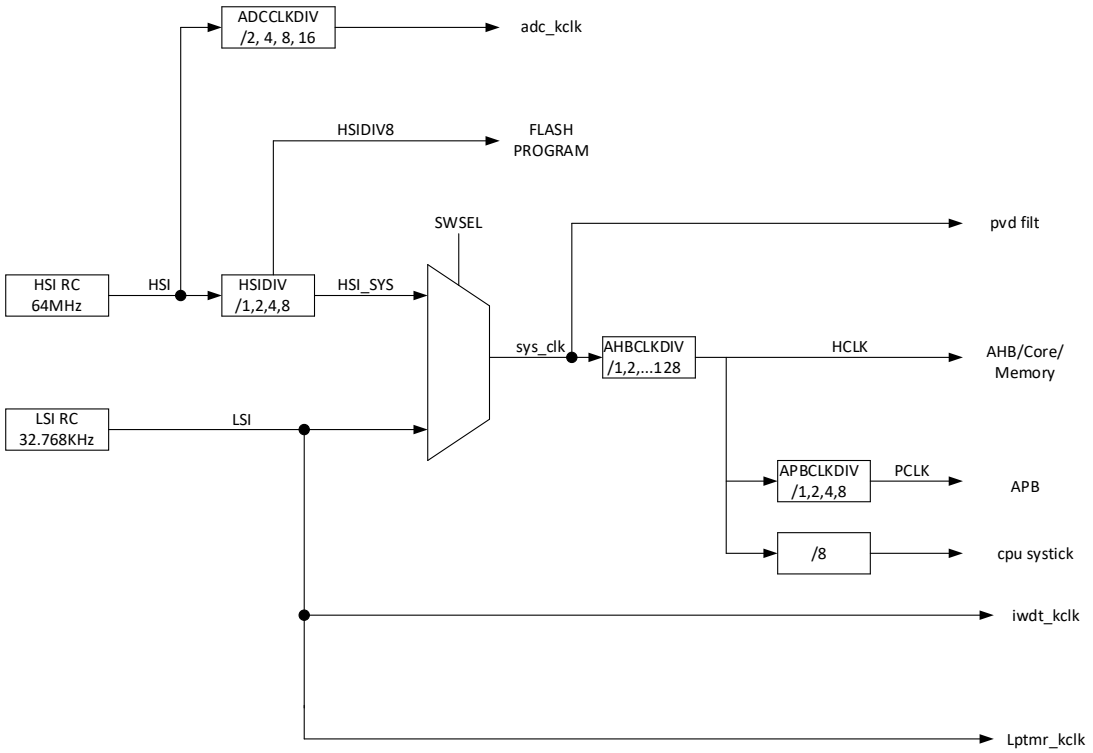
Hardware logic protection:

- The system clock provides hardware logic protection for the LSI. When LSI serves as the system clock source, LSIEN cannot be turned off, and writing 0 is invalid.
- The system has hardware logic protection for IWDT.To ensure the normal operation of IWDT, when turns off LSIEN, the LSIRDY flag will be cleared by hardware along with LSIEN, but IWDT can continue counting, and the CLKOUT function can still output this clock.
- The system has hardware logic protection for NRST filtering. To ensure the normal operation of NRST filtering, when LSIEN is turned off, the flag bit LSIRDY will be reset to zero along with the LSIEN hardware, but the NRST filtering can still work normally. However, compared with turning on LSIEN, the filtering time will be extended by 5 to 6 LSI cycles.
- The system does not provide hardware logic protection for the LPTMR clock source. When LPTMR is enabled but LSIEN is turned off, it will cause LPTMR to stop counting.

5.3.3 Clock tree

Clock tree of G32F031/M3122/M3114/M3115 is shown in the figure below:

Figure 10 G32F031/M3122/M3114/M3115 Clock Tree



5.3.4 System Clock and Switching

HSI or LSI can be selected as the system clock. Among them, HSI can select its 1, 2, 4, and 8 frequency divisions as the system clock branches. The maximum frequency is 64MHz. When the chip is powered on for the first time, HSI is selected as the system clock by default. Switching steps:

(1) Ensure both the original clock and the new clock are ready

(2) Configure SWSEL in RCC_CFG to the corresponding configuration

(3) When the SWSTS flag in RCC_CFG changes to the new clock source, it indicates a successful switch

Note:

(1) As the clock source of the system clock, the hardware has logical protection and the software cannot be turned off.

(2) When switching the system clock source, if the target clock has not been turned on or is not yet stable, the system will keep the current clock source working and wait until the new clock source stabilizes before switching. During the switching process, the hardware automatically protects and cannot turn off the clock source. After the switch is completed, users can turn off the original clock source as needed.

(3) The HSI used as the frequency division HSIDIV for the system clock does not affect the duty cycle.

(4) The STOP low-power mode does not affect HSIDIV. If HSIDIV is configured before entering STOP, the wake-up will still maintain this frequency division, which will cause the STOP wake-up to slow down.

(5) The system clock can be used as the filtering clock for the PVD function. When the PVD filtering function is enabled, it is not recommended to use the LSI as the system clock to prevent the PVD filtering from being too slow. When entering the STOP state, due to the system clock being turned off, the software needs to disable the PVD filtering function before entering STOP.

5.3.5 Bus clock

Built-in AHB and APB buses. The clock source of AHB is the system clock. It performs frequency division based on the configuration bit AHBCLKDIV[2:0], with a maximum frequency of 64MHz and a maximum frequency division coefficient of 128. The clock source of APB is the AHB clock. It performs frequency division based on the configuration bit APBCLKDIV[1:0], with a maximum frequency of 64MHz and a maximum frequency division coefficient of 8.

5.3.6 Clock source selection of CLKOUT

The internal clock can be output to GPIO by configuring the CLKOUTDIV, CLKOUTEN, and CLKOUTSEL bits of the RCC_CFG register. Clock output sources include the following:

- System clock
- HSI clock
- LSI clock
- HCLK clock (CPU)

Note:

(1) When this function is enabled or the output source is switched, clock glitches may occur.

(2) When the clock source is unstable, CLKOUT cannot output the corresponding clock.

(3) This function can be used in STOP mode as needed, but it does not support working in STANDBY mode.

5.3.7 Clock source selection of IWDT

When IWDT (independent watchdog) is enabled, LSICLK oscillator will be enabled by force, and when it is stable, it will provide the clock signal to IWDT. When IWDT is working, it is recommended to turn on LSIEN, but IWDT can still count normally after the software turns off LSIEN.

5.3.8 Clock source selection of LPTMR

When the LPTMR is in operation, the LSI is used as the working clock for counting.

LPTMR clock configuration steps:

(1) Enable LSIEN and confirm that LSIRDY is set to 1.

(2) Configure the LPTMREN of RCC_AONCSR to enable this peripheral.

(3) Configure the relevant Settings in the LPTMR peripheral to work.

Note: When LPTMR is in operation, make sure LSIEN is enabled; otherwise, the module will stop counting.

5.3.9 Clock source selection of ADC

The ADC's working clock source is the system clock, and it operates with HSI 64MHz as the working clock. The frequency division configuration is controlled by ADCCLKDIV[1:0], with the frequency division coefficients being 2, 4, 8, and 16 respectively. When the HSI clock serves as the ADC clock source, the software must ensure that HSIEN is enabled; otherwise, the ADC cannot operate.

5.3.10 SYSTICK clock

The CPU SYSTICK function clock is provided by HCLK's 8-way frequency division.

5.4 Register address mapping

Table 17 RCC Register Address Mapping

Register name	Description	Offset address
RCC_KEY	RCC read-write protection register	0x00
RCC_CR	Clock control register	0x04
RCC_CFG	Clock configuration register	0x08
RCC_IER	Clock interrupt enable register	0x0C
RCC_ISR	Clock interrupt flag register	0x10
RCC_AHBRST	AHB peripheral reset register	0x14
RCC_APBRST	APB peripheral reset register	0x18
RCC_AHBCLKEN	AHB peripheral clock enable register	0x1C
RCC_APBCLKEN	APB peripheral clock enable register	0x20
RCC_ADCCR	ADC clock control register	0x24
RCC_RSTCSR	Reset control/status register	0x28
RCC_AONCSR	AON domain control register	0x2C

5.5 Register functional description

5.5.1 RCC read-write protection register (RCC_KEY)

Offset address: 0x00

Reset value: 0x0000 00000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:17	Reserved		
16	LOCKFLG	R	RCC register write protect flag 0: The RCC register is in the protected state 1: The RCC register is in readable and writable state Note: When 0x87E4 is written to the LOCKKEY, this position 1; When other values are written to the LOCKKEY, this bit is reset to zero.
15:0	LOCKKEY[15:0]	W	Password protection configuration for RCC register write operations Writing 0x87E4 to unlock, and LOCKFLG is automatically set 1 at the same time. When writing any other value, a lock is added and LOCKFLG is reset to zero at the same time.

5.5.2 Clock control register (RCC_CR)

Offset address: 0x04

Reset value: 0x0000 0003

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:2	Reserved		
1	HSIRDY	R	HSI clock stability flag 0: Unstable 1: Stable When HSIEN is turned off, HSIRDY is cleared after 4 HSI clock cycles.
0	HSIEN	R/W	HSI clock enable 0: Disable 1: Enable When HSI is used as the system clock, writing 0 is invalid.

5.5.3 Clock configuration register (RCC_CFG)

Offset address: 0x08

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:23	Reserved		

Field	Name	R/W	Description
22:20	CLKOUTDIV	R/W	Clock output division 000: CLKOUT 1 frequency division 001: CLKOUT 2 frequency division 010: CLKOUT 4 frequency division 011: CLKOUT 8 frequency division 100: CLKOUT 16 frequency division 101: CLKOUT 32 frequency division 110: CLKOUT 64 frequency division 111: CLKOUT 128 frequency division
19	CLKOUTEN	R/W	Clock output enable 0: Disable 1: Enable
18	Reserved		
17:16	CLKOUTSEL	R/W	Clock output select 00: System clock 01: HSI clock 10: LSI clock 11: CPU HCLK clock
15:14	Reserved		
13:12	HSIDIV	R/W	Frequency division of the HSI branch of the system clock 00: No frequency division 01: 2 frequency division 10: 4 frequency division 11: 8 frequency division
11:10	Reserved		
9:8	APBCLKDIV	R/W	APB clock division 00: HCLK 01: HCLK 2 frequency division 10: HCLK 4 frequency division 11: HCLK 8 frequency division
7	Reserved		
6:4	AHBCLKDIV	R/W	AHB clock division 000: System clock 001: System clock 2 frequency division 010: System clock 4 frequency division 011: System clock 8 frequency division 100: System clock 16 frequency division 101: System clock 32 frequency division 110: System clock 64 frequency division 111: System clock 128 frequency division
3:2	Reserved		
1	SWSTS	R	Current system clock flag 0: HSI is the system clock 1: LSI is the system clock

Field	Name	R/W	Description
0	SWSEL	R/W	System clock select 0: HSI 1: LSI

5.5.4 Clock interrupt enable register (RCC_IER)

Offset address: 0x0C

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:3	Reserved		
2	HSIRDYIEN	R/W	HSI clock stabilization interrupt enable 0: Disable 1: Enable
1	Reserved		
0	LSIRDYIEN	R/W	LSI clock stabilization interrupt enable 0: Disable 1: Enable

5.5.5 Clock interrupt flag register (RCC_ISR)

Offset address: 0x10

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:3	Reserved		
2	HSIRDYFLG	R/W	HSI clock stabilization interrupt flag 0: No stabilization interrupt occurs 1: Stabilization interrupt occurs Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.
1	Reserved		
0	LSIRDYFLG	R/W	LSI clock stabilization interrupt flag 0: No stabilization interrupt occurs 1: Stabilization interrupt occurs Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.

5.5.6 AHB peripheral reset register (RCC_AHBRST)

Offset address: 0x14

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:18	Reserved		
17	GPIOBRST	R/W	GPIOB reset 0: Not reset 1: Reset
06	GPIOARST	R/W	GPIOA reset 0: Not reset 1: Reset
15:5	Reserved		
4	DIVRST	R/W	Divider reset 0: Not reset 1: Reset
3	CRCRST	R/W	CRC reset 0: Not reset 1: Reset
2:1	Reserved		
0	DMARST	R/W	DMA reset 0: Not reset 1: Reset

5.5.7 APB peripheral reset register (RCC_APBRST)

Offset address: 0x18

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:25	Reserved		
24	COMP1RST	R/W	COMP1 reset 0: Not reset 1: Reset
23	COMP0RST	R/W	COMP0 reset 0: Not reset 1: Reset
22	OPARST	R/W	OPAMP reset 0: Not reset 1: Reset
21	ADCRST	R/W	ADC reset 0: Not reset 1: Reset
20:18	Reserved		
17	I2CRST	R/W	I2C reset 0: Not reset 1: Reset

Field	Name	R/W	Description
16	Reserved		
15	UARTRST	R/W	UART reset 0: Not reset 1: Reset
14	USARTRST	R/W	USART reset 0: Not reset 1: Reset
13	Reserved		
12	SPIRST	R/W	SPI reset 0: Not reset 1: Reset
11	WWDTRST	R/W	WWDT reset 0: Not reset 1: Reset
10:8	Reserved		
7	BTMR1RST	R/W	BTMR1 reset 0: Not reset 1: Reset
6	BTMR0RST	R/W	BTMR0 reset 0: Not reset 1: Reset
5	Reserved		
4	GTMRRST	R/W	GTMR reset 0: Not reset 1: Reset
3	ATMRRST	R/W	ATMR reset 0: Not reset 1: Reset
2	EINTRST	R/W	EINT reset 0: Not reset 1: Reset
1:0	Reserved		

5.5.8 AHB peripheral clock enable register (RCC_AHBCLKEN)

Offset address: 0x1C

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:18	Reserved		

Field	Name	R/W	Description
17	GPIOBEN	R/W	GPIOB clock enable 0: Disable 1: Enable
16	GPIOAEN	R/W	GPIOA clock enable 0: Disable 1: Enable
15:5	Reserved		
4	DIVEN	R/W	Divider clock enable 0: Disable 1: Enable
3	CRCEN	R/W	CRC clock enable 0: Disable 1: Enable
2:1	Reserved		
0	DMAEN	R/W	DMA clock enable 0: Disable 1: Enable

5.5.9 APB peripheral clock enable register (RCC_APBCLKEN)

Offset address: 0x20

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:25	Reserved		
24	COMP1EN	R/W	COMP1 clock enable 0: Disable 1: Enable
23	COMP0EN	R/W	COMP0 clock enable 0: Disable 1: Enable
22	OPAEN	R/W	OPAMP clock enable 0: Disable 1: Enable
21	ADCEN	R/W	ADC clock enable 0: Disable 1: Enable
20:18	Reserved		
17	I2CEN	R/W	I2C clock enable 0: Disable 1: Enable
16	Reserved		

Field	Name	R/W	Description
15	UARTEN	R/W	UART clock enable 0: Disable 1: Enable
14	USARTEN	R/W	USART clock enable 0: Disable 1: Enable
13	Reserved		
12	SPIEN	R/W	SPI clock enable 0: Disable 1: Enable
11	WWDTEN	R/W	WWDT clock enable 0: Disable 1: Enable Note: This bit is not write-protected. If 0 is written, WWDT will stop counting.
10	IWDTEN	R/W	IWDT clock enable 0: Disable 1: Enable The software can only write 1. Writing 0 is invalid.
9:8	Reserved		
7	BTMR1EN	R/W	BTMR1 clock enable 0: Disable 1: Enable
6	BTMR0EN	R/W	BTMR0 clock enable 0: Disable 1: Enable
5	Reserved		
4	GTMREN	R/W	GTMR clock enable 0: Disable 1: Enable
3	ATMREN	R/W	ATMR clock enable 0: Disable 1: Enable
2	EINTEN	R/W	EINT clock enable 0: Disable 1: Enable
1:0	Reserved		

5.5.10 ADC clock control register (RCC_ADCCR)

Offset address: 0x24

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:2	Reserved		
1:0	ADCCLKDIV	R/W	ADCCLK clock divide 00: System clock 2 frequency division 01: System clock 4 frequency division 10: System clock 8 frequency division 11: System clock 16 frequency division

5.5.11 Reset control/status register (RCC_RSTCSR)

Offset address: 0x28

Reset value: 0x0000 0080

Power domain: AON domain

Reset range: POR reset

Field	Name	R/W	Description
31:16	Reserved		
15	LOCKUPRSTEN	R/W	LOCKUP system reset enable 0: Disable LOCKUP system reset 1: When CPU generates LOCKUP, the chip performs a system reset
14	PVDRSTEN	R/W	PVD system reset enable 0: Disable PVD system reset 1: When the PVD monitoring voltage is lower than the threshold, the chip performs a system reset
13:10	Reserved		
9:8	NRSTFLTSEL	R/W	NRST filter select 00: No filtering is performed; only 50ns filtering is simulated 01: Digital filtering 1ms 10: Digital filtering 5ms 11: Digital filtering 10ms Note: When the LSI is turned off, the NRST filtering time is 5 LSI clock cycles longer than the configured value. When an NRST reset event occurs while configuring this bit, the filtering time will be uncontrollable.
7	PORRSTFLG	R/W	POR/PDR reset flag 0: No POR/PDR reset occurred 1: POR/PDR reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.
6	LOCKUPRSTFLG	R/W	LOCKUP system reset flag 0: No LOCKUP system reset occurred 1: LOCKUP system reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.

Field	Name	R/W	Description
5	WWDTRSTFLG	R/W	Window watchdog WWDT system reset flag 0: No WWDT system reset occurred 1: WWDT system reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.
4	IWDTRSTFLG	R/W	Independent watchdog IWDT system reset flag 0: No IWDT system reset occurred 1: IWDT system reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.
3	SFTRSTFLG	R/W	Software system reset flag 0: No software system reset occurred 1: Software system reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.
2	PVDRSTFLG	R/W	PVD system reset mark 0: No PVD system reset occurred 1: PVD system reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.
1	NRSTRSTFLG	R/W	NRST pin system reset mark 0: No NRST pin system reset occurred 1: NRST pin system reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.
0	OPTRSTFLG	R/W	Option byte load system reset flag 0: No option byte loading system reset occurred 1: Option byte load system reset occurred Note: Set 1 by hardware, cleared by software writing 0, writing 1 is invalid.

5.5.12 AON domain control register（RCC_AONCSR）

Offset address: 0x2C

Reset value: 0x0000 0000

Power domain: AON domain

Reset range: POR reset and NRST reset will reset all bits of this register, and LPTMR reset will reset LPTMREN. When entering STANDBY mode 0, all these registers are reset. When entering STANDBY Mode 1, this register is not affected.

Field	Name	R/W	Description
31:17	Reserved		
16	LPTMRRST	R/W	LPTMR reset 0: Not reset 1: Reset

Field	Name	R/W	Description
15	LPTMREN	R/W	LPTMR clock enable 0: Disable 1: Enable
14:2	Reserved		
1	LSIRDY	R	LSI clock stability flag 0: Unstable 1: Stable When LSIEN is turned off, LSIRDY is cleared after 2 LSI clock cycles.
0	LSIEN	R/W	LSI clock enable 0: Disable 1: Enable When LSI is used as the system clock, writing 0 is invalid.

6 Power management unit (PMU)

6.1 Full Name and Abbreviation Description of Terms

Table 18 Full Name and Abbreviation Description of Terms

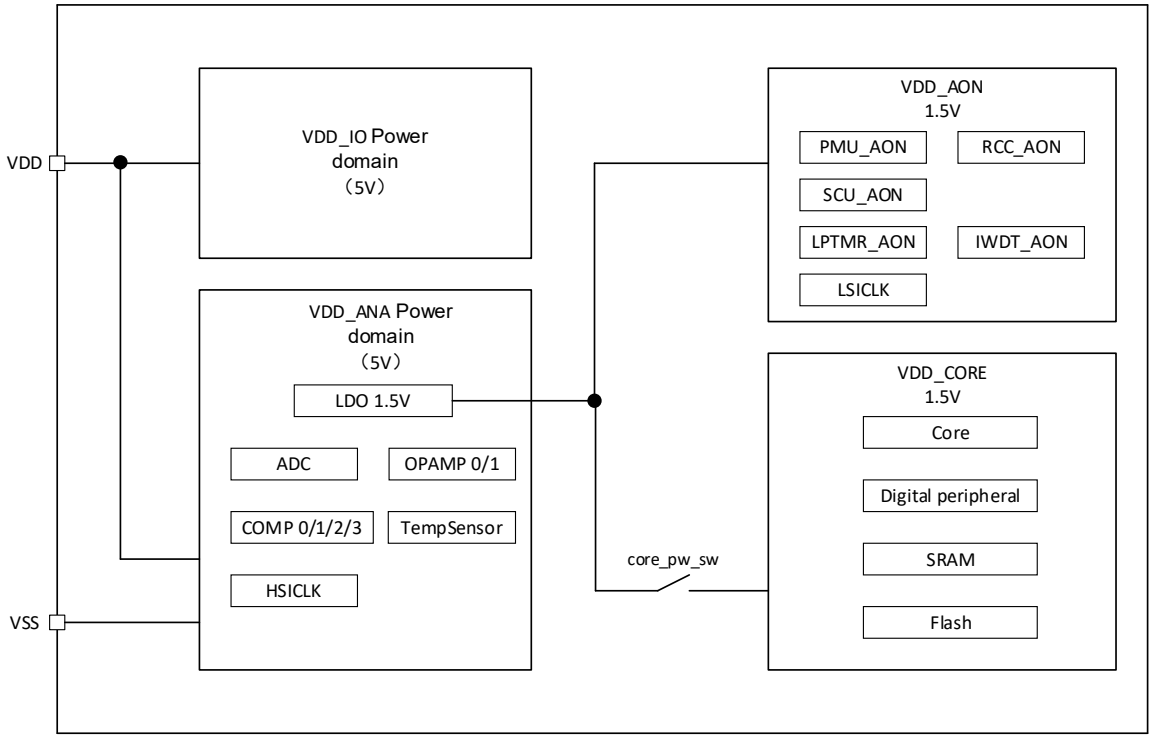
Full name in English	English abbreviation
Power Management Unit	PMU
Power On Reset	POR
Power Down Reset	PDR
Power Voltage Detector	PVD

6.2 Introduction

The power supply is the foundation for stable operation of a system, with an operating voltage of 1.8~5.5V, and 1.5V power supply can be provided by the built-in voltage regulator.

6.3 Structure block diagram

Figure 11 Power Supply Control Structure Block Diagram



6.4 Functional description

6.4.1 Power domain

The power domain of the product includes: analog IO power domain, analog power domain, AON power domain and CORE power domain.

Table 19 Power Domains

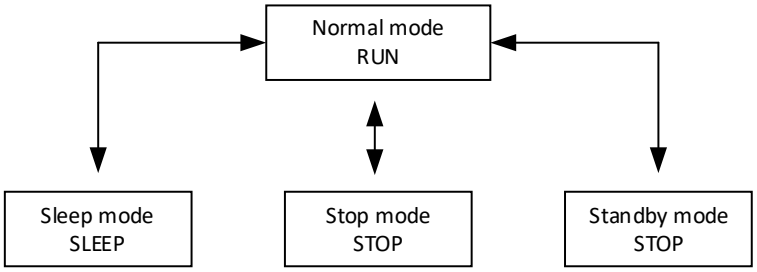
Name	Voltage range (V)	Description
IO power domain	5V	Power supply for all analog I/O is included.
analog power domain	5V	It includes analog modules such as ADC, OPAMP, COMP, TS, HSI clock, and provides an LDO with a 1.5V power supply.
AON power domain	1.5V	It includes all digital non-power-off modules and LSI clocks.
CORE power domain	1.5V	It includes a core, FLASH memory, SRAM memory, and digital peripheral modules.

6.4.2 Power control

6.4.2.1 Reduce the power in low-power mode

There are three low-power modes: sleep mode, stop mode and standby mode. The power is reduced by disabling the core and clock source and setting the voltage regulator. The state transition between low-power mode and normal mode (RUN) is as shown in the figure below.

Figure 12 Low-power Mode Transition



SLEEP mode only stops the CPU core clock, and other peripherals work normally; the working status of peripherals in STOP mode, STANDBY0 mode and STANDBY1 mode is shown in the table.

Table 20 The peripheral working states of STOP mode, STANDBY0 mode and STANDBY1 mode

	STOP		STANDBY0		STANDBY1	
	state (1)	wake-up source	state (1)	wake-up source	state (1)	wake-up source
CPU core	-	-	-	-	-	-
FLASH	-	-	-	-	-	-

	STOP		STANDBY0		STANDBY1	
	state (1)	wake-up source	state (1)	wake-up source	state (1)	wake-up source
SRAM	-（2）	-	-（2）	-	-	-
HSI	-	-	-	-	-	-
LSI	O	-	-	-	O	-
POR/PDR	Y	Y	Y	Y	Y	Y
SYSTICK	-	-	-	-	-	-
PVD	O	O	-	-	-	-
DIV_SHIFT	-	-	-	-	-	-
CRC	-	-	-	-	-	-
GPIO	O	O（3）	O	O（3）	O	O（3）
DMA	-	-	-	-	-	-
LPTMR	O	O	-	-	O	O
COMP	-	-	-	-	-	-
OPAMP	-	-	-	-	-	-
ADC	-	-	-	-	-	-
I2C	-	-	-	-	-	-
UART	-	-	-	-	-	-
USART	-	-	-	-	-	-
SPI	-	-	-	-	-	-
WWDT	-	-	-	-	-	-
IWDT	O	O	-	-	O	O
BTMR1	-	-	-	-	-	-
BTMR0	-	-	-	-	-	-
GTMR	-	-	-	-	-	-
ATMR	-	-	-	-	-	-
EINT	O	O	-	-	-	-

Note:

(1) Y indicates normal operation, O indicates configurable operation (off by default, can be enabled by software), - indicates unavailable.

(2) In the STOP mode of SRAM, data is retained. In the STANDBY mode, when power is lost, data is not retained.

(3) The GPIO pins can be awakened in STOP mode by configuring the EINT module. In STANDBY mode, the PMU module can be configured for wake-up. The pins that support STANDBY wake-up are PA7 and PB2 respectively. In addition, the NRST pin can also be awakened, but the reset range is different. For details, please refer to the RCC section.

Normal mode

When the chip is powered on for the first time or exits from low power mode, it enters RUN mode by default. The system clock is HSICLK by default.

Note: When the system is first powered on or awakened from STANDBY, the system clock frequency is 64MHz. When the system is normally awakened from STOP, since the system configuration HSIDIV is not affected by STOP, the system clock will be configured to different frequencies according to its frequency division.

Sleep mode

The characteristics of sleep mode are shown in the table below

Table 21 Characteristics of Sleep Mode

Characteristics	Description
Enter	Set the SLEEPDEEP bit of the core register to 0 Confirm there are no pending interrupts (WFI) or events (WFE) Execute the WFI or WFE instruction to enter SLEEP mode immediately
After entering	All modules except the CPU core work normally
Wake-up	If entering SLEEP mode by executing WFI, it will be woken up by any interrupt If entering SLEEP mode by executing WFE, it will be woken up by any event
Wake up delay	Several HCLK cycles
After wake-up	The system restores the original working state

Note: Any system reset can wake up the sleep mode, and the system re-executes the program after waking up.

Stop mode

The characteristics of stop mode are shown in the table below:

Table 22 Characteristics of Stop Mode

Characteristics	Description
Enter	Set the SLEEPDEEP bit of the core register to 1 Set the LPM bit of PMU_LPCR in the PMU module to 0 Confirm there are no pending interrupts (WFI) or events (WFE) Execute the WFI or WFE instruction to enter STOP mode immediately
After entering	All modules except IWDT and LPTMR stop working, HSI clock is turned off, LSI clock remains unchanged, FLASH enters DEEP STANDBY state, and SRAM data is retained
Wake-up	If WFI is executed to enter the STOP mode, it will be awakened through the relevant interrupt configured by EINT. If it is awakened through PVD or LPTMR, the corresponding peripheral interrupt needs to be enabled separately. If you perform the WFE into the STOP mode, awakened by EINT configuration related events
Wake up delay	HSI analog clock establishment and stabilization time + FLASH exit DEEP STANDBY time (Awakened with HSI64MHz, MIN 20μs)
After wake-up	The system clock automatically switches to the HSI clock and operates at the corresponding frequency based on the HSIDIV configuration If it is necessary to use LSI as the system clock after waking up, software operation is required for switching

Note: Any system reset can wake up the shutdown mode. After waking up, the system will re-execute the program. Since HSIDIV will be reset by the system, the system will operate at 64MHz.

Standby mode

The characteristics of standby mode are shown in the table below:

Table 23 Characteristics of Standby Mode

Characteristics	Description
Enter	Set the SLEEPDEEP bit of the core register to 1 Set the LPM bit of PMU_LPCR in the PMU module to 1 Confirm there are no pending interrupts (WFI) or events (WFE) Confirm that no STANDBY wake-up event has occurred Execute the WFI or WFE instruction to enter STANDBY mode immediately
After entering	When configured to STANDBY0 mode, all peripherals stop working, the CORE domain power down, all clocks are turned off, FLASH power down, and SRAM power down When configured in STADNBY1 mode, IWDT and LPTMR are not affected, while the rest of the modules stop working. The CORE domain power down, the HSI clock turns off, the LSI clock remains unaffected, FLASH power down, and SRAM power down
Wake-up	When in STANDBY0 mode, it is awakened only through the rising/falling/double-edge of the WKUP pin When in STANDBY1 mode, it can be awakened through the WKUP pin, IWDT and LPTMR
Wake up delay	100μs
After wake-up	The system clock automatically switches to the HSI_SYS_CLK clock. Since HSIDIV has been reset, the frequency is fixed at 64MHz The system re-executes the program

Note:

(1) NRST can wake up STANDBY, but it will reset some of the AON registers additionally. For details, please refer to the RCC section

(2) When the system enters STANDBY, the FLASH will power down, the SRAM will power down and the data will not be retained. In STANDBY0 mode, LPTMR and IWDT will be forcibly reset. In STANDBY1 mode, it will continue to work. An additional peripheral wake-up enable needs to be configured to wake up the STANDBY mode. After waking up, the IWDT will be reset and the LPTMR can continue to work.

(3) Before the system enters STANDBY, the wake-up enable and pull-down mode of the wake-up pin need to be configured according to the requirements. This configuration bit is in the PMU_WKCR register. Once the wake-up pin is enabled, the function of the corresponding wake-up pin will not be affected by the configuration in the GPIO controller. The input enable of this pin is turned on and the output enable is turned off. The pull-down mode is controlled by PMU_WKCR. After entering STANDBY, the pin configuration in the GPIO controller will be reset.

(4) The STANDBY wake-up pin has a fixed 50ns analog filter, and this filter function does not take effect on other functions of this pin.

The software operation process for STANDBY entry:

(1) In the configuration of PMU_WKCR, the pull-up and pull-down requirements for the wake pin are WKPUSx, and the wake polarity of the wake pin is WKPOLx

(2) Enable the wake-up pin, set WKENx to 1. (Note that after enabling, the corresponding pull-down configuration and polarity configuration will have write protection, and the corresponding pin will force input to enable and output to turn off. The pull-down relationship will be replaced by the configuration of the IO controller with the pull-down configuration for PMU wake-up.) To avoid false triggering of wake-up due to filtering, the wake-up detection function will be delayed by 6 system clocks after configuring WKENx.

(3) Confirm that there is no wake flag in PMU_WKSR, that is, all WKFLGx are 0

(4) Configure LPCFG=1 in PMU_LPCR and perform WFI/WFE for deep sleep to enter STANDBY

6.4.3 Programmable Voltage Detector

Power-On Reset (POR), Power-Down Reset (PDR) and Programmable Voltage Detector (PVD) circuits are integrated inside the chip.

POR/PDR reset is always working. When the power supply voltage is detected to be lower than the threshold (V_{POR}/PDR), the system remains in the reset state and all registers will be reset.

The PVD function can monitor the comparison between V_{DD} and the V_{PVD} threshold. The PVD function can be enabled through PVDEN, and the threshold can be selected through the PVDTHSEL[2:0] bit in PMU_PVDCSR.

Function	Description
PVD status	The PVDSTS flag in PMU_PVDCSR can indicate the current VDD state. To avoid false triggering, when the PVD function is turned off or just turned on for about 30μs (during the function establishment time), this flag remains 0. Users can identify whether the function has been established by the PVDRDYFLG flag, and then read the correct PVDSTS flag.
PVD interrupt	PVDHT and PVDLT in PMU_PVDCSR can monitor above/below the threshold. If a monitoring event is triggered, PVDFLG will be automatically set to 1. If the channel interrupt is configured through the EINT module, the system will respond to the interrupt, and the MCU can be set to a safe state through the interrupt service program.
PVD system reset	Configure PVDRSTEN in RCC_RSTCSR to enable the PVD system reset function. When VDD is lower than the threshold, the system enters system reset until VDD returns to above the threshold voltage, then it resumes normal mode and re-executes the program.

Note:

(1) When users use PVD interrupts and PVD system resets, they do not need to wait for the PVDRDYFLG stability flag. The system will automatically handle the establishment delay to avoid false triggering.

(2) PVD includes a system clock for filtering, but the filtering function does not support operation in STOP mode. Before entering the STOP mode, the user must turn off the filtering function to ensure that the PVD continues to work in the STOP mode.

6.5 Register address mapping

Table 24 RCC Register Address Mapping

Register name	Description	Offset address
PMU_KEY	PMU read-write protection register	0x00
PMU_LPCR	PMU low-power mode register	0x04
PMU_WKCR	PMU low-power wake-up control register	0x08
PMU_WKSR	PMU low-power wake-up state register	0x0C
PMU_PVDCSR	PVD control/ state register	0x10

6.6 Register functional description

6.6.1 PMU read-write protection register (PMU_KEY)

Offset address: 0x00

Reset value: 0x0000 00000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset, system reset

Field	Name	R/W	Description
31:17	Reserved		
16	LOCKFLG	R	PMU register write protect flag 0: The PMU register is in the protected state 1: The PMU register is in readable and writable state Note: When 0x87E4 is written to the LOCKKEY, this bit 1; When other values are written to the LOCKKEY, this bit is reset to zero.
15:0	LOCKKEY[15:0]	W	Password protection configuration for RCC register write operations Writing 0x87E4 to unlock, and LOCKFLG is automatically set 1 at the same time. When writing any other value, a lock is added and LOCKFLG is reset to zero at the same time.

6.6.2 PMU low-power mode register (PMU_LPCR)

Offset address: 0x04

Reset value: 0x0000 00000

Power domain: AON domain

Reset range: POR reset, NRST reset

Field	Name	R/W	Description
31:2	Reserved		
1	STANDBYCFG	R/W	STANDBY work mod 0: After the system enters standby mode 0, the LSI is shut down, and the LPTMR and IWDT are reset and do not work 1: After the system enters standby mode 1, tne LSI remains, LPTMR and IWDT continue to work, and STANDBY can be wake-up Note: In this mode, a STANDBY wake-up will reset IWDT.
0	LPCFG	R/W	Low power mode 0: STOP mode 1: STANDBY mode

6.6.3 PMU low-power wake-up control register (PMU_WKCR)

Offset address: 0x08

Reset value: 0x0000 00000

Power domain: AON domain

Reset range: POR reset, NRST reset

Field	Name	R/W	Description
31:26	Reserved		
25	LPTMRWKEN	RW	LPTMR wake up STANDBY enable 0: LPTMR does not wake up STANDBY 1: Wake up STANDBY in STANDBY1 mode Note: LPTMR cannot be wake up in STANDBY0 mode
24	IWDTWKEN	RW	IWDT wake up STANDBY enable 0: IWDT does not wake up STANDBY 1: Wake up STANDBY in STANDBY1 mode Note: IWDT cannot be wake up in STANDBY0 mode
23:20	Reserved		
19:18	WKPUS1	RW	Up and down pull control of wake pin 1 in STANDBY mode 0x: No up or down pull 10: Pull up 11: Pull down Note: Once WKEN1 is enabled, this bit cannot be modified.
17:16	WKPUS0	RW	Up and down pull control of wake pin0 in STANDBY mode 0x: No up or down pull 10: Pull up 11: Pull down Note: Once WKEN0 is enabled, this bit cannot be modified.
15:12	Reserved		
11:10	WKPOL1	RW	Wake-up polarity of the STANDBY pair on the wake-up pin 1 00: Rising edge 01: Falling edge 1x: Double edges Note: Once WKEN1 is enabled, this bit cannot be modified.

Field	Name	R/W	Description
9:8	WKPOL0	RW	Wake-up polarity of the STANDBY pair on the wake-up pin 0 00: Rising edge 01: Falling edge 1x: Double edges Note: Once WKEN0 is enabled, this bit cannot be modified.
7:2	Reserved		
1	WKEN1	R/W	Wake pin 1 for STANDBY enable 0: Disable 1: Enable
0	WKEN0	R/W	Wake pin 0 for STANDBY enable 0: Disable 1: Enable

6.6.4 PMU low-power wake-up state register (PMU_WKSR)

Offset address: 0x0C

Reset value: 0x0000 00000

Power domain: AON domain

Reset range: POR reset, NRST reset

Field	Name	R/W	Description
31:26	Reserved		
25	LPTMRWKFLG	RW	LPTMR for the STANDBY wake-up flag 0: No LPTMR wake-up occurred 1: LPTMR wake-up occurred Note: It is recommended to use it in STANDBY1 mode. In STANDBY0 mode, setting this flag to one will not wake up STANDBY. Set the hardware to 1, write the software to 0 to clear it, and writing to 1 is invalid.
24	IWDTWKFLG	RW	IWDT for the STANDBY wake-up flag 0: No IWDT wake-up occurred 1: IWDT wake-up occurred Note: It is recommended to use it in STANDBY1 mode. In STANDBY0 mode, setting this flag to one will not wake up STANDBY. Set the hardware to 1, write the software to 0 to clear it, and writing to 1 is invalid.
23:5	Reserved		
4	STANDBYFLG	RW	STANDBY event flag 0: No STANDBY entry event occurred 1: A STANDBY entry event occurred Set the hardware to 1, write the software to 0 to clear it, and writing to 1 is invalid.
3:2	Reserved		

Field	Name	R/W	Description
1	WKFLG1	RW	Wake pin 1 pairs with the STANDBY wake-up flag 0: No wake-up of this pin occurred 1: This pin has been wake-up Set the hardware to 1, write the software to 0 to clear it, and writing to 1 is invalid.
0	WKFLG0	RW	Wake pin 0 pairs with the STANDBY wake-up flag 0: No wake-up of this pin occurred 1: This pin has been wake-up Set the hardware to 1, write the software to 0 to clear it, and writing to 1 is invalid.

6.6.5 PVD control/status register (PMU_PVDCSR)

Offset address: 0x10

Reset value: 0x0000 0000

Power domain: CORE domain

Reset range: POR reset, NRST reset, STANDBY reset

Field	Name	R/W	Description
31:17	Reserved		
16	PVDRDYFLG	R	PVD function establishes stable flag 0: The PVD function is turned off or is within the startup setup period 1: The PVD function is now ready Note: When PVDEN is not enabled or during the setup startup time, this flag remains at 0.
15	PVDSTS	R	PVD monitoring results output 0: VBB is lower than PVD threshold voltage 1: VBB is higher than PVD threshold voltage Note: When PVDEN is not enabled or during the setup startup time, this flag remains 0.
14	PVDFLG	R/W	PVD interrupt event flag 0: No monitoring events occurred 1: A monitoring incident occurred Note: Set the hardware to 1, write 0 to clear the software, and writing 1 is invalid.
13	Reserved		
12:10	PVDFLTSEL	R/W	PVD filter length select 000:64 system clocks (1us at 64 MHZ) 001:128 system clocks (2us at 64MHz) 010:192 system clocks (3us at 64MHz) 011:320 system clocks (5us at 64MHz) 100:640 system clocks (10us at 64MHz) 101:1280 system clocks (20us at 64MHz) 110:1920 system clocks (30us at 64MHz) 111:3200 system clocks (50us at 64MHz) Note: Once PVDEN is enabled, this bit cannot be modified.

Field	Name	R/W	Description
9	PVDFLTEN	R/W	PVD filter enable 0: Disable 1: Enable Note: Once PVDEN is enabled, this bit cannot be modified.
8	PVDIEN	R/W	PVD monitoring event interrupts the enable 0: Disable 1: Enable
7:6	Reserved		
5	PVDHT	R/W	PVD above threshold voltage monitoring 0: Do not monitor VDD voltages above the threshold 0: When VDD is higher than the threshold voltage, PVDFLG is set to 1, which can trigger an interrupt Note: After PVDEN is enabled, this bit cannot be modified.
4	PVDLT	R/W	PVD below threshold voltage monitoring 0: Do not monitor VDD voltages below the threshold 1: When VDD is lower than the threshold voltage, PVDFLG is set to 1, which can trigger an interrupt Note: After PVDEN is enabled, this bit cannot be modified.
3:1	PVDTHSEL[2:0]	R/W	PVD threshold voltage configure 000: Threshold voltage 1.9V/2.0V 001: Threshold voltage 2.3V/2.4V 010: Threshold voltage 2.7V/2.8V 011: Threshold voltage 3.1V/3.2V 100: Threshold voltage 3.5V/3.6V 101: Threshold voltage 3.9V/4.0V 110: Threshold voltage 4.3V/4.4V 111: Threshold voltage 4.7V/4.8V Note: After PVDEN is enabled, this setting cannot be modified.
0	PVDEN	R/W	PVD enable 0: Disable 1: Enable

7 Nested Vector Interrupt Controller (NVIC)

7.1 Full Name and Abbreviation Description of Terms

Table 25 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Non Maskable Interrupt	NMI

7.2 Introduction

The Cortex-M0+ core in the product integrates nested vectored interrupt controller (NVIC), which is closely coupled with the core, and can handle exceptions and interrupts and power management control efficiently and with low delay. Please see Cortex-M0+ Technical Reference Manual for more instructions about NVIC.

7.3 Main characteristics

(1) 32 maskable interrupt channels

(2) 4 programmable priority levels

(3) Power management control

(4) Low-delay exception and interrupt processing

(5) Realization of system control register

7.4 Interrupt and exception vector table

Table 26 Interrupt and Exception Vector Table

Name	Vector No.	Priority	Vector address	Description
	-	-	0x0000_0000	Reserved
Reset	-	-3	0x0000_0004	Reset
NMI	-	-2	0x0000_0008	Non-maskable interrupt
HardFault	-	-1	0x0000_000c	Various hardware faults
SVCall	-	Can set	0x0000_002C	System service called by general SWI instruction
PendSV	-	Can set	0x0000_0038	Pending system service
SysTick	-	Can set	0x0000_003C	System tick timer
WWDT	0	Can set	0x0000_0040	Window watchdog interrupt
PVD	1	Can set	0x0000_0044	Supply voltage detection interrupt
-	2	-	0x0000_0048	Reserved
FLASH	3	Can set	0x0000_004C	Flash memory global interrupt

Name	Vector No.	Priority	Vector address	Description
RCC	4	Can set	0x0000_0050	RCC interrupt
EINT0_1	5	Can set	0x0000_0054	EINT line [1:0] interrupt
EINT2_3	6	Can set	0x0000_0058	EINT line [3:2] interrupt
EINT4_15	7	Can set	0x0000_005C	EINT line [15:4] interrupt
-	8	-	0x0000_0060	Reserved
DMA_CH0	9	Can set	0x0000_0064	DMA channel 0 interrupt
DMA_CH1	10	Can set	0x0000_0068	DMA channel 1 interrupt
-	11	-	0x0000_006C	Reserved
ADC	12	Can set	0x0000_0070	ADC interrupt
ATMR_BRK_ UP_TRG_COM	13	Can set	0x0000_0074	Atimer_BRK/UP/TRG and COM interrupt
ATMR_CC	14	Can set	0x0000_0078	Atimer compare interrupt
GTMR	15	Can set	0x0000_007C	Gtimer interrupt
-	16	-	0x0000_0080	Reserved
BTMR0	17	Can set	0x0000_0084	Btimer 0 interrupt
BTMR1	18	Can set	0x0000_0088	Btimer 1 interrupt
LPTMR	19	Can set	0x0000_008C	LPtimer interrupt
-	20	-	0x0000_0090	Reserved
COMP0	21	Can set	0x0000_0094	COMP0 interrupt
COMP1/2/3	22	Can set	0x0000_0098	COMP1/2/3 interrupt
I2C	23	Can set	0x0000_009C	I2C interrupt
-	24	-	0x0000_00A0	Reserved
SPI	25	Can set	0x0000_00A4	SPI interrupt
-	26	-	0x0000_00A8	Reserved
UART	27	Can set	0x0000_00AC	UART interrupt
USART	28	Can set	0x0000_00B0	USART interrupt
-	29	-	0x0000_00B4	Reserved
-	30	-	0x0000_00B8	Reserved
-	31	-	0x0000_00BC	Reserved

8 External interrupt and event controller (EINT)

8.1 Introduction

The interrupts/events are divided into internal interrupts/events and external interrupts/events. In this manual, external interrupt refers to the interrupt/event caused by I/O pin input signal, which is EINTx in interrupt vector table; other interrupts are internal interrupts/events.

The events can be divided into hardware events and software events. Hardware events are generated by external/core hardware signals, while software events are generated by instructions.

Interrupts need to go through the interrupt handler function to implement the work to be processed, while events do not need to go through interrupt handler function, and the preset work can be triggered by hardware. The external events can output pulse by events such as GPIO, while the internal events trigger another timer to work, for example, by an update event of a timer.

8.2 Main characteristics

(1) Support 18 event/interrupt requests

(2) Each event/interrupt line can be masked independently

(3) The internal line is automatically disabled when the system is not in the stop mode

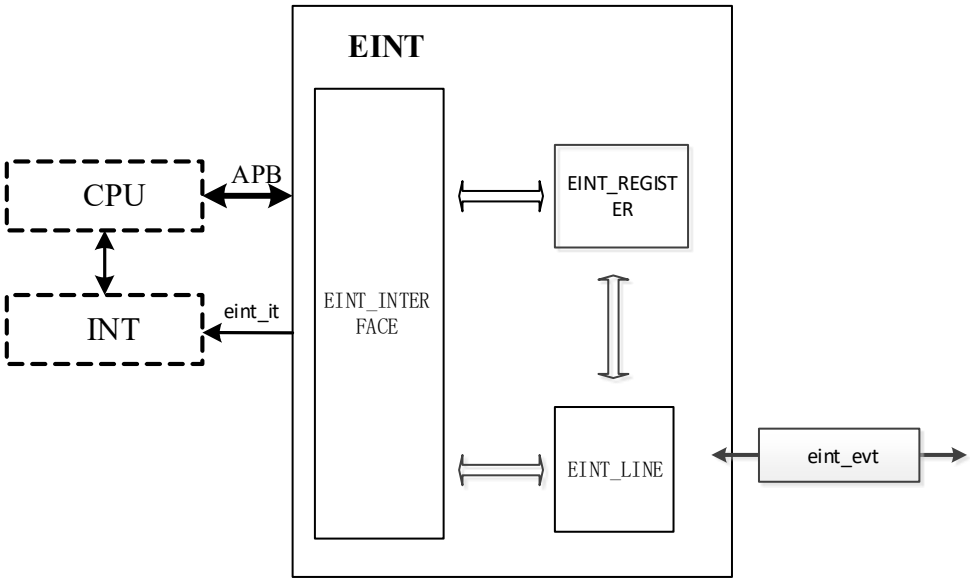
(4) Each external event/interrupt line can be triggered independently

(5) Each external interrupt line has dedicated status bit

(6) Detect external signals with pulse width lower than APB clock width

8.3 Structure block diagram

Figure 13 Structure Block Diagram



8.4 Functional description

8.4.1 Classification and difference of "external interrupt and event"

"External interrupt and event" can be classified into external hardware interrupt, external hardware event, external software event and external software interrupt according to trigger source, configuration and execution process. The differences are shown in the table below:

Table 27 Classification and Differences of "External Interrupts and Events"

Name	Trigger source	Configuration and execution process
External hardware interrupt	External signal	(1) Set the trigger mode, allow the interrupt request, and enable corresponding peripheral interrupt line (enable in NVIC). (2) When an edge consistent with the configuration is generated on the external interrupt line, an interrupt request will be generated, and the corresponding pending bit will be set to 1; write 1 to the corresponding bit of the pending register and the interrupt request will be cleared.
External hardware event	External signal	(1) Set the trigger mode and enable the event line. (2) When an edge consistent with the configuration is generated on the external event line, an event request pulse will be generated, and the corresponding pending bit will not be set to 1.
External software event	Software interrupt register/transmit event (SEV) instruction	(1) Enable the event line. (2) Write 1 to the software interrupt event register of the corresponding event line to generate an event request pulse, and the corresponding pending bit will not be set to 1.

Name	Trigger source	Configuration and execution process
External software interrupt	Software interrupt register	(1) Allow interrupt request, and enable the corresponding peripheral interrupt line (enable in NVIC). (2) Write 1 to the software interrupt event register of the corresponding interrupt line to generate an interrupt request, the corresponding pending bit will be set to 1; write 1 to the corresponding bit of the pending register and the interrupt request will be cleared.

8.4.2 Wake-up

Set trigger registers RTEN or FTEN for the required edge detection. When the edge on the external interrupt line occurs, the edge is locked and a wake-up signal is output.

Before the PMU enters STOP, it is necessary to confirm that there are no suspended interrupts (WFI) or events (WFE). If so, the software needs to write the IPEND register to clear the suspend bit to avoid being unable to enter the STOP.

8.4.3 Event

To generate an event, first the event line shall be configured and enabled. Set 2 trigger registers according to the required edge detection, and write "1" in the corresponding bit of the event mask register to enable event requests. When the required edge occurs on the event line, an event request pulse is generated, and the corresponding pending bit is not set to "1".

8.4.4 Interrupt

To generate an interrupt, first the interrupt line shall be configured and enabled. Set 2 trigger registers according to the required edge detection, and write "1" in the corresponding bit of the interrupt mask register to allow interrupt requests. When the expected edge occurs on the external interrupt line, an interrupt request is generated, and the corresponding pending bit is set to "1". Writing "1" in the corresponding bit of the suspend register will clear the interrupt request.

8.4.5 External interrupt and event line mapping

Table 28 External Interrupt and Event Line Mapping

External Interrupt and Event Channel Name	External Interrupt and Event Line No.
PA0/PB0	EINT 0
PA1/PB1	EINT 1
...	...
PA12/PB12	EINT 12
PA13	EINT 13
PA14	EINT 14

External Interrupt and Event Channel Name	External Interrupt and Event Line No.
PA15	EINT 15
PVD output	EINT 16
LPTMR	EINT 17
Reserved	EINT 18~ EINT 31

8.4.6 Direction for use

(1) Configure EINT_CFG0/ EINT_CFG1 and select the PAD channel

(2) Configure the EINT_IMASK/ EINT_EMASK register and turn on the interrupt or event masking switch

(3) Configure the EINT_RTEN/ EINT_FTEN registers and select rising edge triggering or falling edge triggering

(4) If the software is to trigger interrupts or events, the EINT_SWIEN register can be configured

(5) After the interrupt occurs, the EINT_IPEND register is pulled up. The software can clear 0 by changing the polarity of the edge detection or by writing 1 to this bit

(6) Before switching the PAD channel, turn off EINT_RTEN/ EINT_FTEN first

8.5 Register address mapping

Table 29 External Interrupt/Event Controller Register Address Mapping

Register name	Description	Offset address
EINT_IMASK	Interrupt mask register	0x00
EINT_EMASK	Event mask register	0x04
EINT_RTEN	Enable the rising edge to trigger the register	0x08
EINT_FTEN	Enable the falling edge to trigger the register	0x0C
EINT_SWIEN	Software interrupt event register	0x10
EINT_IPEND	Interrupt pending register	0x14
EINT_CFG0	Configuration register 0	0x18
EINT_CFG1	Configuration register 1	0x1C

8.6 Register functional description

8.6.1 Interrupt mask register (EINT_IMASK)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:0	IMASKx	R/W	Interrupt Request Mask on Line x (x=0~17) 0: Mask 1: Open

8.6.2 Event mask register (EINT_EMASK)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:0	EMASKx	R/W	Event Request Mask on Line x (x=0~17) 0: Mask 1: Open

8.6.3 Enable the rising edge to trigger the register (EINT_RTEN)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:0	RTENx	R/W	Rising Trigger Event Enable and Interrupt of Line x (x=0~17) 0: Disable 1: Enable

8.6.4 Enable the falling edge to trigger the register (EINT_FTEN)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:0	FTENx	R/W	Falling Trigger Event Enable and Interrupt of Line x (x=0~17) 0: Disable 1: Enable

8.6.5 Software interrupt event register (EINT_SWIEN)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:0	SWIENx	R/W	Software Interrupt Event on Line x (x=0~17) Set 1 by software, write 1 or clear 0 for the corresponding bit of EINT_IPEND.

Field	Name	R/W	Description
			When this bit is 0, the pending bit of EINT_IPEND can be set by writing 1. If EINT_IMASK (EINT_EMASK) is set to open the interrupt (event) request, an interrupt (event) will be generated. 0: No effect 1: Software generates an interrupt (event)

8.6.6 Interrupt pending register (EINT_IPEND)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:0	IPENDx	RC_W1	Interrupt Pending Occur of Line x Flag (x=0~17) Whether the selectable trigger request occurs 0: No 1: Occurred When a request is triggered by the corresponding edge of EINT_RTEN/EINT_FTEN on the external interrupt line, set 1 by hardware; clear 0 by changing the polarity of the edge detection or clear 0 by writing 1 to this bit.

8.6.7 Configuration register 0 (EINT_CFG0)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:29	Reserved		
28	EINT7	R/W	EINT7 pin Configure 0: PA7 1: PB7
27:25	Reserved		
24	EINT6	R/W	EINT6 pin Configure 0: PA6 1: PB6
23:21	Reserved		
20	EINT5	R/W	EINT5 pin Configure 0: PA5 1: PB5
19:17	Reserved		
16	EINT4	R/W	EINT4 pin Configure 0: PA4 1: PB4
15:13	Reserved		
12	EINT3	R/W	EINT3 pin Configure 0: PA3 1: PB3

Field	Name	R/W	Description
11:9	Reserved		
8	EINT2	R/W	EINT2 pin Configure 0: PA2 1: PB2
7:5	Reserved		
4	EINT1	R/W	EINT1 pin Configure 0: PA1 1: PB1
3:1	Reserved		
0	EINT0	R/W	EINT0 pin Configure 0: PA0 1: PB0

8.6.8 Configuration register 1 (EINT_CFG1)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:29	Reserved		
28	EINT15	R/W	EINT15 pin Configure 0: PA15 1: not have
27:25	Reserved		
24	EINT14	R/W	EINT14 pin Configure 0: PA14 1: not have
23:21	Reserved		
20	EINT13	R/W	EINT13 pin Configure 0: PA13 1: not have
19:17	Reserved		
16	EINT12	R/W	EINT12 pin Configure 0: PA12 1: PB12
15:13	Reserved		
12	EINT11	R/W	EINT11 pin Configure 0: PA11 1: PB11
11:9	Reserved		
8	EINT10	R/W	EINT10 pin Configure 0: PA10 1: PB10

Field	Name	R/W	Description
7:5	Reserved		
4	EINT9	R/W	EINT9 pin Configure 0: PA9 1: PB9
3:1	Reserved		
0	EINT8	R/W	EINT8 pin Configure 0: PA8 1: PB8

9 Direct memory access (DMA)

9.1 Full Name and Abbreviation Description of Terms

Table 30 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Global	G
Transfer	T
Half	H
Complete	C
Error	E
Channel	CH
Circular	CIR
Peripheral	PER
Increment	I
Memory	M
Priority	PRI
Number	N
Address	ADDR

9.2 Introduction

DMA (Direct Memory Access) can realize direct data transmission between peripheral devices and memory or between memory and memory without CPU intervention, thus saving CPU resources for other operations.

The product has a DMA controller, with 2 channels in total. Each channel of DMA has exactly the same function. The request selection module selects among 16 peripheral requests and handshake with peripherals, triggers the channel control module to control the entire data transmission process, and FIFO realizes data buffering and read-write bit width conversion. The AHB master interface respectively realizes the conversion between channel transmission and AHB interface protocol. Because the AHB bus can only serve one channel at a time, an arbiter must be set between the AHB master interface and the 2 channels. The DMA configuration register interface uses the AHB slave interface. DMA will generate interrupts according to interrupt events. DMA uses a fully synchronous design, with AHB BUS clock as the reference clock, no frequency division clock.

9.3 Main characteristics

(1) DMA has 2 channels

(2) One AHB master interface, one AHB slave interface

(3) There are three data transmission modes: peripheral to memory, memory to peripheral, and memory to memory

(4) Each channel has a special hardware DMA request for connection

(5) Support software priority and hardware priority when multiple requests occur at the same time

(6) Each channel has 5 event flags and independent interrupts

(7) The configurable source and target transmission width is byte, half word or word

(8) Support circular transmission mode

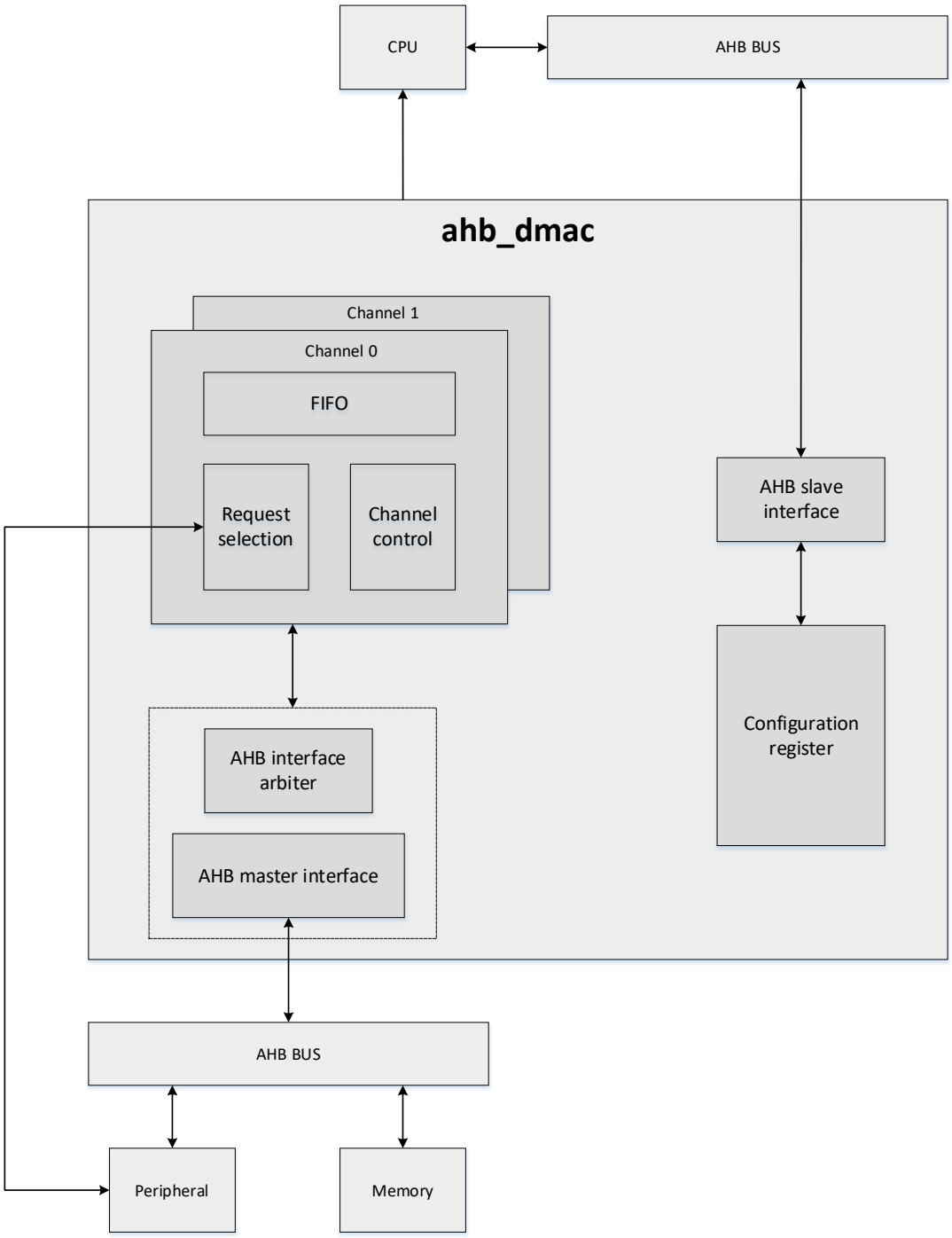
(9) Support source and target incremental modes

(10) The configurable burst increment size is single time, 4, 8 or 16 ticks

(11) Programmable data transmission number, up to 65535, is determined by PSIZECFG bits

9.4 Structure block diagram

Figure 14 Structure Block Diagram



9.5 Functional description

9.5.1 DMA request

If the peripheral or memory needs to transmit data using DMA, it is required to first transmit DMA request and after it is approved by DMA, data transmission

can be started.

DMA has 2 channels. Each channel is connected with different peripherals, and each channel has 5 event flags (DMA half transmission, DMA transmission completion, DMA transmission error, DMA FIFO error, and direct mode error). The logic of the 5 event flags may become a separate interrupt request, and they all support software trigger.

When multiple peripherals request the same channel, it is required to configure the corresponding register to enable or disable the request of each peripheral, so as to ensure that one channel can only enable one peripheral request.

Table 31 DMA Request Mapping Table

Peripheral		Channel 0	Channel 1
Peripheral0	0000	ADC	ADC
Peripheral1	0001	-	-
Peripheral2	0010	-	-
Peripheral3	0011	-	-
Peripheral4	0100	GTMR_CH3	GTMR_CH0
Peripheral5	0101	GTMR_UP	GTMR_CH1
Peripheral6	0110	GTMR_TRIG	GTMR_CH2
Peripheral7	0111	-	-
Peripheral8	1000	-	-
Peripheral9	1001	-	-
Peripheral10	1010	-	-
Peripheral11	1011	USART_TX	USART_RX
Peripheral12	1100	UART_RX	UART_TX
Peripheral13	1101	SPI_TX	SPI_RX
Peripheral14	1110	I2C_RX	I2C_TX
Peripheral15	1111	SPI_RX	SPI_TX

9.5.2 Arbiter

When multiple DMA channel requests (requests to the AHB master interface of DMA) occur, an arbiter is needed to manage the response sequence. Management is divided into two stages: the first stage is software stage, which is divided into two priorities of high and low; the second stage is hardware stage, and under the condition of the same software priority, the lower the channel number is, the higher the priority is.

9.5.3 FIFO

FIFO is used to temporarily store data before the source data is transmitted to the destination address. Each channel has an independent 4-word FIFO, and the FIFO threshold can be controlled by software to be 1/4, 1/2, 3/4 or full.

There are two DMA transmission modes. The first is direct mode, in which a

single transmission will be started to the memory immediately after each peripheral request. If DMA is configured to transmit data from the memory to the peripheral, DMA will store a data in FIFO, and once the peripheral triggers the DMA request, it will transmit the data. The direct mode requires the same data width configuration for the source and destination addresses, and does not support burst mode or memory-to-memory transmission mode. The second is FIFO mode, in which, FIFO threshold is configured first, and when the data storage reaches the threshold, FIFO content will be transmitted to the destination address; FIFO mode is applicable when the data width of source address and destination address is different, and it supports burst mode; FIFO can store the data first and output them as required.

9.5.4 Port

The DMA controller transmits data to the memory and peripherals through the AHB Master port. The AHB Master of DMA is connected to the AHB matrix bus and accesses the internal Flash, internal SRAM, AHB peripherals, and APB peripherals according to the bus relationship.

9.5.5 Interrupt

Each channel has 5 types of interrupt events: half transmission, transmission completion, transmission error, FIFO error and direct mode error. There are corresponding configuration registers to control the enabling of interrupts.

- Half-transmission interrupt: Triggered only when DMA is used for flow control, not an accurate interrupt. Triggered when the transmission is halfway, and at the latest not later than the moment when the data amount of half of the data plus the FIFO threshold is transmitted.
- Transmission completion interrupt: Triggered when the transmission is completed.
- Transmission error interrupt: Triggered when a bus read-write error occurs during DMA transmission; in dual-buffer mode, it is also triggered when writing to M0ADDR when CTARG is 0, or writing to M1ADDR when CTARG is 1.
- FIFO error interrupt: Triggered when FIFO underflow or overrun occurs; triggered when the FIFO threshold and memory Burst type belong to an unsupported transmission type and the channel is enabled.
- Direct mode error interrupt: In direct mode, for peripheral-to-memory transmission, if MINCM is set to 0 (i.e., the memory is a fixed address), and there is data in the current FIFO that has not been transmitted to the memory, and a DMA transmission request is generated at this time, it is triggered.

9.5.6 Direction for use

9.5.6.1 DMA initialization parameter configuration

Transmission Mode

DMA supports three transmission modes: peripheral-to-memory mode, memory-to-peripheral mode and memory-to-memory mode.

The transmission mode can be controlled through DIRCFG bit of DMA_SCFG register.

Increment mode

The increment mode of peripheral and memory is controlled through PINCM and MINCM bits of DMA_SCFG register. When both bits are set to 1, it is configured as the increment mode and the increment is the value of PSIZECFG and MSIZECFG bits of DMA_SCFG register. The PSIZECFG and MSIZECFG bits are used to set the data size of peripheral and memory to byte, half word or word.

Single transmission and burst mode

Burst transmission refers to the high-speed transmission that increases the data volume transmitted each time at the transmission stage so as to improve the transmission speed. In the process of burst transmission, AHB bus will be occupied.

Single and burst transmissions can be controlled through the PBCFG and MBCFG bits of DMA_SCFG register, and it can be configured as single transmission, incremental burst transmission of 4 ticks, incremental burst transmission of 8 ticks and incremental burst transmission of 16 ticks. This increment is determined by the value of PSIZECFG and MSIZECFG bits. The burst mode can be enabled only when the increment mode is supported.

The burst mode shall be used in combination with FIFO, and the selected FIFO threshold shall be suitable for the burst size of memory, as shown in the table below.

Table 32 FIFO Threshold Configuration

MSIZECFG	FIFO threshold	MBCFG=01	MBCFG=10	MBCFG=11
Byte	1/4	One-time burst of 4 ticks	Disable	Disable
	1/2	Two-time burst of 4 ticks	One-time burst of 8 ticks	
	3/4	Three-time burst of 4 ticks	Disable	

MSIZECFG	FIFO threshold	MBCFG=01	MBCFG=10	MBCFG=11
	Full	Four-time burst of 4 ticks	Two-time burst of 8 ticks	One-time burst of 16 ticks
Half word	1/4	Disable	Disable	Disable
	1/2	One-time burst of 4 ticks		
	3/4	Disable		
	Full	Two-time burst of 4 ticks	One-time burst of 8 ticks	
Word	1/4	Disable	Disable	
	1/2			
	3/4			
	Full	One-time burst of 4 ticks		

Circular mode

The circular mode is used to process the circular buffer area and continuous channel. The circular mode will automatically configure the number of data items as the initial value after the transmission ends, and continue the data transmission.

The circular mode can be controlled through CIRCMEN bit of DMA_SCFG register.

Double buffer mode

Setting the DBM of the DMA_SCFG register to 1 can enable the dual-buffer mode and automatically activate the loop mode. In the dual-buffer mode, the DMA_M1ADDR register is activated. After the storage area corresponding to the address pointer of the DMA_M0ADDR register is transmitted, It will switch to the storage area corresponding to the address pointer of the DMA_M1ADDR register for continued transmission, and will be called in a loop. When the DMA accesses the DMA_M1ADDR, the CTARG position of the DMA_SCFG register is 1, and at the same time, data can be written to or read from the DMA_M0ADDR register.

This mode does not support memory-to-memory transfer.

Channel configuration process

(1) Determine whether the channel is enabled by reading CHEN. If CHEN=1, first configure CHEN to 0, then read and confirm, and finally reset each interrupt state to zero

(2) Configure the channel register according to the transmission type requirements

(3) Configure CHEN to 1 to enable the channel

9.6 Register address mapping

Table 33 DMA Register Address Mapping Table

Register name	Description	Offset address
DMA_ISR	DMA interrupt status register	0x00
DMA_IFCLR	DMA interrupt flag clear register	0x08
DMA_SCFGx	DMA Channel x configuration register	0x10+0x18 x
DMA_NDATAx	DMA channel x data item number register	0x14+0x18 x
DMA_PADDRx	DMA Channel x peripheral address register	0x18+0x18 x
DMA_M0ADDRx	DMA channel x memory 0 address register	0x1C+0x18 x
DMA_M1ADDRx	DMA channel x memory 1 address register	0x20+0x18 x
DMA_FIFOCRx	DMA channel x FIFO control register	0x24+0x18 x

9.7 Register functional description

9.7.1 DMA interrupt status register (DMA_ISR)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:12	Reserved		
11	TXCIFLG1	R	Channel 1 Transfer Complete Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No transmission completion event 1: Transmission completion event is generated.
10	HTXIFLG1	R	Channel 1 Half Transfer Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No half-transmission event 1: Half-transmission event occurs
9	TXEIFLG1	R	Channel 1 Transfer Error Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No transmission error 1: Transmission error occurs
8	DMEIFLG1	R	Channel 1 Direct Mode Error Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register.

Field	Name	R/W	Description
			0: No direct mode error 1: Direct mode error occurs
7	Reserved		
6	FEIFLG1	R	Channel 1 FIFO Error Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No FIFO error event 1: FIFO error event occurs
5	TXCIFLG0	R	Channel 0 Transfer Complete Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No transmission completion event 1: Transmission completion event is generated.
4	HTXIFLG0	R	Channel 0 Half Transfer Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No half-transmission event 1: Half-transmission event occurs
3	TXEIFLG0	R	Channel 0 Transfer Error Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No transmission error 1: Transmission error occurs
2	DMEIFLG0	R	Channel 0 Direct Mode Error Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No direct mode error 1: Direct mode error occurs
1	Reserved		
0	FEIFLG0	R	Channel 0 FIFO Error Interrupt Flag These bits are set to 1 by hardware; write 1 and set to 0 by software on the corresponding bit of DMA_IFCLR register. 0: No FIFO error event 1: FIFO error event occurs

9.7.2 DMA interrupt flag clear register (DMA_IFCLR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:12	Reserved		
11	CTXCIFLG1	W	Channel 1 Clear Transfer Complete Interrupt Flag 0: Invalid 1: The TXCIFLG1 flag in DMA_ISR register is cleared to 0
10	CHTXIFLG1	W	Channel 1 Clear Half Transfer Interrupt Flag 0: Invalid

Field	Name	R/W	Description
			1: The HTXIFLG1 flag in DMA_ISR register is cleared to 0
9	CTXEIFLG1	W	Channel 1 Clear Transfer Error Interrupt Flag 0: Invalid 1: The TXEIFLG1 flag in DMA_ISR register is cleared to 0
8	CDMEIFLG1	W	Channel 1 Clear Direct Mode Error Interrupt Flag 0: Invalid 1: The DMEIFLG1 flag in DMA_ISR register is cleared to 0
7	Reserved		
6	CFEIFLG1	W	Channel 1 Clear FIFO Error Interrupt Flag 0: Invalid 1: The FEIFLG1 flag in DMA_ISR register is set to 0
5	CTXCIFLG0	W	Channel 0 Clear Transfer Complete Interrupt Flag 0: Invalid 1: The TXCIFLG0 flag in DMA_ISR register is cleared to 0
4	CHTXIFLG0	W	Channel 0 Clear Half Transfer Interrupt Flag 0: Invalid 1: The HTXIFLG0 flag in DMA_ISR register is cleared to 0
3	CTXEIFLG0	W	Channel 0 Clear Transfer Error Interrupt Flag 0: Invalid 1: The TXEIFLG0 flag in DMA_ISR register is cleared to 0
2	CDMEIFLG0	W	Channel 0 Clear Direct Mode Error Interrupt Flag 0: Invalid 1: The DMEIFLG0 flag in DMA_ISR register is cleared to 0
1	Reserved		
0	CFEIFLG0	W	Channel 0 Clear FIFO Error Interrupt Flag 0: Invalid 1: The FEIFLG0 flag in DMA_ISR register is set to 0

9.7.3 DMA channel x configuration register (DMA_SCFGx) (x=0,1)

Offset address: 0x10+0x18 x

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:29	Reserved		
28:25	PSEL	R/W	Peripheral Selection 0000: choose a peripheral 0 0001: select peripherals 1 0010: select peripherals 2 0011: select peripherals 3 0100: select peripherals 4 0101: select peripherals 5 0110: select peripherals 6 0111: select peripherals 7 1000: select peripherals 8 1001: select peripherals 9

Field	Name	R/W	Description
			1010: Select peripherals 10 1011: select peripherals 11 1100: select peripherals 12 1101: select peripherals 13 1110: select peripherals 14 1111: select peripherals 15 These bits can be written only when CHEN bit is 0.
24:23	MBCFG	R/W	Memory Burst Transfer Configure 00: Single transmission 01: INCR4 (4-tick increment burst transmission) 10: INCR8 (8-tick increment burst transmission) 11: INCR16 (16-tick increment burst transmission) This bit can be written only when CHEN bit is 0. In direct mode, these bits will be forced to 0.
22:21	PBCFG	R/W	Peripheral Burst Transfer Configure 00: Single transmission 01: INCR4 (4-tick increment burst transmission) 10: INCR8 (8-tick increment burst transmission) 11: INCR16 (16-tick increment burst transmission) This bit can be written only when CHEN bit is 0. In direct mode, these bits will be forced to 0.
20	Reserved		
19	CTARG	R/W	Current Target (only in double buffer mode) This bit can be set to 1 or cleared to 0 by hardware, or be written by software. 0: The current target memory is Memory 0 1: The current target memory is Memory 1 These bits can be written only when CHEN bit is 0.
18	DBM	R/W	Double Buffer Mode 0: Do not switch the buffer when the transmission ends 1: Switch the target memory when DMA transmission ends This bit can be written only when CHEN bit is 0.
17	Reserved		
16	PRILCFG	R/W	Priority Level Configure 0: Low 1: High This bit can be written only when CHEN bit is 0.
15	PERIOSIZE	R/W	Peripheral increment offset size 0: The offset used to calculate the peripheral address is related to PSIZECFG 1: The offset used to calculate the peripheral address is fixed to be 4 If PINCM bit is 0, this bit is meaningless, and it can be written only when CHEN bit is 0.

Field	Name	R/W	Description
			If the direct mode is selected or the PBCFG bit is not configured to 00, and the channel is enabled, this bit will be forced to low level by hardware.
14:13	MSIZECFG	R/W	Memory Data Size Configure 00: Byte (8 bits) 01: Half word (16 bits) 10: Word (32 bits) 11: Reserved These bits can be written only when CHEN bit is 0. In direct mode, when CHEN bit is 1, MSIZECFG bit will be forced to be of the same value as that of PSIZECFG bit.
12:11	PSIZECFG	R/W	Peripheral Data Size Configure 00: Byte (8 bits) 01: Half word (16 bits) 10: Word (32 bits) 11: Reserved These bits can be written only when CHEN bit is 0.
10	MINCM	R/W	Memory Increment Mode 0: The memory address pointer is fixed 1: After each data transmission, the memory address pointer will increase This bit can be written only when CHEN bit is 0.
9	PINCM	R/W	Peripheral Increment Mode 0: The peripheral address pointer is fixed 1: After each data transmission, the peripheral address pointer will increase This bit can be written only when CHEN bit is 0.
8	CIRCMEN	R/W	Circular Mode Enable This bit can be set to 1 or 0 by software, or be set to 0 by hardware. 0: Disable 1: Enable If DMA transmission is ended, switch the target memory area, enable the channel, and this bit will be automatically forced to 1 by the hardware.
7:6	DIRCFG	R/W	Data Transfer Direction Configure 00: From peripheral to memory 01: From memory to peripheral 10: From memory to memory 11: Reserved These bits can be written only when CHEN bit is 0.
5	Reserved		
4	TXCIEN	R/W	Transfer Complete Interrupt Enable 0: Disable 1: Enable
3	HTXIEN	R/W	Half Transfer Interrupt Enable 0: Disable

Field	Name	R/W	Description
			1: Enable
2	TXEIEN	R/W	Transfer Error Interrupt Enable 0: Disable 1: Enable
1	DMEIEN	R/W	Direct Mode Error Interrupt Enable 0: Disable 1: Enable
0	CHEN	R/W	Channel Enable 0: Disable 1: Enable This bit shall be set to 0 by hardware in any of the following situations: When DMA transmission ends. When transmission error occurs to AHB main bus. When the FIFO threshold on the memory AHB port is incompatible with the burst size.

9.7.4 DMA channel x data item number register (DMA_NDATA) (x=0,1)

Offset address: 0x14+0x18 x

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	NDATA	R/W	Number Of Data Items To Transfer The number of data items to be transmitted is 0-65535. This register can be operated only when the channel is disabled. After the channel is enabled, this register is read-only to indicate the number of remaining data items to be transmitted. After each DMA transmission, this register will decrease. After the transmission is completed, this register becomes 0. In loop mode, the initial value is automatically reloaded after the transmission is completed.

9.7.5 DMA channel x peripheral address register (DMA_PADDR) (x=0,1)

Offset address: 0x18+0x18 x

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	PADDR	R/W	Peripheral Address Base address of peripheral data register of read/write data. This bit can be written only when CHEN bit is 0.

9.7.6 DMA channel x memory 0 address register (DMA_M0ADDR) (x=0,1)

Offset address: 0x1C+0x18 x

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	M0ADDR	R/W	Memory 0 Address Base address of memory 0 of read/write data. These bits are write-protected, and can be written only in any of the following circumstances: Disable channel Enable the channel and set CTARG bit of DMA_SCFG register to 1

9.7.7 DMA channel x memory 1 address register (DMA_M1ADDR) (x=0,1)

Offset address: 0x20+0x18 x

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	M1ADDR	R/W	Memory 1 Address Base address of memory 1 of read/write data. This register is only applicable to double-buffer mode. These bits are write-protected, and can be written only in any of the following circumstances: Disable channel Enable the channel and set CTARG bit of DMA_SCFG register to 0

9.7.8 DMA channel x FIFO control register (DMA_FIFOCR) (x=0,1)

Offset address: 0x24+0x18 x

Reset value: 0x0000 0020

Field	Name	R/W	Description
31:8	Reserved		
7	FEIEN	R/W	FIFO Error Interrupt Enable 0: Disable 1: Enable
6	Reserved		
5:3	FSTS	R	FIFO Status 000: 0<fifo_level<1/4 001: 1/4≤fifo_level<1/2 010: 1/2≤fifo_level<3/4 011: 3/4≤fifo_level<full 100: FIFO is empty 101: FIFO is full Others: Meaningles These bits are invalid in direct mode.
2	DMDEN	R/W	Direct Mode Disable 0: Enable direct mode 1: Disable direct mode This bit can be written only when the CHEN bit is 0; when the memory-to-memory mode is selected and CHEN bit is 1, this bit will be set to 1 by hardware.

Field	Name	R/W	Description
1:0	FTHSEL	R/W	FIFO Threshold Select 00: 1/4 of FIFO capacity 01: 1/2 of FIFO capacity 10: 3/4 of FIFO capacity 11: Full FIFO capacity In direct mode, these bits are invalid, and they can be written only when CHEN bit is 0.

10 General-Purpose Input/Output Pin (GPIO)

For M3122/M3114/M3115, when using the pin multiplexing function, please note that some pins are already connected to the internal gate driver. For details, refer to the system block diagram or the datasheet.

10.1 Full Name and Abbreviation Description of Terms

Table 34 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
P-channel Metal Oxide Semiconductor	P-MOS
N-channel Metal Oxide Semiconductor	N-MOS

10.2 Introduction

Embedded with up to 29 GPIO pins. All IOs can work under 5V, supporting 4 modes: input, output, multiplexing, and analog. In input mode, they can be configured as high-impedance input. All of them can switch between input, output, or multiplexing functions. Most GPIO pins are shared with multiplexed peripherals. In addition, some pins have redefinition functions, such as analog input, external interrupt, and input/output of chip peripherals, but only one function can be mapped to a pin at a time.

10.3 Main characteristics

(1) Input mode
- Floating input
- Pull-up input
- Pull-down input

(2) Output mode
- Push-pull output
- Open-drain output

(3) Multiplexing mode
- Push-pull multiplexing function
- Open-drain multiplexing function

(4) Analog mode

(5) GPIO can be used as external interrupt/wakeup line

(6) Output rate: up to 20MHz

(7) Configurable output drive capability in two levels: 3mA/6mA

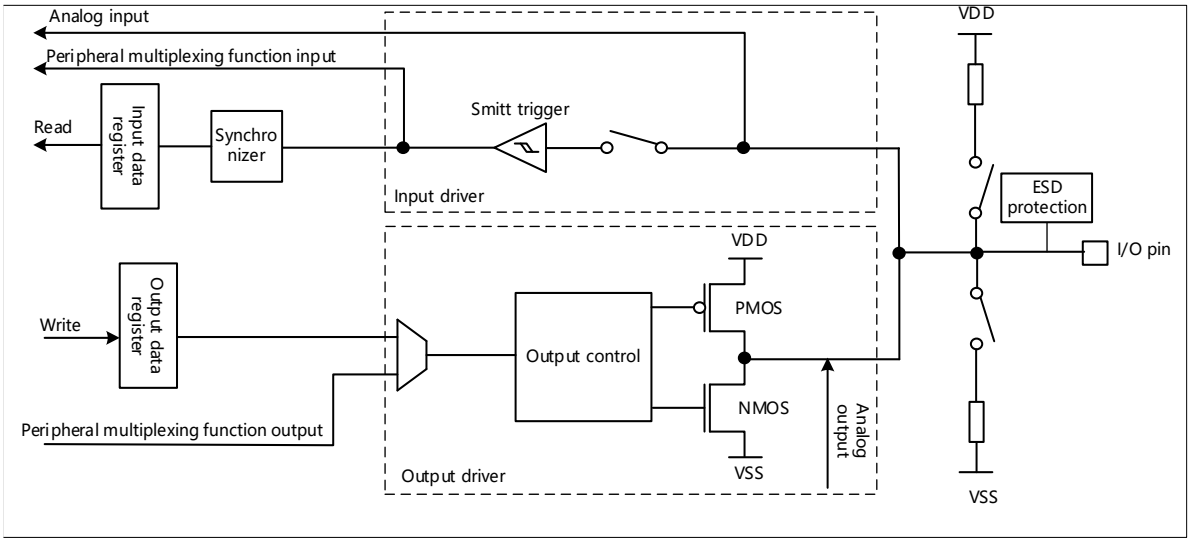
(8) Configurable latch protection: GPIO-related configuration registers have latch protection/unlock functions

(9) Flexible multiplexing functions: each IO has 8 groups of multiplexing configurations

(10) The motor-related pins PA10 to PA15 can be further remapped

10.4 Structure block diagram

Figure 15 GPIO Structure Block Diagram



10.5 Functional description

Each pin of GPIO can be configured as pull-up, pull-down, floating and analog input, or push-pull/open-drain output and input mode and multiplexing function by software. All GPIO interfaces have external interrupt capability.

10.5.1 IO status during reset and just after reset

During and just after GPIO reset, if the multiplexing function is not enabled, the I/O port will be configured as floating input mode.

10.5.2 Input mode

In the input mode, it can be set as pull-up, pull-down, floating and analog input.

When GPIO is configured as input mode, all GPIO pins have an internal weak pull-up and pull-down resistor, which can be activated or broken.

Pull-up, pull-down, and floating modes

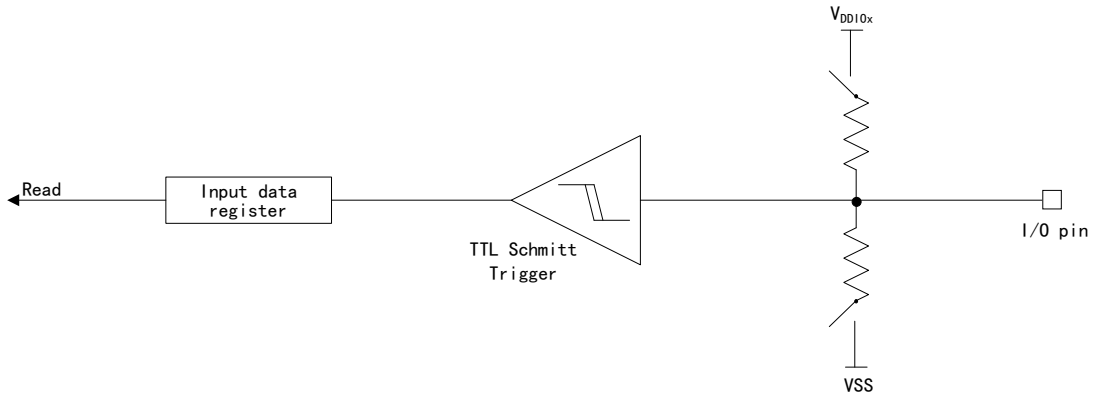
In (pull-up, pull-down, floating) input mode

- Schmitt trigger is enabled
- Disable output buffer
- By configuring the pull-up/pull-down register GPIOx_PUPDR, select whether to use pull-up/pull-down resistor
- The input data register GPIOx_INDR captures the data on I/O pin in each AHB clock cycle.
- Read I/O state by the input data register GPIOx_INDR

The initial level state of the floating input mode is uncertain and is easy to be disturbed by the outside; when connecting the equipment, it is determined by the external input level (except for the very high impedance).

The initial level state of pull-up/pull-down input mode is high if pull-up, and low if pull-down; when connecting the equipment, it is determined by the external input level and load impedance.

Figure 16 Input Mode I/O Structure



10.5.3 Output mode

In the output mode, it can be set as push-pull output and open-drain output.

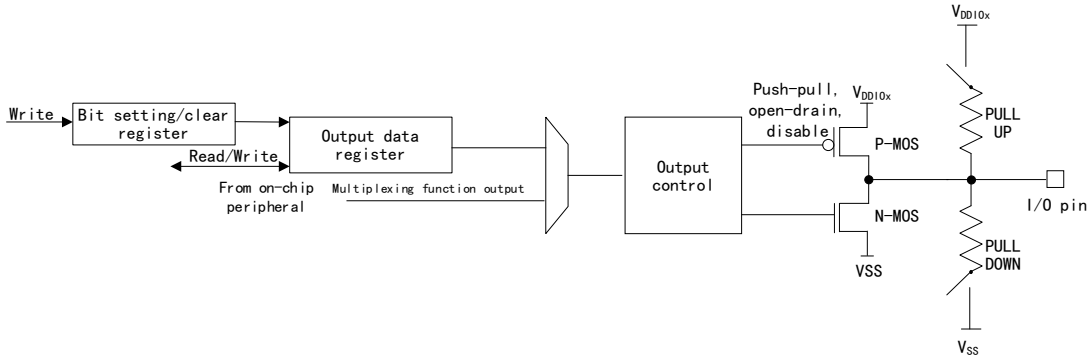
When GPIO is configured as the output pin, the output speed of the port can be configured and the output drive mode (push-pull/open-drain) can be selected.

In output mode

- Schmitt trigger is opened
- Activate the output buffer
- By configuring the pull-up/pull-down register GPIOx_PUPDR, select whether to use pull-up/pull-down resistor
- Push-pull mode:
  - Double MOS transistor works by turns and the output data register can control the high and low level of I/O output;
  - Read the finally written value through the output data register GPIOx_OUTDR
- Open-drain mode:
  - Only N-MOS works, and the output data register can control I/O output high-resistance state or low level

- The input data register GPIOx_INDR captures the data on I/O pin in each AHB clock cycle
- Read the actual I/O state through the input data register GPIOx_INDR

Figure 17 Output Mode I/O Structure



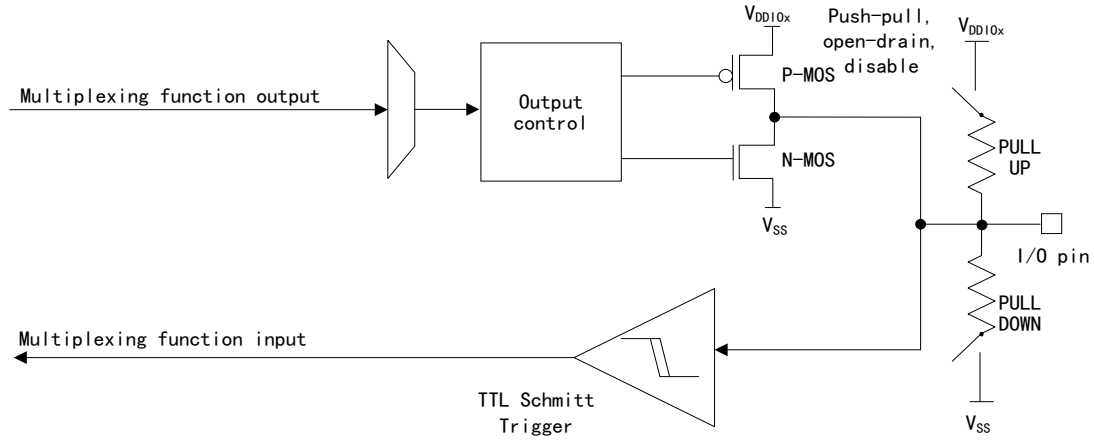
10.5.4 Multiplexing mode

In multiplexing mode, it can be set as push-pull multiplexing and open-drain multiplexing

In push-pull/open drain multiplexing mode

- Enable the output buffer
- Output buffer is driven by peripheral
- Activate Schmitt trigger input
- By configuring the pull-up/pull-down register GPIOx_PUPDR, select whether to use pull-up/pull-down resistor
- The data on the I/O pin is sampled in each AHB clock cycle and stored in the port input status register
- Read the actual I/O state through the input data register GPIOx_INDR

Figure 18 Multiplexing Mode I/O Structure

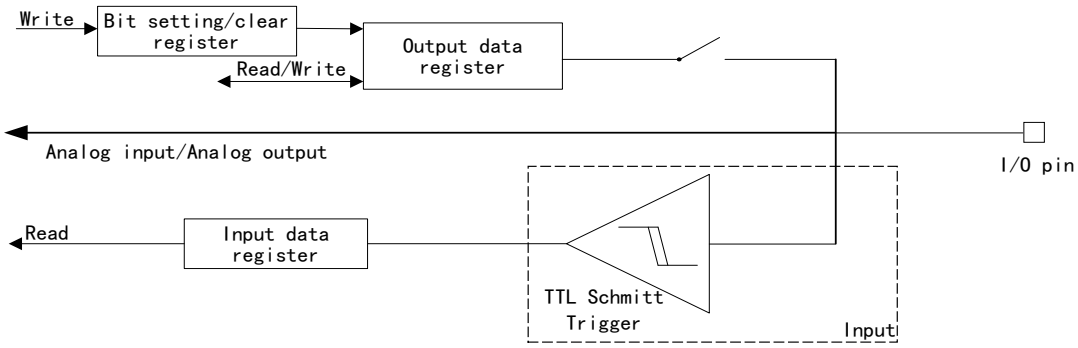


10.5.5 Analog mode

In analog function mode

- Disable output buffer
- The input of Schmitt trigger is disabled, and the output value of Schmitt trigger is forced to be 0
- Weak pull-up and pull-down resistors are disabled
- Read the value of the input status register to be 0

Figure 19 Analog Function I/O Structure



10.5.6 External interrupt/wake-up line

All GPIO ports have external interrupt function. If you want to use external interrupt line, the port must be configured as input mode.

10.5.7 I/O data bit processing

GPIO port set/reset register (GPIOx_BSRR) allows set/reset operation for each bit of the output data register (GPIOx_OUTDR). The valid data width of the set/reset register is double the valid data width of GPIOx_OUTDR.

Writing 0 to any bit in GPIOx_BSRR will not affect the value of the GPIOx_OUTDR register. PAxBS and PAxBC bits of GPIOx_BSRR are set to 1 for the same time, operation of PAxBS bit has the priority. GPIOx_ BSRR register can change the corresponding bit of the GPIOx_OUTDR register, and GPIOx_OUTDR bit can be accessed directly from GPIOx_BSRR register.

When the access mechanism is set or reset by GPIOx_OUTDR through GOIOx_BSRR register, it is not necessary to disable the interrupt by software to access GPIOx_OUTDR.

10.5.8 Multiplexing function and remapping

Multiplexer

The multiplexer is used to connect the I/O port line of the device to the embedded peripheral module, and it can only be one-to-one at the same time.

Each I/O pin is equipped with a multiplexer. a maximum of 8 (AF0-AF7) are used, which are configured by GPIOx_AFSELR0 and GPIOx_ AFSELR1 registers. When I/O pin is reset, all pin ports are connected to AF0.

Remapping

Each peripheral has multiple multiplexing functions, but only one multiplexing function input can be selected for a pin, so the multiplexing function of the peripheral can be mapped to other I/O pins, that is, the multiplexing function signal can be reassigned to a pin address.

The multiplexing function and remapping address table of pins are shown in the datasheet.

I/O multiplexing configuration

When I/O port is connected to the peripheral multiplexing function, the following debugging needs to be done:

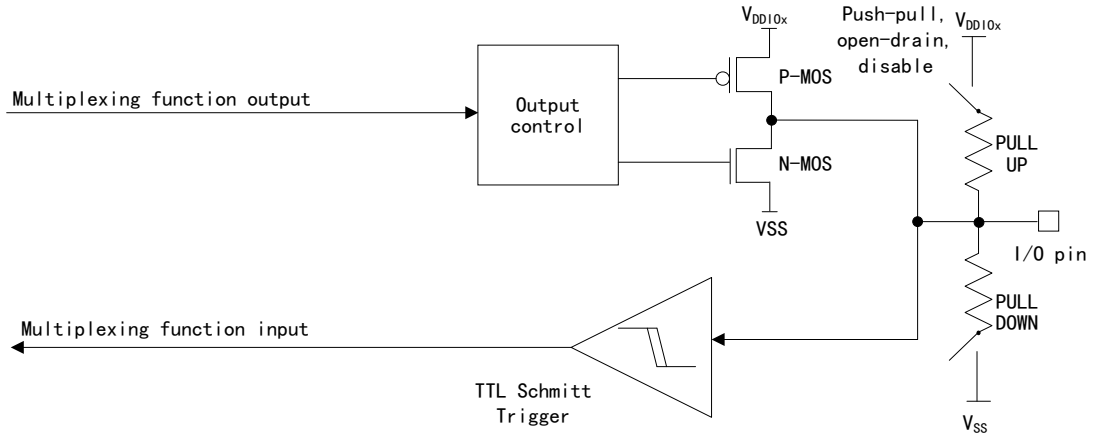
- After reset, the pin is configured with multiplexing function
- I/O port is configured as input, output or analog input
- The I/O port is connected to the defined AFx
- Configure pin pull-up/pull-down and output speed
- Configure I/O as multiplexing function in GPIOx_MDR

When the I/O port is configured with multiplexing function, its input and output mode is as follows:

- Enable the output buffer
- Output buffer is driven by peripheral
- Activate Schmitt trigger input
- By configuring the pull-up/pull-down register GPIOx_PUPDR, select whether to use pull-up/pull-down resistor
- The data on the I/O pin is sampled in each AHB clock cycle and stored in the port input status register
- Read the actual I/O state through the input data register GPIOx_INDR

The multiplexing mode I/O structure is shown in the figure below:

Figure 20 Multiplexing Mode I/O Structure



Reset pin (PA6) multiplexing function switching

Before PA6 switches from other functions to NRST, digital filtering needs to be enabled first. Digital filtering needs to be enabled before entering the standby mode.

10.5.9 GPIO locking function

The locking mechanism of GPIO can protect the configuration of I/O port.

Write sequence (specific) to GPIOx_LOCK register so as to freeze the control register of Port A and Port B. If you want to write GPIOx_LOCK register, a specific write/read sequence should be transmitted.

I/O configuration can be locked by configuring the lock register (GPIOx_LOCK). When a port bit executes the locking program, the configuration of port bit cannot be modified before the next reset.

10.5.10 Direction for use

10.5.10.1 Input configuration

The configuration process is as follows:

(1) Unlock by configuring 0xA5A5_5A5A through GPIO_LOCK

(2) Configure GPIO_MDR to 01 input mode

(3) ull-down operations can be performed by configuring GPIO_PUPDR

(4) Obtain the I/O status by reading GPIO_INDR

10.5.10.2 Output configuration

The configuration process is as follows:

(1) Unlock by configuring 0xA5A5_5A5A through GPIO_LOCK

(2) Configure GPIO_MDR to 10 output mode

(3) Pull-down operations can be performed by configuring GPIO_PUPDR

(4) The push-pull/open-leak output type can be selected by configuring GPIO_OTR

(5) The driver capability can be selected by configuring GPIO_DSR

(6) Obtain the I/O status by reading GPIO_INDR

When programming I/O for output:

- The output path is open
- Read access to the output data register can obtain the last written value

10.5.10.3 Multiplex configuration

The configuration process is as follows:

(1) Unlock by configuring 0xA5A5_5A5A through GPIO_LOCK

(2) Configure GPIO_MDR to 11 multiplex mode

(3) Configure GPIO_AFSELRx for multiplexing selection

(4) The multiplexed output (data and enables) comes from peripheral control

(5) Pull-down operations can be performed by configuring GPIO_PUPDR

(6) The push-pull/open-leak output type can be selected by configuring GPIO_OTYRCFG

(7) The driver capability can be selected by configuring GPIO_DSR

(8) Obtain the I/O status by reading GPIO_INDR

10.5.10.4 Analog configuration

The configuration process is as follows:

(1) Unlock by configuring 0xA5A5_5A5A through GPIO_LOCK

(2) Configure GPIO_MDR to 00 analog mode

When programming I/O in analog mode:

- Output path disconnected
- Input disconnected
- Up and down pulling prohibited
- Read access to the input data register is 0

10.6 Register address mapping

GPIOA base address:0x4800 0000

GPIOB base address:0x4800 0400

Table 35 GPIO Register Address Mapping

Register name	Description	Offset address
GPIOA_MDR	GPIOA mode register	0x00
GPIOA_INENR	GPIOA input enable register	0x04
GPIOA_PUPDR	GPIOA pull-up/down register	0x0C
GPIOA_OTR	GPIOA output type register	0x10
GPIOA_DSR	GPIOA driving strength configuration register	0x14
GPIOA_INDR	GPIOA input data register	0x18
GPIOA_OUTDR	GPIOA output data register	0x1C
GPIOA_BSRR	GPIOA data bit operation register	0x20

GPIOA_BRR	GPIOA data bit clear register	0x24
GPIOA_LOCK	GPIOA lock register	0x28
GPIOA_AFSELR0	GPIOA multiplex select register 0	0x2C
GPIOA_AFSELR1	GPIOA multiplex select register 1	0x30
GPIOA_AF3RMP	GPIOA ATMR multiplexing function remaps register	0x54
GPIOB_MDR	GPIOB mode register	0x00
GPIOB_INENR	GPIOB input enable register	0x04
GPIOB_PUPDR	GPIOB pull-up/down register	0x0C
GPIOB_OTR	GPIOB output type register	0x10
GPIOB_DSR	GPIOB driving strength configuration register	0x14
GPIOB_INDR	GPIOB input data register	0x18
GPIOB_OUTDR	GPIOB output data register	0x1C
GPIOB_BSRR	GPIOB data bit operation register	0x20
GPIOB_BRR	GPIOB data bit clear register	0x24
GPIOB_LOCK	GPIOB lock register	0x28
GPIOB_ AFSELR0	GPIOB multiplex select register 0	0x2C
GPIOB_ AFSELR1	GPIOB multiplex select register 1	0x30

10.7 Register functional description

10.7.1 GPIOA Mode Register (GPIOA_MDR)

Offset address: 0x00

Reset value: 0x5555 7555

Field	Name	R/W	Description
31:30	PA15MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
29:28	PA14MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
27:26	PA13MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode

Field	Name	R/W	Description
25:24	PA12MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
23:22	PA11MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
21:20	PA10MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
19:18	PA9MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
17:16	PA8MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
15:14	PA7MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
13:12	PA6MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
11:10	PA5MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
9:8	PA4MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode

Field	Name	R/W	Description
7:6	PA3MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
5:4	PA2MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
3:2	PA1MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
1:0	PA0MD	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode

10.7.2 GPIOA Input Enable Register (GPIOA_INENR)

Offset address: 0x04

Reset value: 0x0000 0040

Field	Name	R/W	Description
31:16	Reserved		
x	PAxINEN	R/W	Port enable (x=0~15) 0: Disable 1: Enable Note: Effective in non-analog mode.

10.7.3 GPIOA Pull-Up/Pull-Down Register (GPIOA_PUPDR)

Offset address: 0x0C

Reset value: 0x0000 1000

Field	Name	R/W	Description
2x+1	PAxPUS	R/W	Up and down pull select (x=0~15) It takes effect after pull up and down is enabled. 0: Pull-up 1: Pull-down
2x	PAxPUEN	R/W	Up and down pull enable (x=0~15) 0: Disable 1: Enable

10.7.4 GPIOA Output Type Register (GPIOA_OTR)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
x	PAxOT	R/W	Output type configure (x=0~15) 0: Push-pull 1: Open-drain

10.7.5 GPIOA Drive Strength Configuration Register (GPIOA_DSR)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
x	PAxDS	R/W	Output driving capability Configure (x=0~15) 0: Low drive 1: High drive

10.7.6 GPIOA input data register (GPIOA_INDR)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
x	PAxIND	R	Input data (x=0~15) 0: Low level 1: High level

10.7.7 Port output data register (GPIOx_OUTDR)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
x	PAxOUTD	R/W	Output data (x=0~15) 0: Output low level 1: Output high level Bitwise output 1: Controlled by PAxBS Bitwise output 0: Controlled by PAxBR and PAxBRR

10.7.8 GPIOA Data Bit Operation Register (GPIOA_BSRR)

Offset address: 0x20

Reset value: 0x0000 0000

Field	Name	R/W	Description
x+16	PAxBR	W	PAxOUTD Reset (x=0~15) Write-only, reading these bits returns 0. 0: No operation on the corresponding PAxOUTD bit 1: Reset the corresponding PAxOUTD bit Note: If PAxBR and PAx_BS are set at the same time, PAx_BS has higher priority.
x	PAxBS	W	PAxOUTD set (x=0~15) Write-only, reading these bits returns 0. 0: No operation on the corresponding PAxOUTD bit 1: Set the corresponding PAxOUTD bit to 1

10.7.9 GPIOA Data Bit Clear Register (GPIOA_BRR)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
x	PAxBRR	W	PAxOUTD Reset (x=0~15) Write-only, reading these bits returns 0. 0: No operation on the corresponding PAxOUTD bit 1: Reset the corresponding PAxOUTD bit

10.7.10 GPIOA Lock Register (GPIOA_LOCK)

Offset address: 0x28

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	KEY	R/W	GPIO Lock Register Writing 0xA5A5 5A5A to this register allows writing to GPIO registers. After configuring the GPIO registers, writing other values to this register prohibits writing to other GPIO registers. Reset default lock Reading 0: indicates the locked state Reading 1: indicates the unlocked state

10.7.11 GPIOA Multiplex Select Register 0 (GPIOA_AFSELR0)

Offset address: 0x2C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31	Reserved		
30:28	PA7AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5

Field	Name	R/W	Description
			110：AF6 111：AF7
27	Reserved		
26:24	PA6AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
23	Reserved		
22:20	PA5AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
19	Reserved		
18:16	PA4AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
15	Reserved		
14:12	PA3AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
11	Reserved		
10:8	PA2AFSEL	R/W	multiplex select

Field	Name	R/W	Description
			000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
7	Reserved		
6:4	PA1AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
3	Reserved		
2:0	PA0AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7

10.7.12 GPIOA Multiplex Select Register 1 (GPIOA_AFSELR1)

Offset address: 0x30

Reset value: 0x0000 0000

Field	Name	R/W	Description
31	Reserved		
30:28	PA15AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
27	Reserved		

Field	Name	R/W	Description
26:24	PA14AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
23	Reserved		
22:20	PA13AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
19	Reserved		
18:16	PA12AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
15	Reserved		
14:12	PA11AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
11	Reserved		
10:8	PA10AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3

Field	Name	R/W	Description
			100：AF4 101：AF5 110：AF6 111：AF7
7	Reserved		
6:4	PA9AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
3	Reserved		
2:0	PA8AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7

10.7.13 GPIOA ATMR multiplexing function remaps register （GPIOA_AF3RMP）

Offset address: 0x54

Reset value:0x0314 2500

Field	Name	R/W	Description
31	Reserved		
30:28	PA15AF3RMP	R/W	Remapping select 000：ATMR_CH0(Default) 001：ATMR_CH1 010：ATMR_CH2 011：ATMR_CH0N 100：ATMR_CH1N 101：ATMR_CH2N 110：ATMR_CH0 111：ATMR_CH0
27	Reserved		
26:24	PA14AF3RMP	R/W	Remapping select 000：ATMR_CH0 001：ATMR_CH1

Field	Name	R/W	Description
			010：ATMR_CH2 011：ATMR_CH0N(Default) 100：ATMR_CH1N 101：ATMR_CH2N 110：ATMR_CH0 111：ATMR_CH0
23	Reserved		
22:20	PA13AF3RMP	R/W	Remapping select 000：ATMR_CH0 001：ATMR_CH1(Default) 010：ATMR_CH2 011：ATMR_CH0N 100：ATMR_CH1N 101：ATMR_CH2N 110：ATMR_CH0 111：ATMR_CH0
19	Reserved		
18:16	PA12AF3RMP	R/W	Remapping select 000：ATMR_CH0 001：ATMR_CH1 010：ATMR_CH2 011：ATMR_CH0N 100：ATMR_CH1N(Default) 101：ATMR_CH2N 110：ATMR_CH0 111：ATMR_CH0
15	Reserved		
14:12	PA11AF3RMP	R/W	Remapping select 000：ATMR_CH0 001：ATMR_CH1 010：ATMR_CH2(Default) 011：ATMR_CH0N 100：ATMR_CH1N 101：ATMR_CH2N 110：ATMR_CH0 111：ATMR_CH0
11	Reserved		
10:8	PA10AF3RMP	R/W	Remapping select 000：ATMR_CH0 001：ATMR_CH1 010：ATMR_CH2 011：ATMR_CH0N 100：ATMR_CH1N 101：ATMR_CH2N(Default) 110：ATMR_CH0

Field	Name	R/W	Description
			111：ATMR_CH0
7:0	Reserved		

10.7.14 GPIOB Mode Register (GPIOB_MDR)

Offset address: 0x00

Reset value: 0x0155 57D5

Field	Name	R/W	Description
31:26	Reserved		
2x+1:2x	PBxMD	R/W	Work mode configure (x=0~12) 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode

10.7.15 GPIOB Input Enable Register (GPIOB_INENR)

Offset address: 0x04

Reset value: 0x0000 0018

Field	Name	R/W	Description
31:13	Reserved		
x	PAxINEN	R/W	Port enable (x=0~12) 0: Disable 1: Enable Note: Effective in non-analog mode.

10.7.16 GPIOB Pull-Up/Pull-Down Register (GPIOB_PUPDR)

Offset address: 0x0C

Reset value: 0x0000 01C0

Field	Name	R/W	Description
31:26	Reserved		
2x+1	PBxPUS	R/W	Up and down pull select (x=0~12) It takes effect after pull up and down is enabled. 0: Pull-up 1: Pull-down
2x	PBxPUEN	R/W	Up and down pull enable (x=0~12) 0: Disable 1: Enable

10.7.17 GPIOB Output Type Register (GPIOB_OTR)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		

Field	Name	R/W	Description
x	PBxOT	R/W	Output type configure (x=0~12) 0: Push-pull 1: Open-drain

10.7.18 GPIOB Drive Strength Configuration Register (GPIOB_DSR)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
x	PBxDS	R/W	Output driving capability configure (x=0~12) 0: Low drive 1: High drive

10.7.19 GPIOB Input data Register (GPIOB_INDR)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
x	PBxIND	R	Input data (x=0~12) 0: Low level 1: High level

10.7.20 GPIOB output data register (GPIOB_OUTDR)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
x	PBxOUTD	R/W	Output data (x=0~12) 0: Output low level 1: Output high level Bitwise output 1: Controlled by PBxBS Bitwise output 0: Controlled by PBxBR and PBxBRR

10.7.21 GPIOB Data Bit Operation Register (GPIOB_BSRR)

Offset address: 0x20

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:29	Reserved		
x+16	PAxBR	W	PBxOUTD Reset (x=0~12) Write-only, reading these bits returns 0. 0: No operation on the corresponding PBxOUTD bit 1: Reset the corresponding PBxOUTD bit Note: If PAxBR and PAxBS are set at the same time, PAxBS has higher priority.

Field	Name	R/W	Description
15:13	Reserved		
x	PAxBS	W	PBxOUTD set (x=0~12) Write-only, reading these bits returns 0. 0: No operation on the corresponding PBxOUTD bit 1: Set the corresponding PBxOUTD bit to 1

10.7.22 GPIOB Data Bit Clear Register (GPIOB_BRR)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
x	PAxBRR	W	PBxOUTD Reset (x=0~12) Write-only, reading these bits returns 0. 0: No operation on the corresponding PBxOUTD bit 1: Reset the corresponding PBxOUTD bit

10.7.23 GPIOB Lock Register (GPIOB_LOCK)

Offset address: 0x28

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	KEY	R/W	GPIO Lock Register Writing 0xA5A5 5A5A to this register allows writing to GPIO registers. After configuring the GPIO registers, writing other values to this register prohibits writing to other GPIO registers.

10.7.24 GPIOB Multiplex Select Register 0 (GPIOB_AFSELR0)

Offset address: 0x2C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31	Reserved		
30:28	PB7AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
27	Reserved		
26:24	PB6AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2

Field	Name	R/W	Description
			011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
23	Reserved		
22:20	PB5AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
19	Reserved		
18:16	PB4AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
15	Reserved		
14:12	PB3AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
11	Reserved		
10:8	PB2AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7

Field	Name	R/W	Description
7	Reserved		
6:4	PB1AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
3	Reserved		
2:0	PB0AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7

10.7.25 GPIOB Multiplex Select Register 1 (GPIOB_AFSELR1)

Offset address: 0x30

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:19	Reserved		
18:16	PB12AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
15	Reserved		
14:12	PB11AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7

Field	Name	R/W	Description
11	Reserved		
10:8	PB10AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
7	Reserved		
6:4	PB9AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7
3	Reserved		
2:0	PB8AFSEL	R/W	multiplex select 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7

11 Advanced Timer (ATIMER)

11.1 Introduction

The advanced timer ATIMER takes the time base unit as the core, with the functions of output compare and braking input, and has a 16-bit autoload counter. Compared with other timers, the advanced timer supports complementary output, repeat count and programmable dead zone insertion function, and is more suitable for motor control.

11.2 Main characteristics

(1) Timebase unit
- Counter: 16-bit counter, count-up, count-down and central alignment count
- Prescaler: 16-bit programmable prescaler
- Repeat counter: 16-bit repeat counter
- Autoreload function

(2) Clock source selection
- Internal clock
- External trigger
- Internal trigger

(3) Counting function

(4) Output compare function

(5) PWM output mode

(6) Forced output mode

(7) Independent output mode

(8) Single-pulse mode

(9) Complementary output and dead zone insertion

(10) Timing function

(11) Braking function

(12) Master/Slave mode controller of timer
- Timers can be synchronized and cascaded
- Support multiple slave modes and synchronization signals

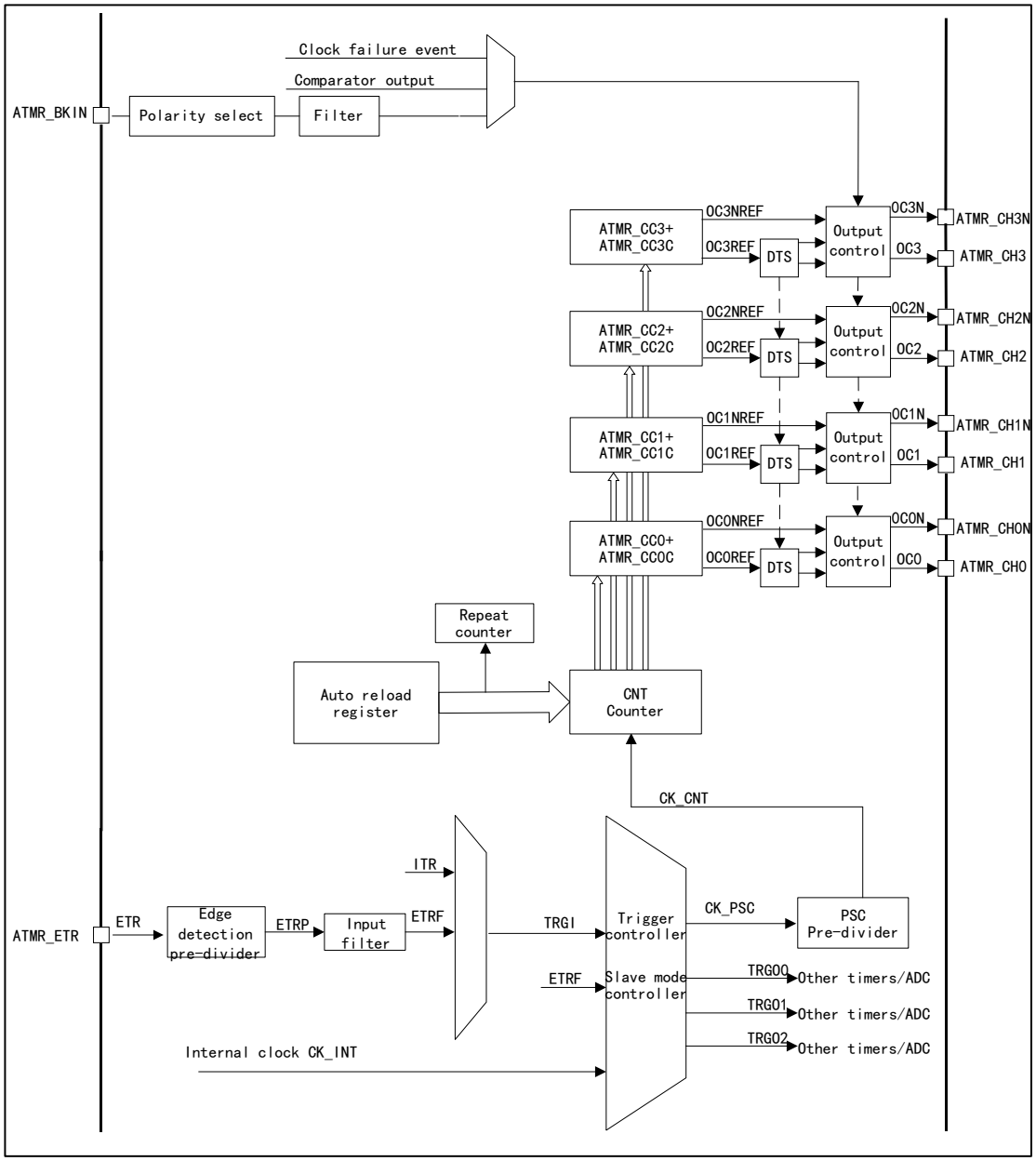
(13) Interrupt output request events
- Update event (counter overrun/underrun, counter initialization)
- Trigger event (counter start, stop, internal/external trigger)

- Comparison events
- Braking signal input event

(14) Supports ETR input (external trigger input) function, which can be used as external clock or cycle-by-cycle current management

11.3 Structure block diagram

Figure 21 Structure Block Diagram



11.4 Functional description

11.4.1 Clock source selection

The advanced timer has 3 clock sources.

Internal clock

It is ATMR_CLK from RCC, namely the driving clock of the timer; when the slave mode controller is disabled, the clock source CK_PSC of the prescaler is driven by the internal clock CK_INT.

External clock mode

After polarity selection, frequency division and filtering, the signal from external trigger interface (ETR) is connected to the slave mode controller through trigger input selector to control the work of the counter.

Internal trigger input

The timer is set to work in slave mode, and the clock source is the output signal of other timers. At this time, the clock source has no filtering, and the synchronization or cascading between timers can be realized. The master mode timer can reset, start, stop or provide clock for the slave mode timer.

11.4.2 Timebase unit

The timebase unit in the advanced timer contains four registers

- Counter register (CNT) 16 bits
- Autoreload register (AUTORLD) 16 bits
- Prescaler (PSC) 16 bits
- Repetition count register (REPCNT) 8 bits

Counter CNT

There are three count modes for the counter in the advanced timer

- Count-up mode
- Count-down mode
- Central alignment mode

Count-up mode

Set to the count-up mode by configuring CNTDIR bit of control register (ATMR_CR1).

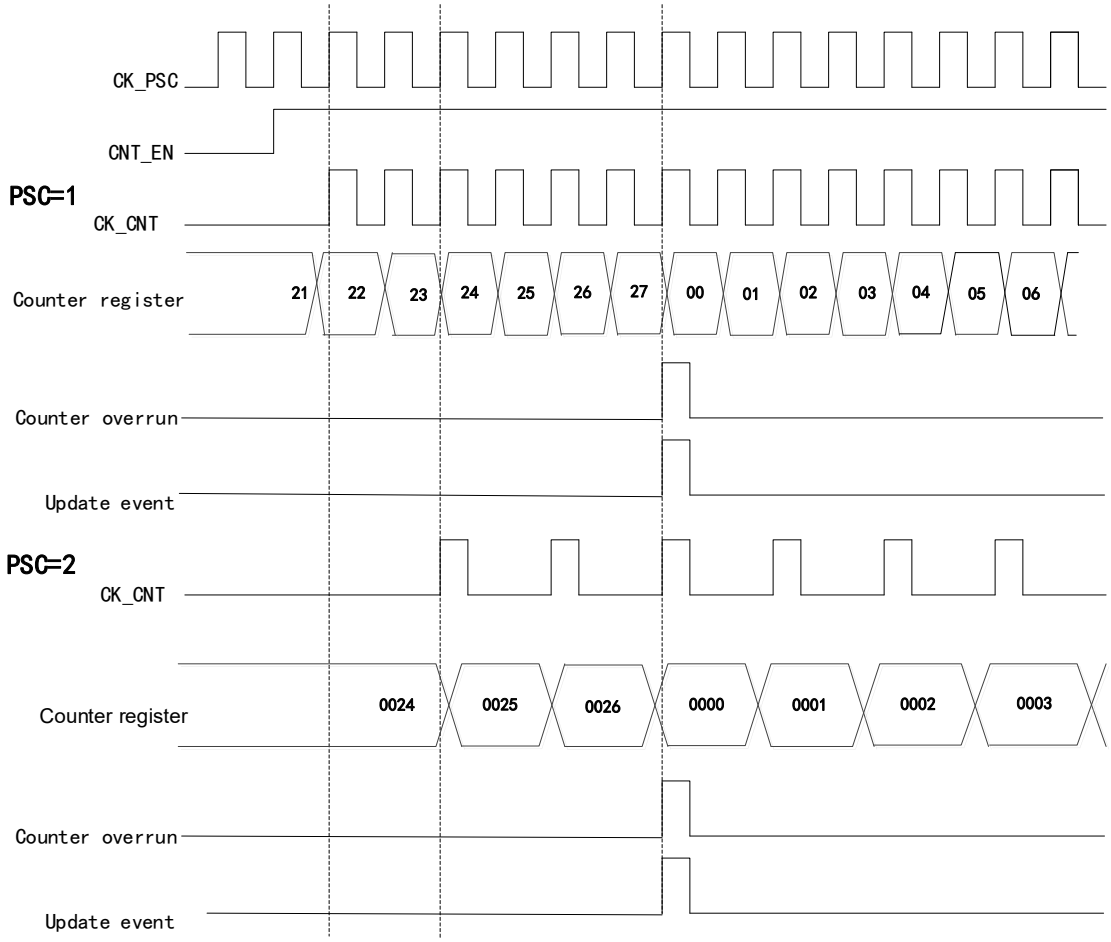
When the counter is in count-up mode, the counter will count up from 0; every time a pulse is generated, the counter will increase by 1 and when the value of the counter (ATMR_CNT) is equal to the value of the auto reload (ATMR_AUTORLD), the counter will start to count from 0 again, a count-up overrun event will be generated, and the value of the auto reload (ATMR_AUTORLD) is written in advance.

If a repeat counter is used, an update event will be generated when the number of count-up repetitions reaches the number in the repeat counter register plus one time (ATMR_REPCNT+1). Otherwise, an update event will be generated

every time the counter overruns. At this time, the repeat count shadow register, the auto reload shadow register and the prescaler buffer will be updated. The update event can be disabled by configuring UDISEN bit of control register ATMR_CR1.

The figure below is the timing diagram of count-up mode when the division factor is 1 or 2.

Figure 22 Timing Diagram of Count-up Mode when Division Factor is 1 or 2



Count-down mode

Set to the count-down mode by configuring CNTDIR bit of control register (ATMR_CR1).

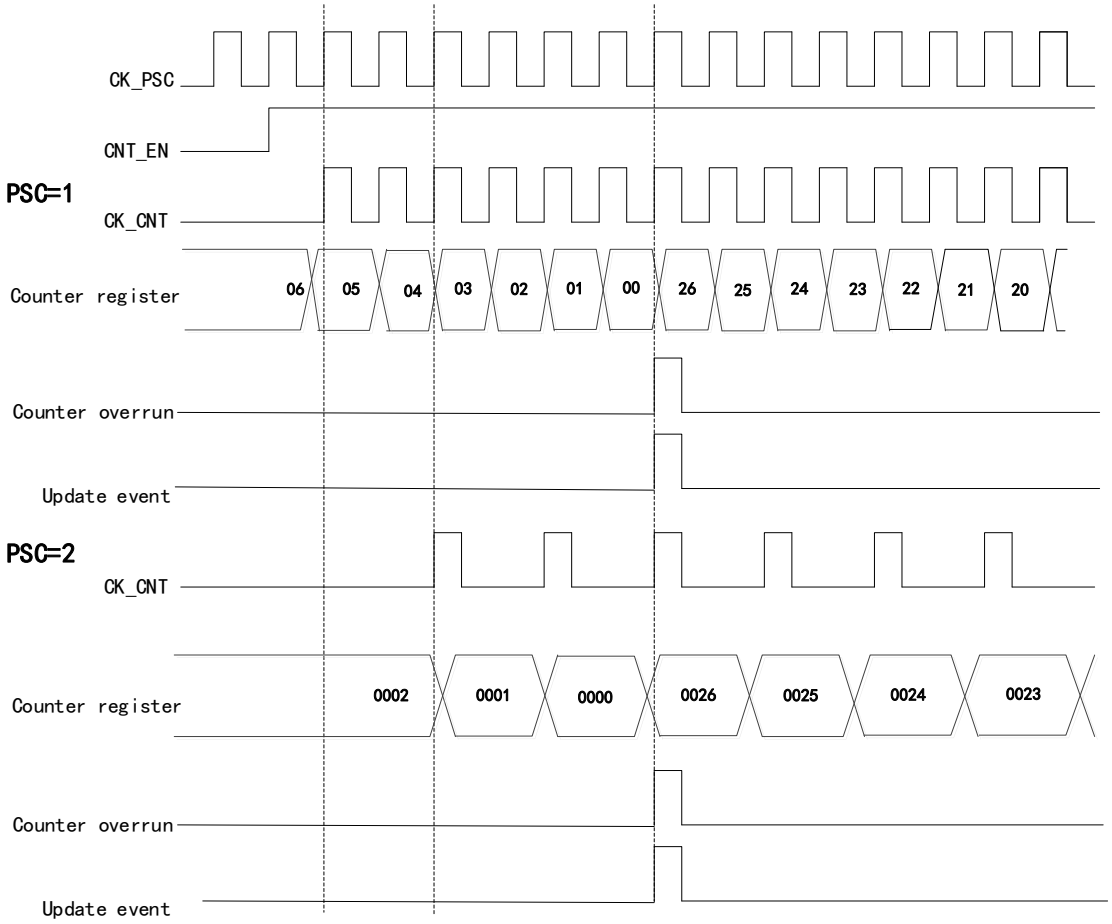
When the counter is in count-down mode, it will start to count down from the value of the auto reload (ATMR_AUTORLD); every time a pulse is generated, the counter will decrease by 1 and when it becomes 0, the counter will start to count again from (ATMR_AUTORLD), meanwhile, a count-down overrun event will be generated, and the value of the auto reload (ATMR_AUTORLD) is written in advance.

If a repeat counter is used, an update event will be generated when the number

of count-down repetitions reaches the number in the repeat counter register plus one time (ATMR_REPCNT+1). Otherwise, an update event will be generated every time the counter underruns. At this time, the repeat count shadow register, the auto reload shadow register and the prescaler buffer will be updated. The update event can be disabled by configuring the UDISEN bit of the ATMR_CR1 register.

The figure below is the timing diagram of count-down mode when the division factor is 1 or 2.

Figure 23 Timing Diagram of Count-down Mode when Division Factor is 1 or 2



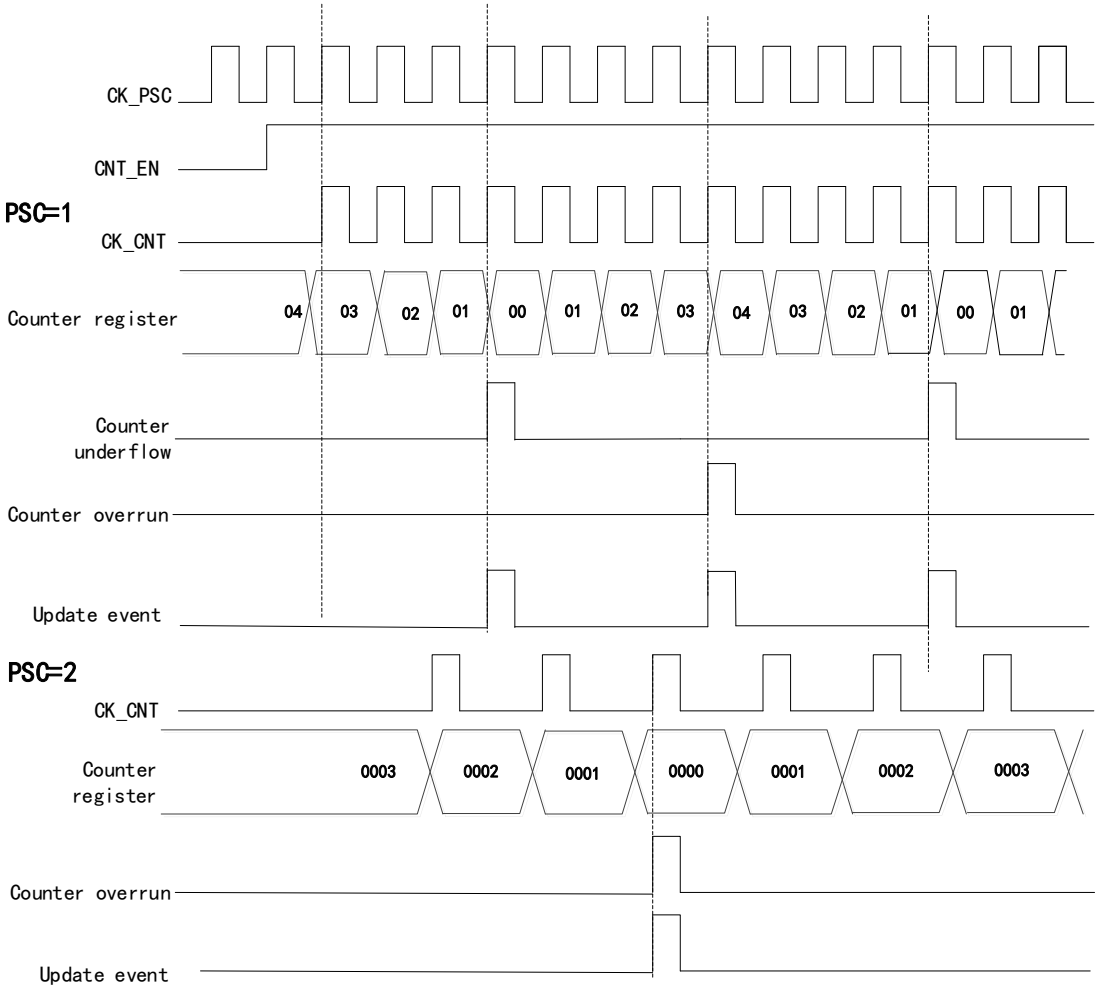
Central alignment mode

Set to the central alignment mode by configuring CAMSEL bit of control register (ATMR_CR1).

When the counter is in center alignment mode, the counter counts up from 0 to the value of auto reload (ATMR_AUTORLD), then counts down to 0 from the value of the auto reload (ATMR_AUTORLD), which will repeat; in counting up, when the counter value is (AUTORLD-1), a counter overrun event will be generated; in counting down, when the counter value is 1, a counter underrun event will be generated.

The figure below is the timing diagram of central alignment mode when the division factor is 1 or 2.

Figure 24 Timing Diagram of Central alignment Mode when Division Factor is 1 or 2



Repeat counter REPCNT

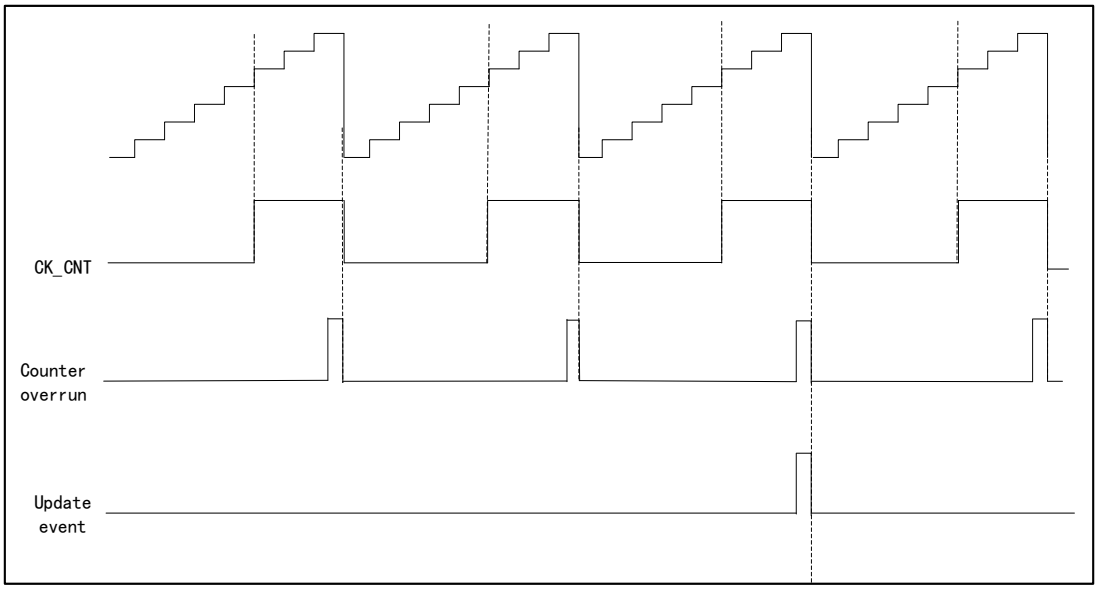
There is no repeat counter REPCNT in the basic/general-purpose timer, which means that when an overrun event or underrun event occurs in the basic/general-purpose timer, an update event will be generated directly; while in the advanced timer, because of the existence of the repeat counter, when an overrun/unerrrun event occurs to the advanced timer, the update event will be generated only when the value of the repeat counter is 0.

For example, if the advanced timer needs to generate an update event when an overrun/underrun event occurs, the value of the repeat counter should be set to 0.

If the repeat counter function is used in the count-up mode, every time the counter counts up to AUTORLD, an overrun event will occur. At this time, the value of the repeat counter will decrease by 1, and an update event will be generated when the value of the repeat counter is 0.

That is, when N+1 (N is the value of repeat counter) overrun/underrun events occur, an update event will be generated.

Figure 25 Timing Diagram of Count-up Mode when Setting REPCNT=2



Prescaler PSC

The prescaler is 16 bits and programmable, and it can divide the clock frequency of the counter to any value within 1~65536 (controlled by ATMR_PSC register), and after frequency division, the clock will drive the counter CNT to count. The prescaler has a buffer, which can be changed during running.

11.4.3 Output compare

There are eight modes of output compare: freeze, channel x is valid level when matching, channel x is invalid level when matching, reverse, forced to invalid, forced to valid, PWM1 and PWM2 mode, which are configured by OCxMOD bit in ATMR_CCMx register and can control the waveform of output signal in output compare mode.

Output compare application

In the output compare mode, the position, polarity, frequency and time of the pulse generated by the timer can be controlled.

When the value of the counter is equal to that of the capture/compare register, the channel output can be set as high level, low level or reverse by configuring the OCxMOD bit in ATMR_CCMx register and the CCxPOL bit in the output polarity ATMR_CCEN register.

When CCxIFLG in the ATMR_SR register is 1, an interrupt occurs if CCxIEN in the ATMR_IER register is 1.

11.4.4 PWM output mode

PWM mode is pulse signal that can be adjusted by external output of the timer. The pulse width of the signal is determined by the value of the compare register CCx, and the cycle is determined by the value of the auto reload AUTORLD.

PWM output mode contains PWM mode 1 and PWM mode 2; PWM mode 1 and PWM mode 2 are divided into count-up, count-down and central alignment counting; in PWM mode 1, if the value of the counter CNT is less than the value of the compare register CCx, the output level will be valid; otherwise, it will be invalid.

Set the timing diagram in PWM mode 1 when CCx=5, AUTORLD=7.

Figure 26 Timing Diagram of PWM1 Count-up Mode

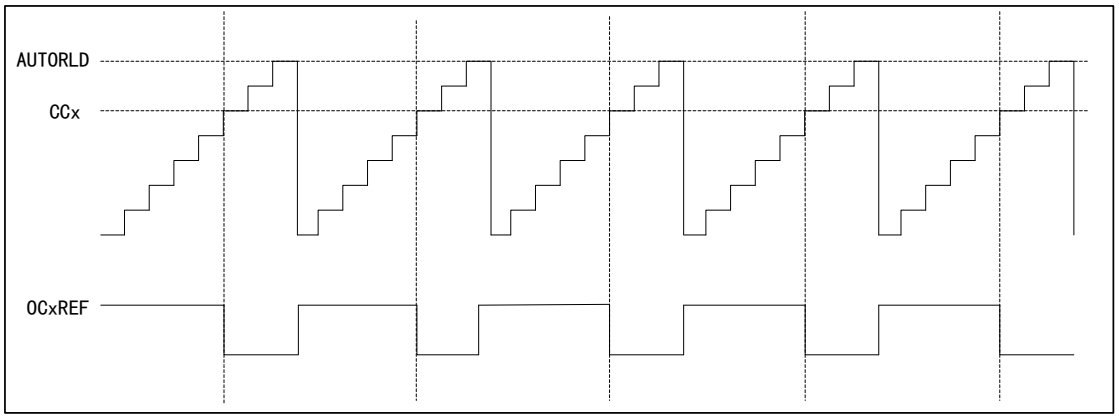


Figure 27 Timing Diagram of PWM1 Count-down Mode

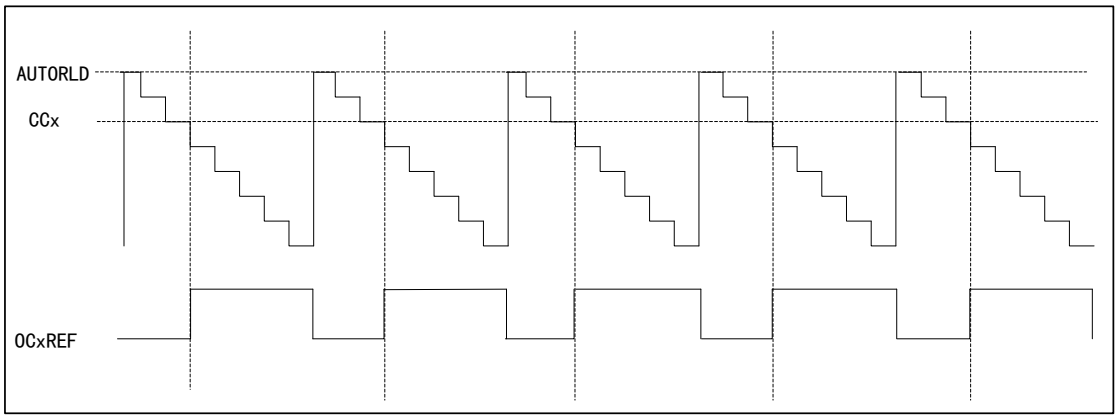
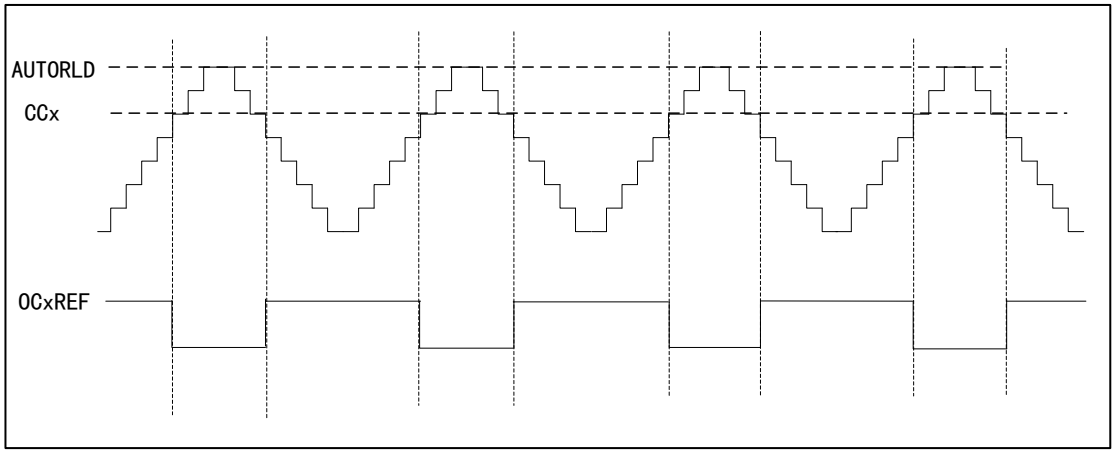


Figure 28 Timing Diagram of PWM1 Central alignment Mode



In PWM mode 2, if the value of the counter CNT is less than that of the compare register CCx, the output level will be invalid; otherwise, it will be valid.

Set the timing diagram of PWM mode 2 when CCx=5, AUTORLD=7.

Figure 29 Timing Diagram of PWM2 Count-up Mode

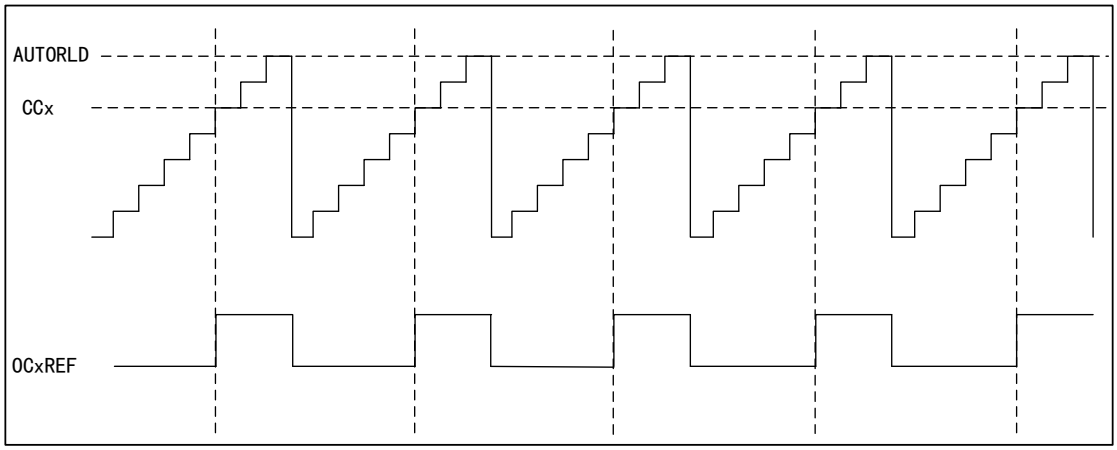


Figure 30 Timing Diagram of PWM2 Count-down Mode

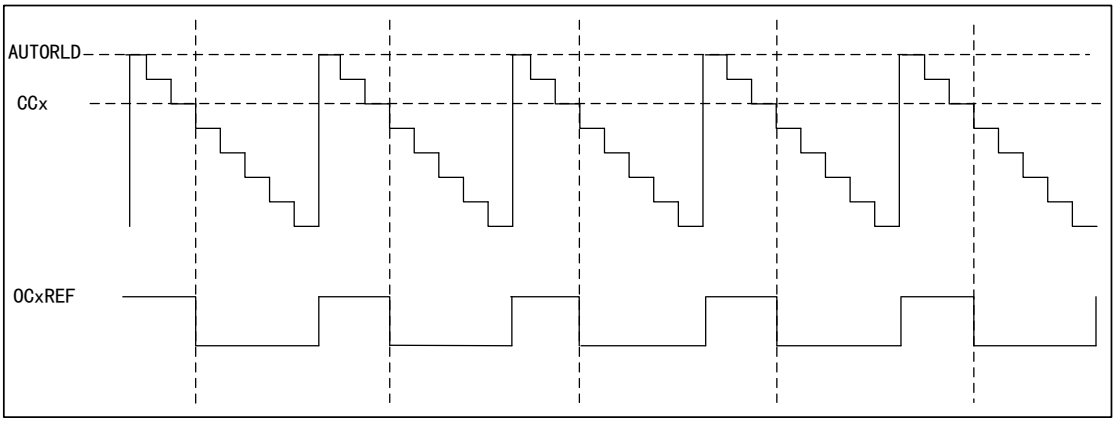
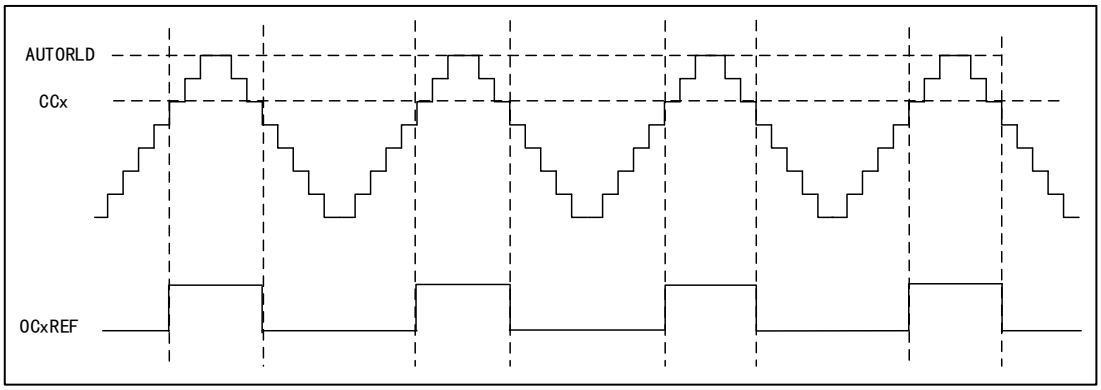


Figure 31 Timing Diagram of PWM2 Central alignment Mode



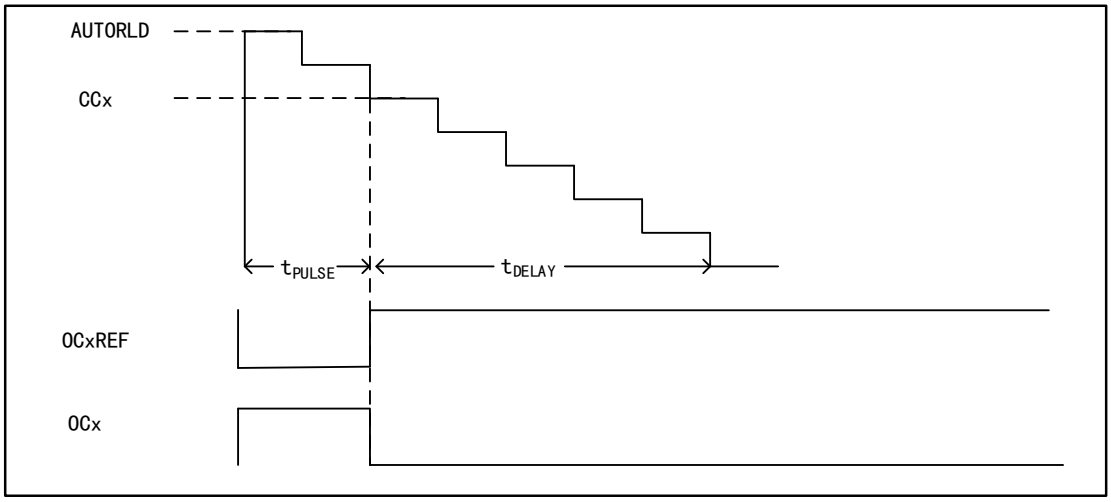
11.4.5 Single-pulse mode

The single-pulse mode is a special case of timer compare output, and is also a special case of PWM output mode.

Set SPMEN bit of ATMR_CR1 register, and select the single-pulse mode. After the counter is started, a certain number of pulses will be output before the update event occurs. When an update event occurs, the counter will stop counting, and the subsequent PWM waveform output will no longer be changed.

After a certain controllable delay, a pulse with controllable pulse width is generated in single-pulse mode through the program. The delay time is defined by the value of ATMR_CCx register; in the count-up mode, the delay time is CCx and the pulse width is AUTORLD-CCx; in the count-down mode, the delay time is AUTORLD-CCx and the pulse width is CCx.

Figure 32 Timing Diagram of Single-pulse Mode



11.4.6 Impact of the register on output waveform

The following registers will affect the level of the timer output waveform. For details, please refer to "Register Functional Description".

(1) CCxEN and CCxNEN bits in ATMR_CCEN register

- CCxNEN=0 and CCxEN=0: The output is disabled (output disabled, invalid)
- CCxNEN=1 and CCxEN=1: The output is enabled (output enabled, normal output)

(2) MOEN bit in ATMR_BDT register

- MOEN=0: Idle mode
- MOEN=1: Run mode

(3) OCxOIS and OCxNOIS bits in ATMR_CR2 register

- OCxOIS=0 and OCxNOIS=0: When idle (MOEN=0), the output level after the dead zone is 0
- OCxOIS=1 and OCxNOIS=1: When idle (MOEN=0), the output level after the dead zone is 1

(4) RMOS bit in ATMR_BDT register

- Application environment of RMOS: In corresponding complementary channel and timer run mode (MOEN=1), the timer is not working (CCxEN=0, CCxNEN=0) or is working (CCxEN=1, CCxNEN=1)

(5) IMOS bit in ATMR_BDT register

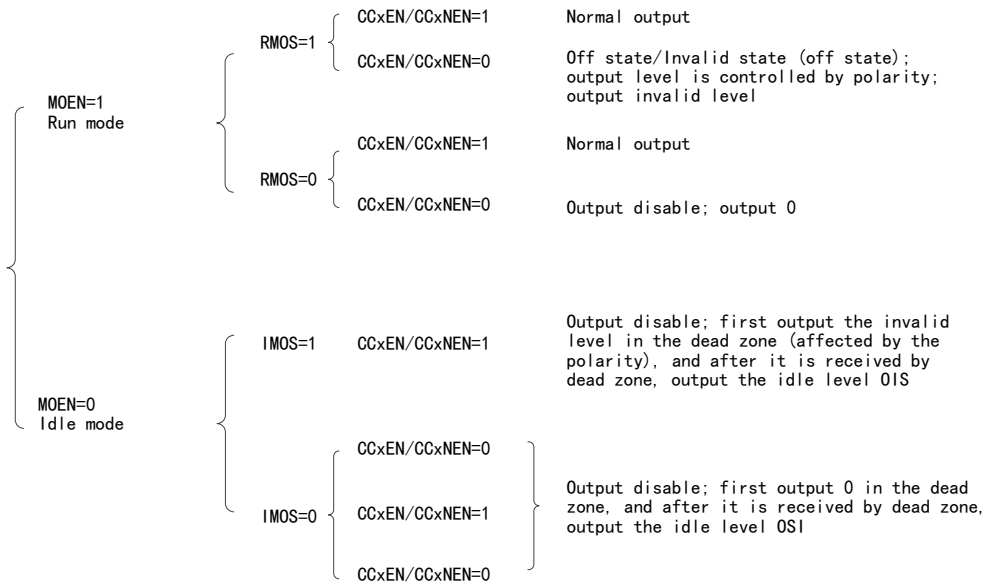
- Application environment of IMOS: In idle mode of corresponding complementary channel and timer (MOEN=0), the timer is not working (CCxEN=0, CCxNEN=0) or is working (CCxEN=1, CCxNEN=1)

(6) CCxPOL and CCxNPOL bits of ATMR_CCEN register

- CCxPOL=0 and CCxNPOL=0: Output polarity, valid at high level
  CCxPOL=1 and CCxNPOL=1: Output polarity, valid at low level

The following figure lists the register structural relationships that affect the output waveform

Figure 33 Register Structural Relationship Affecting Output Waveform



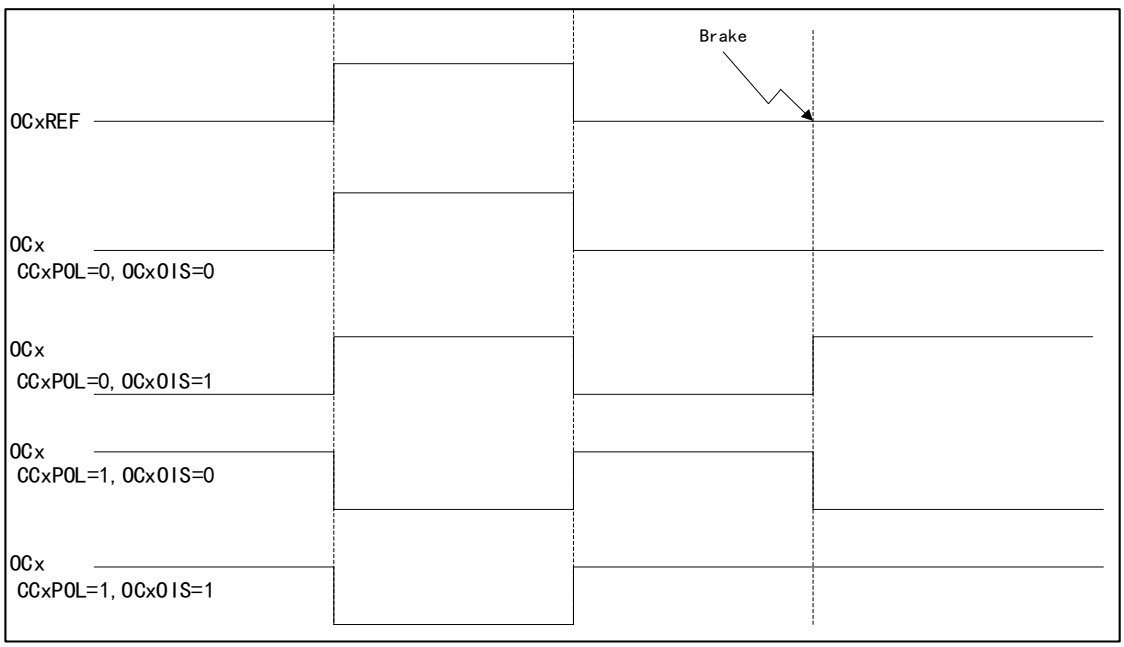
11.4.7 Braking function

The signal source of braking is clock fault event and external input interface. The input filtering function is configured through ATMR_BREAK.

Besides, the BRKEN bit in ATMR_BDT register can enable the braking function, and the BRKPOL bit can configure the polarity of braking input signal.

When a braking event occurs, the output pulse signal level can be modified according to the state of the relevant control bit.

Figure 34 Braking Event Timing Diagram



11.4.8 Complementary output and dead zone insertion

Complementary output is particular output of advanced timer, and the advanced timer has three groups of complementary output channels. The insertion dead time is used to generate complementary output signals to ensure that the two-way complementary signals of channels will not be valid at the same time. Set the dead time according to the output device connected to the timer and its characteristics. Configuring the DTS0, DTS1 bit of the ATMR_BDT register can control the duration of the dead time.

Figure 35 Complementary Output with Dead Zone Insertion

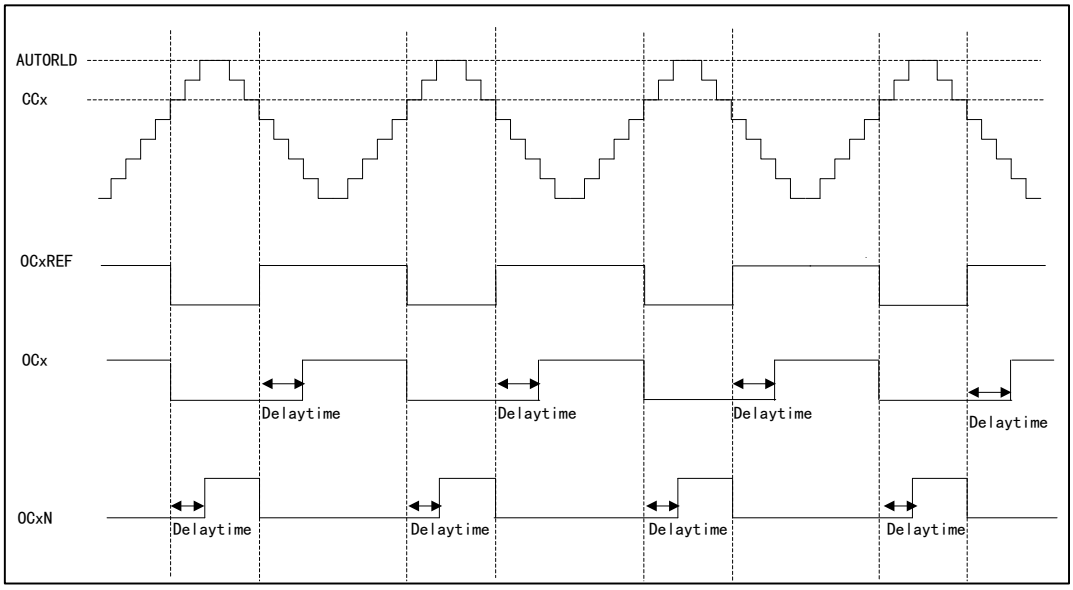
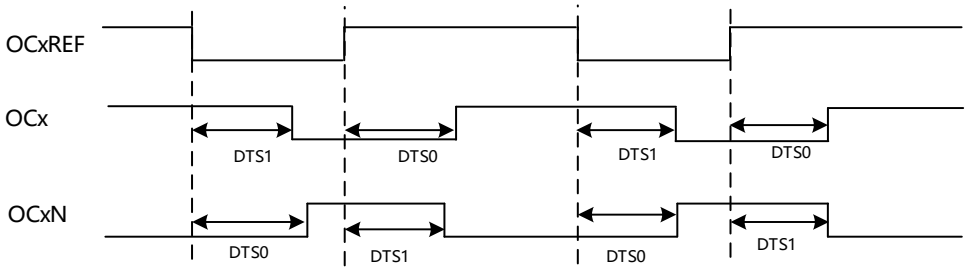


Figure 36 Complementary Output with Double Dead Zone Insertion



11.4.9 Forced output mode

In the forced output mode, the comparison result is ignored, and the corresponding level is directly output according to the configuration instruction.

- CCxSEL=00 for ATMR_CCMx register, set CCx channel as output
- OCxMOD=100/101 for ATMR_CCMx register, set to force OCxREF signal to invalid/valid
- ATMR_OCRx also controls the output. For details, please refer to the corresponding register description

In this mode, the corresponding interrupt will still be generated.

11.4.10 Independent output mode

By configuring the NONCxEN of ATMRx_OCxACR to 1, the channel output can be selected as the independent output mode. At this time, the count comparison value corresponding to CHxN adopts the configuration value of ATMRx_CCxC. The count comparison value corresponding to CHx still adopts the configuration value of ATMRx_CCx.

11.4.11 Slave mode

ATMR timer can synchronize external trigger

- Reset mode
- Gated mode
- Trigger mode

SMFSEL bit in ATMR_SMCR register can be set to select the mode

SMFSEL=100 set the reset mode, SMFSEL=101 set the gated mode, and SMFSEL=110 set the trigger mode.

In the reset mode, when a trigger input event occurs, the counter and prescaler will be initialized, and the rising edge of the selected trigger input (TRGI) will reinitialize the counter and generate a signal to update the register.

In the gated mode, the enable of the counter depends on the high level of the selected input end. When the trigger input is high, the clock of the counter will be enabled. Once the trigger input becomes low, the counter will stop (but not be reset). The start and stop of the counter are controlled.

In the trigger mode, the enable of the counter depends on the event on the selected input, the counter will be enabled at the rising edge of the trigger input (but not be reset), and only the start of the counter is controlled.

11.4.12 Timer interconnection

Each ATMR can be connected to each other to realize synchronization or cascading between timers (i.e., interconnection between GTMR and ATMR). It is required to configure one timer in master mode and the other timer in slave mode.

When the timer is in master mode, it can reset, start, stop and provide clock source for the counter of the slave mode timer.

When the timers are interconnected:

- A timer can be used as the prescaler of other register
- Start the other register by the enable signal of a timer
- Start the other register by the update event of a timer
- Select the other register by the enable of a timer
- Two timers can be synchronized by an external trigger

11.4.13 Interrupt request

The timer can generate an interrupt when an event occurs during operation.

- Update event (counter overrun/underrun, counter initialization)
- Trigger event (counter start, stop, internal/external trigger)
- Capture/Compare event
- Braking signal input event

11.4.14 Debugging mode

ATMR can be configured in debug mode to choose whether to stop or continue working. It depends on the control of the system registers.

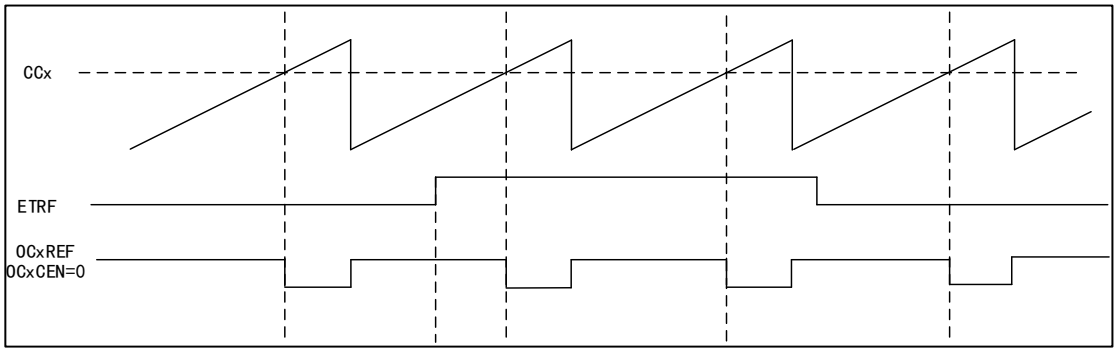
11.4.15 Clear OCxREF signal when an external event occurs

This function is used for output compare and PWM mode.

In one channel, the high level of ETRF input port will reduce the signal of OCxREF to low level, and the OCxCEN bit in capture/compare register ATMR_CCMx is set to 1, and OCxREF signal will remain low until the next update event occurs.

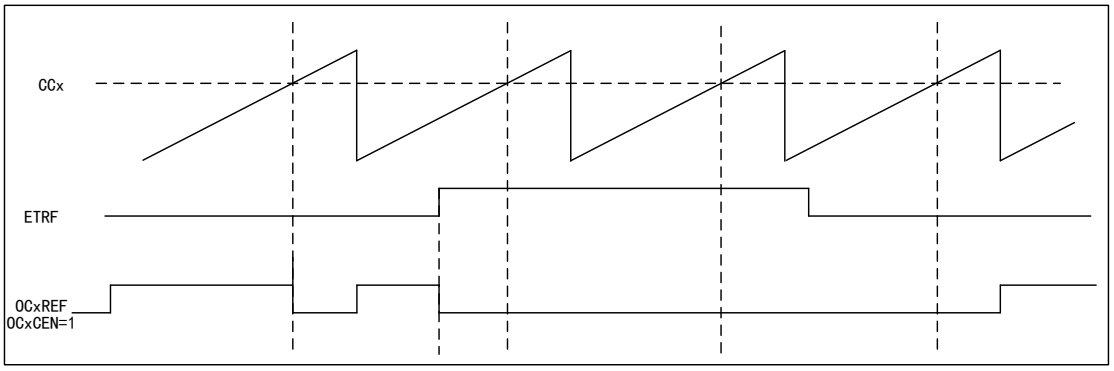
Set ATMR to PWM mode, disable the external trigger prescaler, and disable the external trigger mode 2; when ETRF input is high, set OCxCEN=0, and the output OCxREF signal is shown in the figure below.

Figure 37 OCxREF Timing Diagram



Set ATMR to PWM mode, disable the external trigger prescaler, and disable the external trigger mode 2; when ETRF input is high, set OCxCEN=1, and the output OCxREF signal is shown in the figure below.

Figure 38 OCxREF Timing Diagram



11.4.16 Timer internal trigger connection

Table 36 Timer Internal Trigger Connection

Slave timer	ITR0
ATMR	GTMR
GTMR	ATMR

11.5 Register address mapping

In the following table, all registers of the advanced timer are mapped to a 16-bit addressable (addressing) space.

Table 37 ATIMER Register Address Mapping

Register name	Description	Offset address
ATMR_CR1	Control register 1	0x00
ATMR_CR2	Control register 2	0x04
ATMR_SMCR	Slave mode control register	0x08
ATMR_IER	Interrupt enable register	0x0C
ATMR_SR	Status register	0x10
ATMR_CEG	Control event generation register	0x14
ATMR_CCM1	Compare mode register 1	0x18
ATMR_CCM2	Compare mode register 2	0x1C
ATMR_CCEN	Compare enable register	0x20
ATMR_CNT	Counter register	0x24
ATMR_PSC	Prescale register	0x28
ATMR_AUTORLD	Auto reload register	0x2C
ATMR_REPCNT	Repeat count register	0x30
ATMR_CC0	Channel 0 compare register	0x34

Register name	Description	Offset address
ATMR_CC1	Channel 1 compare register	0x38
ATMR_CC2	Channel 2 compare register	0x3C
ATMR_CC3	Channel 3 compare register	0x40
ATMR_BDT	Braking and dead zone register	0x44
ATMR_OCR1	Output control register 1	0x48
ATMR_OCR2	Output control register 2	0x4C
ATMR_TRGOCR	TRGO control register	0x50
ATMR_BREAK	Break filter register	0x54
ATMR_OCxACR	Lower compare register control register	0x58
ATMR_CC0C	Channel 0 lower compare register	0x5C
ATMR_CC1C	Channel 1 lower compare register	0x60
ATMR_CC2C	Channel 2 lower compare register	0x64
ATMR_CC3C	Channel 3 lower compare register	0x68

11.6 Register functional description

11.6.1 Control register 1 (ATMR_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
12:10	BCS	R/W	BRK source select 000: brk_in 001: comp0_in 010: comp1_in 011: comp2_in 100: comp3_in Others: brk_in Note: The input source of the comparator is valid at a low level, and the default value of the comparator upon power-on is also low. When used as the brake input, the output of the comparator needs to be configured in advance.
9:8	CLKDIV	R/W	Clock Division For the configuration of dead zone and digital filter, CK_INT provides the clock, and the dead time and the clock of the digital filter can be adjusted by this bit. 00: $T_{DTS}=t_{CK\_INT}$ 01: $T_{DTS}=2\times t_{CK\_INT}$ 10: $T_{DTS}=4\times t_{CK\_INT}$ 11: Reserved

Field	Name	R/W	Description
7	ARPEN	R/W	ATMR_AUTORLD register Auto-reload Preload Enable When the buffer is disabled, modification of ATMR_AUTORLD by program will immediately lead to modification of the values loaded to the counter; when the buffer is enabled, modification of ATMR_AUTORLD by program will lead to modification of the values loaded to the counter at the next update event. 0: Disable 1: Enable
6:5	CAMSEL	R/W	Center Aligned Mode Select In the central alignment mode, the counter counts up and down alternately; otherwise, it will only count up or down. Different center alignment modes affect the timing of setting the output comparison interrupt flag bit of the output channel to 1; when the counter is disabled (CNTEN=0), select the center alignment mode. 00: Edge-aligned mode 01: Center alignment mode 1 (the output compare interrupt flag bit of output channel is set to 1 when counting down) 10: Center alignment mode 2 (the output compare interrupt flag bit of output channel is set to 1 when counting up) 11: Center alignment mode 3 (the output compare interrupt flag bit of output channel is set to 1 when counting up/down)
4	CNTDIR	R	Counter Direction This bit is read-only when the counter is configured as central alignment mode. 0: Count up 1: Count down
3	SPMEN	R/W	Single Pulse Mode Enable When an update event is generated, the output level of the channel can be changed; in this mode, the CNTEN bit will be cleared, the counter will be stopped, and the subsequent output level of the channel will no long be changed. 0: Disable 1: Enable
2	URSSEL	R/W	Update Request Source Select If interrupt or DMA is enabled, the update event can generate update interrupt or DMA request. Different update request sources can be selected by this bit. 0: The counter overruns or underruns Set UEG bit Update generated by slave mode controller 1: The counter overruns or underruns
1	UDISEN	R/W	Update Disable Update event can cause AUTORLD, PSC and CCx to generate the value of update setting. 0: Enable update event (UEV) An update event can occur in any of the following situations: The counter overruns/underruns; Set UEG bit; Update generated by slave mode controller.

Field	Name	R/W	Description
			1: Disable update event
0	CNTEN	R/W	Counter Enable 0: Disable 1: Enable When the timer is configured as external clock, gated mode, it is required to write 1 to the bit by software to start regular work; when it is configured as the trigger mode, it can write 1 by hardware.

11.6.2 Control register 2 (ATMR_CR2)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	OC3NOIS	R/W	Configure OC3N output idle state. Refer to OC0NOIS bit Refer to the OC0NOIS position
14	OC3OIS	R/W	Configure OC3 output idle state. Refer to OC0OIS bit Refer to the OC0OIS position
13	OC2NOIS	R/W	Configure OC2N output idle state. Refer to OC0NOIS bit Refer to the OC0NOIS position
12	OC2OIS	R/W	Configure OC2 output idle state. Refer to OC0OIS bit Refer to the OC0OIS position
11	OC1NOIS	R/W	Configure OC1N output idle state. Refer to OC0NOIS bit Refer to the OC0NOIS position
10	OC1OIS	R/W	Configure OC1 output idle state. Refer to OC0OIS bit Refer to the OC0OIS position
9	OC0NOIS	R/W	OC0N Output Idle State Configure Only the level state after the dead time of OC0 is affected when MOEN=0 and OC0N is realized. 0: OC0N=0 1: OC0N=1 Note: When LOCKCFG bit in ATMR_BDT register is at the Level 1, 2 or 3, this bit cannot be modified.
8	OC0OIS	R/W	OC0 Output Idel State Configure Only the level state after the dead time of OC0 is affected when MOEN=0 and OC0N is realized. 0: OC0=0 1: OC0=1 Note: When LOCKCFG bit in ATMR_BDT register is at the Level 1, 2 or 3, this bit cannot be modified.
7:4	MMSEL	R/W	Select the signal for TRGO0 in the timer's main mode The signals of timers working in master mode can be used for TRGO0, to affect the work of timers in slave mode and cascaded with the master timer, and the specific impact is related to the configuration of slave mode timer. 0000: Reset; the reset signal of master mode timer is used for TRGO0

Field	Name	R/W	Description
			0001: Enable; the counter enable signal of master mode timer is used for TRGO0 0010: Update; the update event of master mode timer is used for TRGO0 0011: Compare pulses; when the master mode timer captures/compares successfully (CC0IFLG=1), a pulse signal is output for TRGO0 0100: Compare mode 0; OC0REF is used to trigger TRGO0 0101: Compare mode 1; OC1REF is used to trigger TRGO0 0110: Compare mode 2; OC2REF is used to trigger TRGO0 0111: Compare mode 3; OC3REF is used to trigger TRGO0 1000: TRGO0 is not generated (TRGO0 will be generated only according to MMSZE and MMSPE) 1001: Compare mode 4; OCN3REF is used to trigger TRGO0 1010: OC3REF rising and falling edges generate TRGO0 1011: OC3NREF rising and falling edges generate TRGO0 1100: OC3REF rising edge and OC3NREF rising edge generate TRGO0 1101: OC3REF falling edge and OC3NREF falling edge generate TRGO0 1110: OC3REF rising edge and OC3NREF falling edge generate TRGO0 1111: OC3REF falling edge and OC3NREF rising edge generate TRGO0
3	MMSPE	R/W	TRGO0 Signal is Generated when the Counter Matches the Autoreload Register The generated TRGO is only valid when the counter is in center alignment mode and is only selected when MMSEL is set to 1000 0: TRGO0 is not generated when the counter matches the autoreload register 1: TRGO0 is generated when the counter matches the autoreload register
2	CCUSEL	R/W	Compare Control Update Select Works only when the capture/compare preload is enabled (CCPEN=1), and it works only for complementary output channel. 0: It can only be updated by setting COMG bit 1: It can be updated by setting COMG bit or rising edge on TRGI
1	MMSZE	R/W	When the counter returns to 0, TRGO0 signal is generated The generated TRGO is only valid when the counter is in center-aligned mode, and only when MMSEL (selected when configured as 1000) 1: TRGO0 is generated when the counter returns to 0 0: TRGO0 is not generated when the counter returns to 0
0	CCPEN	R/W	Compare Preloaded Enable This bit affects the change of CCxEN, CCxNEN and OCxMOD values. When preloading is disabled, the program modification will immediately affect the timer setting; when preloading is enabled, it is only updated after COMG is set, so as to affect the setting of the timer; this bit only works on channels with complementary output. 0: Disable 1: Enable

11.6.3 Slave mode control register (ATMR_SMCR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	ETPOL	R/W	External Trigger Polarity Configure This bit decides whether the external trigger ETR is phase-inverting. 0: The external trigger ETR is not phase-inverting, and the high level or rising edge is valid 1: The external trigger ETR is phase-inverting, and the low level or falling edge is valid
14	Reserved		
13:12	ETPCFG	R/W	External Trigger Prescaler Configure The ETR (external trigger input) signal becomes ETRP after frequency division. The signal frequency of ETRP is at most 1/4 of ATMRCLK frequency; when ETR frequency is too high, the ETRP frequency must be reduced through frequency division. 00: Disable the prescaler 01: ETR signal 2 frequency division 10: ETR signal 4 frequency division 11: ETR signal 8 frequency division
11:8	ETFCFG	R/W	External Trigger Filter Configure 0000: Disable filter, sampled by f_{DTS} 0001: DIV=1, N=2 0010: DIV=1, N=4 0011: DIV=1, N=8 0100: DIV=2, N=6 0101: DIV=2, N=8 0110: DIV=4, N=6 0111: DIV=4, N=8 1000: DIV=8, N=6 1001: DIV=8, N=8 1010: DIV=16, N=5 1011: DIV=16, N=6 1100: DIV=16, N=8 1101: DIV=32, N=5 1110: DIV=32, N=6 1111: DIV=32, N=8 Sampling frequency=timer clock frequency/DIV; the filter length=N, and a jump is generated by every N events.
7	MSMEN	R/W	Master/slave Mode Enable 0: Invalid 1: Enable the master/slave mode
6:5	Reserved		
4	TRGSEL	R/W	Trigger Input Signal Select In order to avoid generating false edge detection when changing the value of this bit, it must be changed when SMFSEL=0.

Field	Name	R/W	Description
			0: Internal trigger ITR0 1: External trigger input (ETRF)
3	Reserved		
2:0	SMFSEL	R/W	Slave Mode Function Select 000: Disable the slave mode, the timer can be used as master mode timer to affect the work of slave mode timer; if ATMR_CR1.CNTEN=1, the prescaler is directly driven by the internal clock. 100: Reset mode; the slave mode timer resets the counter after receiving the rising edge signal of TRGI and generates the signal to update the register. 101: Gated mode; when the slave mode timer receives the TRGI high level signal, the counter will start to work; when it receives TRGI low level signal, the counter will stop working; when it receives TRGI high level signal again, the timer will continue to work; the counter is not reset during the whole period. 110: Trigger mode, the slave mode timer starts the counter to work after receiving the rising edge signal of TRGI. Other: Reserved

11.6.4 Interrupt enable register (ATMR_IER)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	BRKIEN	R/W	Break Interrupt Enable 0: Disable 1: Enable
6	TRGIEN	R/W	Trigger Interrupt Enable 0: Disable 1: Enable
5	COMIEN	R/W	COM Interrupt Enable 0: Disable 1: Enable
4	CC3IEN	R/W	Compare Channel 3 Interrupt Enable 0: Disable 1: Enable
3	CC2IEN	R/W	Compare Channel 2 Interrupt Enable 0: Disable 1: Enable
2	CC1IEN	R/W	Compare Channel 1 Interrupt Enable 0: Disable 1: Enable
1	CC0IEN	R/W	Compare Channel 0 Interrupt Enable 0: Disable 1: Enable
0	UIEN	R/W	Update Interrupt Enable

Field	Name	R/W	Description
			0: Disable 1: Enable

11.6.5 Status register (ATMR_SR)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	BRKIFLG	RC_W0	Break Event Interrupt Generate Flag 0: No brake event occurs 1: Brake event occurs When brake input is valid, this bit is set to 1 by hardware; when brake input is invalid, this bit can be cleared to 0 by software.
6	TRGIFLG	RC_W0	Trigger Event Interrupt Generate Flag 0: No trigger event interrupt occurs 1: Trigger event interrupt occurs When a trigger event is generated, this bit is set to 1 by hardware and cleared to 0 by software.
5	COMIFLG	RC_W0	COM Event Interrupt Generate Flag 0: No COM event occurs 1: COM interrupt waits for response After COM event is generated, this bit is set to 1 by hardware and cleared to 0 by software.
4	CC3IFLG	RC_W0	Compare Channel 3 Interrupt Flag Refer to ATMR_SR.CC0IFLG
3	CC2IFLG	RC_W0	Compare Channel 2 Interrupt Flag Refer to ATMR_SR.CC0IFLG
2	CC1IFLG	RC_W0	Compare Channel 1 Interrupt Flag Refer to ATMR_SR.CC0IFLG
1	CC0IFLG	RC_W0	Compare Channel 0 Interrupt Flag 0: No matching occurs 1: The value of ATMR_CNT matches the value of ATMR_CC0 When the content of ATMR_CC0 is greater than that of ATMR_AUTORLD, the counter overflows in the up or up/down counting mode, or the counter overflows in the down counting mode, at this bit 1.
0	UIFLG	RC_W0	Update Event Interrupt Generate Flag 0: No update event interrupt occurs 1: Update event interrupt occurred When the counter value is reloaded or reinitialized, an update event will be generated. The bit is set to 1 by hardware and cleared to 0 by software; update events are generated in the following situations: (1) UDISEN=0 on ATMR_CR1 register, and when the value of the repeat counter overruns/underruns, an update event will be generated;

Field	Name	R/W	Description
			(2) URSSEL=0 and UDISEN=0 on ATMR_CR1 register, configure UEG=1 on ATMR_CEG register to generate an update event, and the counter needs to be initialized by software; (3) URSSEL=0 and UDISEN=0 on ATMR_CR1 register, and an update event will be generated when the counter is initialized by trigger event.

11.6.6 Control event generation register (ATMR_CEG)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	BEG	W	Break Event Generate 0: Invalid 1: Generate brake event This bit is set to 1 by software and cleared to 0 automatically by hardware.
6	TEG	W	Trigger Event Generate 0: Invalid 1: Generate trigger event This bit is set to 1 by software and cleared to 0 automatically by hardware.
5	COMG	W	Compare Control Update Event Generate 0: Invalid 1: Generate compare update event This bit is set to 1 by software and cleared to 0 automatically by hardware. Note: COMG bit is valid only in complementary output channel.
4	CC3EG	W	Compare Channel 3 Event Generation Refer to CC0EG description
3	CC2EG	W	Compare Channel 2 Event Generation Refer to CC0EG description
2	CC1EG	W	Compare Channel 1 Event Generation Refer to CC0EG description
1	CC0EG	W	Compare Channel 0 Event Generation 0: Invalid 1: Generate compare event This bit is set to 1 by software and cleared to 0 automatically by hardware. If Channel 0 is in output mode: When CC0IFLG=1, if CC0IEN and CC0DEN bits are set, the corresponding interrupt and DMA request will be generated. If Channel 0 is in input mode: The value of the capture counter is stored in ATMR_CC0 register; configure CC0IFLG=1, and if CC0IEN and CC0DEN bits are also set, the

Field	Name	R/W	Description
			corresponding interrupt and DMA request will be generated; at this time, if CC0IFLG=1, it is required to configure CC0RCFLG=1.
0	UEG	W	Update Event Generate 0: Invalid 1: Initialize the counter and generate an update event This bit is set to 1 by software, and cleared to 0 by hardware. Note: When an update event is generated, the counter of the prescaler will be cleared to 0, but the prescaler factor remains unchanged. In the count-down mode, the counter reads the value of ATMR_AUTORLD; in central alignment mode or count-up mode, the counter will be cleared to 0.

11.6.7 Compare Mode Register 1 (ATMR_CCM1)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	OC1CEN	R/W	Output Compare Channel 1 Clear Enable
14:12	OC1MOD	R/W	Output Compare Channel 1 Mode
11	OC1PEN	R/W	Output Compare Channel 1 Buffer Enable
10:8	Reserved		
7	OC0CEN	R/W	Output Compare Channel 0 Clear Enable 0: OC0REF is unaffected by ETRF input. 1: When high level of ETRF input is detected, OC0REF=0
6:4	OC0MOD	R/W	Output Compare Channel 0 Mode Configure 000: Freeze The output compare has no effect on OC0REF 001: The output value is high when matching. When the value of counter CNT matches the value CCx of capture/compare register, OC0REF will be forced to be high 010: The output value is low when matching. When the value of the counter matches the value of the capture/compare register, OC0REF will be forced to be low 011: Output reverses when matching. When the value of the counter matches the value of the capture/compare register, reverse the level of OC0REF 100: The output is forced to be low. Force OC0REF to be low 101: The output is forced to be high. Force OC0REF to be high 110: PWM mode 1 (set to high when the counter value<output compare value; otherwise, set to low) 111: PWM mode 2 (set to high when the counter value>output compare value; otherwise, set to low) Note: When the protection level is 3 and the channel is configured as output, this bit cannot be modified. In PWM modes 1 and 2, the OC0REF level changes when the comparison result changes or when the output compare mode changes from freeze mode to PWM mode.
3	OC0PEN	R/W	Output Compare Channel 0 Preload Enable

Field	Name	R/W	Description
			0: Disable preloading function; write the value of ATMR_CC0 register through the program and it will work immediately. 1: Enable preloading function; write the value of ATMR_CC0 register through the program and it will work after an update event is generated. Note: When the protection level is 3 and the channel is configured as output, this bit cannot be modified. When the preload register is uncertain, PWM mode can be used only in single-pulse mode (SPMEN=1); otherwise, the following output compare result is uncertain.
2:0	Reserved		

11.6.8 Compare Mode Register 2 (ATMR_CCM2)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	OC3CEN	R/W	Output Compare Channel 3 Clear Enable
14:12	OC3MOD	R/W	Output Compare Channel 3 Mode Configure
11	OC3PEN	R/W	Output Compare Channel 3 Buffer Enable
10:8	Reserved		
7	OC2CEN	R/W	Output Compare Channel 2 Clear Enable 0: OC2REF is unaffected by ETRF input 1: When high level of ETRF input is detected, OC0REF=0
6:4	OC2MOD	R/W	Output Compare Channel 2 Mode Configure
3	OC2PEN	R/W	Output Compare Channel 2 Preload Enable
2:0	Reserved		

11.6.9 Compare Enable Register (ATMR_CCEN)

Offset address: 0x20

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	CC3NPOL	R/W	Compare Channel 3 Complementary Output Polarity Configure Refer to CCEN_CC0NPOL
14	CC3NEN	R/W	Compare Channel 3 Complementary Output Enable Refer to CCEN_CC0NEN
13	CC3POL	R/W	Compare Channel 3 Output Polarity Refer to CCEN_CC0POL
12	CC3EN	R/W	Compare Channel 3 Output Enable Refer to CCEN_CC0EN
11	CC2NPOL	R/W	Compare Channel 2 Complementary Output Polarity Configure Refer to CCEN_CC0NPOL
10	CC2NEN	R/W	Compare Channel 2 Complementary Output Enable

Field	Name	R/W	Description
			Refer to CCEN_CC0NEN
9	CC2POL	R/W	Compare Channel 2 Output Polarity Configure Refer to CCEN_CC0POL
8	CC2EN	R/W	Compare Channel 2 Output Enable Refer to CCEN_CC0EN
7	CC1NPOL	R/W	Compare Channel 1 Complementary Output Polarity Configure Refer to CCEN_CC0NPOL
6	CC1NEN	R/W	Compare Channel 1 Complementary Output Enable Refer to CCEN_CC0NEN
5	CC1POL	R/W	Compare Channel 1 Output Polarity Configure Refer to CCEN_CC0POL
4	CC1EN	R/W	Compare Channel 1 Output Enable Refer to CCEN_CC0EN
3	CC0NPOL	R/W	Compare Channel 0 Complementary Output Polarity 0: OC0N is active high 1: OC0N is active low Note: When the protection level is 2 or 3, this bit cannot be modified.
2	CC0NEN	R/W	Compare Channel 0 Complementary Output Enable 0: Disable 1: Enable
1	CC0POL	R/W	Compare Channel 0 Output Polarity Configure 0: OC0 is active high 1: OC0 is active low Note: When the protection level is 2 or 3, this bit cannot be modified.
0	CC0EN	R/W	Compare Channel 0 Output Enable 0: Disable output 1: Enable output

11.6.10 Counter register (ATMR_CNT)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CNT	R/W	Counter Value

11.6.11 Prescale register (ATMR_PSC)

Offset address: 0x28

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	PSC	R/W	Prescaler Value Clock frequency of counter (CK_CNT)=f_{CK_PSC}/(PSC+1)

11.6.12 Auto reload register (ATMR_AUTORLD)

Offset address: 0x2C

Reset value: 0x0000 FFFF

Field	Name	R/W	Description
31:16	Reserved		
15:0	AUTORLD	R/W	Auto Reload Value When the value of auto reload is empty, the counter will not count.

11.6.13 Repeat count register (ATMR_REPCNT)

Offset address: 0x30

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7:0	REPCNT	R/W	Repetition Counter Value When the count value of the repeat counter is reduced to 0, an update event will be generated, and the counter will start counting again from the REPCNT value; the new value newly written to this register is valid only when an update event occurs in next cycle.

11.6.14 Channel 0 Compare Register (ATMR_CC0)

Offset address: 0x34

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CC0	R/W	Compare Channel 0 Value When the compare channel 0 is configured as output mode CC0 contains the value currently loaded in the compare register Compare the value CC0 of the capture and compare channel 0 with the value CNT of the counter to generate the output signal on OC0. When the output compare preload is disabled (OC0PEN=0 for ATMR_CCM1 register), the written value will immediately affect the output comparison results; If the output compare preload is enabled (OC0PEN=1 for ATMR_CCM1 register), the written value will affect the output comparison result when an update event is generated.

11.6.15 Channel 1 Compare Register (ATMR_CC1)

Offset address: 0x38

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CC1	R/W	Compare Channel 1 Value Refer to ATMR_CC0

11.6.16 Channel 2 Compare Register (ATMR_CC2)

Offset address: 0x3C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CC2	R/W	Compare Channel 2 Value Refer to ATMR_CC0

11.6.17 Channel 3 Compare Register (ATMR_CC3)

Offset address: 0x40

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CC3	R/W	Compare Channel 3 Value Refer to ATMR_CC0

11.6.18 Brake and dead zone register (ATMR_BDT)

Offset address: 0x44

Reset value: 0x0000 0000

Note: According to the lock setting, AOEN, BRKPOL, BRKEN, IMOS, RMOS and DTS0,DTS1 bits all can be write-protected, and it is necessary to configure them when writing to ATMR_BDT register for the first time.

Field	Name	R/W	Description
31:24	Reserved		
23:16	DTS1	R/W	Dead Time Setup DT is the dead zone duration, and the relationship between DT and register DTS1 is as follows: DTS1[23:21]=0xx=>DT=DTS[23:21]×T_{dts}, T_{dts}=T_{DTS}; DTS1[23:21]=10x=>DT=(64+DTS[21:16])×T_{dts}, T_{dts}=2×T_{DTS}; DTS1[23:21]=110=>DT=(32+DTS[20:16])×T_{dts}, T_{dts}=8×T_{DTS}; DTS1[23:21]=111=>DT=(32+DTS[20:16])×T_{dts}, T_{dts}=16×T_{DTS}; For example: assuming T_{DTS}=125ns (8MHz), the dead time setting is as follows: If the step time is 125ns, the dead time can be set from 0 to 15875ns; If the step time is 250ns, the dead time can be set from 16us to 31750ns; If the step time is 1μs, the dead time can be set from 32μs to 63μs; If the step time is 2μs, the dead time can be set from 64μs to 126μs. Note: Once LOCK level (LOCKCFG bit in ATMR_BDT register) is set to 1, 2 or 3, these bits cannot be modified.
15	MOEN	R/W	PWM Main Output Enable 0: Disable the output of OCx and OCxN or force the output of idle state 1: When CCxEN and CCxNEN bits of the ATMR_CCEN register are set, enable OCx and OCxN output When the brake input is valid, it is cleared to 0 by hardware asynchronously. Note: Setting 1 by software or setting 1 automatically depends on AOEN bit of the ATMR_BDT register.

Field	Name	R/W	Description
14	AOEN	R/W	Automatic Output Enable 0: MOEN can only be set to 1 by software 1: MOEN can be set to 1 by software or be automatically set to 1 at the next update event (braking input is invalid) Note: When the protection level is 1, this bit cannot be modified.
13	BRKPOL	R/W	Brake Polarity Configure 0: The brake input BRK is valid at low level 1: The brake input BRK is valid at high level Note: When the protection level is 1, this bit cannot be modified. Writing to this bit requires an APB clock delay before use.
12	BRKEN	R/W	Break Function Enable 0: Disable 1: Enable Note: When the protection level is 1, this bit cannot be modified.
11	RMOS	R/W	Run Mode Off-state Configure Run mode means MOEN=1; disable means CCxEN=0; this bit describes the impact of different values for this bit on the output waveform when MOEN=1 and CCxEN changes from 0 to 1. 0: Disable OCx/OCxN output 1: OCx/OCxN first outptus invalid level (the specific level value is affected by the polarity configuration)
10	IMOS	R/W	Idle Mode Off-state Configure Idle mode means MOEN=0; disable means CCxEN=0; this bit describes the impact of different values for this bit on the output waveform when MOEN=0 and CCxEN changes from 0 to 1. 0: Disable OCx/OCxN output 1: If CCxEN=1, the invalid level is output during the dead time (the specific level value is affected by the polarity configuration), and the idle level is output after the dead time
9:8	LOCKCFG	R/W	Lock Write Protection Mode Configure 00: No Lock write protection; it cannot be written to the register directly 01: Lock write protection level 1 It cannot be written to DTS, BRKEN, BRKPOL and AOEN bits of ATMR_BDT, and OCxOIS and OCxNOIS bits of ATMR_CR2 register. 10: Lock write protection level 2 It cannot be written to all bits of protection level 1, CCxPOL and OCxNPOL bits in ATMR_CCEN register, and RMOS and IMOS bits in ATMR_BDT register. 11: Lock write protection level 3 It cannot be written to all bits of protection level 2, and OCxMOD and OCxPEN bits of ATMR_CCMx register. Note: After system reset, the lock write protect bit can only be written once.
7:0	DTS0	R/W	Dead Time Setup DT is the dead zone duration, and the relationship between DT and register DTS0 is as follows: DTS0[7:5]=0xx=>DT=DTS[7:0]×T_{dts}, T_{dts}=T_{DTS}; DTS0[7:5]=10x=>DT=(64+DTS[5:0])×T_{dts}, T_{dts}=2×T_{DTS};

Field	Name	R/W	Description
			DTS0[7:5]=110=>DT=(32+DTS[4:0])×T_{dts}, T_{dts}=8×T_{DTS}; DTS0[7:5]=111=>DT=(32+DTS[4:0])×T_{dts}, T_{dts}=16×T_{DTS}; For example: assuming T_{DTS}=125ns (8MHZ), the dead time setting is as follows: If the step time is 125ns, the dead time can be set from 0 to 15875ns; If the step time is 250ns, the dead time can be set from 16us to 31750ns; If the step time is 1μs, the dead time can be set from 32μs to 63μs; If the step time is 2μs, the dead time can be set from 64μs to 126μs. Note: Once LOCK level (LOCKCFG bit in ATMR_BDT register) is set to 1, 2 or 3, these bits cannot be modified.

11.6.19 Output control register 1 (ATMR_OCR1)

Offset address: 0x48

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:9	Reserved		
8	BUFEN	R/W	Output Control Buffering is Enable 0: The modification of output control register takes effect immediately 1: The modification of output control register takes effect at the next update event
7	CH3NFORCEEN	R/W	Complementary Channel Output Control Enable Register for Channel 3 0: Complementary channel of channel 3 outputs PWM waveform 1: The output of complementary channel of channel 3 is controlled by the corresponding bit of output control register 3
6	CH3FORCEEN	R/W	Channel 3 Outputs the Control Enable Register 0: Channel 3 outputs PWM waveform 1: The output of channel 3 is controlled by the corresponding bit of output control register 2
5	CH2NFORCEEN	R/W	Complementary Channel Output Control Enable Register for Channel 2 0: Complementary channel of channel 2 outputs PWM waveform 1: The output of complementary channel of channel 2 is controlled by the corresponding bit of output control register 2
4	CH2FORCEEN	R/W	Channel 2 Outputs the Control Enable Register 0: Channel 2 outputs PWM waveform 1: The output of channel 2 is controlled by the corresponding bit of output control register 2.
3	CH1NFORCEEN	R/W	Complementary Channel Output Control Enable Register for Channel 1 0: Complementary channel of channel 1 outputs PWM waveform 1: The output of complementary channel of channel 1 is controlled by the corresponding bit of output control register 2
2	CH1FORCEEN	R/W	Channel 1 Outputs the Control Enable Register 0: Channel 1 outputs PWM waveform

Field	Name	R/W	Description
			1: The output of channel 1 is controlled by the corresponding bit of output control register 2
1	CH0NFORCEEN	R/W	Complementary Channel Output Control Enable Register for Channel 0 0: Complementary channel of channel 0 outputs PWM waveform 1: The output of complementary channel of channel 0 is controlled by the corresponding bit of output control register 2
0	CH0FORCEEN	R/W	Channel 0 Outputs the Control Enable Register 0: Channel 0 outputs PWM waveform 1: The output of channel 0 is controlled by the corresponding bit of output control register 2

Note: Once the LOCK level (the LOCKCFG bit in the ATMR_BDT register) is set to 1, 2, or 3, this register cannot be modified.

11.6.20 Output control register 2 (ATMR_OCR2)

Offset address: 0x4C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	CH3NFORCEVAL	R/W	Complementary Channel Output Level Register for Channel 3 0: Complementary channel of channel 3 outputs low level 1: Complementary channel of channel 3 outputs high level
6	CH3FORCEVAL	R/W	Channel 3 Output Level Register 0: Channel 3 outputs low level 1: Channel 3 outputs high level
5	CH2NFORCEVAL	R/W	Complementary Channel Output Level Register for Channel 2 0: Complementary channel of channel 2 outputs low level 1: Complementary channel of channel 2 outputs high level
4	CH2FORCEVAL	R/W	Channel 2 Output Level Register 0: Channel 2 outputs low level 1: Channel 2 outputs high level
3	CH1NFORCEVAL	R/W	Complementary Channel Output Level Register for Channel 1 0: Complementary channel of channel 1 outputs low level 1: Complementary channel of channel 1 outputs high level
2	CH1FORCEVAL	R/W	Channel 1 Output Level Register 0: Channel 1 outputs low level 1: Channel 1 outputs high level
1	CH0NFORCEVAL	R/W	Complementary Channel Output Level Register for Channel 0 0: Complementary channel of channel 0 outputs low level 1: Complementary channel of channel 0 outputs high level
0	CH0FORCEVAL	R/W	Channel 0 Output Level Register 0: Channel 0 outputs low level 1: Channel 0 outputs high level

Note: ATMR_OCR2 must be configured first before ATMR_OCR1. If it is necessary to change the output control buffer function and modify the values of

other control bits in ATMR_OCRx: First, write the BUFEN of ATMR_OCR1, and then write the ATMR_OCR2 register and the ATMR_OCR1 register in sequence.

11.6.21 TRGO Control Register (ATMR_TRGOCR)

Offset address: 0x50

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:12	Reserved		
11	MMS2PE	R/W	TRGO2 Signal is Generated when the Counter Matches the Autoreload Register The generated TRGO2 is only valid when the counter is in center alignment mode and is only selected when MMS2 is set to 1000 0: TRGO2 is not generated when the counter overflows 1: TRGO2 is generated when the counter overflows
10	MMS2ZE	R/W	When the counter returns to 0, TRGO2 signal is generated The generated TRGO2 is only valid when the counter is in center alignment mode and is only selected when MMS2 is configured to 1000. 0: No TRGO2 is generated when the counter overflows 1: TRGO2 is generated when the counter overflows
9	MMS1PE	R/W	TRGO1 Signal is Generated when the Counter Matches the Autoreload Register The generated TRGO1 is only valid when the counter is in center alignment mode and is only selected when MMS1 is set to 1000 0: TRGO1 is not generated when the counter overflows 1: TRGO1 is generated when the counter overflows
8	MMS1ZE	R/W	When the counter returns to 0, TRGO 1 signal is generated The generated TRGO1 is only valid when the counter is in center alignment mode and is only selected when MMS2 is configured to 1000. 0: No TRGO1 is generated when the counter overflows 1: TRGO1 is generated when the counter overflows
7:4	MMS2	R/W	Master mode 2 selection Used to select the synchronization information (TRGO2) sent to the slave timer in master mode. Possible combinations are as follows: 0000: Reset – The UEG bit of the ATMR_CEG register is used as the trigger output (TRGO2). If the reset is generated by the trigger input (the slave mode controller is in reset mode), the signal on TRGO2 will have a delay relative to the actual reset. 0001: Enable – The counter enable signal CNT_EN is used as the trigger output (TRGO2). Sometimes it is necessary to start multiple timers at the same time or control the enabling of slave timers within a period of time. The counter enable signal is generated by the logical OR of the CEN control bit and the trigger input signal in gated mode. When the counter enable signal is controlled by the trigger input, there will be a delay on TRGO unless the master/slave mode is selected (see the description of the MSMEN bit in the ATMR_SMCR register). 0010: Update – The update event is selected as the trigger input (TRGO2). For example, the clock of a master timer can be used as the prescaler of a slave timer.

Field	Name	R/W	Description
			011: Compare pulse – When a capture or a comparison is successful, a positive pulse is sent to the trigger output (TRGO2) when the CC0IFLG flag is to be set (even if it is already high). 0100: Compare – The OC0REF signal is used as the trigger output (TRGO2) 0101: Compare – The OC1REF signal is used as the trigger output (TRGO2) 0110: Compare – The OC2REF signal is used as the trigger output (TRGO2) 0111: Compare – The OC3REF signal is used as the trigger output (TRGO2) 1000: It does not produce TRGO2 (it only generates TRGO2 based on MMS2ZE and MMS2PE) 1001: Compare - The OC3NREF (Independent Output Mode) signal is used as the trigger output (TRGO2) 1010: OC3REF rising edge and falling edge generate TRGO2 1011: OC3NREF (Independent Output Mode) rising edge and falling edge generate TRGO2 1100: OC3REF rising edge and OC3NREF rising edge generate TRGO2 1101: OC3REF falling edge and OC3NREF falling edge generate TRGO2 1110: OC3REF rising edge and OC3NREF falling edge generate TRGO2 1111: OC3REF falling edge and OC3NREF rising edge generate TRGO2
3:0	MMS1	R/W	Master mode 1 selection Used to select the synchronization information (TRGO1) sent to the slave timer in master mode. Possible combinations are as follows: 0000: Reset – The UEG bit of the ATMR_CEG register is used as the trigger output (TRGO1). If the reset is generated by the trigger input (the slave mode controller is in reset mode), the signal on TRGO1 will have a delay relative to the actual reset. 0001: Enable – The counter enable signal CNT_EN is used as the trigger output (TRGO1). Sometimes it is necessary to start multiple timers at the same time or control the enabling of slave timers within a period of time. The counter enable signal is generated by the logical OR of the CEN control bit and the trigger input signal in gated mode. When the counter enable signal is controlled by the trigger input, there will be a delay on TRGO1 unless the master/slave mode is selected (see the description of the MSMEN bit in the ATMR_SMCR register). 0010: Update – The update event is selected as the trigger input (TRGO1). For example, the clock of a master timer can be used as the prescaler of a slave timer. 011: Compare pulse – When a capture or a comparison is successful, a positive pulse is sent to the trigger output (TRGO1) when the CC0IFLG flag is to be set (even if it is already high). 0100: Compare – The OC0REF signal is used as the trigger output (TRGO1) 0101: Compare – The OC1REF signal is used as the trigger output (TRGO1) 0110: Compare – The OC2REF signal is used as the trigger output (TRGO1)

Field	Name	R/W	Description
			0111: Compare – The OC3REF signal is used as the trigger output (TRGO1) 1000: It does not produce TRGO1 (it only generates TRGO1 based on MMS1ZE and MMS1PE). 1001: Compare - The OC3NREF (Independent Output Mode) signal is used as the trigger output (TRGO1). 1010: OC3REF rising and falling edges generate TRGO1 1011: OC3NREF rising and falling edges generate TRGO1 1100: OC3REF rising edge and OC3NREF rising edge generate TRGO1 1101: OC3REF falling edge and OC3NREF falling edge generate TRGO1 1110: OC3REF rising edge and OC3NREF falling edge generate TRGO1 1111: OC3REF falling edge and OC3NREF rising edge generate TRGO1

11.6.22 Break Filter Register (ATMR_BREAK)

Offset address: 0x54

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7:2	FILT	R/W	Filter coefficient Based on the APB clock. FILT*APB_CLK
1	FILTEN	R/W	Break_in input filtering function enable 0: Disable 1: Enable
0	ANAFILTEN	R/W	Analog Break_in input filtering function enable 0: Disable 1: Enable

11.6.23 Lower Compare Register Control Register (ATMR_OCxACR)

Offset address: 0x58

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	NONC3EN	R/W	Channel 3 complementary independent select 0: Complementary function 1: Independent function Note: Once the LOCK level (the LOCKCFG bit in the ATMR_BDT register) is set to 3, these bits cannot be modified.
6	NONC2EN	R/W	Channel 2 complementary independent select 0: Complementary function 1: Independent function Note: Once the LOCK level (the LOCKCFG bit in the ATMR_BDT register) is set to 3, these bits cannot be modified.
5	NONC1EN	R/W	Channel 1 complementary independent select 0: Complementary function

Field	Name	R/W	Description
			1: Independent function Note: Once the LOCK level (the LOCKCFG bit in the ATMR_BDT register) is set to 3, these bits cannot be modified.
4	NONC0EN	R/W	Channel 0 complementary independent select 0: Complementary function 1: Independent function Note: Once the LOCK level (the LOCKCFG bit in the ATMR_BDT register) is set to 3, these bits cannot be modified.
3	OC3AEN	R/W	Channel 3 Asymmetric PWM Output Mode Enable Refer to OC0AEN.
2	OC2AEN	R/W	Channel 2 Asymmetric PWM Output Mode Enable Refer to OC0AEN.
1	OC1AEN	R/W	Channel 1 Asymmetric PWM Output Mode Enable Refer to OC0AEN.
0	OC0AEN	R/W	Channel 0 Asymmetric PWM Output Mode Enable Valid only when the counter is in the center symmetry mode and channel 0 is configured as PWM output mode. In the asymmetric PWM mode, when the counter counts up, OC0REF is controlled by CC0, when the counter counts down, OC0REF is controlled by CC0. 1: Enable asymmetric PWM output mode 0: CC0C does not affect OC0REF output Note: Once LOCK level (LOCKCFG bit in ATMR_BDT register) is set to 3, these bits cannot be modified.

Note: When OCxAEN and NONCxEN are enabled simultaneously, CCRxC does not affect the output of OCxREF.

11.6.24 Channel 0 Lower Compare Register (ATMR_CC0C)

Offset address: 0x5C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CCR0C	R/W	Channel 0 Capture/Compare Register Complementary Register When the capture/compare channel 0 is configured as PWM output mode, the counter is in the center alignment mode, CCR0C contains the value currently loaded in the compare register complementary register. When the counter is a count-up counter, compare the value CC0 of the capture/compare channel 0 with the value CNT of the counter, and the output signal is generated on OC0. When the counter counts down, compare the value of CCR0C with the value CNT of the counter to generate the output signal on OC0. When the output compare preload is disabled (OC0PEN=0 in the ATMR_CCM1 register), the written value will immediately affect the output compare result; when the output compare preload is enabled (OC0PEN=1 in the ATMR_CCM1 register), the written value will affect the output compare result when an update event occurs.

11.6.25 Channel 1 Lower Compare Register (ATMR_CC1C)

Offset address: 0x60

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CCR1C	R/W	Channel 1 Compare Register Complementary Register Refer to ATMR_CC0C.

11.6.26 Channel 2 Lower Compare Register (ATMR_CC2C)

Offset address: 0x64

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CCR2C	R/W	Channel 2 Compare Register Complementary Register Refer to ATMR_CC0C.

11.6.27 Channel 3 Lower Compare Register (ATMR_CC3C)

Offset address: 0x68

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CCR3C	R/W	Channel 3 Compare Register Complementary Register Refer to ATMR_CC0C.

12 General Timer (GTIMER)

12.1 Introduction

The general-purpose timer takes the time base unit as the core, and has the functions of input capture and output compare, and can be used to measure the pulse width, frequency and duty cycle, and generate the output waveform. It includes a 32-bit auto reload counter (realize count-up, count-down and central alignment count).

The timers are independent of each other, and they can achieve synchronization and cascading.

12.2 Main characteristics

(1) Timebase unit
- Counter: 32-bit counter, supporting count-up, count-down and central alignment count
- Prescaler: 16-bit programmable prescaler
- Autoreload function

(2) Clock source selection
- Internal clock
- External trigger
- Internal trigger

(3) Input capture function

(4) Counting function

(5) PWM input mode

(6) Encoder interface mode

(7) Output compare function

(8) PWM output mode

(9) Forced output mode

(10) Single-pulse mode

(11) Timing function

(12) Master/Slave mode controller of timer
- Timers can be synchronized and cascaded
- Support multiple slave modes and synchronization signals

(13) Interrupt output and DMA request event
- Update event (counter overrun/underrun, counter initialization)

- Trigger event (counter start, stop, internal/external trigger)
- Capture/Compare event

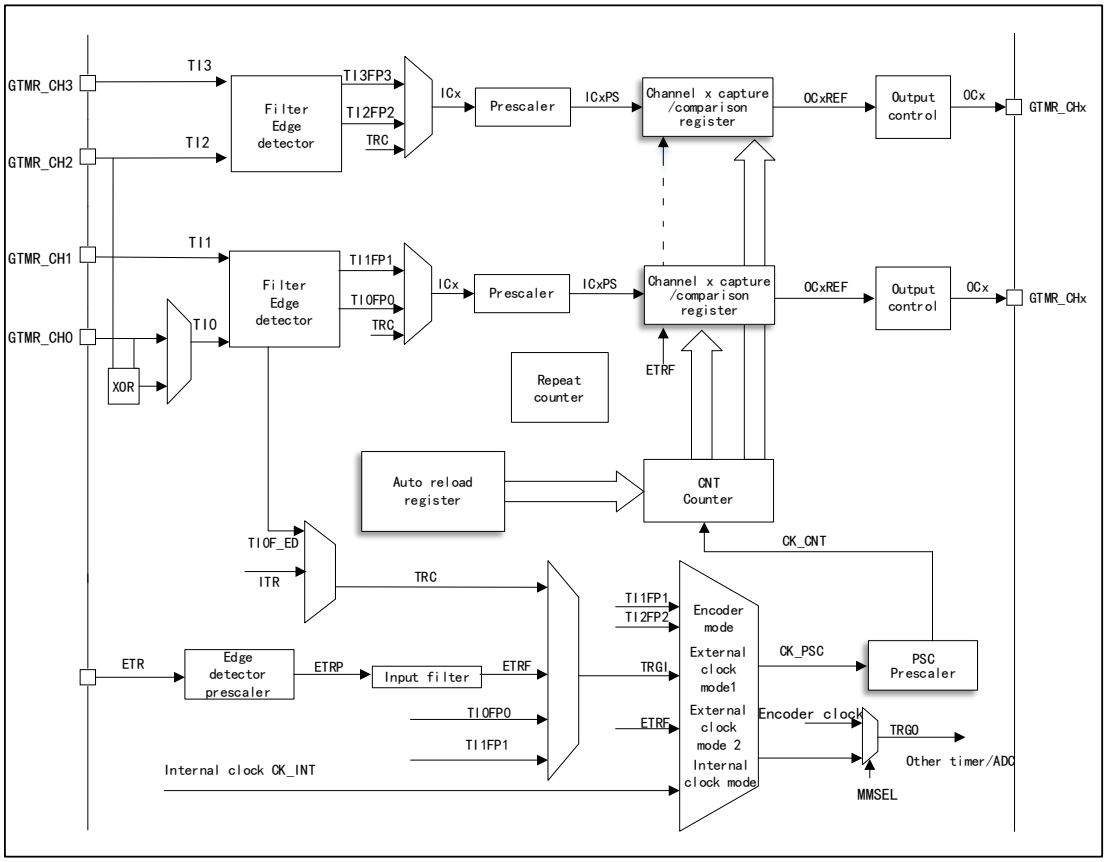
(14) The timer has an independent DMA request mechanism for generation

(15) Support incremental (quadrature) encoder and Hall sensor circuits for positioning

(16) Supports ETR input (external trigger input) function, which can be used as external clock or cycle-by-cycle current management

12.3 Structure block diagram

Figure 39 Structure Block Diagram



12.4 Functional description

12.4.1 Clock source selection

The general-purpose timer has three clock sources.

Internal clock

It is GTMR_CLK from RCC, namely the driving clock of the timer; when the slave mode controller is disabled, the clock source CK_PSC of the prescaler is

driven by the internal clock CK_INT.

External clock mode

After polarity selection, frequency division and filtering, the signal from external trigger interface (ETR) is connected to the slave mode controller through trigger input selector to control the work of the counter.

Internal trigger input

The timer is set to work in slave mode, and the clock source is the output signal of other timers. At this time, the clock source has no filtering, and the synchronization or cascading between timers can be realized. The master mode timer can reset, start, stop or provide clock for the slave mode timer.

12.4.2 Timebase unit

The time base unit in the general-purpose timer contains three registers

- Counter register (CNT) 32 bits
- Autoreload register (AUTORLD) 32 bits
- Prescaler (PSC) 16 bits

Counter CNT

There are three counting modes for the counter in the general-purpose timer

- Count-up mode
- Count-down mode
- Central alignment mode

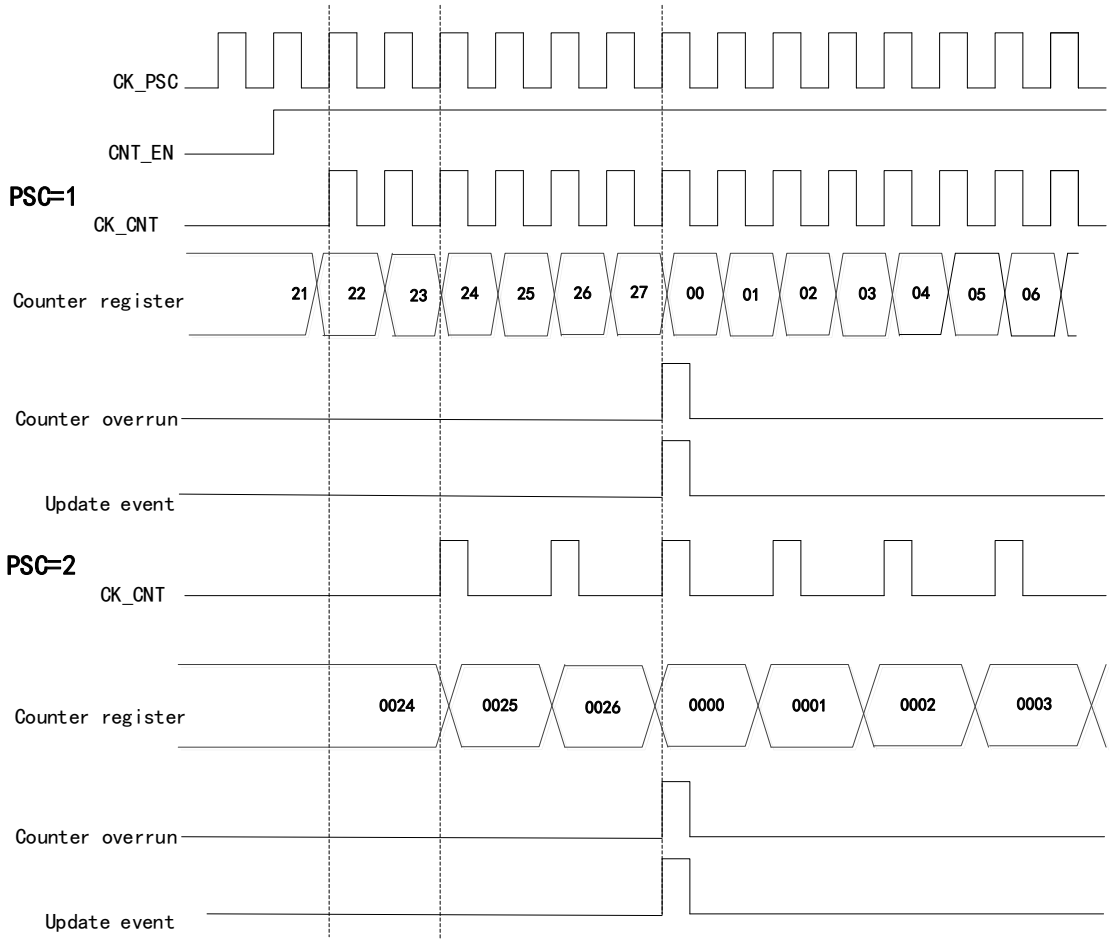
Count-up mode

Set to the count-up mode by configuring CNTDIR bit of control register (GTMR_CR1).

When the counter is in count-up mode, the counter will count up from 0; every time a pulse is generated, the counter will increase by 1 and when the value of the counter (GTMR_CNT) is equal to the value of the auto reload (GTMR_AUTORLD), the counter will start to count from 0 again, a count-up overrun event will be generated, and the value of the auto reload (GTMR_AUTORLD) is written in advance. Otherwise, an update event will be generated every time the counter underruns. At this time, the auto reload shadow register and the prescaler buffer will be updated. The update event can be disabled by configuring UDISEN bit of control register GTMR_CR1.

The figure below is the timing diagram of count-up mode when the division factor is 1 or 2.

Figure 40 Timing Diagram of Count-up Mode when Division Factor is 1 or 2



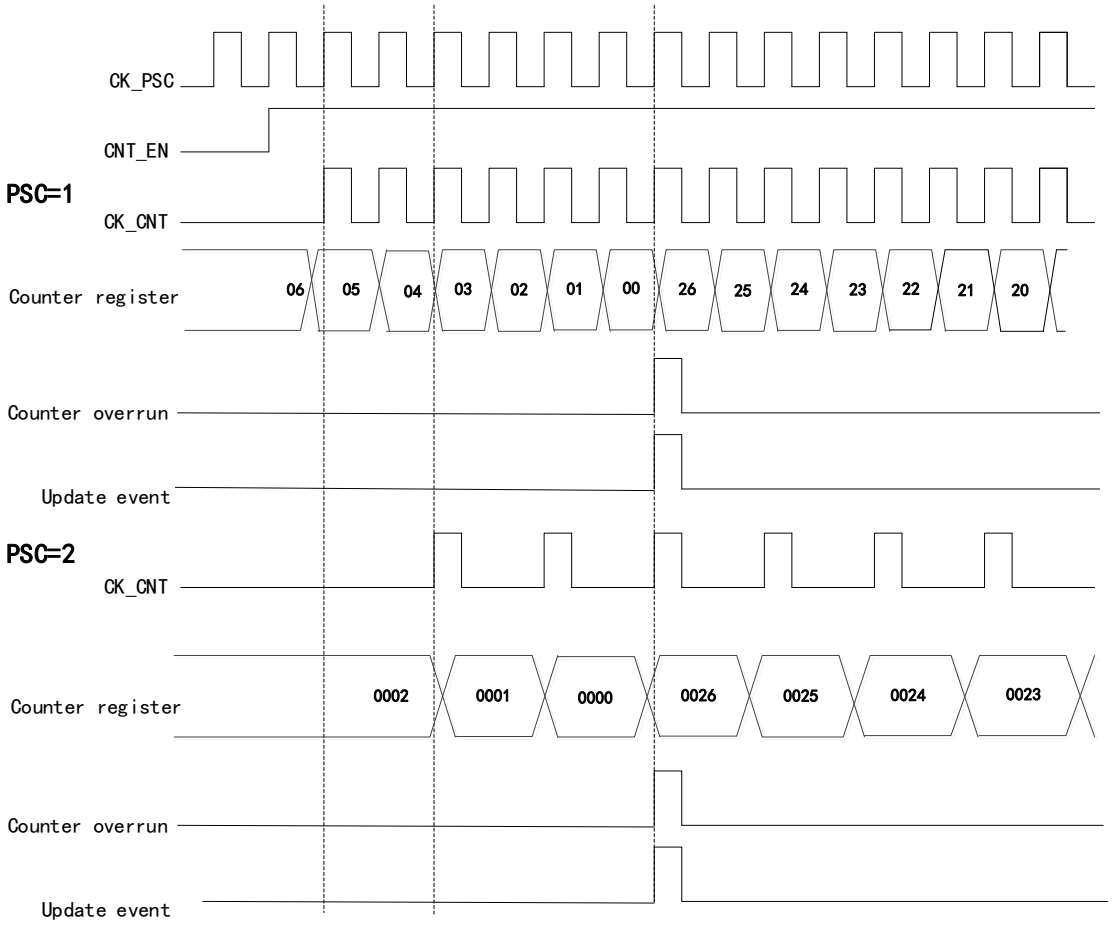
Count-down mode

Set to the count-down mode by configuring CNTDIR bit of control register (GTMR_CR1).

When the counter is in count-down mode, it will start to count down from the value of the auto reload (GTMR_AUTORLD); every time a pulse is generated, the counter will decrease by 1 and when it becomes 0, the counter will start to count again from (GTMR_AUTORLD), meanwhile, a count-down overrun event will be generated, and the value of the auto reload (GTMR_AUTORLD) is written in advance. Otherwise, an update event will be generated every time the counter underruns. At this time, the auto reload shadow register and the prescaler buffer will be updated. The update event can be disabled by configuring the UDISEN bit of the GTMR_CR1 register.

The figure below is the timing diagram of count-down mode when the division factor is 1 or 2.

Figure 41 Timing Diagram of Count-down Mode when Division Factor is 1 or 2



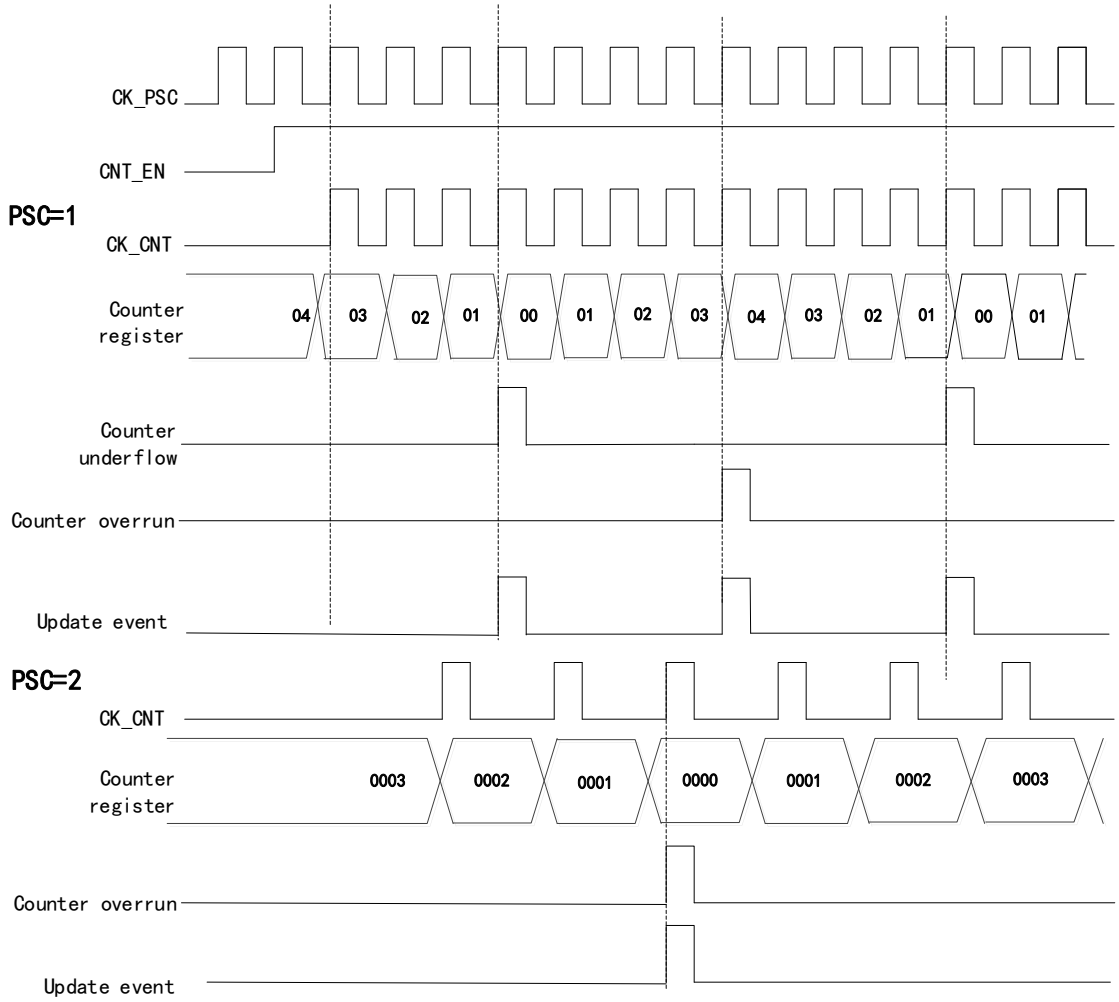
Central alignment mode

Set to the central alignment mode by configuring CAMSEL bit of control register (GTMR_CR1).

When the counter is in center alignment mode, the counter counts up from 0 to the value of auto reload (GTMR_AUTORLD), then counts down to 0 from the value of the auto reload (GTMR_AUTORLD), which will repeat; in counting up, when the counter value is (AUTORLD-1), a counter overrun event will be generated; in counting down, when the counter value is 1, a counter underrun event will be generated.

The figure below is the timing diagram of central alignment mode when the division factor is 1 or 2.

Figure 42 Timing Diagram of Central alignment Mode when Division Factor is 1 or 2



Prescaler PSC

The prescaler is 16 bits and programmable, and it can divide the clock frequency of the counter to any value within 1~65536 (controlled by GTMR_PSC register), and after frequency division, the clock will drive the counter CNT to count. The prescaler has a buffer, which can be changed during running.

12.4.3 Input capture

Input capture channel

The general-purpose timer has four independent capture/compare channels, each of which is surrounded by a capture/compare register.

In the input capture, the measured signal will enter from the external pin TI0/1/2/3 of the timer, first pass through the edge detector and input filter, and then enter the capture channels. Each capture channel has a corresponding capture register. When the capture occurs, the value of the counter CNT will be

latched in the capture register CCx. Before entering the capture register, the signal will pass through the prescaler to set how many events to capture at a time.

Input capture application

Input capture is used to capture external events, and can give the time flag to indicate the occurrence time of the event and measure the pulse jump edge events (measure the frequency or pulse width), for example, if the selected edge appears on the input pin, the GTMR_CCx register will capture the current value of the counter and the CCxIFLG bit of the status register GTMR_SR will be set to 1; if CCxIEN=1, an interrupt will be generated.

In capture mode, the timing, frequency, cycle and duty cycle of a waveform can be measured. In the input capture mode, the edge selection is set to rising edge detection. When the rising edge appears on the capture channel, the first capture occurs, at this time, the value of the counter CNT will be latched in the capture register CCx; at the same time, it will enter the capture interrupt, a capture will be recorded in the interrupt service program and the value will be recorded. When the next rising edge is detected, the second capture occurs, the value of counter CNT will be latched in capture register CCx again, at this time, it will enter the capture interrupt again; read the value of capture register and the cycle of this pulse signal will be obtained by capture.

12.4.4 Output compare

There are eight modes of output compare: freeze, channel x is valid when matching, channel x is invalid when matching, reverse, force to invalid, force to valid, PWM mode 1 and PWM mode 2, which are configured by OCxMOD bit in GTMR_CCMx register and can control the waveform of output signal in output compare mode.

Output compare application

In the output compare mode, the position, polarity, frequency and time of the pulse generated by the timer can be controlled.

When the value of the counter is equal to that of the capture/compare register, the channel output can be set as high level, low level or reverse by configuring the OCxMOD bit in GTMR_CCMx register and the CCxPOL bit in the output polarity GTMR_CCEN register.

When CCxIFLG=1 in GTMR_SR register, if CCxIEN=1 in GTMR_DIER register, an interrupt will be generated; if CCDSEL=1 in GTMR_CR2 register, a DMA request will be generated.

12.4.5 PWM output mode

PWM mode is pulse signal that can be adjusted by external output of the timer.

The pulse width of the signal is determined by the value of the compare register CCx, and the cycle is determined by the value of the auto reload AUTORLD.

PWM output mode contains PWM mode 1 and PWM mode 2; PWM mode 1 and PWM mode 2 are divided into count-up, count-down and edge alignment counting; in PWM mode 1, if the value of the counter CNT is less than the value of the compare register CCx, the output level will be valid; otherwise, it will be invalid.

Set the timing diagram in PWM mode 1 when CCx=5, AUTORLD=7.

Figure 43 Timing Diagram of PWM1 Count-up Mode

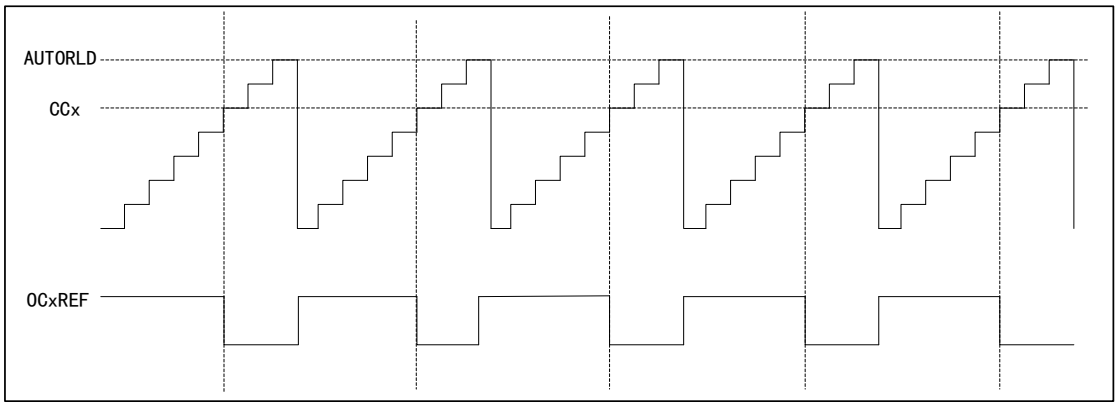


Figure 44 Timing Diagram of PWM1 Count-down Mode

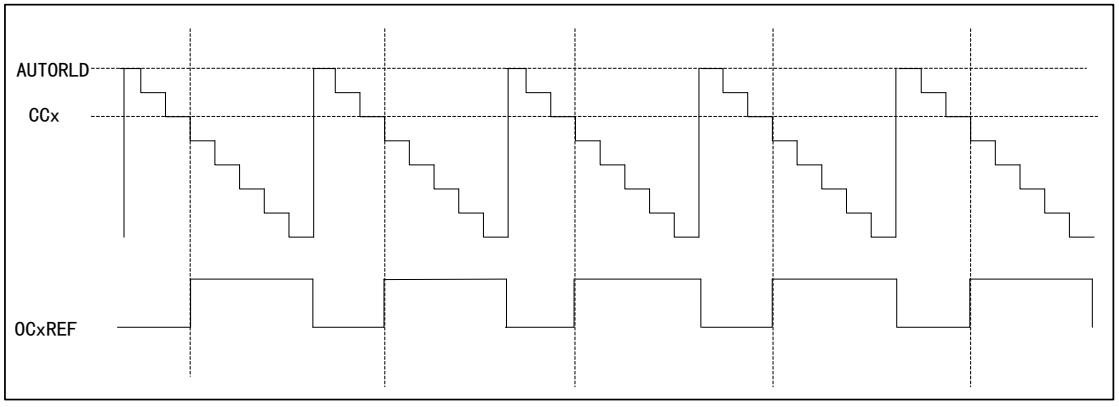
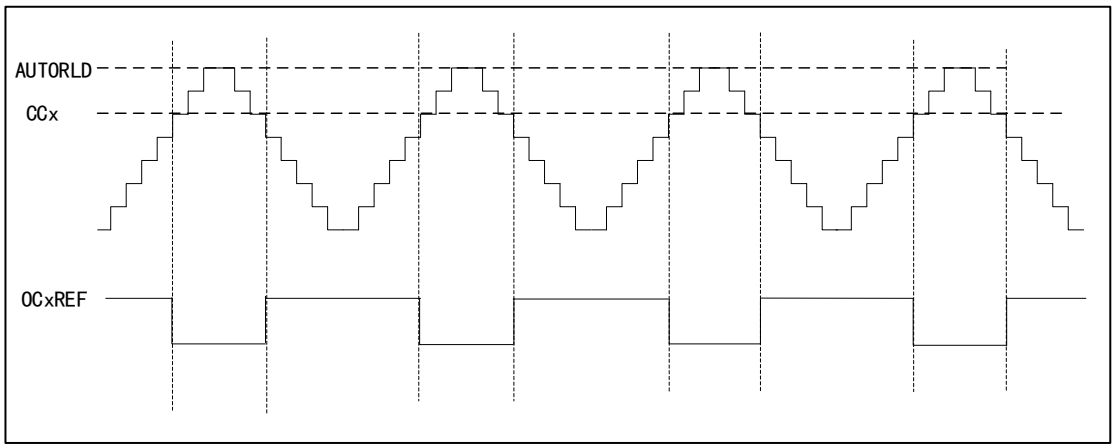


Figure 45 Timing Diagram of PWM1 Central alignment Mode



In PWM mode 2, if the value of the counter CNT is less than that of the compare register CCx, the output level will be invalid; otherwise, it will be valid.

Set the timing diagram in PWM mode 2 when CCx=5, AUTORLD=7.

Figure 46 Timing Diagram of PWM2 Count-up Mode

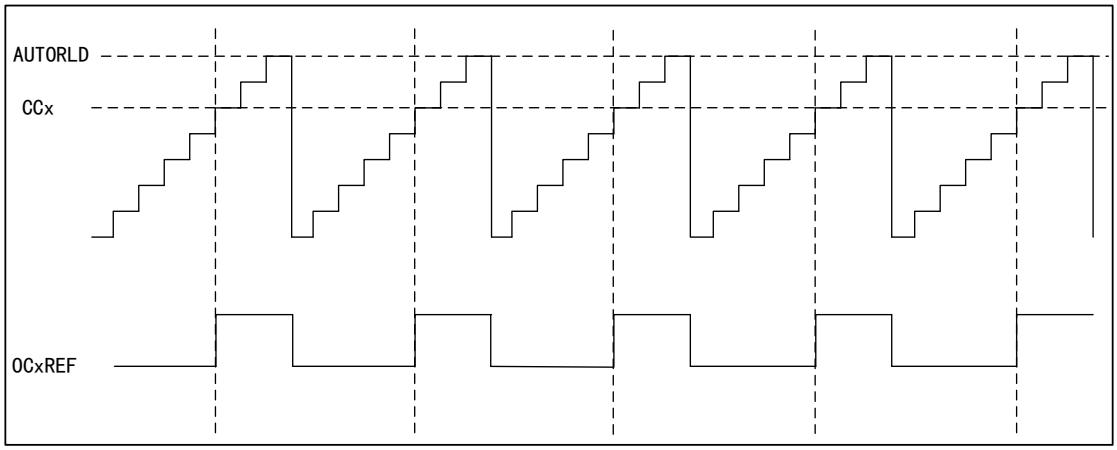


Figure 47 Timing Diagram of PWM2 Count-down Mode

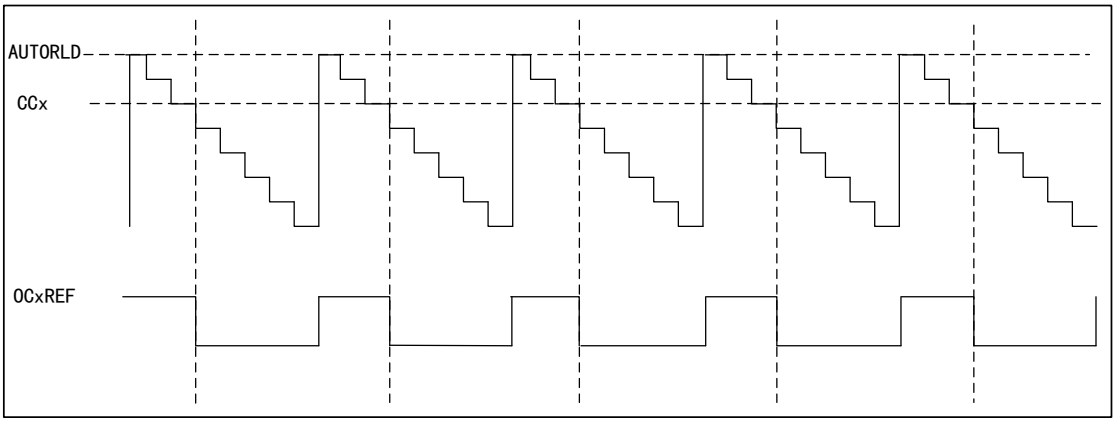
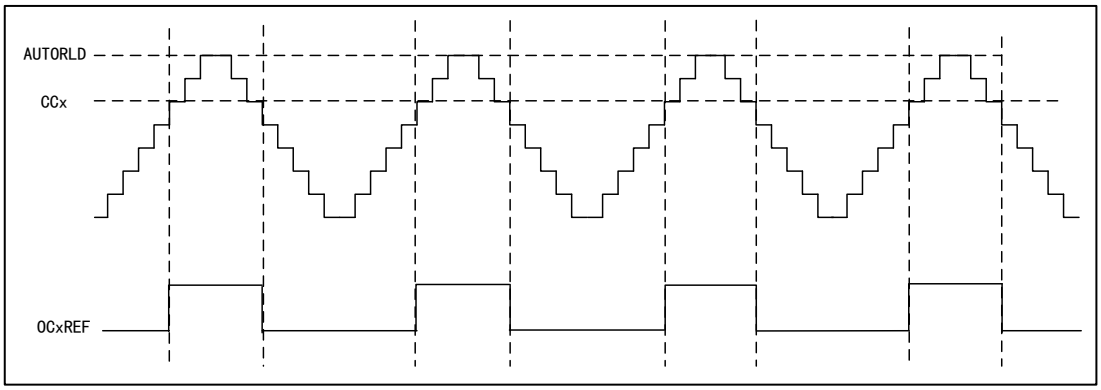


Figure 48 Timing Diagram of PWM2 Central alignment Mode



12.4.6 PWM input mode

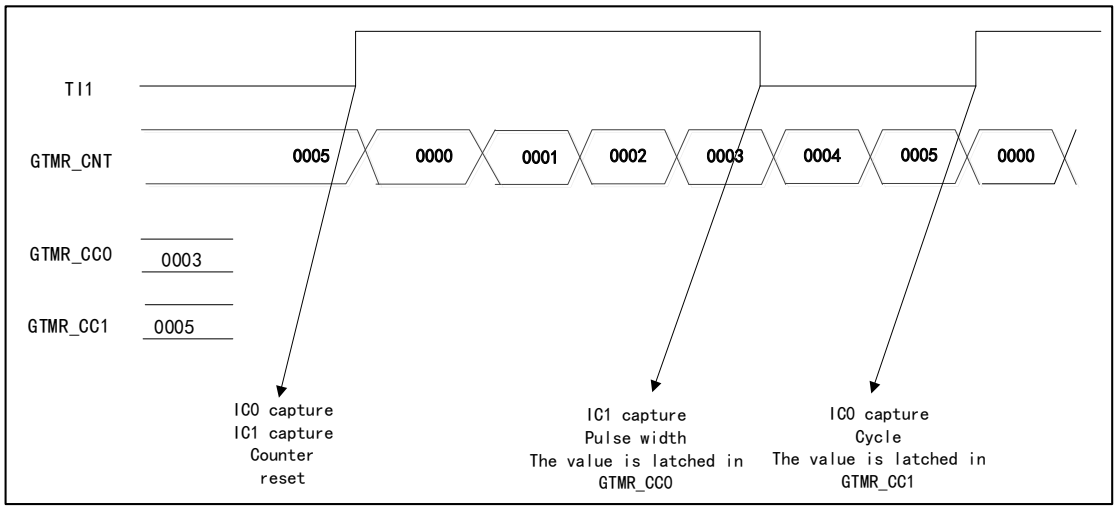
PWM input mode is a particular case of input capture.

In PWM input mode, as only TI0FP0 and TI1FP1 are connected to the slave mode controller, input can be performed only through the channels GTMR_CH0 and GTMR_CH1, which need to occupy the capture registers of CH0 and CH1.

In the PWM input mode, the PWM signal enters from GTMR_CH0, and the signal will be divided into two channels, one can measure the cycle and the other can measure the duty cycle. In the configuration, it is only required to set the polarity of one channel, and the other will be automatically configured with the opposite polarity.

In this mode, the slave mode controller should be configured as the reset mode (SMFSEL bit of GTMR_SMCR register).

Figure 49 Timing Diagram in PWM Input Mode



12.4.7 Single-pulse mode

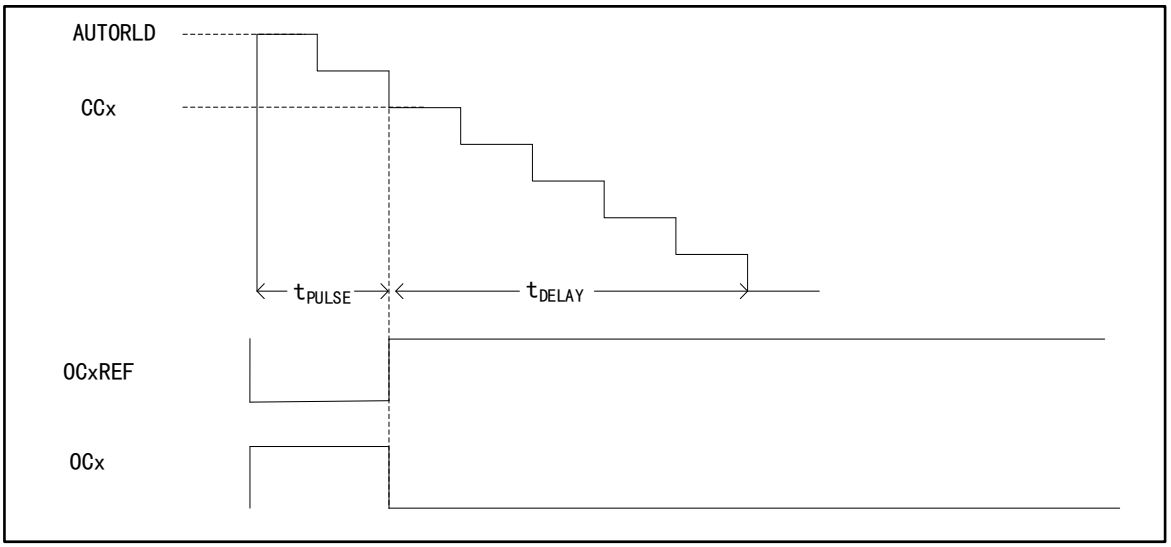
The single-pulse mode is a special case of timer compare output, and is also a

special case of PWM output mode.

Set SPMEN bit of GTMR_CR1 register, and select the single-pulse mode. After the counter is started, a certain number of pulses will be output before the update event occurs. When an update event occurs, the counter will stop counting, and the subsequent PWM waveform output will no longer be changed.

After a certain controllable delay, a pulse with controllable pulse width is generated in single-pulse mode through the program. The delay time is defined by the value of GTMR_CCx register; in the count-up mode, the delay time is CCx and the pulse width is AUTORLD-CCx; in the count-down mode, the delay time is AUTORLD-CCx and the pulse width is CCx.

Figure 50 Timing Diagram of Single-pulse Mode



12.4.8 Forced output mode

In the forced output mode, the comparison result is ignored, and the corresponding level is directly output according to the configuration instruction.

- CCxSEL=00 for GTMR_CCMx register, set CCx channel as output
- OCxMOD=100/101 for GTMR_CCMx register, set to force OCxREF signal to invalid/valid

In this mode, the corresponding interrupt and DMA request will still be generated.

12.4.9 Encoder interface mode

The encoder interface mode is equivalent to an external clock with direction selection. In the encoder interface mode, the content of the timer can always indicate the position of the encoder.

The method of selecting encoder interface is as follows:

- By setting SMFSEL bit of GTMR_SMCR register, set the counter to count on the edge of TI0 channel /TI1 channel, or count on the edge of TI0 and TI1 at the same time.
- Select the polarity of TI0 and TI1 by setting the CC0POL and CC1POL bits of GTMR_CCEN register.
- Select to filter or not by setting the IC0F and IC1F bits of GTMR_CCM1 register.

The two input TI0 and TI1 can be used as the interface of incremental encoder. The counter is driven by the effective jump of the signals TI0FP0 and TI1FP1 after filtering and edge selection in TI0 and TI1.

The count pulse and direction signal are generated according to the input signals of TI0 and TI1

- The counter will count up/down according to the jumping sequence of the input signal
- Set CNTDIR of control register GTMR_CR1 to be read-only (CNTDIR will be re-calculated due to jumping of any input end)

The change mechanism of counter count direction is shown in the figure below

Table 38 Relationship between Count Direction and Encoder

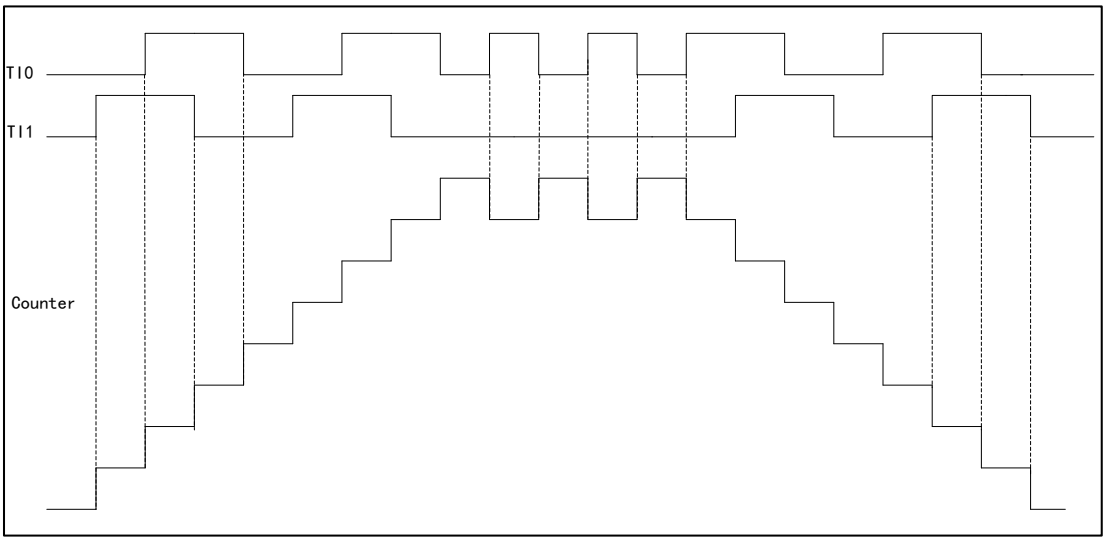
Effective edge		**Count only in TI0**		**Count only in TI1**		**Count in both TI0 and TI1**	
Level of relative signal		High	Low	High	Low	High	Low
TI0FP0	Rising Edge	Count down	Count up	—		Count down	Count up
	Falling Edge	Count up	Count down			Count up	Count down
TI1FP1	Rising Edge	—		Count up	Count down	Count up	Count down
	Falling Edge			Count down	Count up	Count down	Count up

The external incremental encoder can be directly connected with MCU, not needing external interface logic, so the comparator is used to convert the differential output of the encoder to digital signal to increase the immunity to noise interference.

Among the following examples,

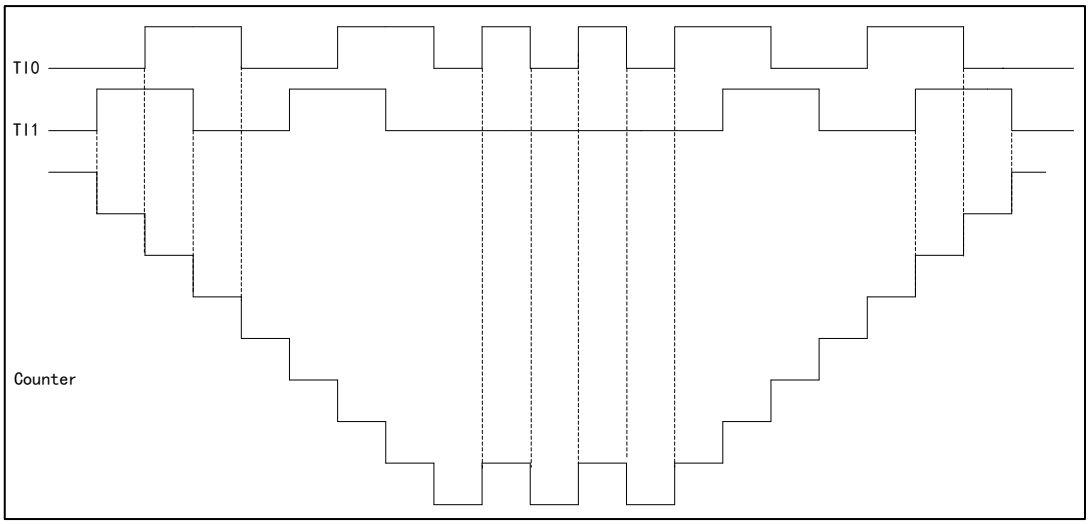
- IC0FP0 is mapped to TI0
- IC1FP1 is mapped to TI1
- Neither IC0FP0 nor IC1FP1 is phase-inverting
- The input signal is valid at the rising edge and falling edge
- Enable the counter

Figure 51 Counter Operation Example in Encoder Mode



For example, when TI0 is at low level, and TI1 is in rising edge state, the counter will count up.

Figure 52 Example of Encoder Interface Mode of IC0FP0 Reversed Phase



For example, when TI0 is at low level, and the rising edge of TI1 jumps, the counter will count down.

12.4.10 Slave mode

GTMR timer can synchronize external trigger

- Reset mode
- Gated mode
- Trigger mode

SMFSEL bit in GTMR_SMCR register can be set to select the mode

SMFSEL=100 set the reset mode, SMFSEL=101 set the gated mode, and SMFSEL=110 set the trigger mode.

In the reset mode, when a trigger input event occurs, the counter and prescaler will be initialized, and the rising edge of the selected trigger input (TRGI) will reinitialize the counter and generate a signal to update the register.

In the gated mode, the enable of the counter depends on the high level of the selected input end. When the trigger input is high, the clock of the counter will be enabled. Once the trigger input becomes low, the counter will stop (but not be reset). The start and stop of the counter are controlled.

In the trigger mode, the enable of the counter depends on the event on the selected input, the counter will be enabled at the rising edge of the trigger input (but not be reset), and only the start of the counter is controlled.

12.4.11 Timer interconnection

Each timer of GTIMER can be connected with each other to realize synchronization or cascading between timers. It is required to configure one timer in master mode and the other timer in slave mode.

When the timer is in master mode, it can reset, start, stop and provide clock source for the counter of the slave mode timer.

When the timers are interconnected:

- A timer can be used as the prescaler of other register
- Start the other register by the enable signal of a timer
- Start the other register by the update event of a timer
- Select the other register by the enable of a timer
- Two timers can be synchronized by an external trigger

12.4.12 Interrupt and DMA request

The timer can generate an interrupt when an event occurs during operation

- Update event (counter overrun/underrun, counter initialization)
- Trigger event (counter start, stop, internal/external trigger)
- Capture/Compare event
- Braking signal input event

Some internal interrupt events can generate DMA requests, and special interfaces can enable or disable trigger DMA requests.

12.4.13 Clear OCxREF signal when an external event occurs

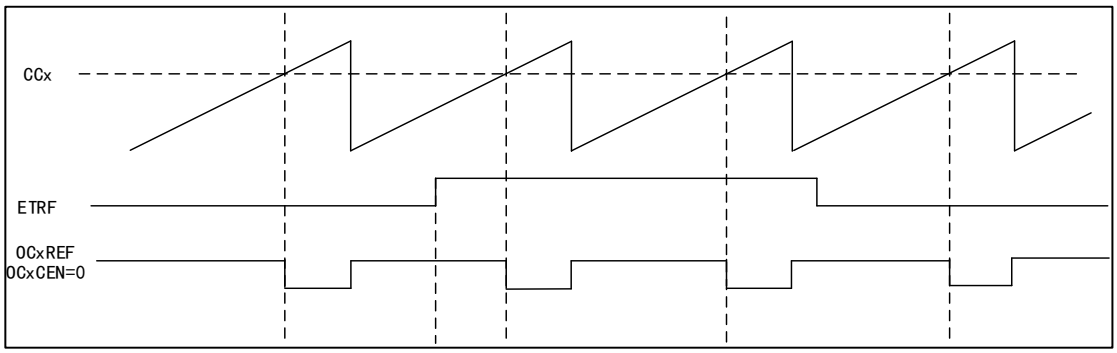
This function is used for output compare and PWM mode.

In one channel, the high level of ETRF input port will reduce the signal of OCxREF to low level, and the OCxCEN bit in capture/compare register GTMR_CCMx is set to 1, and OCxREF signal will remain low until the next update event occurs.

Set GTMR to PWM mode, disable the external trigger prescaler, and disable the

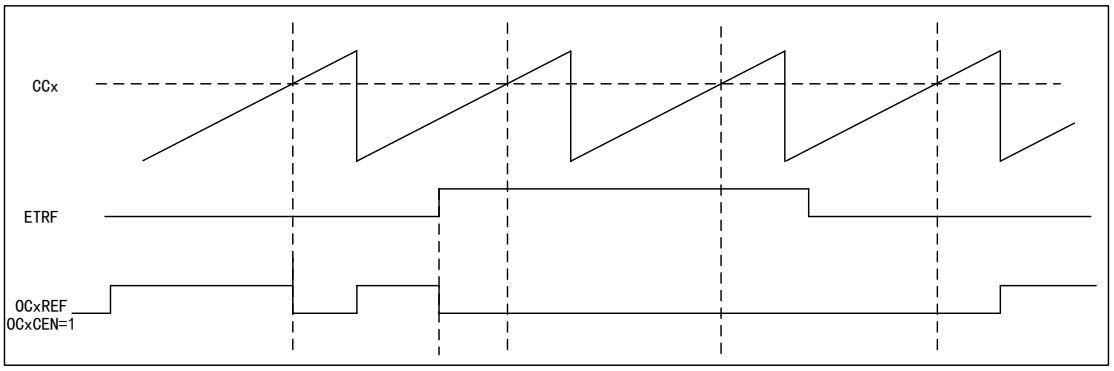
external trigger mode 2; when ETRF input is high, set OCxCEN=0, and the output OCxREF signal is shown in the figure below.

Figure 53 OCxREF Timing Diagram



Set GTMR to PWM mode, disable the external trigger prescaler, and disable the external trigger mode 2; when ETRF input is high, set OCxCEN=1, and the output OCxREF signal is shown in the figure below.

Figure 54 OCxREF Timing Diagram



12.4.14 Timer internal trigger connection

Table 39 Timer Internal Trigger Connection

Slave timer	ITR0
ATMR	GTMR
GTMR	ATMR

12.5 Register address mapping

Table 40 GTIMER Register Address Mapping

Register name	Description	Offset address
GTMR_CR1	Control register 1	0x00
GTMR_CR2	Control register 2	0x04
GTMR_SMCR	Slave mode control register	0x08

Register name	Description	Offset address
GTMR_DIER	DMA/Interrupt enable register	0x0C
GTMR_SR	Status register	0x10
GTMR_CEG	Control event generation register	0x14
GTMR_CCM1	Capture/Compare mode register 1	0x18
GTMR_CCM2	Capture/Compare mode register 2	0x1C
GTMR_CCEN	Capture/Compare enable register	0x20
GTMR_CNT	Counter register	0x24
GTMR_PSC	Prescale register	0x28
GTMR_AUTORLD	Auto reload register	0x2C
GTMR_CC0	Channel 0 capture/compare register	0x34
GTMR_CC1	Channel 1 capture/compare register	0x38
GTMR_CC2	Channel 2 compare register	0x3C
GTMR_CC3	Channel 3 compare register	0x40

12.6 Register functional description

12.6.1 Control register 1 (GTMR_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:10	Reserved		
9:8	CLKDIV	R/W	Clock Division For the configuration of dead zone and digital filter, CK_INT provides the clock, and the dead time and the clock of the digital filter can be adjusted by this bit. 00: $t_{DTS}=t_{CK\_INT}$ 01: $t_{DTS}=2\times t_{CK\_INT}$ 10: $t_{DTS}=4\times t_{CK\_INT}$ 11: Reserved
7	ARPEN	R/W	GTMR_AUTORLD register Auto-reload Preload Enable When the buffer is disabled, modification of GTMR_AUTORLD by program will immediately lead to modification of the values loaded to the counter; when the buffer is enabled, modification of GTMR_AUTORLD by program will lead to modification of the values loaded to the counter at the next update event. 0: Disable 1: Enable
6:5	CAMSEL	R/W	Center Aligned Mode Select In the central alignment mode, the counter counts up and down alternately; otherwise, it will only count up or down. Different center alignment modes affect the timing of setting the output comparison

Field	Name	R/W	Description
			interrupt flag bit of the output channel to 1; when the counter is disabled (CNTEN=0), select the center alignment mode. 00: Edge-aligned mode 01: Center alignment mode 1 (the output compare interrupt flag bit of output channel is set to 1 when counting down) 10: Center alignment mode 2 (the output compare interrupt flag bit of output channel is set to 1 when counting up) 11: Center alignment mode 3 (the output compare interrupt flag bit of output channel is set to 1 when counting up/down)
4	CNTDIR	R/W	Counter Direction This bit is read-only when the counter is configured as central alignment mode or encoder mode. 0: Count up 1: Count down
3	SPMEN	R/W	Single Pulse Mode Enable When an update event is generated, the output level of the channel can be changed; in this mode, the CNTEN bit will be cleared, the counter will be stopped, and the subsequent output level of the channel will no long be changed. 0: Disable 1: Enable
2	URSSEL	R/W	Update Request Source Select If interrupt or DMA is enabled, the update event can generate update interrupt or DMA request. Different update request sources can be selected by this bit. 0: The counter overruns or underruns Set UEG bit Update generated by slave mode controller 1: The counter overruns or underruns
1	UDISEN	R/W	Update Disable Update event can cause AUTORLD, PSC and CCx to generate the value of update setting. 0: Enable update event (UEV) An update event can occur in any of the following situations: The counter overruns/underruns; Set UEG bit; Update generated by slave mode controller. 1: Disable update event
0	CNTEN	R/W	Counter Enable 0: Disable 1: Enable When the timer is configured as external clock, gated mode and encoder mode, it is required to write 1 to the bit by software to start regular work; when it is configured as the trigger mode, it can write 1 by hardware.

12.6.2 Control register 2 (GTMR_CR2)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:23	Reserved		
22:20	COMPETRSEL	R/W	COMP ETR input select 000: Select ETR input 001: Select COMP0 and input it into ETR 010: Select COMP1 and input it into ETR 011: Select COMP2 and input it to ETR 100: Select COMP3 and input it to ETR Others: Select ETR input
19:17	COMPCH3SEL	R/W	COMP channel 3 input select 000: Select Channel 3 for input 001: Select COMP0 and input it to channel 3 010: Select COMP1 input to channel 3 011: Select COMP2 and input it to channel 3 100: Select COMP3 input to channel 3 Others: Select Channel 3 for input
16:14	COMPCH2SEL	R/W	COMP channel 2 input select 000: Select Channel 2 for input 001: Select COMP0 and input it to channel 2 010: Select COMP1 input to channel 2 011: Select COMP2 and input it to channel 2 100: Select COMP3 input to channel 2 Others: Select Channel 2 for input
13:11	COMPCH1SEL	R/W	COMP channel 1 input select 000: Select Channel 1 for input 001: Select COMP0 and input it to channel 1 010: Select COMP1 input to channel 1 011: Select COMP2 and input it to channel 1 100: Select COMP3 input to channel 1 Others: Select Channel 1 for input
10:8	COMPCH0SEL	R/W	COMP channel 0 input select 000: Select Channel 0 for input 001: Select COMP0 and input it to channel 0 010: Select COMP1 input to channel 0 011: Select COMP2 and input it to channel 0 100: Select COMP3 input to channel 0 Others: Select Channel 0 for input
7	TI0SEL	R/W	Timer Input 1 Select 0: GTMR_CH0 pin is connected to TI0 input 1: GTMR_CH0, GTMR_CH1 and GTMR_CH2 pins are connected to TI0 input after exclusive
6:4	MMSEL	R/W	Master Mode Signal Select The signals of timers working in master mode can be used for TRGO, to affect the work of timers in slave mode and cascaded with the master timer, and the specific impact is related to the configuration of slave mode timer. 000: Reset; the reset signal of master mode timer is used for TRGO

Field	Name	R/W	Description
			001: Enable; the counter enable signal of master mode timer is used for TRGO 010: Update; the update event of master mode timer is used for TRGO 011: Compare pulses; when the master mode timer captures/compares successfully (CC0IFLG=1), a pulse signal is output for TRGO 100: Compare mode 1; OC0REF is used to trigger TRGO 101: Compare mode 2; OC1REF is used to trigger TRGO 110: Compare mode 3; OC2REF is used to trigger TRGO 111: Compare mode 4; OC3REF is used to trigger TRGO
3	CCDSEL	R/W	Capture/Compare DMA Select 0: Transmit DMA request of CCx when CCx event occurs 1: Transmit DMA request of CCx when an update event occurs
2:0	Reserved		

12.6.3 Slave mode control register (GTMR_SMCR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	ETPOL	R/W	External Trigger Polarity Configure This bit decides whether the external trigger ETR is phase-inverting. 0: The external trigger ETR is not phase-inverting, and the high level or rising edge is valid 1: The external trigger ETR is phase-inverting, and the low level or falling edge is valid
14	ECEN	R/W	External Clock Enable Mode2 0: Disable 1: Enable Setting ECEN bit has the same function as selecting external clock mode 1 to connect TRGI to ETRF; slave mode (reset, gating, trigger) can be used at the same time with external clock mode 2, but TRGI cannot be connected to ETRF in such case; when external clock mode 1 and external clock mode 2 are enabled at the same time, the input of external clock is ETRF.
13:12	ETPCFG	R/W	External Trigger Prescaler Configure The ETR (external trigger input) signal becomes ETRP after frequency division. The signal frequency of ETRP is at most 1/4 of GTMRCLK frequency; when ETR frequency is too high, the ETRP frequency must be reduced through frequency division. 00: Disable the prescaler 01: ETR signal 2 frequency division 10: ETR signal 4 frequency division 11: ETR signal 8 frequency division
11:8	ETFCFG	R/W	External Trigger Filter Configure 0000: Disable filter, sampled by f_{DTS} 0001: DIV=1, N=2

Field	Name	R/W	Description
			0010: DIV=1, N=4 0011: DIV=1, N=8 0100: DIV=2, N=6 0101: DIV=2, N=8 0110: DIV=4, N=6 0111: DIV=4, N=8 1000: DIV=8, N=6 1001: DIV=8, N=8 1010: DIV=16, N=5 1011: DIV=16, N=6 1100: DIV=16, N=8 1101: DIV=32, N=5 1110: DIV=32, N=6 1111: DIV=32, N=8 Sampling frequency=timer clock frequency/DIV; the filter length=N, and a jump is generated by every N events.
7	MSMEN	R/W	Master/slave Mode Enable 0: Invalid 1: Enable the master/slave mode
6:4	TRGSEL	R/W	Trigger Input Signal Select In order to avoid generating false edge detection when changing the value of this bit, it must be changed when SMFSEL=0. 000~011: Internal trigger ITR0 100: Channel 0 input edge detector TIF_ED 101: Channel 0 post-filtering timer input TI0FP0 110: Channel 1 post-filtering timer input TI1FP1 111: External trigger input (ETRF)
3	Reserved		
2:0	SMFSEL	R/W	Slave Mode Function Select 000: Disable the slave mode, the timer can be used as master mode timer to affect the work of slave mode timer; if GTMR_CR1.CNTEN=1, the prescaler is directly driven by the internal clock. 001: Encoder mode 1; according to the level of TI0FP0, the counter counts at the edge of TI1FP1. 010: Encoder mode 2; according to the level of TI1FP1, the counter counts at the edge of TI0FP0. 011: Encoder mode 3; according to the input level of the other signal, the counter counts at the edge of TI0FP0 and TI1FP1. 100: Reset mode; the slave mode timer resets the counter after receiving the rising edge signal of TRGI and generates the signal to update the register. 101: Gated mode; when the slave mode timer receives the TRGI high level signal, the counter will start to work; when it receives TRGI low level signal, the counter will stop working; when it receives TRGI high level signal again, the timer will continue to work; the counter is not reset during the whole period. 110: Trigger mode, the slave mode timer starts the counter to work after receiving the rising edge signal of TRGI.

Field	Name	R/W	Description
			111: External clock mode 1; select the rising edge signal of TRGI as the clock source to drive the counter to work.

12.6.4 DMA/Interrupt enable register (GTMR_DIER)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:15	Reserved		
14	TRGDEN	R/W	Trigger DMA Request Enable 0: Disable 1: Enable
13	Reserved		
12	CC3DEN	R/W	Capture/Compare Channel 3 DMA Request Enable 0: Disable 1: Enable
11	CC2DEN	R/W	Capture/Compare Channel 2 DMA Request Enable 0: Disable 1: Enable
10	CC1DEN	R/W	Capture/Compare Channel 1 DMA Request Enable 0: Disable 1: Enable
9	CC0DEN	R/W	Capture/Compare Channel 0 DMA Request Enable 0: Disable 1: Enable
8	UDIEN	R/W	Update DMA Request Enable 0: Disable 1: Enable
7	Reserved		
6	TRGIEN	R/W	Trigger Interrupt Enable 0: Disable 1: Enable
5	Reserved		
4	CC3IEN	R/W	Capture/Compare Channel 3 Interrupt Enable 0: Disable 1: Enable
3	CC2IEN	R/W	Capture/Compare Channel 2 Interrupt Enable 0: Disable 1: Enable
2	CC1IEN	R/W	Capture/Compare Channel 1 Interrupt Enable 0: Disable 1: Enable
1	CC0IEN	R/W	Capture/Compare Channel 0 Interrupt Enable 0: Disable

Field	Name	R/W	Description
			1: Enable
0	UIEN	R/W	Update Interrupt Enable 0: Disable 1: Enable

12.6.5 Status register (GTMR_SR)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
12	CC3RCFLG	RC_W0	Capture/Compare Channel 3 Repetition Capture Flag Refer to GTMR_SR.CC0RCFLG
11	CC2RCFLG	RC_W0	Capture/Compare Channel 2 Repetition Capture Flag Refer to GTMR_SR. CC0RCFLG
10	CC1RCFLG	RC_W0	Capture/Compare Channel 1 Repetition Capture Flag Refer to GTMR_SR. CC0RCFLG
9	CC0RCFLG	RC_W0	Captuer/Compare Channel 0 Repetition Capture Flag 0: Repeated capture does not occur 1: Repeated capture occurs The value of the counter is captured to GTMR_CC0 register, and CC0IFLG=1; this bit is set to 1 by hardware and cleared to 0 by software only when the channel is configured as input capture.
8:7	Reserved		
6	TRGIFLG	RC_W0	Trigger Event Interrupt Generate Flag 0: No trigger event interrupt occurs 1: Trigger event interrupt occurs When a trigger event is generated, this bit is set to 1 by hardware and cleared to 0 by software.
5	Reserved		
4	CC3IFLG	RC_W0	Captuer/Compare Channel 3 Interrupt Flag Refer to GTMR_SR. CC0IFLG
3	CC2IFLG	RC_W0	Captuer/Compare Channel 2 Interrupt Flag Refer to GTMR_SR. CC0IFLG
2	CC1IFLG	RC_W0	Capture/Compare Channel 1 Interrupt Flag Refer to GTMR_SR. CC0IFLG
1	CC0IFLG	RC_W0	Captuer/Compare Channel 0 Interrupt Flag When the capture/compare channel 0 is configured as output: 0: No matching occurs 1: The value of GTMR_CNT matches the value of GTMR_CC0 When the capture/compare channel 0 is configured as input: 0: No input capture occurs 1: Input capture occurs

Field	Name	R/W	Description
			When a capture event occurs, set 1 by hardware; clear 0 by software or clear 0 when reading GTMR_CC0 register.
0	UIFLG	RC_W0	Update Event Interrupt Generate Flag 0: No update event interrupt occurs 1: Update event interrupt occurred When the counter value is reloaded or reinitialized, an update event will be generated. The bit is set to 1 by hardware and cleared to 0 by software; update events are generated in the following situations: (1) UDISEN=0 on GTMR_CR1 register, and when the value of the repeat counter overruns/underruns, an update event will be generated; (2) URSSEL=0 and UDISEN=0 on GTMR_CR1 register, configure UEG=1 on GTMR_CEG register to generate an update event, and the counter needs to be initialized by software; (3) URSSEL=0 and UDISEN=0 on GTMR_CR1 register, and an update event will be generated when the counter is initialized by trigger event.

12.6.6 Control event generation register (GTMR_CEG)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:7	Reserved		
6	TEG	R/W	Trigger Event Generate 0: Invalid 1: Generate trigger event This bit is set to 1 by software and cleared to 0 automatically by hardware.
5	Reserved		
4	CC3EG	R/W	Capture/Compare Channel 3 Event Generation Refer to CC0EG description
3	CC2EG	R/W	Capture/Compare Channel 2 Event Generation Refer to CC0EG description
2	CC1EG	R/W	Capture/Compare Channel 1 Event Generation Refer to CC0EG description
1	CC0EG	R/W	Capture/Compare Channel 0 Event Generation 0: Invalid 1: Generate capture/compare event This bit is set to 1 by software and cleared to 0 automatically by hardware. If Channel 0 is in output mode: When CC0IFLG=1, if CC0IEN and CC0DEN bits are set, the corresponding interrupt and DMA request will be generated. If Channel 0 is in input mode:

Field	Name	R/W	Description
			The value of the capture counter is stored in GTMR_CC0 register; configure CC0IFLG=1, and if CC0IEN and CC0DEN bits are also set, the corresponding interrupt and DMA request will be generated; at this time, if CC0IFLG=1, it is required to configure CC0RCFLG=1.
0	UEG	R/W	Update Event Generate 0: Invalid 1: Initialize the counter and generate an update event This bit is set to 1 by software, and cleared to 0 by hardware. Note: When an update event is generated, the counter of the prescaler will be cleared to 0, but the prescaler factor remains unchanged. In the count-down mode, the counter reads the value of GTMR_AUTORLD; in central alignment mode or count-up mode, the counter will be cleared to 0.

12.6.7 Capture/Compare mode register 1 (GTMR_CCM1)

Offset address: 0x18

Reset value: 0x0000 0000

The timer can be configured as input (capture mode) or output (compare mode) by CCxSEL bit. The functions of other bits of the register are different in input and output modes, and the functions of the same bit are different in output mode and input mode. The OCxx in the register describes the function of the channel in the output mode, and the ICxx in the register describes the function of the channel in the input mode.

Output compare mode:

Field	Name	R/W	Description
31:16	Reserved		
15	OC1CEN	R/W	Output Compare Channel 1 Clear Enable
14:12	OC1MOD	R/W	Output Compare Channel 1 Mode
11	OC1PEN	R/W	Output Compare Channel 1 Buffer Enable
10	Used in input mode		
9:8	CC1SEL	R/W	Capture/Compare Channel 1 Select This bit defines the input/output direction and selects the input pin. 00: CC1 channel is output 01: CC1 channel is input, and IC1 is mapped on TI1 10: CC1 channel is input, and IC1 is mapped on TI0 11: CC1 channel is input, and IC1 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN register CC1EN=0).
7	OC0CEN	R/W	Output Compare Channel 0 Clear Enable 0: OC0REF is unaffected by ETRF input. 1: When high level of ETRF input is detected, OC0REF=0
6:4	OC0MOD	R/W	Output Compare Channel 0 Mode Configure 000: Freeze The output compare has no effect on OC0REF

Field	Name	R/W	Description
			001: The output value is high when matching. When the value of counter CNT matches the value CCx of capture/compare register, OC0REF will be forced to be high 010: The output value is low when matching. When the value of the counter matches the value of the capture/compare register, OC0REF will be forced to be low 011: Output reverses when matching. When the value of the counter matches the value of the capture/compare register, reverse the level of OC0REF 100: The output is forced to be low. Force OC0REF to be low 101: The output is forced to be high. Force OC0REF to be high 110: PWM mode 1 (set to high when the counter value<output compare value; otherwise, set to low) 111: PWM mode 2 (set to high when the counter value>output compare value; otherwise, set to low) Note: When the channel is configured as output, this bit cannot be modified. In PWM modes 1 and 2, the OC0REF level changes when the comparison result changes or when the output compare mode changes from freeze mode to PWM mode.
3	OC0PEN	R/W	Output Compare Channel 0 Preload Enable 0: Disable preloading function; write the value of GTMR_CC0 register through the program and it will work immediately. 1: Enable preloading function; write the value of GTMR_CC0 register through the program and it will work after an update event is generated. Note: When the channel is configured as output, this bit cannot be modified. When the preload register is uncertain, PWM mode can be used only in single-pulse mode (SPMEN=1); otherwise, the following output compare result is uncertain.
2	Used in input mode		
1:0	CC0SEL	R/W	Capture/Compare Channel 0 Select This bit defines the input/output direction and selects the input pin. 00: CC0 channel is output 01: CC0 channel is input, and IC0 is mapped on TI0 10: CC0 channel is input, and IC0 is mapped on TI1 11: CC0 channel is input, and IC0 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN register CC0EN=0).

Input capture mode:

Field	Name	R/W	Description
31:16	Reserved		
15:12	IC1F	R/W	Input Capture Channel 1 Filter Configure
11:10	IC1PSC	R/W	Input Capture Channel 1 Prescaler Configure
9:8	CC1SEL	R/W	Capture/Compare Channel 1 Select 00: CC1 channel is output 01: CC1 channel is input, and IC1 is mapped on TI1 10: CC1 channel is input, and IC1 is mapped on TI0

Field	Name	R/W	Description
			11: CC1 channel is input, and IC1 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN register CC1EN=0).
7:4	IC0F	R/W	Input Capture Channel 0 Filter Configure 0000: Disable filter, sampled by f_{DTS} 0001: DIV=1, N=2 0010: DIV=1, N=4 0011: DIV=1, N=8 0100: DIV=2, N=6 0101: DIV=2, N=8 0110: DIV=4, N=6 0111: DIV=4, N=8 1000: DIV=8, N=6 1001: DIV=8, N=8 1010: DIV=16, N=5 1011: DIV=16, N=6 1100: DIV=16, N=8 1101: DIV=32, N=5 1110: DIV=32, N=6 1111: DIV=32, N=8 Sampling frequency=timer clock frequency/DIV; the filter length=N, indicating that a jump is generated by every N events.
3:2	IC0PSC	R/W	Input Capture Channel 0 Prescaler Configure 00: PSC=1 01: PSC=2 10: PSC=4 11: PSC=8 PSC is prescaler factor; capture is triggered once by every PSC events.
1:0	CC0SEL	R/W	Capture/Compare Channel 0 Select 00: CC0 channel is output 01: CC0 channel is input, and IC0 is mapped on TI0 10: CC0 channel is input, and IC0 is mapped on TI1 11: CC0 channel is input, and IC0 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN bit CC0EN=0).

12.6.8 Capture/Compare mode register 2 (GTMR_CCM2)

Offset address: 0x1C

Reset value: 0x0000 0000

Refer to the description of the above CCM1 register.

Output compare mode:

Field	Name	R/W	Description
31:16	Reserved		
15	OC3CEN	R/W	Output Compare Channel 3 Clear Enable

Field	Name	R/W	Description
14:12	OC3MOD	R/W	Output Compare Channel 3 Mode Configure
11	OC3PEN	R/W	Output Compare Channel 3 Buffer Enable
10	Used in input mode		
9:8	CC3SEL	R/W	Capture/Compare Channel 3 Select This bit defines the input/output direction and selects the input pin. 00: CC3 channel is output 01: CC3 channel is input, and IC3 is mapped on TI3 10: CC3 channel is input, and IC3 is mapped on TI2 11: CC3 channel is input, and IC3 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN register CC3EN=0).
7	OC2CEN	R/W	Output Compare Channel 2 Clear Enable 0: OC2REF is unaffected by ETRF input 1: When high level of ETRF input is detected, OC2REF=0
6:4	OC2MOD	R/W	Output Compare Channel 2 Mode Configure
3	OC2PEN	R/W	Output Compare Channel 2 Preload Enable
2	Used in input mode		
1:0	CC2SEL	R/W	Capture/Compare Channel 2 Select This bit defines the input/output direction and selects the input pin. 00: CC2 channel is output 01: CC2 channel is input, and IC2 is mapped on TI2 10: CC2 channel is input, and IC2 is mapped on TI3 11: CC2 channel is input, and IC2 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN register CC2EN=0).

Input capture mode:

Field	Name	R/W	Description
15:12	IC3F	R/W	Input Capture Channel 3 Filter Configure
11:10	IC3PSC	R/W	Input Capture Channel 3 Prescaler Configure
9:8	CC3SEL	R/W	Capture/Compare Channel 3 Select 00: CC3 channel is output 01: CC3 channel is input, and IC3 is mapped on TI3 10: CC3 channel is input, and IC3 is mapped on TI2 11: CC3 channel is input, and IC3 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN register CC3EN=0).
7:4	IC2F	R/W	Input Capture Channel 2 Prescaler Configure
3:2	IC2PSC	R/W	Input Capture Channel 2 Prescaler Configure 00: PSC=1

Field	Name	R/W	Description
			01: PSC=2 10: PSC=4 11: PSC=8 PSC is prescaler factor; capture is triggered once by every PSC events.
1:0	CC2SEL	R/W	Capture/Compare Channel 2 Select 00: CC2 channel is output 01: CC2 channel is input, and IC2 is mapped on TI2 10: CC2 channel is input, and IC2 is mapped on TI3 11: CC2 channel is input, and IC2 is mapped on TRC, and only works in internal trigger input Note: This bit can be written only when the channel is closed (GTMR_CCEN register CC2EN=0).

12.6.9 Capture/Compare enable register (GTMR_CCEN)

Offset address: 0x20

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:14	Reserved		
13	CC3POL	R/W	Capture/Compare Channel 3 Output Polarity Refer to CCEN_CC0POL
12	CC3EN	R/W	Capture/Compare Channel 3 Output Enable Refer to CCEN_CC0EN
11:10	Reserved		
9	CC2POL	R/W	Capture/Compare Channel 2 Output Polarity Configure Refer to CCEN_CC0POL
8	CC2EN	R/W	Capture/Compare Channel 2 Output Enable Refer to CCEN_CC0EN
7:6	Reserved		
5	CC1POL	R/W	Capture/Compare Channel 1 Output Polarity Configure Refer to CCEN_CC0POL
4	CC1EN	R/W	Capture/Compare Channel 1 Output Enable Refer to CCEN_CC0EN
3:2	Reserved		
1	CC0POL	R/W	Capture/Compare Channel 0 Output Polarity Configure When CC0 channel is configured as output: 0: OC0 is active high 1: OC0 is active low When CC0 channel is configured as input: 0: Phase not reversed, capture at the rising edge of IC0; phase not reversed when IC0 is used as external trigger. 1: Phase reversed, capature at the falling edge of ICC0; phase reversed when IC0 is used as external trigger.
0	CC0EN	R/W	Capture/Compare Channel 0 Output Enable When CC0 is configured as output:

Field	Name	R/W	Description
			0: Disable output 1: Enable output When CC0 is configured as input: This bit determines whether the value CNT of the counter can be captured and enter GTMR_CC0 register 0: Disable capture 1: Enable capture

12.6.10 Counter register (GTMR_CNT)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	CNT	R/W	Counter Value

12.6.11 Prescale register (GTMR_PSC)

Offset address: 0x28

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	PSC	R/W	Prescaler Value Clock frequency of counter (CK_CNT)＝f_{CK_PSC}/(PSC+1)

12.6.12 Auto reload register (GTMR_AUTORLD)

Offset address: 0x2C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	AUTORLD	R/W	Auto Reload Value When the value of auto reload is empty, the counter will not count.

12.6.13 Channel 0 capture/compare register (GTMR_CC0)

Offset address: 0x34

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	CC0	R/W	Capture/Compare Channel 0 Value When the capture/compare channel 0 is configured as input mode: CC0 contains the counter value transmitted by the last input capture channel 0 event. When the capture/compare channel 0 is configured as output mode: CC0 contains the value currently loaded in the capture/compare register Compare the value CC0 of the capture and compare channel 0 with the value CNT of the counter to generate the output signal on OC0. When the output compare preload is disabled (OC0PEN=0 for GTMR_CCM1 register), the written value will immediately affect the output comparison results;

Field	Name	R/W	Description
			If the output compare preload is enabled (OC0PEN=1 for GTMR_CCM1 register), the written value will affect the output comparison result when an update event is generated.

12.6.14 Channel 1 capture/compare register (GTMR_CC1)

Offset address: 0x38

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CC1	R/W	Capture/Compare Channel 1 Value Refer to GTMR_CC0

12.6.15 Channel 2 Compare Register (GTMR_CC2)

Offset address: 0x3C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	CC2	R/W	Compare Channel 2 Value Refer to GTMR_CC0

12.6.16 Channel 3 Compare Register (GTMR_CC3)

Offset address: 0x40

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	CC3	R/W	Compare Channel 3 Value Refer to GTMR_CC0

13 Basic Timer (BTIMER)

13.1 Introduction

The basic timers have an unsigned 16-bit counter, auto reload register, prescaler and trigger controller.

13.2 Main characteristics

(1) Timebase unit
- Counter: 16-bit counter, supporting count-up, count-down and central alignment count
- Prescaler: 16-bit programmable prescaler
- Autoreload function

(2) Clock source selection
- Internal clock

(3) Input capture function

(4) Counting function

(5) PWM input mode

(6) Output compare function

(7) PWM output mode

(8) Forced output mode

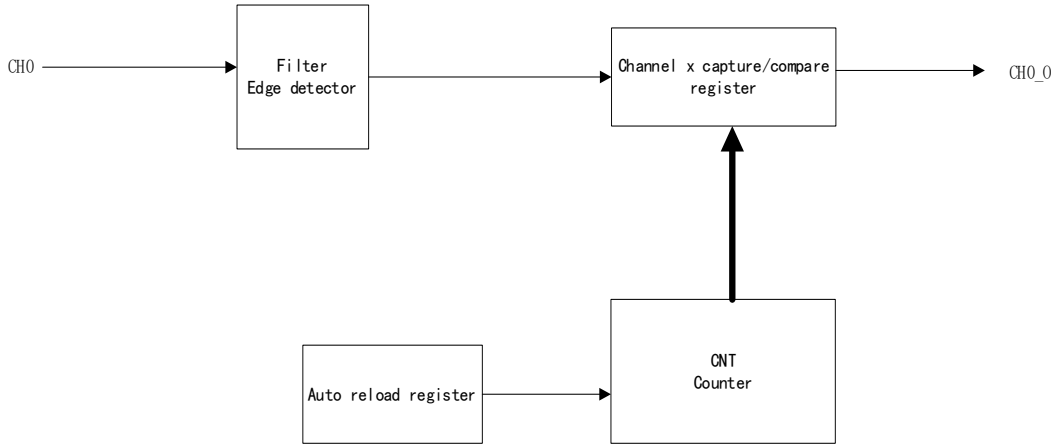
(9) Single-pulse mode

(10) Timing function

(11) Interrupt output request events
- Update event (counter overrun/underrun, counter initialization)
- Capture/Compare event

13.3 Structure block diagram

Figure 55 Structure Block Diagram



13.4 Functional description

13.4.1 Clock source selection

The basic timer is driven by internal clock source BTMR_CLK

Configure the CNTEN bit of BTMR_CR1 register to enable the counter; when CNTEN bit is set, the internal clock CK_INT can generate CK_INT to drive the counter through the controller and prescaler.

13.4.2 Timebase unit

The time base unit in the basic timer contains three registers

- Counter register (CNT) 16 bits
- Autoreload register (AUTORLD) 16 bits
- Prescaler (PSC) 16 bits

Counter CNT

There are three counting modes for the counter in the general-purpose timer

- Count-up mode
- Count-down mode
- Central alignment mode

Count-up mode

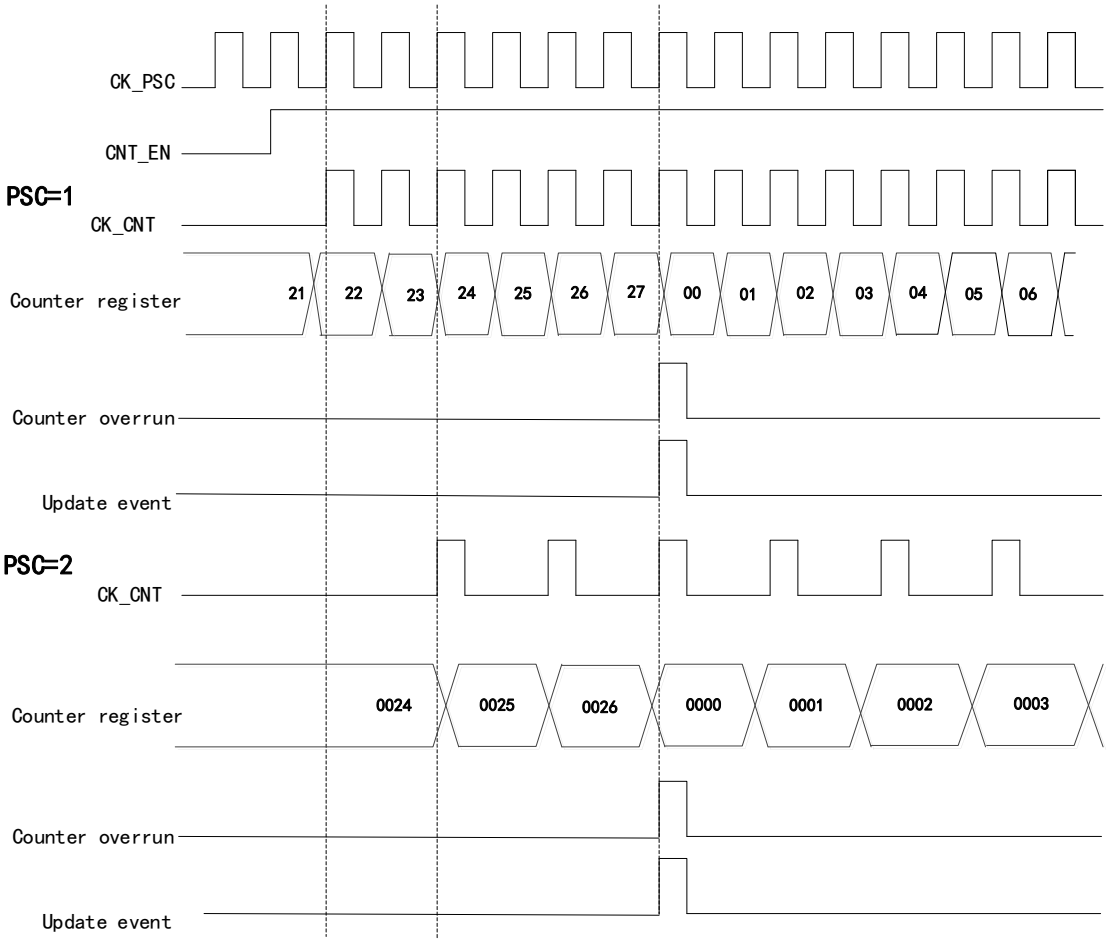
Set to the count-up mode by configuring CNTDIR bit of control register (BTMR_CR1).

When the counter is in count-up mode, the counter will count up from 0; every time a pulse is generated, the counter will increase by 1 and when the value of the counter (BTMR_CNT) is equal to the value of the auto reload

(BTMR_AUTORLD), the counter will start to count from 0 again, a count-up overrun event will be generated, and the value of automatic reloading (BTMR_AUTORLD) is written in advance.

The figure below is the timing diagram of count-up mode when the division factor is 1 or 2.

Figure 56 Timing Diagram of Count-up Mode when Division Factor is 1 or 2



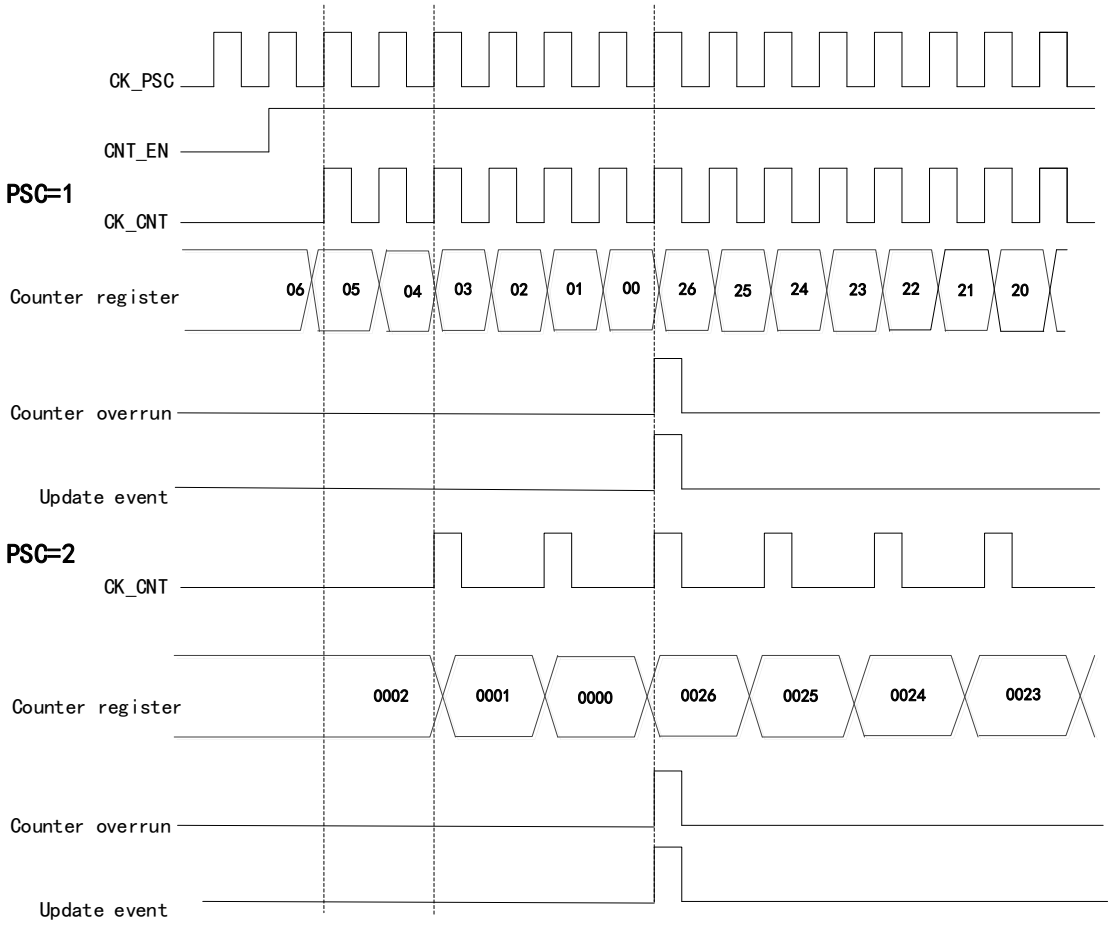
Count-down mode

Set to the count-down mode by configuring CNTDIR bit of control register (BTMR_CR1).

When the counter is in down-counting mode, the counter starts counting down from the auto-reload value (BTMR_AUTORLD). Each pulse causes the counter to decrement by 1, and when it counts down to 0, the counter restarts counting from (BTMR_AUTORLD).

The figure below is the timing diagram of count-down mode when the division factor is 1 or 2.

Figure 57 Timing Diagram of Count-down Mode when Division Factor is 1 or 2



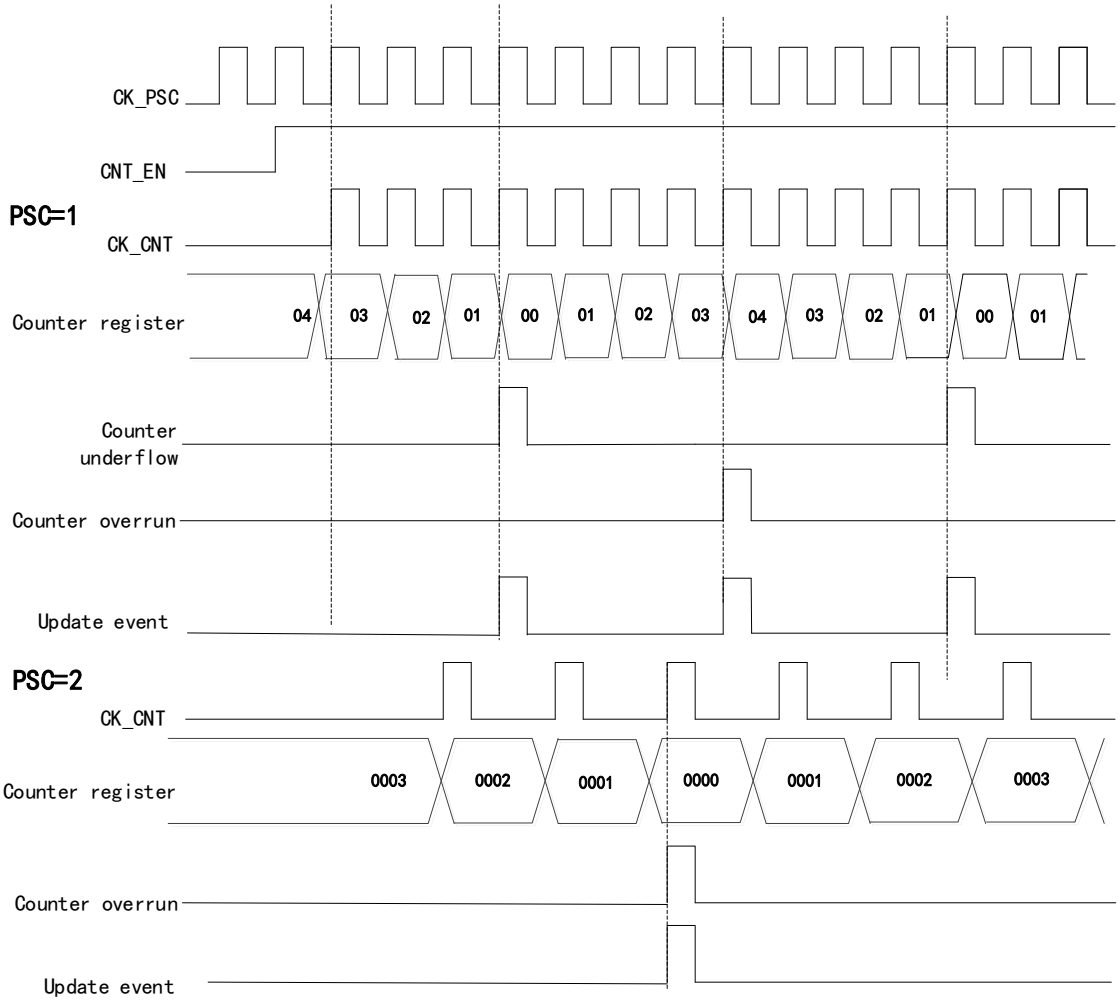
Central alignment mode

Set to the central alignment mode by configuring CAMSEL bit of control register (BTMR_CR1).

When the counter is in center alignment mode, the counter counts up from 0 to the value of auto reload (BTMR_AUTORLD), then counts down to 0 from the value of the auto reload (BTMR_AUTORLD), which will repeat; in counting up, when the counter value is (AUTORLD-1), a counter overrun event will be generated; in counting down, when the counter value is 1, a counter underrun event will be generated.

The figure below is the timing diagram of central alignment mode when the division factor is 1 or 2.

Figure 58 Timing Diagram of Central alignment Mode when Division Factor is 1 or 2



Prescaler PSC

The prescaler is 16 bits and programmable, and it can divide the clock frequency of the counter to any value within 1~65536 (controlled by BTMR_PSC register), and after frequency division, the clock will drive the counter CNT to count. The prescaler has a buffer, which can be changed during running.

13.4.3 Input capture

Input capture channel

The basic timer has one independent capture/compare channels, each of which is surrounded by a capture/compare register.

In the input capture, the measured signal will enter from the external pin TI0 of the timer, first pass through the edge detector and input filter, and then enter the capture channels. Each capture channel has a corresponding capture register. When the capture occurs, the value of the counter CNT will be latched in the capture register CCx. Before entering the capture register, the signal will pass

through the prescaler to set how many events to capture at a time.

Input capture application

Input capture is used to capture external events, and can give the time flag to indicate the occurrence time of the event and measure the pulse jump edge events (measure the frequency or pulse width), for example, if the selected edge appears on the input pin, the BTMR_CCx register will capture the current value of the counter and the CCxIFLG bit of the status register BTMR_SR will be set to 1; if CCxIEN=1, an interrupt will be generated.

In capture mode, the timing, frequency, cycle and duty cycle of a waveform can be measured. In the input capture mode, the edge selection is set to rising edge detection. When the rising edge appears on the capture channel, the first capture occurs, at this time, the value of the counter CNT will be latched in the capture register CCx; at the same time, it will enter the capture interrupt, a capture will be recorded in the interrupt service program and the value will be recorded. When the next rising edge is detected, the second capture occurs, the value of counter CNT will be latched in capture register CCx again, at this time, it will enter the capture interrupt again; read the value of capture register and the cycle of this pulse signal will be obtained by capture.

13.4.4 Output compare

There are eight modes of output compare: freeze, channel x is valid when matching, channel x is invalid when matching, reverse, force to invalid, force to valid, PWM mode 1 and PWM mode 2, which are configured by OCxMOD bit in BTMR_CCxCRx register and can control the waveform of output signal in output compare mode.

Output compare application

In the output compare mode, the position, polarity, frequency and time of the pulse generated by the timer can be controlled.

When the value of the counter is equal to that of the capture/compare register, the channel output can be set as high level, low level or reverse by configuring the OCxMOD bit in BTMR_CCxCRx register and the CCxPOL bit.

13.4.5 PWM output mode

PWM mode is pulse signal that can be adjusted by external output of the timer. The pulse width of the signal is determined by the value of the compare register CCx, and the cycle is determined by the value of the auto reload AUTORLD.

PWM output mode contains PWM mode 1 and PWM mode 2; PWM mode 1 and PWM mode 2 are divided into count-up, count-down and edge alignment counting; in PWM mode 1, if the value of the counter CNT is less than the value of the compare register CCx, the output level will be valid; otherwise, it will be invalid.

Set the timing diagram in PWM mode 1 when CCx=5, AUTORLD=7.

Figure 59 Timing Diagram of PWM1 Count-up Mode

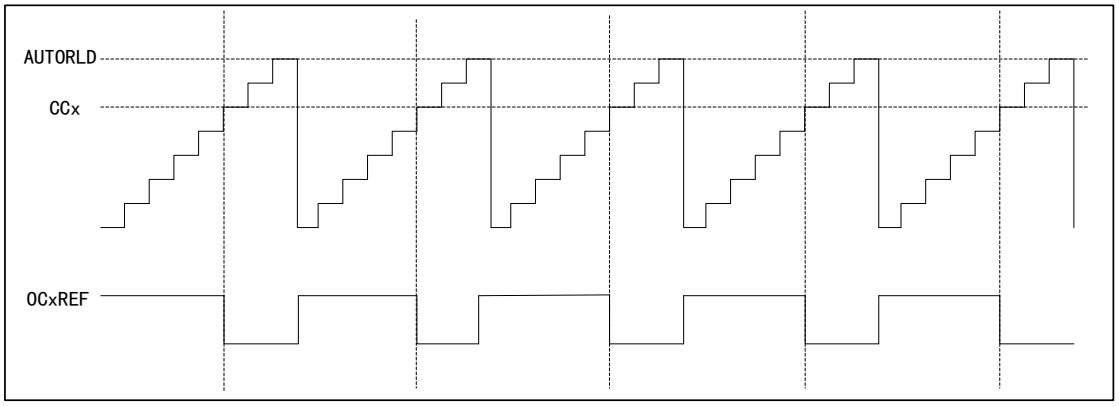


Figure 60 Timing Diagram of PWM1 Count-down Mode

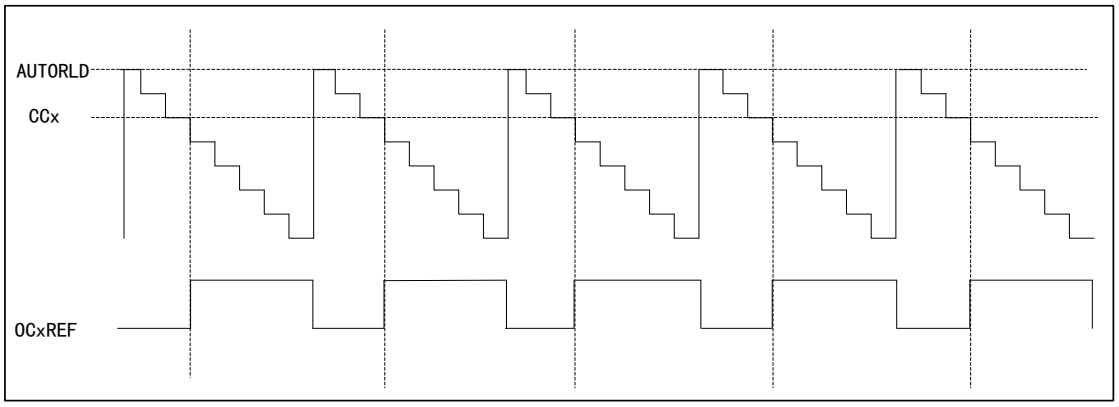
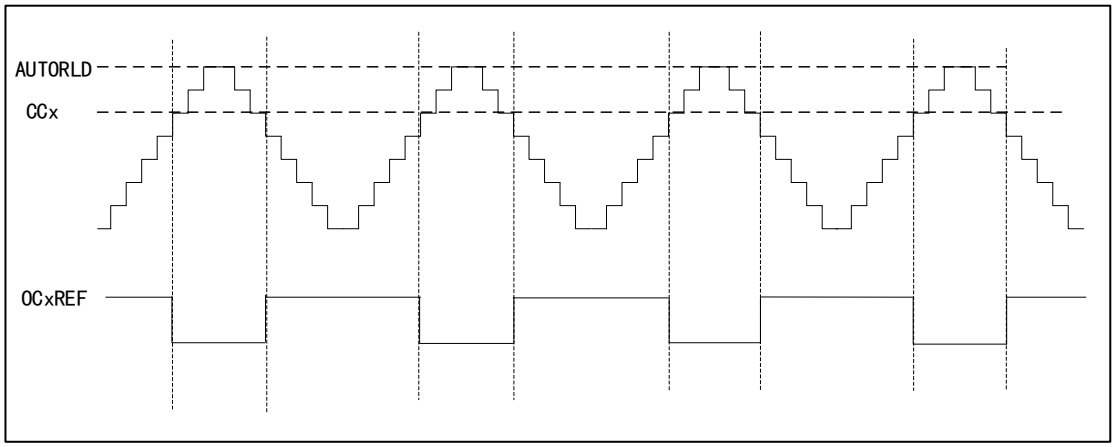


Figure 61 Timing Diagram of PWM1 Central alignment Mode



In PWM mode 2, if the value of the counter CNT is less than that of the compare register CCx, the output level will be invalid; otherwise, it will be valid.

Set the timing diagram in PWM mode 2 when CCx=5, AUTORLD=7.

Figure 62 Timing Diagram of PWM2 Count-up Mode

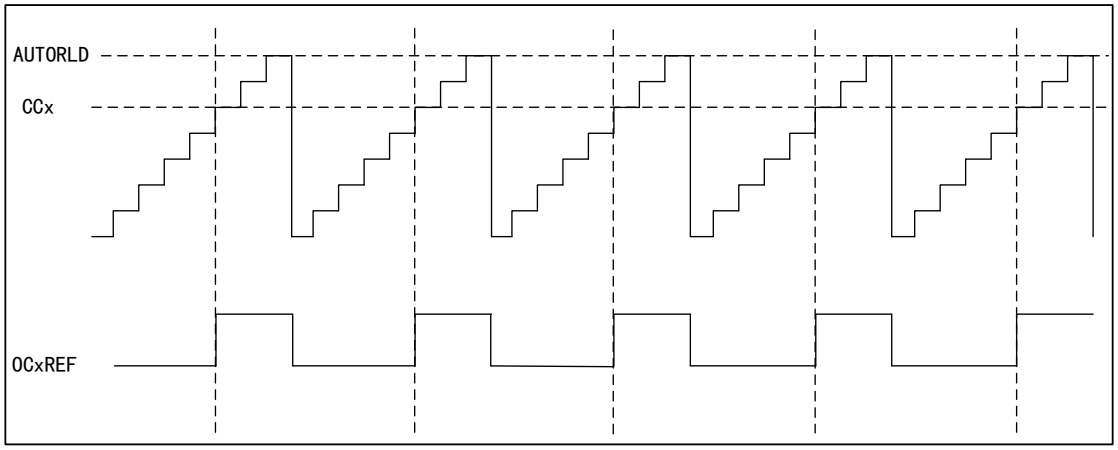


Figure 63 Timing Diagram of PWM2 Count-down Mode

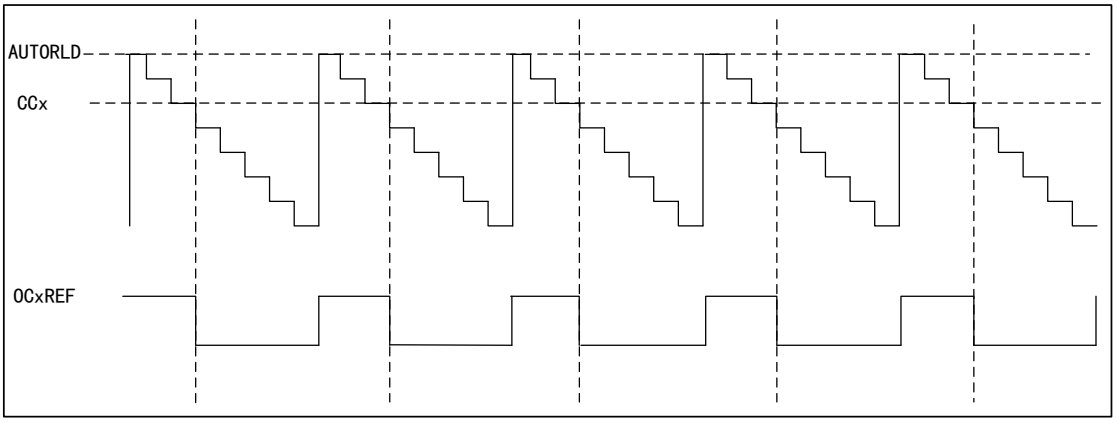
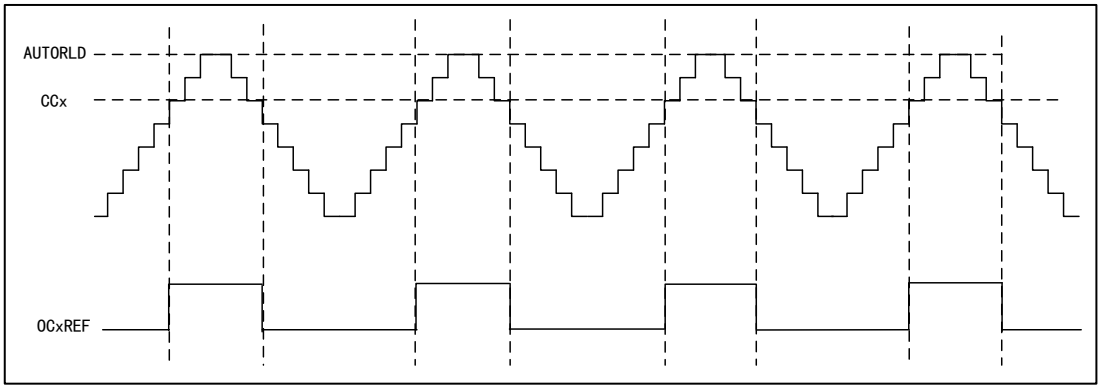


Figure 64 Timing Diagram of PWM2 Central alignment Mode



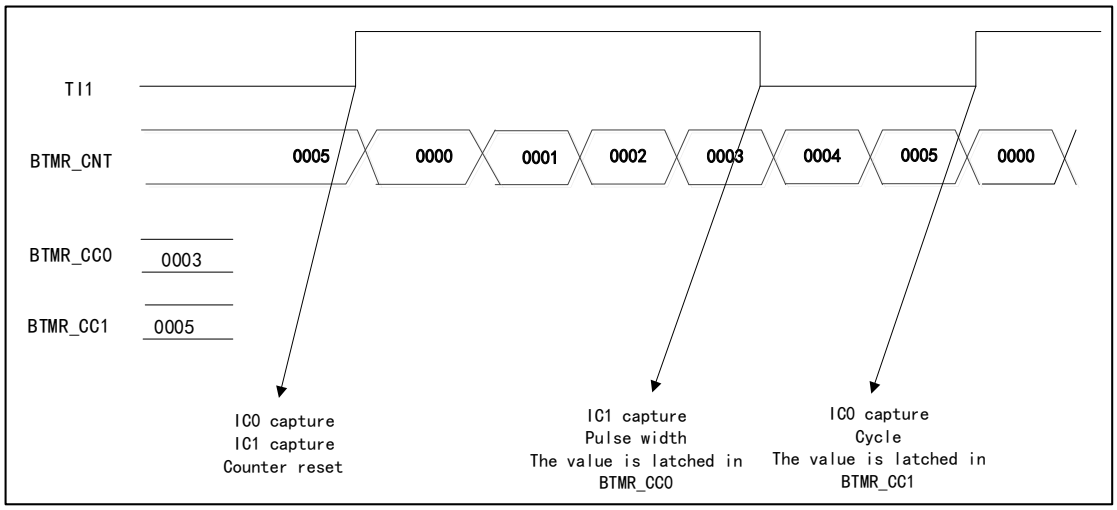
13.4.6 PWM input mode

PWM input mode is a particular case of input capture.

In the PWM input mode, the PWM signal enters from BTMR_CH0, and the signal will be divided into two channels, one can measure the cycle and the other can measure the duty cycle. In the configuration, it is only required to set

the polarity of one channel, and the other will be automatically configured with the opposite polarity.

Figure 65 Timing Diagram in PWM Input Mode



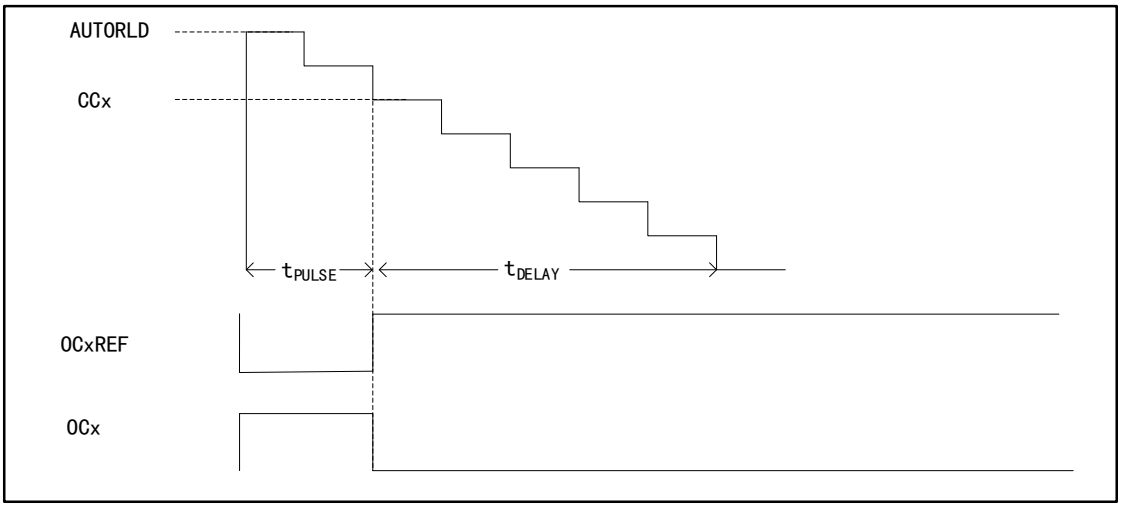
13.4.7 Single-pulse mode

The single-pulse mode is a special case of timer compare output, and is also a special case of PWM output mode.

Set SPMEN bit of BTMR_CR1 register, and select the single-pulse mode. After the counter is started, a certain number of pulses will be output before the update event occurs. When an update event occurs, the counter will stop counting, and the subsequent PWM waveform output will no longer be changed.

After a certain controllable delay, a pulse with controllable pulse width is generated in single-pulse mode through the program. The delay time is defined by the value of BTMR_CCx register; in the count-up mode, the delay time is CCx and the pulse width is AUTORLD+1-CCx; in the count-down mode, the delay time is AUTORLD-CCx and the pulse width is CCx+1.

Figure 66 Timing Diagram of Single-pulse Mode



13.4.8 Forced output mode

In the forced output mode, the comparison result is ignored, and the corresponding level is directly output according to the configuration instruction.

- CCxSEL=1 for BTMR_CCxCR2 register, set CCx channel as output
- OCxMOD=100/101 for BTMR_CCxCR1 register, set to force OCxREF signal to invalid/valid

In this mode, the corresponding interrupt will still be generated.

13.4.9 Interrupt request

The timer can generate an interrupt when an event occurs during operation

- Update event (counter overrun/underrun, counter initialization)
- Capture/Compare event

13.5 Register address mapping

Table 41 BTIMER Register Address Mapping

Register name	Description	Offset address
BTMR_CR1	Control register 1	0x00
BTMR_CCxCR1	Channel x control register 1	0x04
BTMR_CCxCR2	Channel x control register 2	0x08
BTMR_CEG	Control event generation register	0x0C
BTMR_IER	Interrupt enable register	0x10
BTMR_SR	Status register	0x14
BTMR_CNT	Counter register	0x18
BTMR_PSC	Prescale register	0x1C

Register name	Description	Offset address
BTMR_AUTORLD	Auto reload register	0x20
BTMR_CC0	Channel 0 capture/compare register	0x24

13.6 Register functional description

13.6.1 Control register 1 (BTMR_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	PRPEN	R/W	PSC Register Auto-reload Preload Enable When the buffer is disabled, modifying the PSC in the program will immediately change the value loaded into the PSC; when the buffer is enabled, modifying the PSC in the program will change the value loaded into the counter at the next update event. 0: Disable 1: Enable
6	ARPEN	R/W	BTMR_AUTORLD Register Auto-reload Preload Enable When the buffer is disabled, modification of BTMR_AUTORLD by program will immediately lead to modification of the values loaded to the counter; when the buffer is enabled, modification of BTMR_AUTORLD by program will lead to modification of the values loaded to the counter at the next update event. 0: Disable 1: Enable
5:4	CAMSEL	R/W	Central alignment mode select In the central alignment mode, the counter counts up and down alternately; otherwise, it will only count up or down. Different center alignment modes affect the timing of setting the output comparison interrupt flag bit of the output channel to 1; when the counter is disabled (CNTEN=0), select the center alignment mode. 00: Edge-aligned mode 01: Center alignment mode 1 (the output compare interrupt flag bit of output channel is set to 1 when counting down) 10: Center alignment mode 2 (the output compare interrupt flag bit of output channel is set to 1 when counting up) 11: Center alignment mode 3 (the output compare interrupt flag bit of output channel is set to 1 when counting up/down)
3	UDISEN	R/W	Update Disable Update event can cause AUTORLD, PSC and CCx to generate the value of update setting. 0: Enable update event (UEV) An update event can occur in any of the following situations: The counter overruns/underruns; Set UEG bit; 1: Disable update event

Field	Name	R/W	Description
2	SPMEN	R/W	Single Pulse Mode Enable When an update event is generated, the output level of the channel can be changed; in this mode, the CNTEN bit will be cleared, the counter will be stopped, and the subsequent output level of the channel will no long be changed. 0: Disable 1: Enable
1	CNTDIR	R/W	Counter Direction This bit is read-only when the counter is configured as central alignment mode. 0: Count up 1: Count down
0	CNTEN	R/W	BTIMER Enable 0: Disable 1: Enable

13.6.2 Channel x control register 1 (BTMR_CCxCR1)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:10	Reserved		
9:7	OC0MOD	R/W	Output Compare Channel 0 Mode Configure 000: Freeze The output compare has no effect on OC0REF 001: The output value is high when matching. When the value of counter CNT matches the value CCx of capture compare register, OC0REF will be forced to be at high level 010: The output value is low when matching. When the value of the counter matches the value of the capture/compare register, OC0REF will be forced to be low 011: Output reverses when matching. When the value of the counter matches the value of the capture comparison register, flip the level of OC0REF 100: The output is forced to be low. Force OC0REF to be low 101: The output is forced to be high. Force OC0REF to be high 110: PWM mode 1 (set to high when the counter value<output compare value; otherwise, set to low) 111: PWM mode 2 (set to high when the counter value>output compare value; otherwise, set to low) Note: When the channel is configured as output, this bit cannot be modified. In PWM modes 1 and 2, the OC0REF level changes when the comparison result changes or when the output compare mode changes from freeze mode to PWM mode.
6:4	IC0F	R/W	Input filtering coefficient Based on TIMER 000: bypass 001: 1 PCLK 010: 2 PCLK

Field	Name	R/W	Description
			011: 3 PCLK 100: 4 PCLK 101: 5 PCLK 110: 6 PCLK 111: 7 PCLK
3:2	CC0EDGESEL	R/W	Capture effective edge 00: Rising edge is effective 01: Falling edge is effective 10: Both edges are effective 11: Rising edge active
1	CC0POL	R/W	Capture/Compare Channel 0 Output Polarity Configure When CC0 channel is configured as output: 0: Phase not reversed 1: Phase reversed When CC0 channel is configured as input 0: Phase not reversed 1: Phase reversed
0	CC0EN	R/W	CC0 channel input and output enable 0: Not enable 1: Enable

13.6.3 Channel x control register 2 (BTMR_CCxCR2)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:2	Reserved		
1	CC0SEL	R/W	Channel 0 input and output select 0: Analog input mode 1: Output mode
0	CC0RPEN	R/W	CC0 register auto- reloading buffer enable When the buffer is disabled, program modifications will immediately change the value loaded into CC0; when the buffer is enabled, program modifications to CC0 will change the value loaded into the counter at the next update event. 0: Disable 1: Enable

13.6.4 Control event generation register (BTMR_CEG)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:2	Reserved		

Field	Name	R/W	Description
1	CC0EG	W	Capture/Compare Channel 0 Event Generation 0: Invalid 1: Generate capture/compare event This bit is set to 1 by software and cleared to 0 automatically by hardware. If Channel 0 is in output mode When CC0IFLG=1, if CC0IEN is set, the corresponding interrupt is generated. If Channel 0 is in input mode The value of the capture counter is stored in BTMR_CC0 register; configure CC0IFLG=1, and if CC0IEN is also set, the corresponding interrupt will be generated; at this time, if CC0IFLG=1, it is required to configure CC0RCFLG=1.
0	UEG	W	Update Event Generate 0: Invalid 1: Initialize the counter and generate an update event This bit is set to 1 by software, and cleared to 0 by hardware. Note: When an update event is generated, the counter of the prescaler will be cleared to 0, but the prescaler factor remains unchanged. In the count-down mode, the counter reads the value of BTMR_AUTORLD; in central alignment mode or count-up mode, the counter will be cleared to 0.

13.6.5 Interrupt enable register (BTMR_IER)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:2	Reserved		
1	CC0IEN	R/W	Capture/Compare Channel 0 Interrupt Enable 0: Disable 1: Enable
0	UIEN	R/W	Update Interrupt Enable 0: Disable 1: Enable

13.6.6 Status register (BTMR_SR)

Offset address: 0x14

Reset value: 0x0000

Field	Name	R/W	Description
31:3	Reserved		
2	CC0RCFLG	RC_W0	Captuer/Compare Channel 0 Repetition Capture Flag 0: Repeated capture does not occur 1: Repeated capture occurs The value of the counter is captured to GTMR_CC0 register, and CC0IFLG=1; this bit is set to 1 by hardware and cleared to 0 by software only when the channel is configured as input capture.

Field	Name	R/W	Description
1	UIFLG	R_W0C	Update Event Generate Flag 0: No update event occurs 1: Update event occurs When the counter value is reloaded or reinitialized, an update event will be generated. The bit is set to 1 by hardware and cleared to 0 by software.
0	CC0IFLG	RC_W0	Capture/Compare Channel 0 Interrupt Flag When the capture/compare channel 0 is configured as output: 0: No matching occurs 1: The value of BTMR_CNT matches the value of BTMR_CC0 When the content of BTMR_CC0 is greater than that of BTMR_AUTORLD, under the condition of counter overflow in the upcount mode, this bit is 1. When the capture/compare channel 0 is configured as input: 0: No input capture occurs 1: Input capture occurs When a capture event occurs, set 1 by hardware; clear 0 by software or clear 0 when reading BTMR_CC0 register.

13.6.7 Counter register (BTMR_CNT)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CNT	R/W	Counter Value

13.6.8 Prescale register (BTMR_PSC)

Offset address: 0x1C

Reset value: 0x0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	PSC	R/W	Prescaler Value Clock frequency of counter (CK_CNT)=f_{CK_PSC}/(PSC+1)

13.6.9 Auto reload register (BTMR_AUTORLD)

Offset address: 0x20

Reset value: 0x0000 FFFF

Field	Name	R/W	Description
31:16	Reserved		
15:0	AUTORLD	R/W	Auto Reload Value When the value of auto reload is empty, the counter will not count.

13.6.10 Channel 0 capture/compare register (BTMR_CC0)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CC0	R/W	Capture/Compare Channel 0 Value When the capture/compare channel 0 is configured as input mode: CC0 contains the counter value transmitted by the last input capture channel 0 event. When the capture/compare channel 0 is configured as output mode: CC0 contains the value currently loaded in the capture/compare register Compare the value CC0 of the capture and compare channel 0 with the value CNT of the counter to generate the output signal on OC0. When the output compare preload is disabled (OC0PEN=0 for BTMR_CCxCR1 register), the written value will immediately affect the output comparison results; If the output compare preload is enabled (OC0PEN=1 for BTMR_CCxCR1 register), the written value will affect the output comparison result when an update event is generated.

14 Low-power timer (LPTIMER)

14.1 Introduction

The low-power timer features an unsigned 16-bit counter that can operate in a low-power state with very low power consumption. It also supports waking up the system from a low-power mode.

14.2 Main characteristics

(1) 16-bit count-up counter

(2) Configurable clock frequency selection

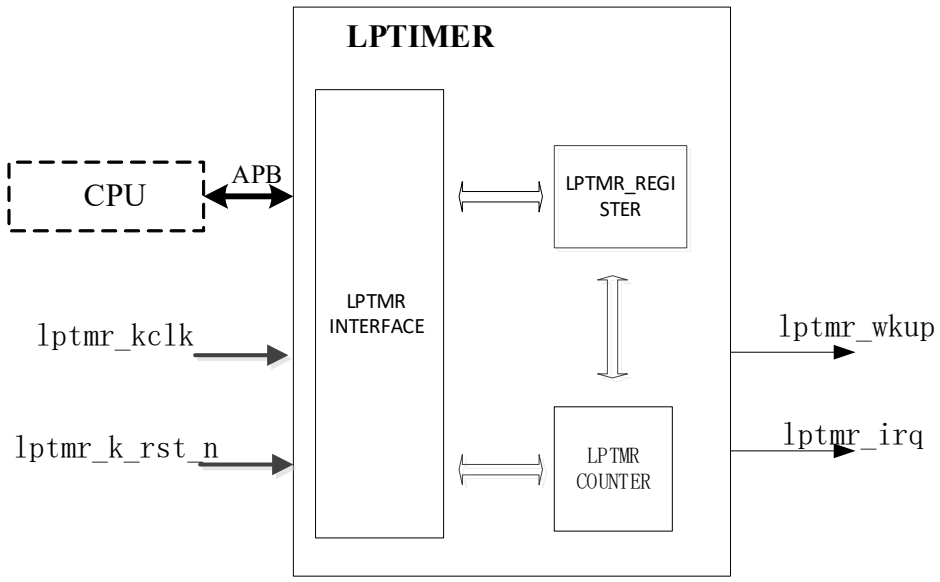
(3) Low-frequency independent clock drive

(4) Wake-up output

(5) Interrupt output

14.3 Structure block diagram

Figure 67 Structure Block Diagram



14.4 Functional description

14.4.1 Direction for use

(1) Configure LPTMR_PSC register

(2) Configure LPTMR_WKVAL register

(3) Configure the CNTEN bit enable module of LPTMR_CR

(4) Configure the LPTMR_CR LPTIEN bit to enable interrupts. Enabling it allows you to enter the interrupt, otherwise it cannot

14.5 Register address mapping

Table 42 LPTIMER Register Address Mapping

Register name	Description	Offset address
LPTMR_CR	Control register	0x00
LPTMR_PSC	Prescale register	0x04
LPTMR_WKVAL	Wake-up count Register	0x08
LPTMR_SR	Status register	0x0C
LPTMR_CNT	Counter register	0x10

14.6 Register functional description

14.6.1 Control register (LPTMR_CR)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:2	Reserved		
1	LPTIEN	R/W	Interrupt enable 0: Disable 1: Enable This bit can only be written after lsi_ready.
0	CNTEN	R/W	Module enable 0: Disable 1: Enable This bit can only be written after lsi_ready.

14.6.2 Prescaler register (LPTMR_PSC)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	PSC	R/W	Prescale select Frequency division of 1~65535 can be selected 0000000000000000: 1 frequency division 0000000000000001: 1 frequency division 0000000000000010: 2 frequency division 0000000000000011: 3 frequency division …… 1111111111111111: 65535 frequency division

14.6.3 Wake-up Count Register (LPTMR_WKVAL)

Offset address: 0x08

Reset value: 0x0000 FFFF

Field	Name	R/W	Description
31:16	Reserved		
15:0	WKVAL	R/W	Wakeup value

14.6.4 Status register (LPTMR_SR)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:3	Reserved		
2	RVU	R	LPTMR Reload Value Update Flag When the counter reload value is updated, set 1 by hardware; after the counter reload value is updated, clear 0 by hardware; the counter reload value is updated only when the RVU bit is cleared to 0.
1	PVU	R	LPTMR Prescaler Value Update Flag When the prescaler factor is updated, set 1 by hardware; after the prescaler factor is updated, clear 0 by hardware; the prescaler factor is updated only when the PVU bit is cleared to 0.
0	WKFLG	R/W	Interrupt status register Write 0 to clear to 0.

14.6.5 Counter register (LPTMR_CNT)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	CNT	R	CNT count value It needs to be read twice, and only when both readings are consistent will it be the current count value.

15 Watchdog timer (WDT)

15.1 Independent watchdog

Note: For independent watchdogs, there should be an interval of 160μs between two feedings.

15.1.1 Introduction

The independent watchdog consists of an 8-bit prescaler IWDT_PSC, 12-bit count-down counter, 12-bit reload register IWDT_RLR, key register IWDT_KEY and status register IWDT_SR.

The independent watchdog has an independent clock source, and even if the master clock fails, it is still valid.

The independent watchdog is applicable when an independent environment is required but the accuracy requirement is not high.

15.1.2 Main characteristics

(1) Configurable 12-bit down counter

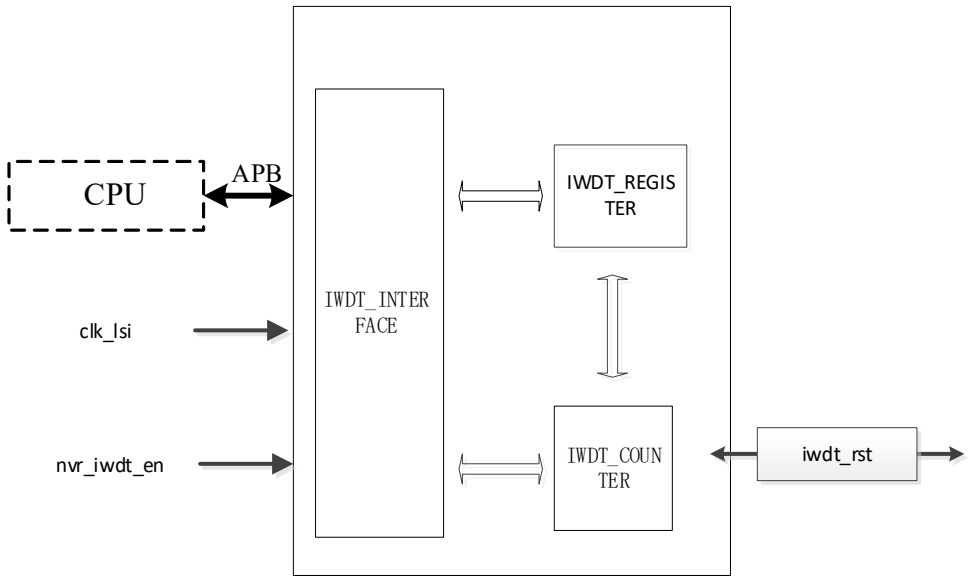
(2) Driven by a low-frequency independent clock (can work in stop modes)

(3) After the watchdog is activated, a reset occurs when the counter drops to 0x00

(4) In debug mode, the counter can be configured to stop counting

15.1.3 Structure block diagram

Figure 68 Block diagram of independent watchdog



15.1.4 Functional description

15.1.4.1 Key register

Write 0xCCCC in the key register to enable the independent watchdog, then the counter starts to count down from the reset value 0xFFF and when the counter counts to 0x000, a reset will be generated.

Write 0xAAAA in the key register, and the value of the reload register will be reloaded to the counter to prevent the watchdog from resetting.

Write 0x5555 to the key register to rewrite the value of the prescaler register and the reload register.

15.1.4.2 Register access protection

The prescaler register IWDT_PSC and reload register IWDT_RLR have the function of write protection. If you want to rewrite these two registers, you need to write 0x5555 in the key register. If you write other value in the key register, the protection of the register will be started again.

Write 0xAAAA to the key register and the write protection function will also be enabled.

The prescaler register and reload register can be observed through the status register.

15.1.4.3 Direction for use

Watch dog initialization steps

(1) Configure the RCC register to enable the IWDT clock;

(2) Configure the RCC register to enable the LSI clock;

(3) Enable IWDT by writing 0xCCCC to the IWDT_KEY register;

(4) Unlock register write protection by writing 0x5555 to the IWDT_KEY register;

(5) Configure the IWDT_PSC register;

(6) Wait for PVU to be 0;

(7) Configure the reload register IWDT_RLR;

(8) Wait for RVU to be 0;

(9) When feeding the dog, refresh the counter value as the reload value by writing 0xAAAA to the IWDT_KEY register.

15.2 Window watchdog

15.2.1 Introduction

The window watchdog contains a 7-bit free-running down counter, prescaler and control register WWDT_CR, configuration register WWDT_CFG and status register WWDT_SR.

The window watchdog clock comes from PCLK, and the counter clock is obtained by pre-frequency division of the CK counter clock (configured by the configuration register).

The window watchdog is applicable when precise timing is needed.

15.2.2 Main characteristics

(1) Configurable 7-bit down counter

(2) Configurable window, with a lower limit value of 0x3F, window period counting can prevent reset

(3) A reset occurs when the decrement counter is reduced to 0x3F, that is, when T6 jumps to 0

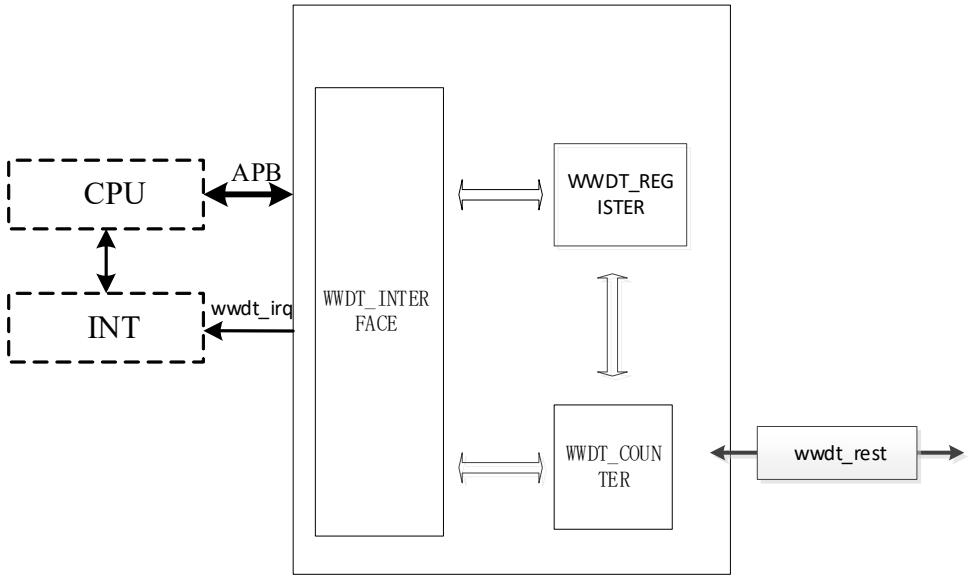
(4) When the counter value is greater than the window value, the counter (with a decrement counter configured) will be reset

(5) An early wake-up interrupt can be generated when the counter value equals 0x40

(6) In modulation mode, the counter will stop counting

15.2.3 Structure block diagram

Figure 69 Window Watchdog Structure Block Diagram



15.2.4 Functional description

Enable window watchdog timer, and the reset conditions are:

- When the counter drops from 0x40 to 0x3F, a reset will be occur.
- When the counter value is greater than the value stored in the window register, if the software reloads the counter, a reset will occur.

15.2.4.1 Enable the watchdog

After reset, the watchdog is always closed. Setting the WWDTEN bit in the WWDT_CR register to 1 enables the watchdog, and clearing it to 0 disables the watchdog. When the watchdog is in the off state, the counter stops counting and reverts to the default value.

15.2.4.2 Configuration Protection

Interrupt enable, prescaler, and window configuration can only be changed when the watchdog is disabled.

15.2.4.3 Window mode

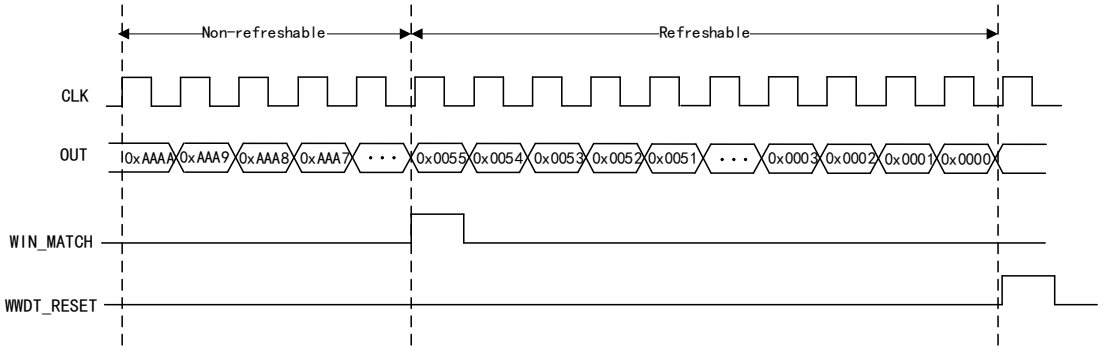
WINDOW bit stores the window value, and the feeding interval of the WWDT watchdog is affected by the WINDOW bit. If feeding occurs when the count value is greater than WINDOW, a reset will occur. Therefore, the feeding interval must be less than or equal to the WINDOW value.

15.2.4.4 Feed the dog

The WWDT feeding action is to directly load the count into the TOUT bit of the WWDT_CR register, and the TOUT restarts counting from the loaded value, as

shown in the figure where the window value is configured as 0x0055.

Figure 70 Timing Diagram



Interrupts

WWDT can be configured with an interrupt. When the count value reaches 0x40, an early wake-up interrupt flag is generated. When EWIEN is enabled, an interrupt is generated. The wake-up flag and interrupt can be cleared by writing 0 to the EWIFLG bit of the WWDT_SR register.

15.2.4.5 Clock

One input is PCLK, and two outputs are the reset WWDT_RST and the interrupt WWDT_IRQ. TOUT[6:0] represents the decrement counter value, and WINDOW[6:0] represents the upper limit value of the window. The counter value and the upper limit of the window can be configured by the user, and the lower limit of the window is 0x3F.

After being frequency-divided by the frequency divider, the PCLK serves as the counting clock of the decrement counter. When WWDTEN is enabled, WWDT begins to work. After starting work, the value of the configuration register WWDT_CFG cannot be changed.

15.2.4.6 Direction for use

Watch dog initialization steps

(1) Enable the WWDT clock;

(2) Configure the frequency division coefficient, window value and decrement counter value;

(3) Configure interrupt enable and NVIC;

(4) Timed loading of the WWDT_CR register to feed the dog must be done during the window period; otherwise, a reset will occur.

(5) When the counter reaches 0x40, an interrupt is generated, and the interrupt handling function is entered to reload the value and clear the interrupt flag.

15.3 IWDT register address mapping

Table 43 IWDT Register Mapping

Register name	Description	Offset address
IWDT_KEY	Key register	0x00
IWDT_PSC	Prescale register	0x04
IWDT_RLR	Counter reload register	0x08
IWDT_SR	Status register	0x0C

15.4 IWDT register functional description

15.4.1 Key register (IWDT_KEY)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	KEY	W	Allow Access IWDT Register Key Value Writing 0x5555 means enabled access to IWDT_PSC, IWDT_RLR registers. When the software writes 0xAAAA, it means to execute the reload counter, which requires at least 4 LSI_CLK writes to prevent the watchdog from resetting. Write 0xCCCC to enable the watchdog (the hardware watchdog is unrestricted by this command word). Once a watchdog is turned on, it cannot be turned off. Only by resetting can it be turned off.

15.4.2 Prescaler register (IWDT_PSC)

Offset address: 0x04

Reset value: 0x0000 0007

Field	Name	R/W	Description
31:3	Reserved		
2:0	PSC	R/W	Prescaler Factor Configure Support write protection function; when writing 0x5555 to the IWDT_KEY register, it is allowed to access the register; in the process of writing to this register, only when PVU=0 for IWDT_SR register, can the prescaler factor be changed; in the process of reading this register, only when PVU=0, can the read-out value of PSC register be valid. 000: PSC=4 001: PSC=8 010: PSC=16 011: PSC=32 100: PSC=64 101: PSC=128 110: PSC=256 111: PSC=256

15.4.3 Counter reload register (IWDT_RLR)

Offset address: 0x08

Reset value: 0x0000 0FFF

Field	Name	R/W	Description
31:12	Reserved		
11:0	RLR	R/W	Watchdog Counter Reload Value Setup It supports write protection function and defines the value loaded to the watchdog counter when 0xAAAA is written to IWDT_KEY register; in the process of writing this register, this register can be modified only when RVU=0. In the process of reading this register, only when RVU=0 in IWDT_SR register, can the read value be valid. The watchdog timeout cycle can be calculated by the reload value and clock prescaler value.

15.4.4 Status register (IWDT_SR)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:2	Reserved		
1	RVU	R	Watchdog Counter Reload Value Update Flag When the counter reload value is updated, set 1 by hardware; after the counter reload value is updated, clear 0 by hardware; the counter reload value is updated only when the RVU bit is cleared to 0.
0	PVU	R	Watchdog Prescaler Value Update Flag When the prescaler factor is updated, set 1 by hardware; after the prescaler factor is updated, clear 0 by hardware; the prescaler factor is updated only when the PVU bit is cleared to 0.

15.5 WWDT register address mapping

Table 44 WWDT Register Address Mapping

Register name	Description	Offset address
WWDT_CR	Control register	0x00
WWDT_CFG	Configuration register	0x04
WWDT_SR	Status register	0x08

15.6 WWDT register functional description

These peripheral registers can be operated by half word (16 bits) or word (32 bits).

15.6.1 Control register (WWDT_CR)

Offset address: 0x00

Reset value: 0x0000 007F

Field	Name	R/W	Description
31:8	Reserved		
7	WWDTEN	R/S	Window Watchdog Enable This bit is set to 1 by software and can be cleared by hardware only after reset. When WWDTEN=1, WWDT can generate a reset. 0: Disable 1: Enable
6:0	TOUT	R/W	Down counter Store the value of the watchdog counter, which decreases once every (4096*2PSC[1:0]) pclk cycle. A reset occurs when it is reduced from 0x40 to 0x3F.

15.6.2 Configuration register (WWDT_CFG)

Offset address: 0x04

Reset value: 0x0000 01FF

Field	Name	R/W	Description
31:10	Reserved		
9	EWIEN	R/W	Early Wakeup Interrupt Enable 0: Meaningless 1: An interrupt is generated when the counter value reaches 0x40; this interrupt can only be configured when WWDT is not enabled.
8:7	PSC	R/W	Timer Base Prescaler Factor Configure Divide the frequency on the basis of PCLK1/4096 00: No frequency division 01: 2 frequency division 10: 4 frequency division 11: 8 frequency division This prescaler can only be configured when WWDT is not enabled.
6:0	WINDOW	R/W	Window Value Setup This window value is 7 bits, which is used to compare with the down counter. This window value can only be configured when WWDT is not enabled.

15.6.3 Status register (WWDT_SR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	EWIFLG	RC_W0	Early Wakeup Interrupt Occur Flag 0: Not occurred 1: When the counter value reaches 0x40, set 1 by hardware; if the interrupt is not enabled, the bit will also be set to 1. It can be cleared by writing 0 by software Writing 1 to this bit is invalid.

16 Universal Synchronous/Asynchronous Transceiver (USART/UART)

16.1 Full Name and Abbreviation Description of Terms

Table 45 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Clear to Send	CTS
Request to Send	RTS
Most Significant Bit	MSB
Least Significant Bit	LSB
Guard	GRD
Overrun	OVR

16.2 Introduction

USART (universal synchronous/asynchronous transceiver) is a serial communication device that can flexibly exchange full-duplex and half-duplex data with external devices, and meets the requirements of external devices for industry standard NRZ asynchronous serial data format. USART also offers a wide range of baud rate options.

USART not only supports the standard asynchronous transceiver mode but also supports synchronous one-way communication and some other serial data exchange modes, such as LIN protocol mode and modem operation (CTS/RTS), it also supports multi-processor communication.

USART also supports DMA function to realize high-speed data communication.

16.3 Main characteristics

(1) Full-duplex asynchronous communication

(2) Single-line half-duplex communication

(3) NRZ standard format

(4) Characteristics of programmable serial port:
- Data bit: 7,8 or 9 bits
- Check bits: Even parity check, odd parity check, no check
- Support 1 and 2 stop bits

(5) Check control

- Transmit the check bit
- Check the received data

(6) Supports 8x/16x oversampling rate

(7) Programmable high or low priority

(8) Independent transmitter and receiver enable bit

(9) Can switch TX/RX pins

(10) Programmable baud rate generator, with the USART baud rate up to 8Mbits, UART baud rate up to 4Mbits

(11) Multiprocessor communication:
- If the address does not match, it will enter the mute mode
- Wake up from mute mode through idle bus detection or address flag detection

(12) Automatic baud rate detection

(13) Synchronous transmission mode

(14) Generation and detection of LIN break frame

(15) Receive frame error detection

(16) Hardware parity detection

(17) 1/16 bit noise detection

(18) Support hardware flow control and RS485 driver enable

(19) DMA can be used for continuous communication

(20) Support ModBus communication
- Timeout detection
- CR/LF character recognition

(21) Status flag bit:
- Transmission detection flag: The transmit register is empty, the receive register is not empty, and transmission is completed
- Error detection flag: Overrun error, noise error, parity error, frame error

(22) Multiple interrupt sources:
- The transmit register is empty
- Transmission completed
- CTS changed
- The receive register is not empty
- Parity check error
- LIN break detection
- Noise error

- Overrun error
- Frame error
- Failed to receive interrupt on time
- Character match

16.4 Structure block diagram

Figure 71 USART Structure Block Diagram

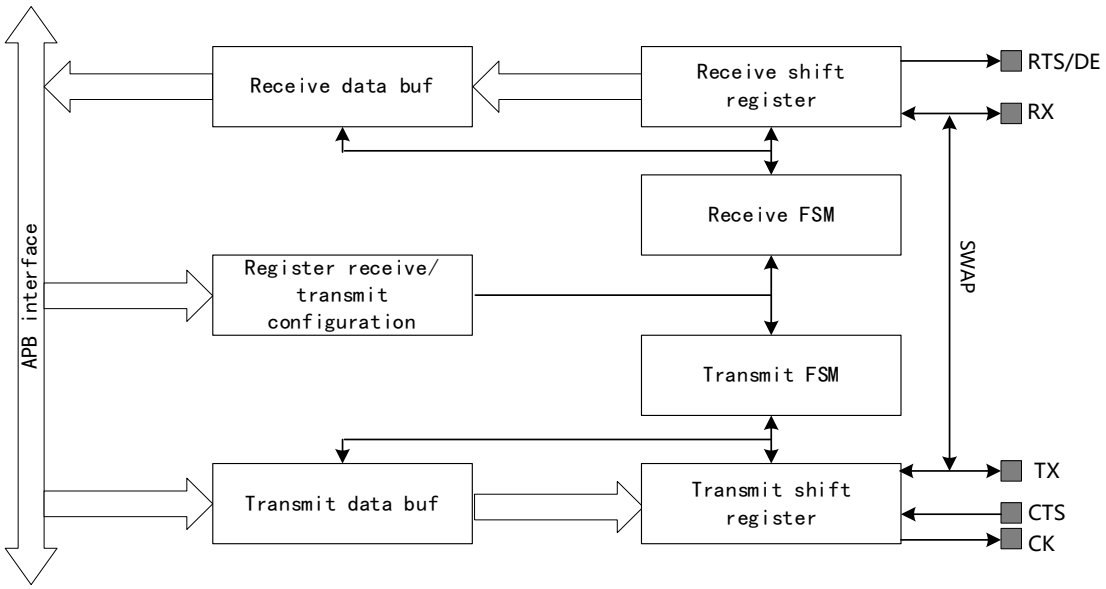
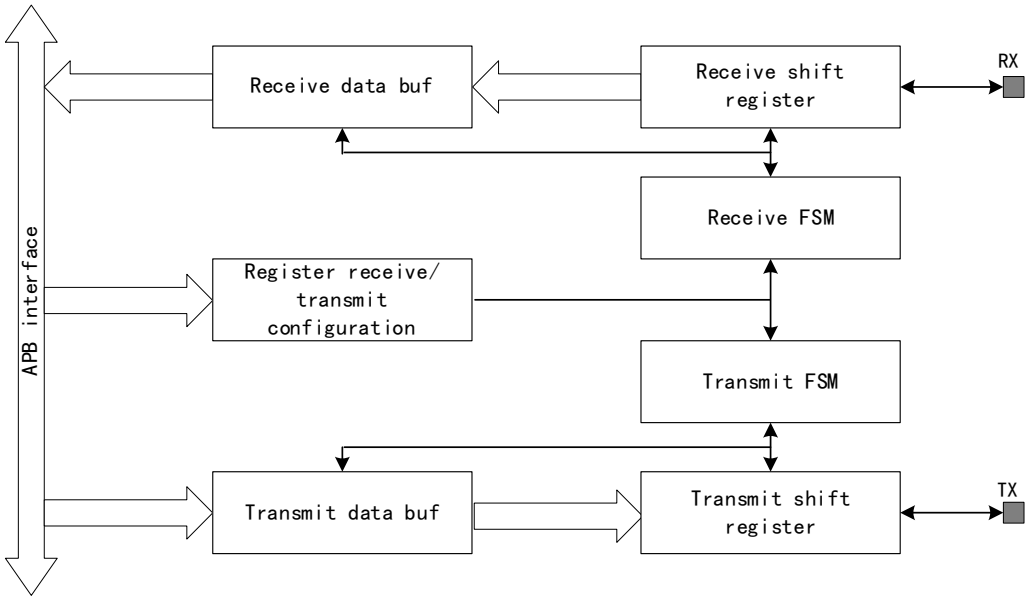


Figure 72 UART Structure Block Diagram



16.5 Functional description

Table 46 USART Pin Description

Pin	Type	Description
USART_RX	Input	Data receiving
USART_TX	Output I/O (single-line mode)	Data transmission When the transmitter is enabled and does not transmit data, the default is high
USART_CK	Output	Clock output
USART_nRTS	Input	Request to send in hardware flow control mode
USART_nCTS	Output	Clear to send in hardware flow control mode
USART_DE	Input	Drive enable activating external transmitter/receiver

16.5.1 Single-line half-duplex communication

HDSEL bit of USART_CR3 register determines whether to enter the single-line half-duplex mode.

When USART enters single-line half-duplex mode:

- The CLKEN bit and LINEN bit in the USART_CR2 register must be cleared to 0.
- RX pin is disabled.
- TX pin should be configured as open-drain output and connected with RX pin inside the chip.
- Transmitting data and receiving data can not be carried out at the same time. The data cannot be received before they are transmitted. To receive data, enable receiving can be turned on only after TCFLG bit of USART_SR register is set to 1.
- If there is data collision on the bus, software is required to manage the distributed communication process.

16.5.2 Frame format

The frame format of data frame is controlled by USART_CR1 register

- M bit controls the character length, which can be set to 7,8 or 9 bits
- The PCEN bit controls whether to enable the check bit
- The PSEL bit controls the check bit to determine if it is odd or even

Table 47 USART Frame Format

M bit	PCEN bit	USART data frame
00	0	Start bit+8-bit data+stop bit
00	1	Start bit+7-bit data+odd-even parity check bit+stop bit
01	0	Start bit+9-bit data+stop bit

M bit	PCEN bit	USART data frame
01	1	Start bit+8-bit data+odd-even parity check bit+stop bit
10	0	Start bit+7-bit data+stop bit
10	1	Start bit+6-bit data+parity check bit+stop bit

Note: Only USART supports a character length of 7 characters.

Configurable stop bit

2 different stop bits can be configured through STOP bit of USART_CR2 register.

- 1 stop bit: The default stop bit
- 2 stop bits: Used in normal mode, single-line mode and hardware flow control mode

Check bit

PSEL bit of USART_CR1 determines the parity check bit; when PSEL=0, it is even parity check, on the contrary, it is odd parity check.

- Even check: When the number of frame data and check bit '1' is even, the even check bit is 0; otherwise it is 1.
- Odd check: When the number of frame data and check bit '1' is even, the odd check bit is 1; otherwise it is 0.
- Check generation: When transmitting data, set PCEN bit of USART_CR1 register, and the check bit will replace the MSB bit of the data and be transmitted.
- Parity check:
  - If the parity check fails, the PEFLG flag bit of USART_SR register will be set.
  - If the check control is enabled, corresponding interrupt will be triggered.

16.5.3 Transmitter

When TEN bit of the register USART_CR1 is set, the transmit shift register will output data through TX pin and the corresponding clock pulses will be output through CK pin.

16.5.3.1 Character transmission

During transmission period of USART, the least significant bit of the data will be moved out by TX pin first. In this mode, USART_DR register has a buffer between the internal bus and the transmit shift register.

A data frame is composed of the start bit, character and stop bit, so there is a low-level start bit in front of each character; then there is a high-level stop bit whose number is configurable.

Transmission configuration steps

(1) Set UEN bit of USART_CR1 register to enable USART

(2) Decide the word length by setting M bit of USART_CR1 register

(3) Decide the number of stop bits by setting STOP bit of USART_CR2 register

(4) If multi-buffer communication is selected, DMA should be enabled in USART_CR3 register

(5) Set the baud rate of communication in USART_BRR register

(6) Enable TEN bit in USART_CR1 register, and transmit an idle frame

(7) Write data to USART_DR register (if DMA is not enabled, repeat step 7 for each byte to be transmitted)

(8) Wait for TCFLG bit of USART_SR register to be set to 1, indicating transmission completion

Note: TEN bit cannot be reset during data transmission; otherwise, the data on TX pin will be destroyed, which is because if the baud rate generator stops counting, the data being transmitted will be lost.

16.5.3.2 Single-byte communication

TXBEFLG bit can be cleared to 0 by writing to USART_DR register. When the TXBEFLG bit is set by hardware, the shift register will receive the data transferred from the data transmit register, then the data will be transmitted, and the data transmit register will be cleared. The next data can be written in the data register without overwriting the previous data.

(1) If TXBEIEN in USART_CR1 register is set to 1, an interrupt will be generated.

(2) If USART is in the state of transmitting data, write to the data register to save the data to the DATA register, and transfer the data to the shift register at the end of the current data transmission.

(3) If USART is in idle state, write to the data register, put the data into the shift register, start transmitting data, and set TXBEFLG bit to 1.

(4) When a data transmission is completed and TXBEFLG bit is set, TCFLG bit will be set to 1; at this time if TCIEN bit in USART_CR1 register is set to 1, an interrupt will be generated.

(5) After the last data is written to the USART_TXDATA register, before entering the low-power mode or before disabling the USART module, wait to set TCFLG to 1.

16.5.3.3 Break frame

The break frames are regarded to all receive '0' within one frame period. One break frame can be transmitted by setting the SBK bit of USART_CR1 register, and the length of the break frame is determined by the M bit of USART_CR1 register. If the SBK bit is set, after completion of transmission of current data, the TX line will transmit a break frame, and after completion of transmission of break frame, this bit will be reset. At the end of the break frame, the transmitter inserts one or two stop bits to respond to the start bit.

Note: If the SBK bit is reset before transmission of the break frame starts, the break frame will not be transmitted. To transmit two consecutive break frames, the SBK bit should be set after the stop bit of the previous disconnection symbol.

16.5.3.4 Idle frame

The idle frame is regarded as a complete data frame composed entirely of '1', followed by the start bit of the next frame containing the data. Set TEN bit of USART_CR1 register to 1 and one idle frame can be transmitted before the first data frame.

16.5.4 Receiver

16.5.4.1 Character receiving

During receiving period of USART, RX pin will first introduce the least significant bit of the data. In this mode, USART_DR register has a buffer between the internal bus and the receive shift register. The data is transmitted to the buffer bit by bit. When fully receiving the data, the corresponding receive register is not empty, then the user can read USART_DR.

Receiving configuration steps

(1) Set UEN bit of USART_CR1 register to enable USART

(2) Decide the word length by setting M bit of USART_CR1 register

(3) Decide the number of stop bits by setting STOP bit of USART_CR2 register

(4) If multi-buffer communication is selected, DMA should be enabled in USART_CR3 register

(5) Set the baud rate of communication in USART_BRR register

(6) Set REN bit of USART_CR1 to enable receiving

Note:

(1) REN bit cannot be reset during data receiving period; otherwise, the bytes being received will be lost.

(2) In the process of the receiver receiving a data frame, if an overrun error, noise error or frame error is detected, the error flag will be set to 1.

(3) When data is transferred from the shift register to USART_DR register, the RXBNEFLG bit of USART_SR will be set by hardware.

(4) An interrupt will be generated if RXBNEIEN bit is set.

(5) In single-buffer mode, the RXBNEFLG bit can be cleared by reading USART_DR register by software or by writing 0.

(6) In multi-buffer mode, after each byte is received, the RXBNEFLG bit of USART_SR register will be set to 1, and can be cleared to 0 by reading the data register by DMA.

16.5.4.2 Break frame

When the receiver receives a break frame, USART will handle it as receiving a frame error.

16.5.4.3 Idle frame

When the receiver receives an idle frame, USART will handle it as receiving an ordinary data frame; if IDLEIEN bit of USART_CR1 is set, an interrupt will be generated.

16.5.4.4 Oversampling rate

OSMCFG bit of USART_CR1 register determines the oversampling rate.
If the oversampling rate is 8 times of the baud rate, the speed is higher, but the clock tolerance is smaller. If it is 16 times, the speed is lower, but the clock tolerance is bigger.

Note: Only USART supports an 8x sampling rate.

16.5.4.5 Overrun error

When RXBNEFLG bit of USART_SR register is set to 1 and a new character is received at the same time, an overrun error will be caused. Only after REN is reset, can the data be transferred from the shift register to DATA register. After the byte is received, the RXBNEFLG bit will be set to 1. This bit needs to be reset before receiving the next data or servicing the previous DMA request; otherwise, an overrun error will occur.

When an overrun error occurs

- The OREFLG bit of USART_SR is set to 1
- The data in DATA register will not be lost
- The data in the shift register previously received will be overwritten, but the data received later will not be saved
- If RXBNEIEN bit or EIEN bit of USART_CR1 is set, an interrupt will be generated
- When OREFLG bit is set, it means there are data lost. There are two possibilities:

- When RXBNEFLG=1, the previous valid data is still on DATA register, and can be read
- When RXBNEFLG=0, there is no valid data in DATA register

- The OREFLG bit can be reset by reading USART_SR and USART_DR registers

16.5.4.6 Noise error

When noise is detected in the receiving process of the receiver:

- Set NEFLG flag on the rising edge of RXBNEFLG bit of USART_SR register
- Invalid data is transmitted from the shift register to USART_DR register
- In single- byte communication, no interrupt will be generated, but in multi-buffer communication, an interrupt will be generated by setting the EIEN bit of USART_CR3 register

16.5.4.7 Frame error

If the stop bit is not received and recognized at the expected receiving time due to excessive noise or lack of synchronization, a frame error will be detected.

When a frame error is detected by the receiver in the receiving process:

- Set the FEFLG bit of USART_SR register
- Invalid data is transmitted from the shift register to USART_DR register
- In single- byte communication, no interrupt will be generated, but in multi-buffer communication, an interrupt will be generated by setting the EIEN bit of USART_CR3 register

16.5.5 TX and RX pin swap

The TX/RX pin functions can be swapped by enabling the SWAPEN bit in the USART_CTLR1 register. After enabling, the functions of the TX and RX pins are swapped

16.5.6 Baud rate generator

The baud rates of the receiver and transmitter in the integer and decimal registers of the USARTDIV should be set to be the same.

$$TX/RX\ baud = \frac{fCK}{8 \times (2 - \mathrm{OSMCFG}) \times USARTDIV}$$

This f_{CK} refers to the clock for peripheral devices.

After writing to USART_BRR, the baud rate counter will be replaced by the new value of the baud rate register. Therefore, the value of the baud rate register cannot be changed during communication. USART must be enabled after the system clock is enabled in the clock control unit.

Table 48 Baud Rate Error Calculation (USART_PCLK = 64MHz)

Baud rate (bps)	Actual Value	BRR (decimal)	Error rate
2400	2399.97	26667	0.0012 %
9600	9599.52	6667	0.005 %
19200	19201.92	3333	0.01 %
57600	57605.76	1111	0.01 %
115200	115107.9	556	0.08 %
230400	230215.83	278	0.08 %
460800	457142.86	139	0.08 %
921600	927536	69	0.6 %
2250000	2285714.3	28	1.6 %

Note: When the chip's operating voltage is lower than 2V, the HSI as the system clock will have a 10% deviation, which may cause errors in USART communication.

16.5.7 Automatic baud rate detection

When a character is received, USART can detect and automatically set the value of the USART_BRR register. Automatic baud rate detection functions when the communication speed of the system is unknown, the clock source with low precision is used, or the clock deviation is not measured to obtain the correct bit rate. The clock source must be compatible with the expected communication speed.

A non-zero baud rate must be written for initialization; confirm the character content, and then enable automatic baud rate detection. The character content can be 0x55 or 0x155. First detect the baud rate of the start bit, then detect the baud rate at the end of Bit 0 data, and finally detect the baud rate at the end of Bit 6 data. Take samples of Bit 0, bits 1 to 6 and Bit 6 respectively. During the automatic baud rate detection process, the received character frame will be checked. If the final check result is not the expected value of 0x55, an automatic baud rate detection error will be generated; meanwhile, if the updated baud rate value is not between 16 and 65535(at 16 times oversampling), nor within 8 and 65,536 clock cycles (at 8 times oversampling), an automatic baud rate detection error will also be generated.

ABEN bit of USART_CR2 register determines whether to enable automatic baud rate detection. After the automatic baud rate detection is enabled, wait for the first character on RX line. After detection, the ABCFLG flag bit of USART_SR register will be set.

Note:

(1) If the line noise is too loud, correct baud rate cannot be guaranteed. In this case, the BR value may be damaged and the ABERRFLG flag bit will be set. This situation can also happen if the communication speed and automatic baud rate detection are not compatible.

(2) RXBNEFLG interrupt will be generated after detection.

(3) At any time, automatic baud rate detection may be restarted by resetting the ABCFLG flag (writing a 0).

(4) USART cannot be disabled during automatic baud rate detection; otherwise, the BR value may be damaged.

16.5.8 Multi-processor Communication(Only supported by USART)

In multi-processor communication, multiple USARTs are connected to form a network. In this network, two devices communicate with each other, and the mute mode can be enabled for other devices not participating in the communication in order to reduce the burden of USART. In mute mode, the LINEN bit of USART_CR2 register, and the HDSEL bit of USART_CR3 register are cleared, any receive state bit will not be set, and all receive interrupts will be disabled. When mute mode is enabled, there are two ways to exit the mute mode:

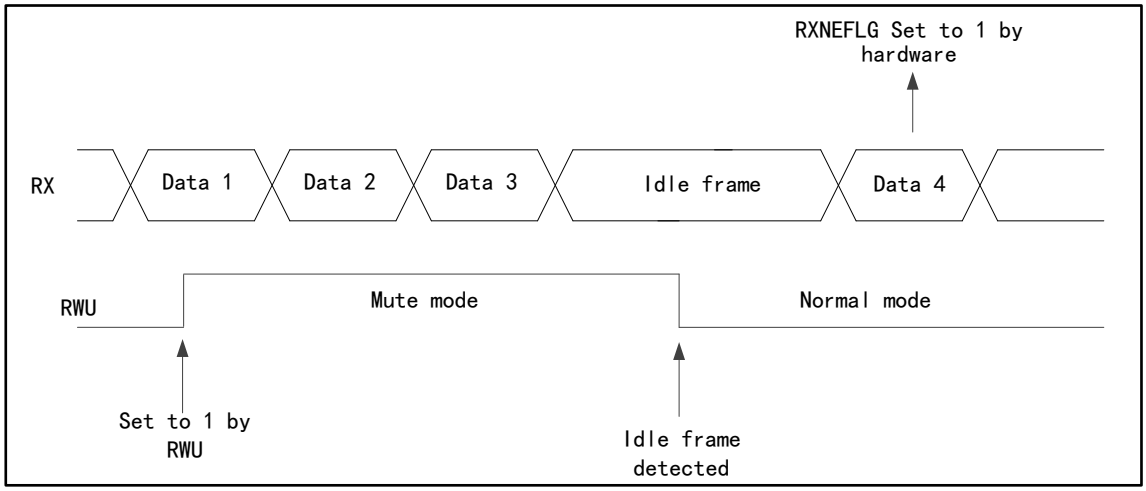
(1) WKSEL bit is cleared and the bus is idle to exit the mute mode.

(2) WKSEL bit is set and after receiving the address flag, it can exit the mute mode.

Idle bus detection (WKSEL=0)

When RXWFMUTE is set to 1, USART enters the mute mode, and it can be waken up from the mute mode when an idle frame is detected, meanwhile, the RXWFMUTE bit will be cleared by the hardware. RXWFMUTE can also be cleared by software.

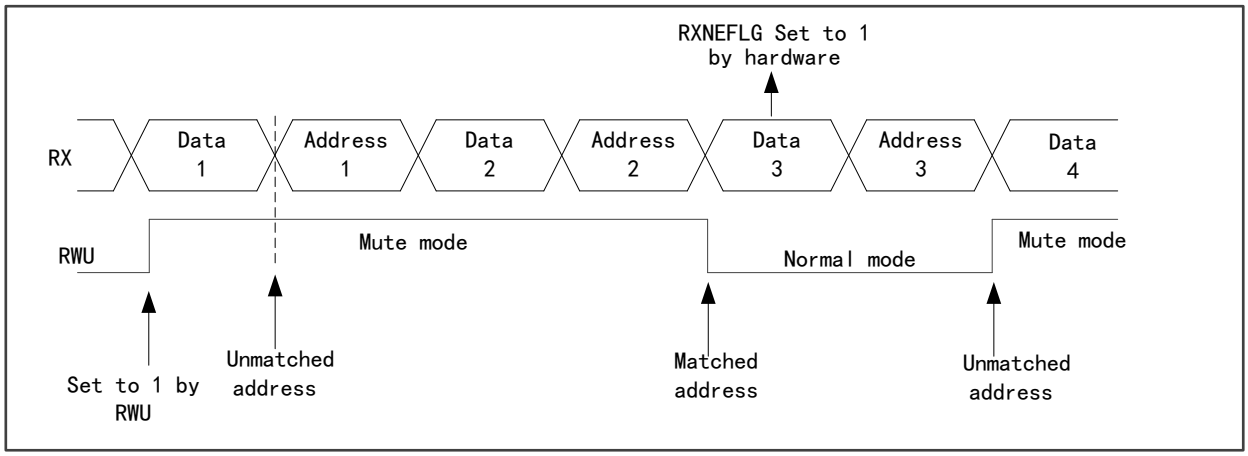
Figure 73 Idle Bus Exit Mute Mode



Address flag detection (WKSEL=1)

If the address flag bit is 1, this byte is regarded as the address. The storage address of lower four bits of the address bytes will first be compared with its own address when the receiver receives the address byte. If the addresses do not match, the receiver will enter the mute mode. If the addresses match, the receiver will wake up from the mute mode and be ready to receive the next byte. If the address byte is received again after exiting the mute mode, but the address does not match its own address, the receiver will enter the mute mode again.

Figure 74 Address Flag Exit Mute Mode



16.5.9 Synchronous Mode (Only supported by USART)

The synchronous mode supports full duplex synchronous serial communication in master mode, and has one more signal line USART_CK which can output synchronous clock than the asynchronous mode.
CLKEN bit of USART_CR2 register decides whether to enter the synchronous mode.
When USART enters the synchronous mode:

- The LINEN bit of USART_CR2 register, and HDSEL bits of USART_CR3 registers must be cleared
- The start bit and stop bit of the data frame have no clock output
- Whether the last data bit of the data frame generates USART_CK clock is determined by LBCL bit of the register USART_CR2
- The clock polarity of USART_CK is decided by CPOL bit of USART_CR2 register
- The phase of USART_CK is decided by the CPHA bit of USART_CR2
- The external CK clock cannot be activated when the bus is idle or the frame is disconnected

Figure 75 USART Synchronous Transmission Example

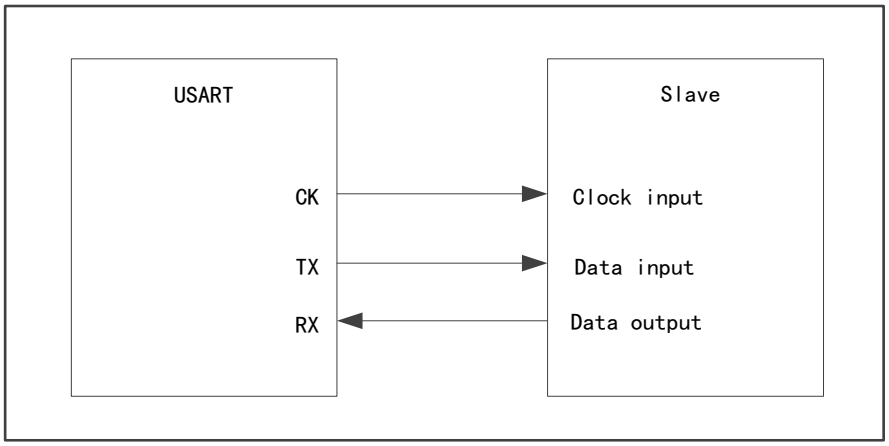


Figure 76 USART Synchronous Transmission Timing Diagram (M=10)

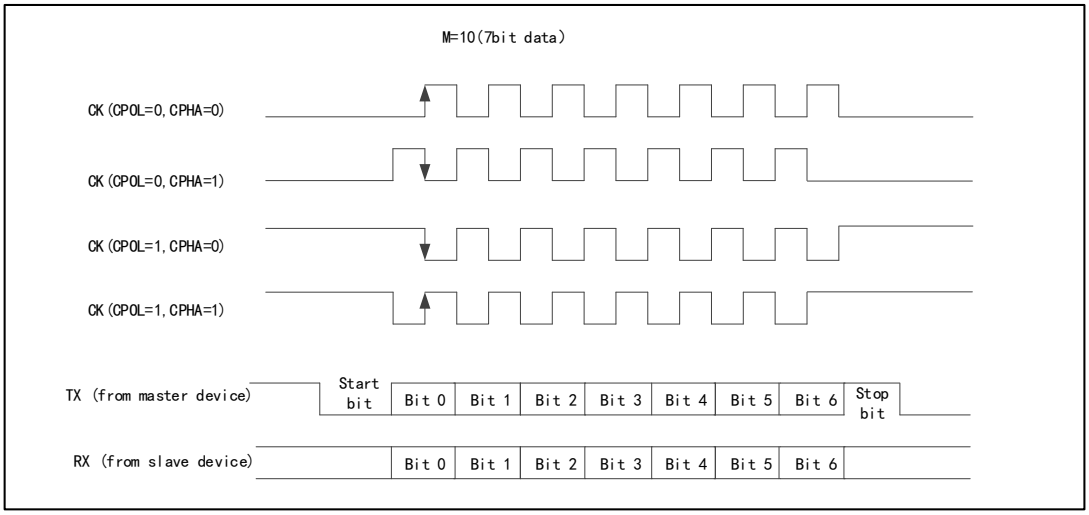


Figure 77 USART Synchronous Transmission Timing Diagram (M=00)

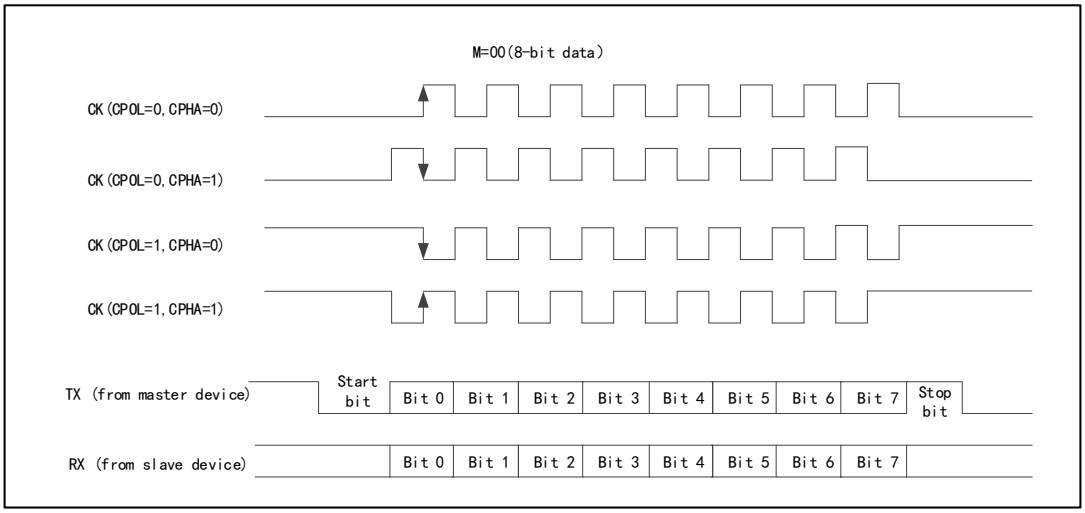
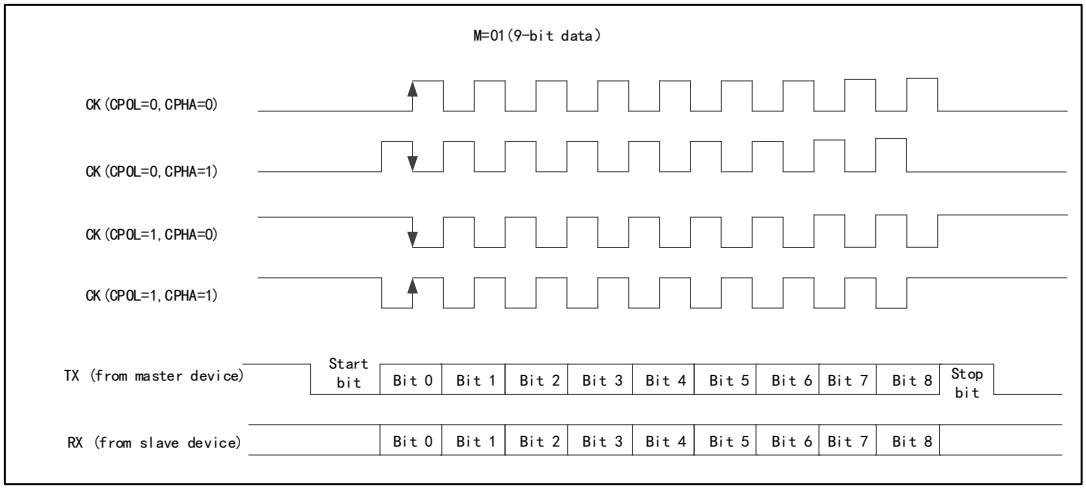


Figure 78 USART Synchronous Transmission Timing Diagram (M=01)



16.5.10 LIN mode

LINEN bit of USART_CR2 register decides whether to enter LIN mode.

When entering LIN mode:

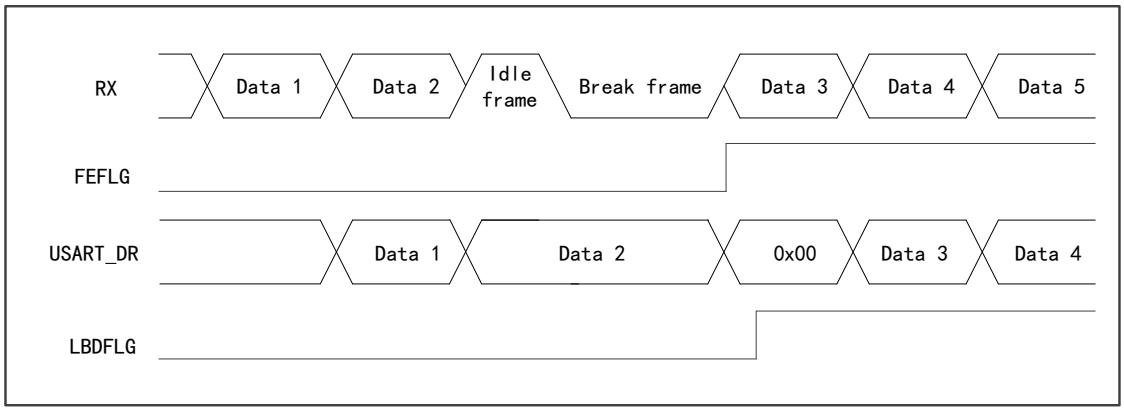
- Each data frame includes 8 data bits and 1 stop bit
- The OSMCFG bit of the register of USART_CR1, CLKEN bit and of USART_CR2 register and HDSEL bit and STOP bit of USART_CR3 register need to be cleared to 0.

In LIN master mode, USART can generate break frame, and the detection length of break frame can be set to 10 or 11 bits through LBDL bit of USART_CR2. The break frame detection circuit is independent of USART receiver, and whether in idle state or in data transmission state, RX pin can detect the break frame, and the LBDFLG bit of USART_SR register is set to 1; at this time, if the LBDIEN bit of USART_CR2 is enabled, an interrupt will be generated.

Detection of break frame in idle state

In idle state, if a break frame is detected on RX pin, the receiver will receive a data frame of 0 and generate FEFLG error.

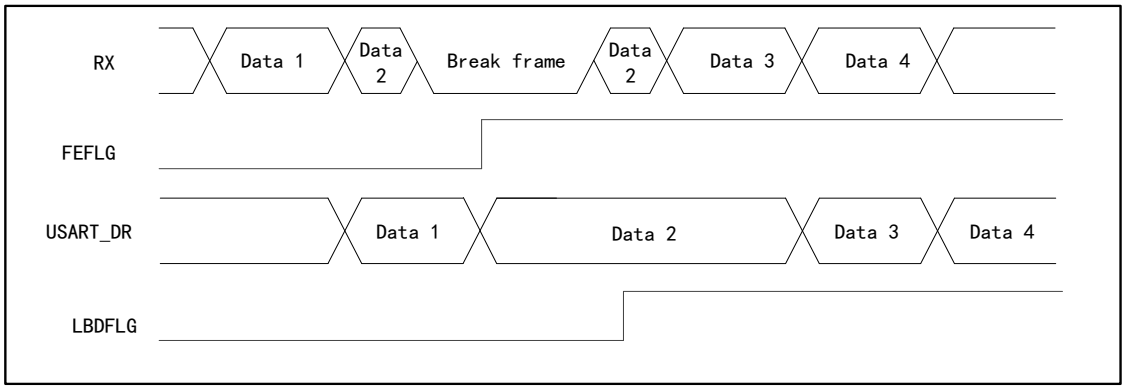
Figure 79 Break Frame Detection in Idle State



Detection of break frame in data transmission state

In the process of data transmission, if the RX pin detects the break frame, the currently transmitted data frame will generate FEFLG error.

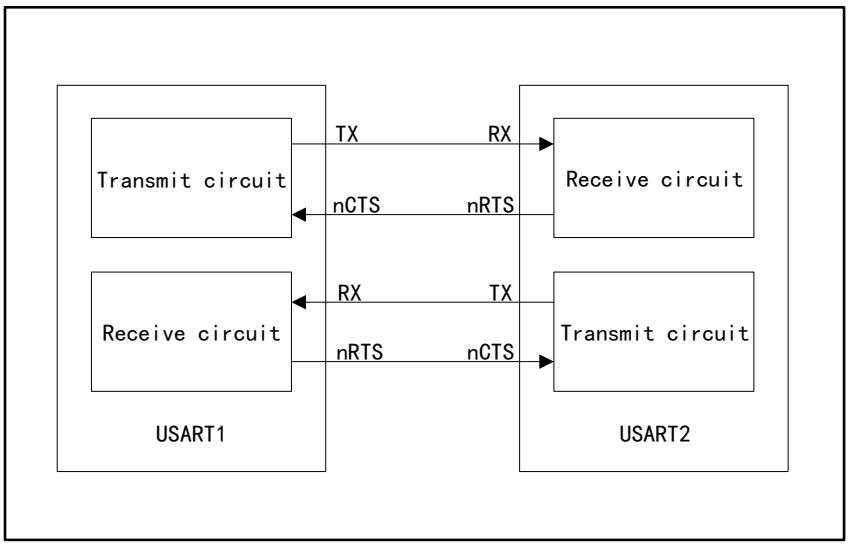
Figure 80 Break Frame Detection in Data Transmission State



16.5.11 Hardware Flow Control and RS485 Drive Enable (Only supported by USART)

The function of hardware flow control is to control the serial data flow between two devices through nCTS pin and nRTS pin.

Figure 81 Hardware Flow Control between Two USARTs



CTS flow control

CTSEN bit of USART_CR3 register determines whether CTS flow control is enabled. If CTS flow control is enabled, the transmitter will detect whether the data frame of nCTS pin can be transmitted. If TXBEFLG bit=0 for USART_SR register and nCTS is pulled to low level, the data frame can be transmitted. If nCTS becomes high during transmission, the transmitter will stop transmitting after the current data frame is transmitted.

RTS flow control

RTSEN bit of USART_CR3 register determines whether RTS flow control is enabled. If RTS flow control is enabled, when the receiver receives data, nRTS will be pulled to low level. When a data frame is received, nRTS will becomes high to inform the transmitter to stop transmitting data frame.

RS485 driver enable

DEMEN bit of USART_CR3 register determines whether to turn on the driver enable function, and this function can allow DE signal to turn on the control terminal of the external transceiver.
Lead time: The time interval between the driver enable signal and the start bit of the first byte. Controlled by DEAT[4:0] of USART_CR1 controller.
Lag time: The time interval between the stop bit of the last byte and the release DE signal. Controlled by DEDT[4:0] of USART_CR1 register.

16.5.12 DMA multi-buffer communication

To reduce the burden of processors, USART can access the data buffer in DMA mode.

Transmission in DMA mode

The DMAT bit of USART_CR3 register determines whether to transmit in DMA mode. When transmitting by DMA, the data in the designated SRAM will be transmitted to the buffer by DMA.

Configuration steps of transmission by DMA:

(1) Clear the TCFLG flag bit of USART_SR register to 0

(2) Set the address of SRAM memory storing data as DMA source address

(3) Set the address of USART_DR register as DMA destination address

(4) Set the number of data bytes to be transmitted

(5) Set channel priority

(6) Set interrupt enable

(7) Enable DMA channel

(8) Wait for TCFLG bit of USART_SR register to be set to 1, indicating transmission completion

Receive by DMA

The DMAR bit of USART_CR3 register determines whether to receive by DMA. When receiving by DMA, every time one byte is received, the data in the receive buffer will be transmitted to the designated SRAM area by DMA.

Configuration steps of receiving by DMA:

(1) Set the address of USART_DR register as DMA source address

(2) Set the address of SRAM memory storing data as DMA destination address

(3) Set the number of data bytes to be transmitted

(4) Set channel priority

(5) Set interrupt enable

(6) Enable DMA channel

16.5.13 Modbus Communication (Only supported by USART)

USART supports ModBus/RTU and ModBus/ASCII protocols, and ModBus/RTU is a half duplex block transmission protocol. Control part of the protocol can be realized only in software. USART supports end of block detection, not requiring software or other condition.

16.5.13.1 ModBus/RTU

This function can be realized through programmable timeout function. In this mode, the end of one block is regarded as one free line with the length greater than two characters. RXTOIEN bit and RXTODEN bit of USART_CR1 register control the timeout function and corresponding interrupts. Write a timeout number to USART_RXTOR register, and when the idle state of the receiving line reaches this length, an interrupt will be generated, indicating the completion of block receiving.

16.5.13.2 Modbus/ASCII

In this mode, the end of one block is identified by one specific (CR/LF) character sequence. USART uses character matching function to manage this mechanism. Program ASCII code of LF in ADDR[7:0] field and activate this character to match the interrupt (CMIEN=1). When a LF character is received, the software will be informed to check CR/LF in DMA buffer.

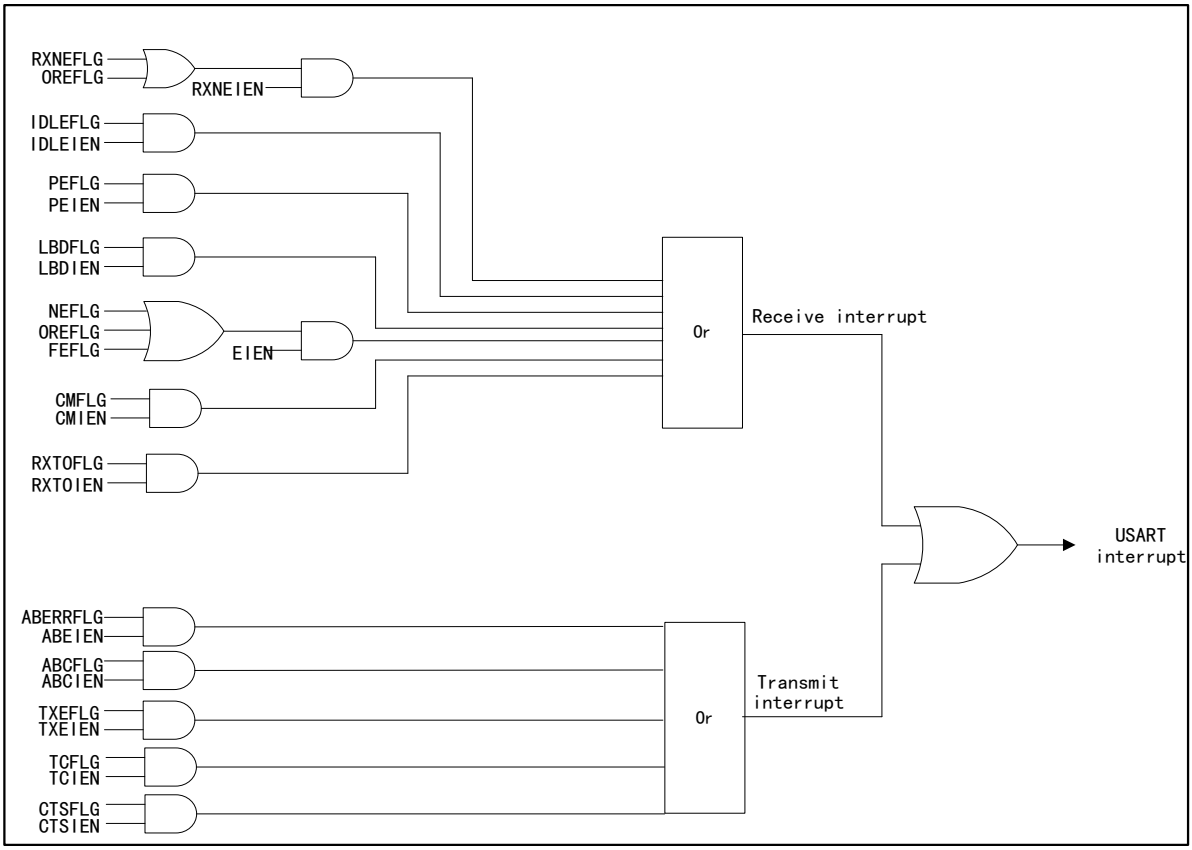
16.5.14 Interrupt request

Table 49 USART Interrupt Request

Interrupt event		Event flag bit	Enable bit
The receive register is not empty		RXBNEFLG	RXBNEIEN
Overrun error		OREFLG	
Idle line is detected		IDLEFLG	IDLEIEN
Parity check error		PEFLG	PEIEN
LIN break error		LBDFLG	LBDIEN
Receiving error in DMA mode	Noise error	NEFLG	EIEN
	Overrun error	OREFLG	
	Frame error	FEFLG	
Matching character		CMFLG	CMIEN
Error of failing to receive on time		RXTOFLG	RXTOIEN
Data transmit register is empty		TXBEFLG	TXBEIEN
Transmission completed		TCFLG	TCIEN
Automatic baud rate error flag		ABERRFLG	ABEIEN
Automatic baud rate completion flag		ABCFLG	ABCIEN
CTS flag		CTSFLG	CTSIEN

All interrupt requests of USART are connected to the same interrupt controller, and the interrupt requests have logical or relation before they are transmitted to the interrupt controller.

Figure 82 USART Interrupt Mapping



16.5.15 Comparison of supported functions between USART and UART

Table 50 Comparison of supported functions between USART and UART

USART mode	USART	UART
Half duplex (single-line mode)	√	√
Mode supporting automatic baud rate detection	√	√
Multiprocessor communication	√	-
Wake up from mute mode	√	-
The maximum baud rate at 16 times oversampling (Mbit/s)	4	4
The maximum baud rate at 8 times oversampling (Mbit/s)	8	-
Synchronous mode	√	-
LIN mode	√	√
Hardware flow control	√	-
RS485 driver enable	√	-
Multi-buffer communication (DMA)	√	√
Receiving timeout interrupt	√	√
Modbus communication	√	-

Note:

(1) " √ " means this function is supported, while "—" means that this function is not supported.

16.6 Register address mapping

Table 51 USART Register Address Mapping

Register name	Description	Offset address
USART_SR	Status register	0x00
USART_DR	Data register	0x04
USART_BRR	Baud rate register	0x08
USART_CR1	Control register 1	0x0C
USART_CR2	Control register 2	0x10
USART_CR3	Control register 3	0x14
USART_RXTOR	Receive timeout register	0x1C

16.7 Register functional description

16.7.1 Status register (USART_SR)

Offset address: 0x00

Reset value: 0x0000 00C0

Field	Name	R/W	Description
31:14	Reserved		
13	CMFLG	R	Character Match Flag 0: No character matches 1: There is matching character When the received character matches the value set by ADDR[7:0], the hardware sets it to 1. The software clears the 0. Writing 0 to this bit can clear this flag. If the CMIEN bit in the USART_CR1 register is 1, an interrupt request will be generated. Note: This bit is only applicable to the USART module.
12	ABERRFLG	RC_W0C	Auto Baud Rate Detection Error Flag 0: No automatic baud rate detection error 1: Automatic baud rate detection error (baud rate exceeding the range of 16~65535 or receiving data error) Set to 1 by hardware; cleared to 0 by software (writing 0 to this bit).
11	ABCFLG	RC_W0C	Auto Baud Rate Detection Flag 0: Automatic baud rate detection is not completed 1: Automatic baud rate detection is completed Set to 1 by hardware; cleared to 0 by software (writing 0 to this bit).

Field	Name	R/W	Description
10	RXTOFLG	RC_W0C	Receiver Timeout Flag 0: Not timed out 1: Timed out If no start bit is detected within the duration set by the RXTO bit, it is set to 1 by hardware; cleared to 0 by software (writing 0 to this bit).
9	CTSFLG	RC_W0C	CTS Change Flag 0: No change on nCTS state line 1: There is change on nCTS state line If the CTSEN bit is set, when the nCTS input is switched, it is set to 1 by the hardware. Reset it to zero by the software. If CTSIEN in USART_CR3 is' 1 ', an interrupt is generated. Note: This bit is only applicable to the USART module.
8	LBDFLG	RC_W0C	LIN Break Detected Flag 0: LIN break is not detected 1: LIN break is detected Set to 1 by hardware when a LIN break is detected; cleared to 0 by software (writing 0 to this bit).
7	TXEFLG	R	Transmit Data Register Empty Flag 0: The transmit data register is not empty 1: The transmit data register is empty Set to 1 by hardware when the shift register receives data transferred from the transmit data register; cleared to 0 by software by writing to the USART_DR register.
6	TCFLG	RC_W0C	Transmit Data Complete Flag 0: Transmitting data is not completed 1: Transmitting data is completed Set to 1 by hardware when the last frame of data transmission is completed and TXBEFLG is set; cleared to 0 by software by first reading the USART_SR register and then writing to the USART_DR register, or by writing 0 to this bit. This clearing procedure is only recommended in multi-buffer communication.
5	RXNEFLG	RC_W0C	Receive Data Register Not Empty Flag 0: The receive data register is empty 1: The receive data register is not empty Set to 1 by hardware when the data register receives data transferred from the receive shift register; cleared to 0 by software by reading the USART_DR register or writing 0 to this bit. This clearing procedure is only recommended in multi-buffer communication.
4	IDLEFLG	R	IDLE Line Detected Flag 0: Idle bus is not detected 1: Idle bus is detected Set to 1 by hardware when an idle bus is detected; cleared to 0 by first reading the USART_SR register and then reading the USART_DR register. This bit will not be set high again until RXBNEFLG is set.
3	OREFLG	R	Overrun Error Occur Flag 0: No overrun error

Field	Name	R/W	Description
			1: Overrun error is detected Set to 1 by hardware when RXBNEFLG is set and the data in the shift register is to be transferred to the receive register; cleared to 0 by first reading the USART_SR register and then reading the USART_DR register.
2	NEFLG	R	Noise Error Occur Flag 0: No noise 1: Noise is detected Set to 1 by hardware when a noise error occurs; cleared to 0 by first reading the USART_SR register and then reading the USART_DR register. This bit does not generate an interrupt because it appears together with RXNE, and the hardware generates an interrupt when the RXNE flag is set. In multi-buffer communication mode, if the EIEN bit is set, an interrupt is generated when the NEFLG flag is set.
1	FEFLG	R	Frame Error Occur Flag 0: No frame error 1: Frame error or break symbol is detected Set to 1 by hardware when a synchronization error, excessive noise, or break character occurs; cleared to 0 by first reading the USART_SR register and then reading the USART_DR register. This bit does not generate an interrupt because it appears together with RXNE, and the hardware generates an interrupt when the RXNE flag is set. If the currently transmitted data generates both a frame error and an overrun error, the hardware will still continue the transmission of this data and only set the OREFLG flag bit. In multi-buffer communication mode, if the EIEN bit is set, an interrupt is generated when the FEFLG flag is set.
0	PEFLG	R	Parity Error Occur Flag 0: No error 1: Parity error is detected Set to 1 by hardware when a parity error occurs in receive mode; cleared to 0 by first reading the USART_SR register and then reading the USART_DR register. Software must wait for the RXNEFLG flag bit to be set to 1 before clearing PEFLG.

16.7.2 Data register (USART_DR)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:9	Reserved		
8:0	DATA	R/W	Data Value Setup Transmit or receive the data value; read data when receiving data, and write data to the register when transmitting data. When the parity bit is enabled, for 9 data bits, the 8 bit of DATA is parity bit; for 8 data bits, the 7 bit of DATA is parity bit.

Field	Name	R/W	Description
			Note: This register only supports 32-bit access.

16.7.3 Baud rate register (USART_BRR)

This register can be set only when USART is not enabled. This bit may be reset by hardware during automatic baud rate detection.

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:4	IBR	R/W	Integer of USART Baud Rate Divider Factor The integral part of USART baud rate divider factor is determined by these 12 bits. If TEN or REN is disabled respectively, the baud rate counter stops counting.
3:0	FBR	R/W	Fraction of USART Baud Rate Divider Factor The decimal part of USART baud rate divider factor is determined by these four bits. If TEN or REN is disabled respectively, the baud rate counter stops counting.

16.7.4 Control register 1 (USART_CR1)

Offset address: 0x0C

Reset value: 0x0000 0000

Applicable to USART:

Field	Name	R/W	Description
31:30	Reserved		
29	CMIEN	R/W	Character Match Interrupt Enable 0: Disable 1: Generate an interrupt when CMFLG is set Set 1 or clear 0 by software.
28	SWAPEN	R/W	Swap TX/RX Pins Function Enable 0: Use according to standard allocation 1: The functions of TX and RX pins can be exchanged for use Set or cleared to 0 by software. This bit can be set only when USART is not enabled.
27	RXTOIEN	R/W	Receiver Timeout Interrupt Enable 0: Disable 1: Generate an interrupt when RXTOFLG is set Set or cleared to 0 by software.
26	RXTODEN	R/W	Receive Timeout Detection Function Enable 0: Disable 1: Enable Set or cleared to 0 by software. Set this bit, and when it is found that the RX line is idle for the length of time configured by RXTO register, the RXTOFLG bit will be set by hardware.
25:21	DEAT[4:0]	R/W	Driver Lead Time Enable This bit field is the time interval between DE signal and the first start

Field	Name	R/W	Description
			bit during transmission. Its unit is sampling time, determined by oversampling rate. This bit field can be set only when USART is not enabled.
20:16	DEDT[4:0]	R/W	Driver De-lead Time Enable This bit field is the time interval between the last stop bit and DE signal during transmission. Its unit is sampling time, determined by oversampling rate. This bit field can be set only when USART is not enabled.
15	OSMCFG	R/W	Oversampling Mode Configure 0: 16-time oversampling 1: 8-time oversampling This bit can be set only when USART is not enabled. In LIN mode, this bit must remain reset to zero.
14	UEN	R/W	USART Enable 0: Disable USART frequency divider and output 1: Enable USART module When this bit is cleared, the USART's frequency divider and output stop working after the current byte transmission is completed to reduce power consumption. This bit is set and cleared to 0 by software.
13:12	M	R/W	Data Bits Length Configure M[1:0]=00: 1 start bit, 8 data bits, n stop bits M[1:0]=01: 1 start bit, 9 data bits, n stop bits M[1:0]=10: 1 start bit, 7 data bits, n stop bits Set 1 or clear 0 by software. This bit cannot be modified during transmission of data.
11	WKSEL	R/W	Wakeup Method Configure 0: Idle bus wakeup 1: Address tag wakeup Set 1 or clear 0 by software. This bit can be set only when USART is not enabled.
10	PCEN	R/W	Parity Control Enable 0: Disable 1: Enable If this bit is set, a check bit will be inserted in the most significant bit (MSB) when transmitting data; when receiving data, check whether the check bit of the received data is correct. The check control will not take effect until the current transmission of bytes is completed.
9	PSEL	R/W	Odd/Even Parity Configure 0: Even parity check 1: Odd parity check The selection will not take effect until the current transmission of bytes is completed.
8	PEIEN	R/W	Parity Error interrupt Enable 0: Disable interrupt generation 1: An interrupt will be generated when PEFLG is set

Field	Name	R/W	Description
7	TXEIEN	R/W	Transmit register Empty Interrupt Enable 0: Disable interrupt generation 1: Generate an interrupt when TXBEFLG is set
6	TCIEN	R/W	Transmit Complete Interrupt Enable 0: Disable 1: An interrupt will be generated when TCFLG is set
5	RXNEIEN	R/W	Receive register Not Empty Interrupt Enable 0: Disable 1: An interrupt will be generated when OREFLG or RXBNEFLG is set
4	IDLEIEN	R/W	IDLE Interrupt Enable 0: Disable 1: An interrupt will be generated when IDLEFLG is set
3	TEN	R/W	Transmit Enable 0: Disable 1: Enable If there is a 0 pulse on this bit at any time of transmitting data, an idle bus will be transmitted after the current data is transmitted. After this bit is set, the data will be transmitted after delay of one-bit time.
2	REN	R/W	Receive Enable 0: Disable 1: Enable, and start to detect the start bit on RX pin
1	RWU	R/W	Receive Mute Mode Enable 0: Normal working mode 1: Can switch between normal mode and mute mode Set 1 or clear 0 by software.
0	SBK	R/W	Transmit Break Frame 0: Not transmit 1: Will transmit This bit can be set by software and cleared to 0 by hardware when the stop bit of the break frame is transmitted.

Applicable to UART:

Field	Name	R/W	Description
31:29	Reserved		
28	SWAPEN	R/W	Swap TX/RX Pins Function Enable 0: Use according to standard allocation 1: The functions of TX and RX pins can be exchanged for use Set or cleared to 0 by software. This bit can be set only when USART is not enabled.
27	RXTOIEN	R/W	Receiver Timeout Interrupt Enable 0: Disable 1: Generate an interrupt when RXTOFLG is set Set or cleared to 0 by software.
26	RXTODEN	R/W	Receive Timeout Detection Function Enable 0: Disable 1: Enable Set or cleared to 0 by software.

Field	Name	R/W	Description
			Set this bit, and when it is found that the RX line is idle for the length of time configured by RXTO register, the RXTOFLG bit will be set by hardware.
25:14	Reserved		
13	UEN	R/W	UART Enable 0: Disable UART frequency divider and output 1: Enable UART module When this bit is cleared, the UART's frequency divider and output stop working after the current byte transmission is completed to reduce power consumption. This bit is set and cleared to 0 by software.
12	M	R/W	Data Bits Length Configure M[1:0]=00: 1 start bit, 8 data bits, n stop bits M[1:0]=01: 1 start bit, 9 data bits, n stop bits M[1:0]=10: 1 start bit, 7 data bits, n stop bits Set 1 or clear 0 by software. This bit cannot be modified during transmission of data.
11	Reserved		
10	PCEN	R/W	Parity Control Enable 0: Disable 1: Enable If this bit is set, a check bit will be inserted in the most significant bit (MSB) when transmitting data; when receiving data, check whether the check bit of the received data is correct. The check control will not take effect until the current transmission of bytes is completed.
9	PSEL	R/W	Odd/Even Parity Configure 0: Even parity check 1: Odd parity check The selection will not take effect until the current transmission of bytes is completed.
8	PEIEN	R/W	Parity Error interrupt Enable 0: Disable interrupt generation 1: An interrupt will be generated when PEFLG is set
7	TXEIEN	R/W	Transmit register Empty Interrupt Enable 0: Disable interrupt generation 1: Generate an interrupt when TXBEFLG is set
6	TCIEN	R/W	Transmit Complete Interrupt Enable 0: Disable 1: An interrupt will be generated when TCFLG is set
5	RXNEIEN	R/W	Receive register Not Empty Interrupt Enable 0: Disable 1: An interrupt will be generated when OREFLG or RXBNEFLG is set
4	IDLEIEN	R/W	IDLE Interrupt Enable 0: Disable 1: An interrupt will be generated when IDLEFLG is set

Field	Name	R/W	Description
3	TEN	R/W	Transmit Enable 0: Disable 1: Enable If there is a 0 pulse on this bit at any time of transmitting data, an idle bus will be transmitted after the current data is transmitted. After this bit is set, the data will be transmitted after delay of one-bit time.
2	REN	R/W	Receive Enable 0: Disable 1: Enable, and start to detect the start bit on RX pin
1	Reserved		
0	SBK	R/W	Transmit Break Frame 0: Not transmit 1: Will transmit This bit can be set by software and cleared to 0 by hardware when the stop bit of the break frame is transmitted.

16.7.5 Control register 2 (USART_CR2)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:24	ADDR	R/W	USART Device Node Address High Setup These bits are used to specify the address of the USART node in silent mode or the character code to be recognized in running mode. In mute mode: They are used to detect wake-up from mute mode through 4-bit / 7-bit address markers during multi-processor communication. The MSB of the character sent by the transmitter should be 1. In 4-bit address marking detection, only ADDR[3:0] bits are used. Note: This bit is only applicable to the USART module.
23:19	Reserved		
18	ABEIEN	R/W	Automatic baud rate detection error interrupt enable 0: Disable 1: Enable
17	ABCIEN	R/W	Automatic baud rate completion interrupt enable 0: Disable 1: Enable
16	ABEN	R/W	Auto Baud Rate Detection Enable 0: Disable 1: Enable Set or cleared to 0 by software.
15	MSBFIRST	R/W	Most Significant Bit First Enable 0: The data of No. 0 bit immediately follows the start bit 1: The data of the most significant bit immediately follows the start bit Set or cleared by software. This bit can be set only when USART is not enabled.

Field	Name	R/W	Description
			Note: This bit is only applicable to the USART module.
14	LINEN	R/W	LIN Mode Enable 0: Disable 1: Enable This bit is set or cleared by software. In LIN mode, the SBK bit in the USART_CR1 register can be used to send LIN synchronization break characters (13 low bits) and detect LIN synchronization break characters.
13	STOP	R/W	STOP Bit Configure 0: 1 stop bit 1: 2 stop bits Note: This bit is only applicable to the UART module.
12	STOP	R/W	STOP Bit Configure 0: 1 stop bit 1: 2 stop bits Note: This bit is only applicable to the USART module.
11	CLKEN	R/W	Clock Enable (CK pin) 0: Disable 1: Enable This bit can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.
10	CPOL	R/W	Clock Polarity Configure The state of CK pin when USART is in idle state 0: Low level 1: High level This bit is valid only in synchronous mode. This bit can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.
9	CPHA	R/W	Clock Phase Configure This bit indicates on the edge of which clock sampling is conducted 0: The first 1: The second This bit is valid only in synchronous mode. This bit can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.
8	LBCL	R/W	Last Bit Clock Pulse Output Enable 0: Not output from CK 1: Output from CK This bit is valid only in synchronous mode. This bit can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.
7	Reserved		
6	LBDIEN	R/W	LIN Break Detection Interrupt Enable 0: Disable 1: Generate an interrupt when LBDFLG bit is set.

Field	Name	R/W	Description
5	LBDL	R/W	LIN Break Detection Length Configure This bit is used to select 11-bit or 10-bit break character detection. 0: 10 bits 1: 11 bits
4	ADDRM7	R/W	Slave Address Length Configure 0: 4-bit address 1: 7-bit address This bit field can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.
3:0	Reserved		

16.7.6 Control register 3 (USART_CR3)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
12	DEMEN	R/W	Driver Enable Users are allowed to activate the control terminal of external transceiver through DE signal. 0: DE function disabled 1: DE function enabled, DE signal output on RTS pin This bit can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.
11	ONEBIT	R/W	Sample Method Configure 0: Sampling for three times 1: Single sampling; flag of noise detection disabled This bit can be set only when USART is not enabled.
10	CTSIEN	R/W	CTS Interrupt Enable 0: Disable 1: Generate an interrupt when CTSFLG is set Note: This bit is only applicable to the USART module.
9	CTSEN	R/W	CTS Function Enable 0: Disable 1: Enable CTS: Clear To Send, which is input signal When CTS input signal is at low level, the data can be transmitted; otherwise, the data cannot be transmitted; if CTS signal is pulled to high during data transmission, the data transmission will be stopped after the data transmission is completed; if write operation is performed for the data register when CTS is high, the data will not be transmitted until CTS is valid. This bit can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.

Field	Name	R/W	Description
8	RTSEN	R/W	RTS Function Enable 0: Disable 1: Enable RTS interrupt RTS: Require To Send, which is output signal, indicating it has been ready to receive. Request is made to receive data only when there is space in the receive buffer; when data can be received, RTS output is pulled to low level. This bit can be set only when USART is not enabled. Note: This bit is only applicable to the USART module.
7	DMAT	R/W	DMA Transmit Enable 0: Disable 1: Enable This bit is set or cleared by software.
6	DMAR	R/W	DMA Receive Enable 0: Disable 1: Enable This bit is set or cleared by software.
5:4	Reserved		
3	HDSEL	R/W	Half-duplex Mode Select 0: Do not select half-duplex mode 1: Select half-duplex mode
2:1	Reserved		
0	EIEN	R/W	Error interrupt Enable 0: Disable 1: Enable; when DMAR is set and one among FEFLG, OREFLG or NEFLG is set, an interrupt will be generated.

16.7.7 Receive timeout register (USART_RXTOR)

Offset address: 0x1C

Reset value: 0x0000

Field	Name	R/W	Description
31:24	Reserved		
23:0	RXTO	R/W	Receiver Timeout Value Setup This field specifies the receive timeout value in the unit of baud clock. In standard mode, after the last byte is received, if no new start bit is detected within the duration of RXTO value, RXTOFLG will be set by hardware.

17 Internal integrated circuit interface (I2C)

17.1 Full Name and Abbreviation of Terms

Table 52 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Serial Data	SDA
Serial Clock	SCL
System Management Bus	SMBus
Clock	CLK
Serial Clock High	SCLH
Serial Clock Low	SCLL
Address Resolution Protocol	ARP
Negative Acknowledgement	NACK
Packet Error Checking	PEC
Address Resolution Protocol	ARP

17.2 Introduction

I2C is a short-distance bus communication protocol. In physical implementation, I2C bus is composed of two signal lines (SDA and SCL) and a ground wire. These two signal lines can be used for bidirectional transmission.

- Two signal lines, SCL clock line and SDA data line. SCL provides timing for SDA, and SDA transmits/receives data in series
- Both SCL and SDA signal lines are bidirectional
- The ground is common when the two systems use I2C bus for communication

17.3 Main characteristics

(1) Multi-master function

(2) The master can generate the clock, start bit and stop bit

(3) Slave function
- Programmable I2C address detection
- Double-address mode
- Detection of stop bit

(4) 7-bit and 10-bit addressing mode

(5) Response to broadcast

(6) Three communication speeds
- Standard mode
- Fast mode
- Fast mode plus

(7) Programmable clock extension

(8) State flag
- Transmitter/Receiver mode flag
- Flag for end of byte transmission
- Busy bus flag

(9) Error flag
- Arbitration loss
- Acknowledgment error
- Detection of wrong start bit or stop bit
- Overflow is prohibited when the clock function is extended

(10) One interrupt vector

(11) Interrupt source
- Address/Data communication succeeded
- Error interrupt

(12) Support DMA function

(13) Programmable digital noise filter

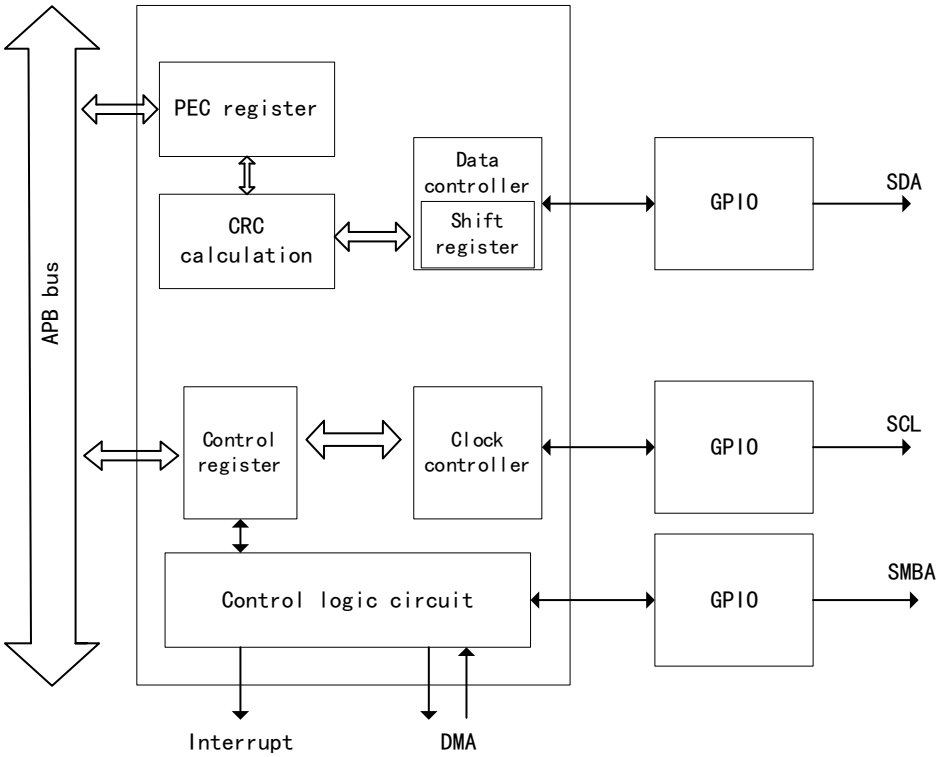
(14) Programmable PEC
- Final transmission in transmission mode
- PEC error check after the last byte is received

(15) SMBus specific function
- 25 ms clock low timeout delay
- The cumulative clock of the 10 ms master device has a low expansion time
- The cumulative clock of the 25 ms slave device has a low expansion time
- Hardware PEC
- Address resolution protocol

(16) Compatible with PMBus

17.4 Structure block diagram

Figure 83 I2C Structure Block Diagram



The interface can be configured to the following modes:

- Slave transmitting
- Slave receiving
- Master transmitting
- Master receiving

In the initial state of I2C interface, the working mode is slave mode. After I2C interface transmits the start signal, it will automatically switch from slave mode to master mode.

17.5 Functional Description

Table 53 Description of Proper Noun of I2C Bus

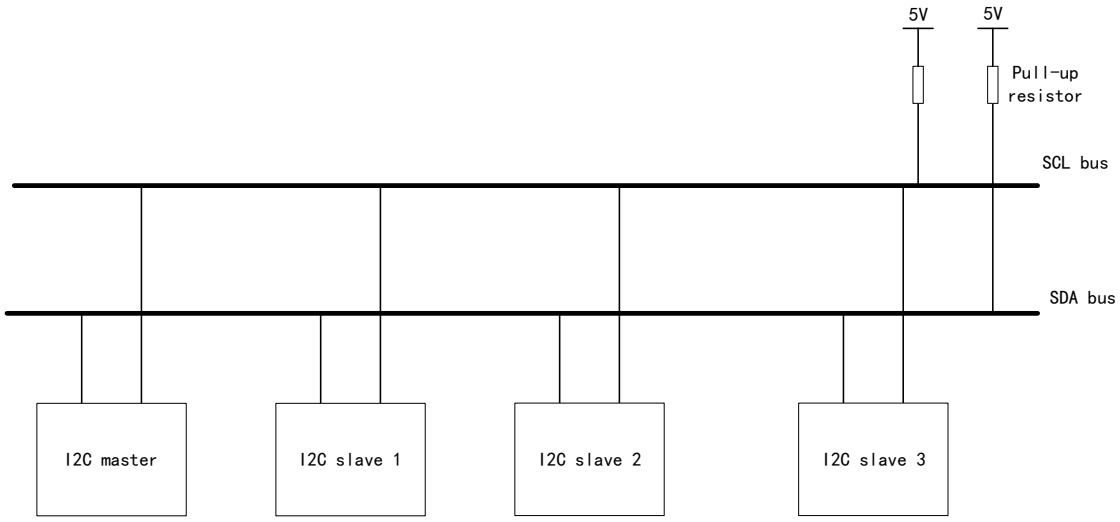
Proper nouns	Description
Transmitter	Device transmitting data to the bus
Receiver	Device receiving data from the bus
Master	Device that initiates data transmission, generates clock signals and ends data transmission
Slave	Device addressed by master

Multiple masters	Multiple masters that control the bus at the same time without destroying information
Synchronize	The process of synchronizing the clock signals between two or more devices
Arbitration	If more than one master tries to control the bus at the same time, only one master can control the bus, and the information of the controlled master will not be destroyed

17.5.1 I2C physical layer

The commonly used connection modes between I2C communication devices are shown in the figure below:

Figure 84 Common I2C Communication Connection Diagram



Characteristics of physical layer:

(1) It supports the buses of multiple devices (signal line shared by multiple devices), which, in I2C communication bus, can connect multiple communication masters and communication slaves.

(2) An I2C bus only uses two bus lines, namely, a bidirectional serial data line (SDA) and a serial clock line (SCL). The data line is used for data transmission, and the clock line is used for synchronous receiving and transmission of data.

(3) Each device connected to the bus has an independent address (seven or ten bits), and the master addresses and accesses the slave devices according to the address of the device.

(4) The bus needs to connect the pull-up resistor to the power supply. When I2C bus is idle, the output is in high-impedance state. When all devices are idle, the output is in high-impedance state, and the pull-up resistor pulls the bus to high.

(5) Three communication modes: Standard mode (up to 100KHz), fast mode (up to 400KHz), and fast mode plus (up to 1MHz).

(6) When the bus is used by multiple masters at the same time, to prevent data collision, the bus arbitration mode is adopted to determine which device occupies the bus.

(7) Able to program the setup and hold time, and program the high-level time and low-level time of SCL in I2C.

17.5.2 I2C protocol layer

Characteristics of protocol layer

(1) Data is transmitted in the form of frame, and each frame is composed of 1 byte (8 bits).

(2) In the rising edge phase of SCL, SDA needs to keep stable and SDA changes when SCL is low.

(3) In addition to data frame, I2C bus also has start signal, stop signal and acknowledgment signal.

- Start bit: During the stable high level period of SCL, a falling edge of SDA starts transmission.
- Stop bit: During the stable high level period of SCL, a rising edge of SDA stops transmission.
- Acknowledge bit: Used to indicate successful transmission of one byte. After the bus transmitter (regardless of the master or slave) transmits 8-bit data, SDA will release (from output to input). During the ninth clock pulse period, the receiver will pull down SDA to acknowledge receiving of data.

I2C communication reading and writing process

Figure 85 Master Writes Data to Slave

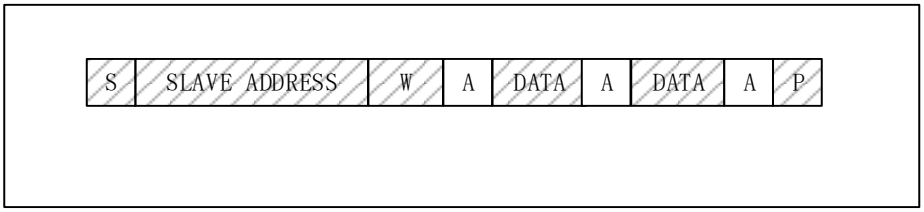
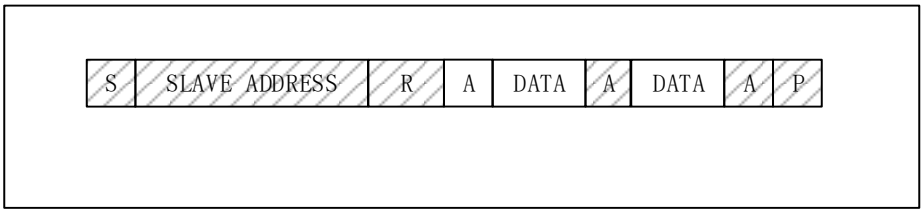


Figure86 Master Reads Data from Slave



Remarks:

(1)  : This data is transmitted from master to slave

(2) S: Start signal

(3) SLAVE ADDRESS: Slave address

(4)  : This data is transmitted from slave to master

(5) R/W: Selection bit of transmission direction

(6) 1 means read

(7) 0 means write

(8) P: Stop signal

After the start signal is generated, all slaves will wait for the slave address signal transmitted by the master. In I2C bus, the address of each device is unique. When the address signal matches the device address, the slave will be selected, and the unselected slave will ignore the future data signal.

When the master direction is writing data

After broadcasting the address and receiving the acknowledge signal, the master will transmit data to the slave, the data length is one byte, and every time the master transmits one byte of data, it needs to wait for the acknowledge signal transmitted by the slave. After all the bytes have been transmitted, the master will transmit a stop signal (STOP) to the slave, indicating that the transmission is completed.

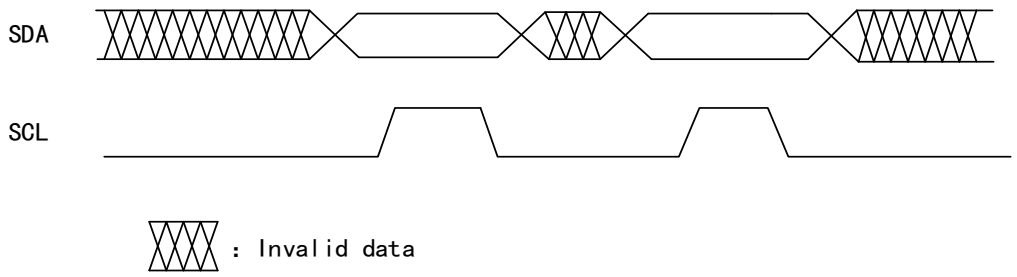
When the master direction is reading data

After broadcasting the address and receiving the acknowledge signal, the slave will transmit the data to the master. The size of the data package is 8 bits. Every time the slave transmits one byte of data, it needs to wait for the acknowledge signal of the master. When the master needs to stop receiving data, it needs to return a non-acknowledge signal to the slave, then the slave will stop transmitting the data automatically.

17.5.3 Data validity

In the process of data transmission, the data on SDA line must be stable when the clock signal SCL is at high level. Only when the SCL is at the low level, can the level state of SDA be changed, and the bit transmission of each data needs a clock pulse.

Figure 87 SDA Timing Diagram



17.5.4 Start and stop signals

All data transmission must have start signal (START) and stop signal (STOP).

Figure 88 START signal is defined as: when SCL is at high level, SDA will convert from high level to low level

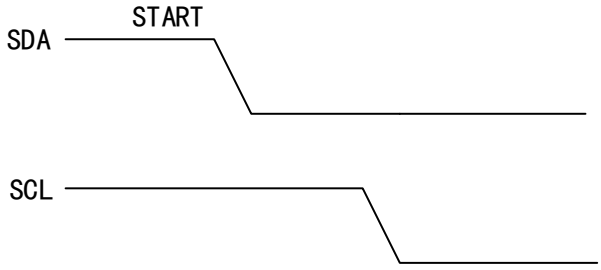
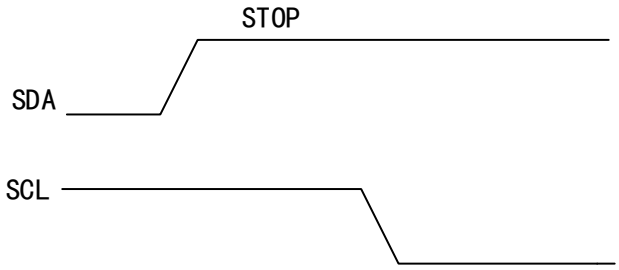


Figure 89 STOP signal is defined as: when SCL is at high level, SDA will convert from low level to high level



17.5.5 Arbitration

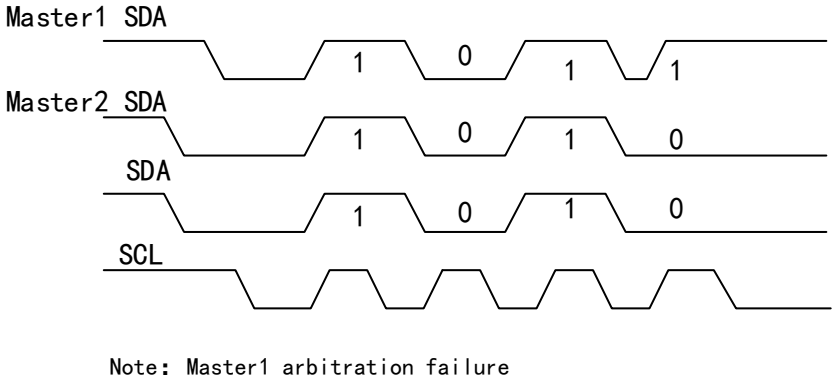
Arbitration is also used to solve the bus control conflict in case of multiple masters. The arbitration process takes place on the master and has nothing to do with the slave.

The master can start transmission only when the bus is idle. Two masters may generate an effective START signal on the bus within the shortest hold time of the START signal. In this situation, it is required that arbitration should decide which master completes the transmission.

Arbitration is conducted by bit. During each arbitration, when SCL is high, each master will check whether the SDA level is the same as that transmitted by itself. The arbitration process needs to last for many bits. Theoretically, if two masters transmit exactly the same content, they can successfully transmit

without arbitration failure. If one master transmits high level, but it is detected that SDA is at low level, an arbitration failure error will occur, the SDA output of the master will be disabled, and the other master will complete its own transmission.

Figure 90 SDA Timing Diagram



Note: Master1 arbitration failure

17.5.6 SDA/SCL line control

(1) If the clock is allowed to be extended

- Transmitter mode: If TXBEFLG=1 and BTCFLG=1: The I2C interface keeps the clock line low before transmission to wait for the software to read I2C_SR1, and then writes the data into the data register (both the buffer and the shift register are empty).
- Receiver mode: If RXNE =1 and BTCFLG=1: The I2C interface keeps the clock line low after receiving data bytes, waiting for the software to read I2C_SR1, and then read the data register I2C_DR (both the buffer and the shift register are full).

(2) If clock extension is disabled in the slave mode

- If RXBNEFLG=1, an overload error occurs if the DATA has not been read out before the next byte is received. The last byte received was lost.
- If TXBEFLG=1 and no new data is written into I2C_DR before the next byte must be sent, an underload error occurs. The same bytes will be sent out repeatedly.
- Do not control duplicate write conflicts.

17.5.7 SMBus specific function

System management bus (SMBus) is a simple single-end double-wire bus, which can meet the requirements of lightweight communication.

SMBus is commonly used in computer motherboard, mainly for power transmission ON/OFF instructions. SMBus is the derivative bus of I2C. It is mainly used for communication of low-bandwidth devices on computer motherboard, and power-related chip.

Address resolution protocol

SMBus specification includes an address resolution protocol, which can realize dynamic address assignment. Dynamic recognition hardware and software enable the bus to support hot plugging, and the bus devices will be automatically identified and assigned with a unique address.

SMBus alarm

SMBus alarm is an optional signal with an interrupt line. It is used for pins that are sacrificed to extend their control ability.

17.5.8 Error flag bit

Table 54 The following several error flag bits exist in I2C communication

Error flag bit	Description of error flag bit
Answer error flag bit (AEFLG)	No acknowledgment received
Bus error flag bit (BERRFLG)	An external stop or start condition is detected
Arbitration loss flag bit (ALFLG)	Arbitration loss is detected by the interface
Overrun/Underrun error flag bit (OVRURFLG)	In slave mode, the received data is not read out, the next data has arrived, and an overrun error occurs. The transmitting data clock has arrived, but the data has not been written to the DATA register, and an underrun error occurs.
Timeout or Tlow error flag bit (TTEFLG)	SCL is pulled down for more than a certain time
PEC comparison error flag bit (PECEFLG)	CRC values are not equal

17.5.9 Packet error check (PEC)

I2C module has a PEC module, which checks the message of I2C data by CRC-8 calculator. The CRC-8 polynomial used by the calculator is: C(x)= X8+X2+X+1.

When PECEN bit is set to 1 and PEC function is enabled, PEC module will calculate all data transmitted by I2C bus, including address data.

17.5.10 DMA mode

According to the software process of I2C, when the transmit register is empty or the receiver register is full, MCU needs to write or read bytes, then we can complete the operation more quickly through the DMA function of I2C.

DMA transmission

Set the DMAEN bit in I2C_CR2 register to enable the DMA mode. When the transmit register is empty (TXBEFLG is set to 1), the data will be directly loaded from the memory area to the DATA register through DMA.

DMA receiving

Set DMAEN bit in I2C_CR2 register to enable DMA mode. When the receive register is full (RXBNEFLG is set to 1), DMA will transmit DATA register data to the set storage area.

17.5.11 Programmable noise filter

In Fm mode, the I2C standard requires that the spikes on SDA and SCL lines should be suppressed to a length of 50ns.

The analog noise filter is implemented by SDA and SCL I/O. The analog noise filter is enabled by default and can be disabled by setting ANFEN bit of I2C_FILTER register.

The digital noise filter can be enabled and its filtering capability can be configured by DNFCFG bit of I2C_FILTER register.

Table 55 The maximum value of DNFCFG[3:0] that meets the communication requirements

PCLK frequency (MHz)	DNFCFG[3:0] maximum value		
	Sm mode	Fm mode	Fm plus mode
Fpclk = 2	1	/	/
Fpclk = 4	8	0	/
Fpclk = 8	15	2	/
Fpclk = 16	15	7	0
Fpclk = 32	15	15	1
Fpclk = 64	15	15	2

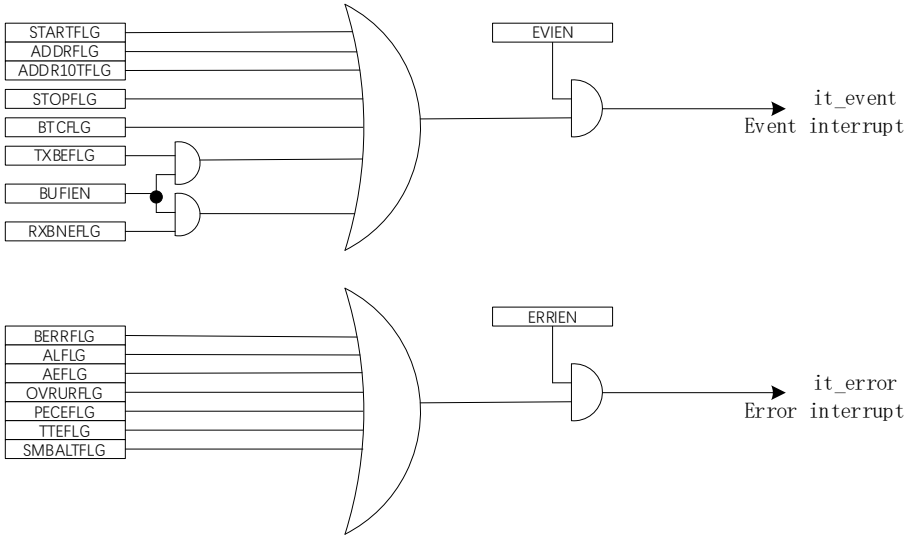
17.5.12 I2C interrupt

Table 56 I2C Interrupt Request

Interrupt event	Event flag bit	Interrupt control bit
Start bit transmission completed	STARTFLG	EVIEN
Transmission completed/Address matching address signal	ADDRFLG	
10-bit address head segment transmission completed	ADDR10FLG	
Stop signal received	STOPFLG	
Data byte transmission completed	BTCFLG	
Receive buffer not empty	RXBNEFLG	EVIEN and BUFIEN
Transmit buffer empty	TXBEFLG	
Bus error	BERRFLG	ERRIEN
Arbitration loss	ALFLG	

Interrupt event	Event flag bit	Interrupt control bit
Answer failed	AEFLG	
Overrun/Underrun	OVRURFLG	
PEC error	PECEFLG	
Timeout or Tlow error	TTEFLG	
SMBus reminder	SMBALTFLG	

Figure 91 I2C interrupt mapping



Note: Event interrupts and error interrupts are aggregated into one interrupt output.

17.6 Register address mapping

Table 57 I2C Register Address Mapping

Register name	Description	Offset Address
I2C_CR1	Control register 1	0x00
I2C_CR2	Control register 2	0x04
I2C_SADDR1	Slave mode address register 1	0x08
I2C_SADDR2	Slave mode address register 2	0x0C
I2C_DR	Data register	0x10
I2C_SR1	State register 1	0x14
I2C_SR2	State register 2	0x18
I2C_CLKCR	Master clock control register	0x1C
I2C_RISETMAX	Maximum rising time register	0x20
I2C_FILTER	Filter control register	0x24

17.7 Register functional description

17.7.1 Control register 1 (I2C_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	SWRSTS	R/W	Software Configure I2C under Reset State 0: Not reset 1: Reset; before I2C is reset, ensure that I2C pin is released and the bus is in idle state.
14	Reserved		
13	ALERTEN	R/W	SMBus Alert Enable This bit can be set to 1 or cleared to 0 by software; when I2CEN=0, it is cleared to 0 by hardware. 0: Release the SMBAlert pin to make it higher, and transmit the response address header immediately after prompt is given to transmit the NACK signal 1: Drive SMBAlert pin to make it lower, and transmit the response address header immediately after prompt is given to transmit the ACKEN signal
12	PEC	R/W	Packet Error Check Transfer Enable This bit can be set to 1 or cleared to 0 by software; after PEC, start bit or stop bit is transmitted, or when I2CEN=0, it is cleared to 0 by hardware. 0: Disable 1: Enable
11	ACKPOS	R/W	Acknowledge /PEC Position Configure This bit can be set to 1 or cleared to 0 by software; when I2CEN=0, it is cleared by hardware. 0: When receiving current byte, whether transmitting NACK/ACK, whether PEC is in shift register 1: When receiving next byte, whether transmitting NACK/ACK and whether PEC is in the next byte of shift register
10	ACKEN	R/W	Acknowledge Transfer Enable This bit can be set to 1 or cleared to 0 by software; when I2CEN=0, it is cleared by hardware. 0: Not transmit 1: Transmit
9	STOP	R/W	Stop Bit Transfer This bit can be set to 1 or cleared to 0 by software; when transmitting the stop bit, it is cleared to 0 by hardware; when timeout error is detected, it is set to 1 by hardware. 0: Not transmit 1: Transmit

Field	Name	R/W	Description
8	START	R/W	Start Bit Transfer This bit can be set to 1 and cleared to 0 by software; when transmitting the start bit or 2CEN=0, it is cleared to 0 by hardware. 0: Not transmit 1: Transmit
7	CLKSTRETCHD	R/W	Slave Mode Clock Stretching Disable 0: Enable 1: Disable In slave mode, enabling extending the low-level time of the clock can avoid overrun and underrun errors.
6	SRBEN	R/W	Slave Responds Broadcast Enable 0: Disable 1: Enable Note: The broadcast address is 0X00
5	PECEN	R/W	PEC Enable 0: Disable 1: Enable
4	ARPEN	R/W	ARP Enable 0: Disable 1: Enable If SMBTCFG=0, use the default address of SMBus device If SMBTCFG=1, use the master address of SMBus
3	SMBTCFG	R/W	SMBus Type Configure 0: SMBus device 1: SMBus master
2	Reserved		
1	SMBEN	R/W	SMBus Mode Enable 0: I2C mode 1: SMBus mode
0	I2CEN	R/W	I2C Enable 0: Disable 1: Enable

17.7.2 Control register 2 (I2C_CR2)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
12	LTCFG	R/W	DMA Last Transfer Configure Configure whether the EOT of the next DMA is the last transmission received, and only used for the master receiving mode. 0: No 1: Yes

Field	Name	R/W	Description
11	DMAEN	R/W	DMA Requests Enable 0: Disable 1: When TXBEFLG=1 or RXBNEFLG=1, enable DMA request
10	BUFIEN	R/W	Buffer Interrupt Enable 0: Disable 1: Enable; when the bit of any of the following state registers is set to 1, the interrupt will be generated: TXBEFLG and RXBNEFLG
9	EVIEN	R/W	Event Interrupt Enable 0: Disable 1: Enable; when the bit of any of the following state registers is set to 1, the interrupt will be generated: STARTFLG, ADDRFLG, ADDR10FLG, STOPFLG, BTCFLG, TXBEFLG is set to 1 and BUFIEN is set to 1, RXBNEFLG is set to 1 and BUFIEN is set to 1.
8	ERRIEN	R/W	Error Interrupt Enable 0: Disable 1: Enable; when the bit of any of the following state registers is set to 1, this interrupt will be generated: SMBALTFLG, TTEFLG, PECEFLG, OVRURFLG, AEFLG, ALFLG, and BERRFLG
7:0	CLKFCFG	R/W	I2C Clock Frequency Configure The clock frequency is frequency of the clock of I2C module, namely, the clock input from APB bus. 0: Disable 1: Disable 2: 2MHz ... 64: 64MHz Minimum clock frequency of the I2C bus: 2MHz in standard mode, 4MHz in fast mode, and 10MHz in ultra-fast mode.

17.7.3 Slave mode address register 1 (I2C_SADDR1)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	ADDRLEN	R/W	Slave Address Length Configure 0: 7-bit address mode 1: 10-bit address mode
14:10	Reserved		
9:8	ADDR[9:8]	R/W	Slave Address Setup When the address mode is 7 bits, the bit is invalid; when the address mode is 10 bits, this bit is the 9:8 bit of the address.
7:1	ADDR[7:1]	R/W	Slave Address Setup The 7:1 bit of slave address
0	ADDR[0]	R/W	Slave Address Setup When the address mode is 7 bits, the bit is invalid; when the address mode is 10 bits, this bit is the 0 bit of the address.

17.7.4 Slave mode address register 2 (I2C_SADDR2)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7:1	ADDR2[7:1]	R/W	Slave Dual Address Mode Address Setup The 7:1 bit of the address in double-address mode
0	ADDRNUM	R/W	Slave Address Number Configure In 7-bit address mode, the slave can be configured to identify the single-address mode and double-address mode; only ADDR1 is identified in single-address mode; both ADDR1 and ADDR2 can be identified in double-address mode Single or double-address registers can be identified in 7-bit address mode, specifically as follows: 0: Identify 1 address (ADDR1) 1: Identify 2 addresses (ADDR1 and ADDR2)

17.7.5 Data register (I2C_DR)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7:0	DATA	R/W	Data Register In I2C transmission mode, write the data to be transmitted to this register; in I2C receiving mode, read the received data from this register.

17.7.6 State register 1 (I2C_SR1)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	SMBALTFLG	RC_W0	SMBus Alert Occur Flag 0: SMBus master mode, without alarm; SMBus slave mode, without alarm, SMBAlert pin level unchanged 1: SMBus master mode, with an alarm generated on the pin; SMBus slave mode, receiving an alarm, causing SMBAlert pin level to become low This bit is set to 1 by hardware; this bit can be cleared by writing 0 by software; when I2CEN=0, it can be cleared to 0 by hardware.
14	TTEFLG	RC_W0	Timeout or Tlow Error Flag 0: No timeout error 1: When a timeout error occurs, in slave mode, the slave is reset and the bus is released; in master mode, the hardware transmits the stop bit. This bit can be set to 1 by hardware when timeout error occurs in any of the following situations: (1) SCL maintains low level for more than 25ms;

Field	Name	R/W	Description
			(2) SCL low-level extension time of the master device is more than 10ms; (3) SCL low-level extension time of the slave device is more than 25ms. This bit can be cleared by writing 0 by software; and be cleared to 0 by hardware when I2CEN=0.
13	Reserved		
12	PECEFLG	RC_W0	PEC Error in Reception Flag 0: No PEC error: when ACKEN=1, after PEC is received, the receiver will return ACKEN 1: There is PEC error: regardless of the value of ACKEN, as long as PEC is received, the receiver will return NACK This bit can be cleared by writing 0 by software; and be cleared to 0 by hardware when I2CEN=0.
11	OVRURFLG	RC_W0	Overrun/Underrun Flag 0: Not occur 1: Occurred This bit can be set to 1 by hardware when CLKSTRETCHD=1 and any of the following conditions is met: (1) In the slave receiving mode, when the data in the DATA register is not read out, but a new data is received (this data will be lost), overrun occurs; (2) In the slave transmission mode, no data is written in the data register but data still needs to be transmitted (the same data is transmitted twice), and then underrun occurs. This bit can be cleared by writing 0 by software; and be cleared to 0 by hardware when I2CEN=0.
10	AEFLG	RC_W0	Acknowledge Error Flag 0: No acknowledgment error 1: Acknowledgment error occurred This bit is set to 1 by hardware; this bit can be cleared by writing 0 by software; when I2CEN=0, it can be cleared to 0 by hardware.
9	ALFLG	RC_W0	Master Mode Arbitration Lost Flag 0: No arbitration loss 1: In case of arbitration loss, I2C interface will automatically switch back to slave mode "Arbitration loss in master mode" means the master loses the control of buses; this bit can be set to 1 by hardware; this bit can be cleared by writing 0 by software; when I2CEN=0, it can be cleared to 0 by hardware.
8	BERRFLG	RC_W0	Bus Error Flag 0: No bus error 1: Bus error occurred Bus error means exception of start bit or stop bit; when an error is detected, this bit can be set to 1 by hardware; this bit can be cleared by writing 0 by software; when I2CEN=0, it can be cleared to 0 by hardware.
7	TXBEFLG	R	Transmit Buffer Empty Flag 0: The transmit buffer is not empty

Field	Name	R/W	Description
			1: The transmit buffer is empty This bit can be set to 1 by hardware when the content of DATA register is empty; When the software writes the first data to the DATA register, it will immediately move the data to the shift register, then the data in the DATA register is empty and this bit cannot be cleared; This bit can be cleared after the software writes data to DATA register; after transmitting the start bit and stop bit, or when I2CEN=0, it can be cleared to 0 by hardware.
6	RXBNEFLG	R	Receive Buffer Not Empty Flag 0: The receive buffer is empty 1: The receive buffer is not empty This bit can be set to 1 by hardware when there is data in DATA register; When BTCFLG is set to 1, since the data register is still full, the RXBNEFLG bit cannot be cleared by reading DATA register; This bit can be cleared after the software reads and writes DATA register; when I2CEN=0, it can be cleared to 0 by hardware.
5	Reserved		
4	STOPFLG	R	Stop Bit Detection Flag 0: Not detected 1: Detected If ACKEN=1, after one answer, when the slave detects the stop bit on the bus, it will be set to 1 by hardware; This bit can be cleared after the software first reads I2C_SR1 register and then writes the I2C_CR1 register; when I2CEN=0, it can be cleared to 0 by hardware.
3	ADDR10FLG	R	10-Bit Address Header Transmit Flag 0: Not transmit 1: Transmitted The bit can be set to 1 by hardware; this bit can be cleared after the software first reads I2C_SR1 register and then writes the DATA register; when I2CEN=0, it can be cleared to 0 by hardware.
2	BTCFLG	R	Byte Transfer Complete Flag 0: Not completed 1: Completed When receiving data, if the data received in DATA register fails to be read , and a new data is received then, it will be set to 1 by hardware; When transmitting data, if the DATA register is empty, it will be set to 1 by hardware to transmit the data in the shift register. This bit can be cleared after the software first reads I2C_SR1 register, and then reads or writes the DATA register; this bit can be cleared to 0 by hardware by transmitting a start bit and stop bit during the transmission, or when I2CEN=0.
1	ADDRFLG	R	Address Transfer Complete /Receive Match Flag Whether the matching address is received in slave mode: 0: Not received

Field	Name	R/W	Description
			1: Received Whether master mode address transmission is completed: 0: Not completed 1: Completed The bit can be set to 1 by hardware; this bit can be cleared after the software first reads I2C_SR1 register and then reads I2C_SR2 register; when I2CEN=0, it can be cleared to 0 by hardware.
0	STARTFLG	R	Start Bit Sent Finished Flag 0: Not transmit 1: Transmitted When the start bit is transmitted, this bit can be set to 1 by hardware; this bit can be cleared after the software first reads I2C_SR1 register and then writes the DATA register; when I2CEN=0, it can be cleared to 0 by hardware.

17.7.7 State register 2 (I2C_SR2)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:8	PECVAL	R	Save Packet Error Checking Value When PECEN=1, the internal PEC value is saved in PECVAL.
7	DUALADDRFLG	R	Slave Mode Received Dual Address Match Flag 0: The received address matches the content of ADDR1 register 1: The received address matches the content of ADDR2 register This bit can be set to 1 by hardware; and cleared to 0 by hardware when any of the following conditions is met: (1) Stop bit is generated (2) Repeated start bit is generated (3) I2CEN=0
6	SMMHADDRFLG	R	SMBus Device Received Master Header Flag in Slave Mode 0: Failed to receive the master head address 1: Received the master head address when SMBTSEL=1 and ARPEN=1 This bit can be set to 1 by hardware; and cleared to 0 by hardware when any of the following conditions is met: (1) Stop bit is generated (2) Repeated start bit is generated (3) I2CEN=0
5	SMBDADDRFLG	R	SMBus Device Received Default Address Flag in Slave Mode 0: Failed to receive the default address 1: Received the default address when ARPEN=1 This bit can be set to 1 by hardware; and cleared to 0 by hardware when any of the following conditions is met: (1) Stop bit is generated (2) Repeated start bit is generated

Field	Name	R/W	Description
			(3) I2CEN=0
4	GENCALLFLG	R	Slave Mode Received General Call Address Flag 0: Failed to receive the broadcast address 1: Received the Broadcast address This bit can be set to 1 by hardware; and cleared to 0 by hardware when any of the following conditions is met: (1) Stop bit is generated (2) Repeated start bit is generated (3) I2CEN=0
3	Reserved		
2	TRFLG	R	Transmitter / Receiver Mode Flag 0: The device is in receiver mode (read) 1: The device is in transmitter mode (write) Decide the bit value according to R/W bit; This bit can be cleared to 0 by hardware when any of the following conditions is met: (1) Stop bit is generated (2) Repeated start bit is generated (3) Bus arbitration is lost (4) I2CEN=0
1	BUSBSYFLG	R	Bus Busy Flag 0: The bus is idle (no communication) 1: The bus is busy (in the progress of communication) This bit can be set to 1 by hardware when SDA or SCL is at low level; and cleared to 0 by hardware after the stop bit is generated.
0	MSFLG	R	Master Slave Mode Flag 0: Slave mode 1: Master mode This bit can be set to 1 by hardware when I2C is configured as master mode; This bit can be cleared to 0 by hardware when any of the following conditions is met: (1) Stop bit is generated (2) Bus arbitration is lost (3) I2CEN=0

17.7.8 Master clock control register (I2C_CLKCR)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	SPEEDCFG	R/W	Master Mode Speed Configure 0: Standard mode 1: Fast mode

Field	Name	R/W	Description
14	FDUTYCFG	R/W	Fast Mode Duty Cycle Configure Duty cycle = t_{low}/t_{high} 0: SCLK duty cycle is 2 1: SCLK duty cycle is 16/9
13:12	Reserved		
11:0	CLKCFG	R/W	Clock Setup in Fast/Standard Master Mode In I2C standard mode or SMBus mode: T_{high}=CLKCFG × T_{PCLK1} T_{low}=CLKCFG × T_{PCLK1} In I2C fast mode: When FDUTYCFG=0: T_{high}=CLKCFG×T_{PCLK1} T_{low}=2×CLKCFG× T_{PCLK1} When FDUTYCFG=1: T_{high}=9 × CLKCFG × T_{PCLK1} T_{low}=16 × CLKCFG × T_{PCLK1}

17.7.9 Maximum rising time register (I2C_RISETMAX)

Offset address: 0x20

Reset value: 0x0000 0002

Field	Name	R/W	Description
31:8	Reserved		
7:0	RISETMAX	R/W	Master Mode Maximum Rise Time in Fast/Standard Mode The time is in T_{PCLK1}, and RISETMAX is the maximum rising time unit of SCL plus 1.

17.7.10 Filter control register (I2C_FILTER)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:5	Reserved		
4	ANFDISEN	R/W	Analog Noise Filter Disable 0: Enable 1: Disable
3:0	DNFCFG	R/W	Digital Noise Filter Filtering Capability Configure 0000: Disable 0001：1 × T_{PCLK1} …… 1111：15 × T_{PCLK1} Note: These bits can be configured only when I2CEN=0.

18 Serial peripheral interface (SPI)

18.1 Full Name and Abbreviation Description of Terms

Table 58 Full Name and Abbreviation Description of SPI Terms

Full name in English	English abbreviation
Most Significant Bit	MSB
Least Significant Bit	LSB
Master Out Slave In	MOSI
Master In Slave Out	MISO
Serial Clock	SCK
Serial Data	SD
Master Clock	MCK
Word Select	WS
Pulse-code Modulation	PCM
Inter-IC Sound	I2S
Transmit	TX
Receive	RX
Busy	BSY

18.2 Introduction

Serial peripheral interface (SPI) provides data transmitting and receiving functions based on SPI protocol, which allows chips to communicate with external devices in full duplex, synchronous and serial modes, and can work in master or slave mode.

18.3 Main characteristics of SPI

(1) Master and slave operation with 4-wire full duplex synchronous transmission and receiving

(2) Select 2- wire half-duplex communication

(3) Select 8-bit or 16-bit transmission frame format

(4) Supports SPI MOTOROLA protocol mode

(5) Support special transmission and receiving mark and can trigger interrupts

(6) Have SPI bus busy state flag

(7) Clock polarity and phase are programmable

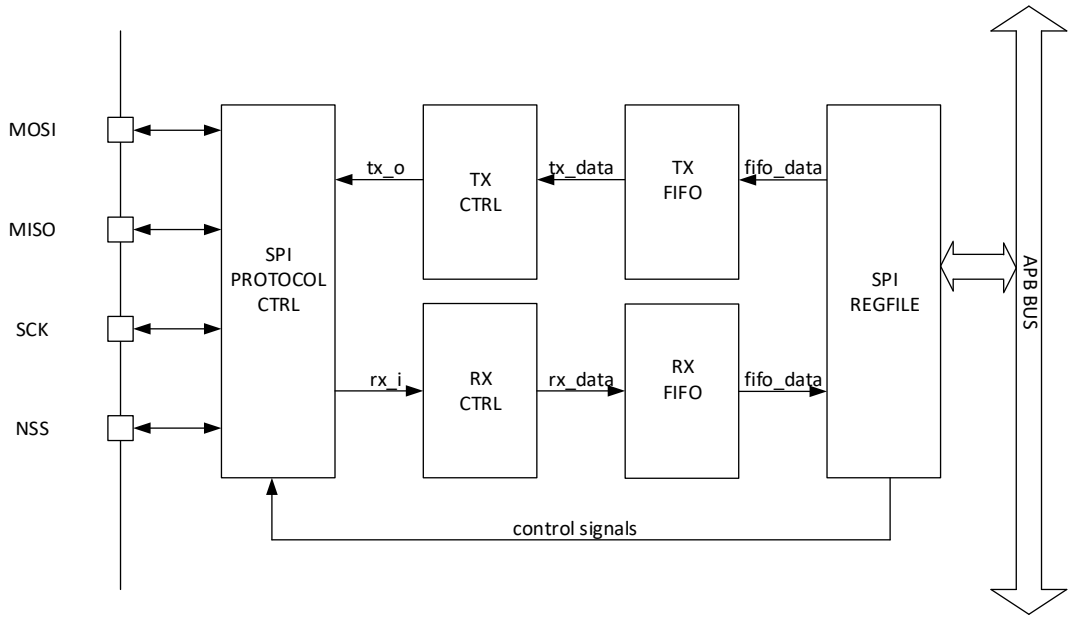
(8) Independent transmit and receive FIFO

(9) Data sequence is programmable; select MSB or LSB in front

(10) With DMA transmission

18.4 Structure block diagram

Figure 92 SPI Structure Block Diagram



18.5 SPI functional description

18.5.1 Description of SPI signal line

Table 59 SPI Signal Line Description

Pin name	Description
SCK	Master device: SPI clock output Slave device: SPI clock input
MISO	Master device: Input the pin and receive data Slave device: Output the pin and transmit data Data direction: From slave device to master device
MOSI	Master device: Output the pin and transmit data Slave device: Input the pin and receive data Data direction: From master device to slave device

Pin name	Description
NSS	Master hardware NSS mode: NSS output Slave hardware NSS mode: NSS input Note: Multi-master mode is not supported. When the slave machine is in software NSS mode, this pin can be used for other purposes.

18.5.2 Communication format

In SPI communication, receiving data and transmitting data can be carried out at the same time. SCK transmits and samples the data on the data line synchronously. The communication format depends on the clock phase, clock polarity and data frame format. If the communication is normal, the master device and the slave device must be in the same communication format.

18.5.2.1 Phase and polarity of clock signal

The clock polarity and clock phase are CPOL and CPHA bits of SPI_CR1 register.

Clock polarity CPOL means the level signal of SCK signal line when SPI is in idle state.

- When CPOL=0, SCK signal line is low in idle state
- When CPOL=1, SCK signal line is high in idle state

Clock phase CPHA means the sampling moment of data

- When CPHA=0, the signal on MOSI or MISO data line will be sampled by the "odd edge" on SCK clock line.
- When CPHA=1, the signal on MOSI or MISO data line will be sampled by the "even edge" on SCK clock line.

SPI can be divided into four modes according to the states of clock phase CPHA and clock polarity CPOL.

Table 60 Four Modes of SPI

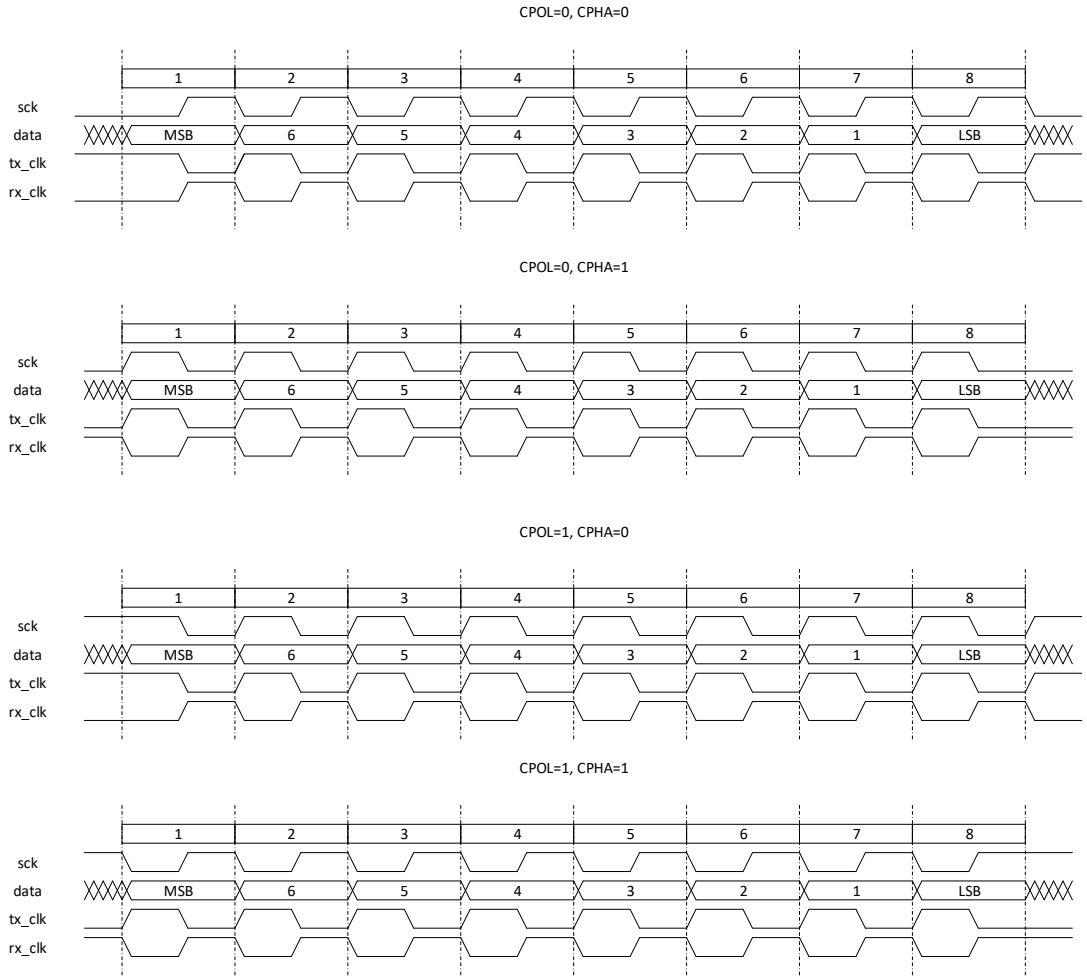
SPI mode	CPHA	CPOL	Sampling moment	Idle SCK clock
0	0	0	Odd edge	Low level
1	0	1	Odd edge	High level
2	1	0	Even edge	Low level
3	1	1	Even edge	High level

Note:

(1) To change CPOL and CPHA bits, SPI must be cleared to 0 and disabled by SPIEN bit.

(2) When SCK is in idle state, if CPOL=1, pull up SCK; if CPOL=0, pull down SCK.

Figure 93 Clock phase and polarity



18.5.2.2 Data frame format

You can first configure the DFF bit selection of SPI_CR1 to pass through the word length, and then configure the LSBFIRST bit selection of the SPI_CR1 register to determine whether LSB or MSB comes first. When accessing SPI_DR register, the data frames are always right aligned. In the process of communication, only the bits within the data word length range will be output with the clock.

18.5.3 NSS mode

Software NSS mode: Select to enable or disable this mode by configuring SSM bit of SPI_CR1 register, and the internal NSS signal level is driven by SSI bit of SPI_CR1 register.

Hardware NSS mode:

- Enable NSS output: When SPI is in master mode, enable SSM bit, NSS pin will be pulled to low and SPI will automatically enter the slave mode.
- Disable NSS output: Operation is allowed in multi-master environments.

18.5.4 SPI mode

18.5.4.1 Initialization of SPI master mode

In master mode, serial clock is generated on SCK pin.

Configure master mode

- Configure MSTR=1 in SPI_CR1 register, and set to master mode
- Select the serial clock baud rate by configuring BR bit in SPI_CR1 register
- Select the polarity and phase by configuring CPOL and CPHA bits in SPI_CR1 register
- Select the transmission mode by configuring BIDIOEN and BIDIMEN bits in SPI_CR1 register
- Select data frame length by configuring DFF bit of SPI_CR1 register
- Select LSB or MSB first by configuring LSBFIRST in SPI_CR1 register
- Configure SPIEN bit in SPI_CR1 register to enable SPI

In master mode: MOSI pin is data output, while MISO is data input.

Note:

(1) To prevent communication errors, when SPIEN=1, BR, CPOL, CPHA, DFF, LSBFIRST, and MSTR all have write protection.

(2) If it is necessary to change the clock polarity for communication, it is recommended to keep SPIEN=0 and then change CPOL and CPHA, and then set SPIEN to 1 to avoid communication errors when the clock polarity is switched.

18.5.4.2 Initialization of SPI slave mode

In slave mode, SCK pin receives the serial clock from the master device.

Configuration of slave mode

- Configure MSTR=0 in SPI_CR1 register, and set to slave mode
- Select the polarity and phase by configuring CPOL and CPHA bits in SPI_CR1 register
- Select the transmission mode by configuring BIDIOEN and BIDIMEN bits in SPI_CR1 register
- Select data frame length by configuring DFF bit of SPI_CR1 register
- Select LSB or MSB first by configuring LSBFIRST in SPI_CR1 register
- NSS configuration:
  - In hardware mode: NSS pin must be low in the whole data frame transmission process
  - In software mode: Set SSM bit in SPI_CR1 register and clear SSI bit
- Configure SPIEN bit in SPI_CR1 register to enable SPI

In slave mode: MOSI pin is data input, while MISO is data output.

18.5.4.3 Full-duplex communication of SPI

Usually, SPI is configured as full-duplex communication, and the master and the slave shift registers are connected through two unidirectional lines MOSI and MISO. During SPI communication, synchronous data transmission is conducted according to SCK clock edge. The data of the master are transmitted to the slave through MOSI pin, and the data of the slave are transmitted to the master through MISO pin. When the data transmission is completed, it means that the information is exchanged successfully.

18.5.4.4 Communication of multiple slave devices of SPI

SPI can be operated by multiple slave devices. The master device uses GPIO pin to manage the chip selection line of the slave device, and can control two or more independent slave devices.

The master device decides using which slave device to transmit data by pulling down the NSS pin of the slave device.

18.5.5 Data transmission and receiving process in different modes of SPI

Table 61 Run Mode of SPI

Mode	Configuration	Data pin
Full-duplex mode of master device	BIDIMEN=0, RXONLY=0	MISO receives; MOSI transmits
One-way receiving mode of the master device	BIDIMEN=0, RXONLY=1	MISO receives; MOSI is not used
Bidirectional transmission mode of the master device	BIDIMEN=1，BIDIOEN=1	MISO is not used; MOSI transmits
Bidirectional receiving mode of the master device	BIDIMEN =1，BIDIOEN=0	MISO is not used; MOSI receives
Full-duplex mode of slave device	BIDIMEN=0, RXONLY=0	MISO transmits; MOSI receives
One-way receiving mode of the slave device	BIDIMEN=0, RXONLY=1	MISO is not used; MOSI receives
Bidirectional transmission mode of the slave device	BIDIMEN=1，BIDIOEN=1	MISO transmits; MOSI is not used
Bidirectional receiving mode of the slave device	BIDIMEN =1，BIDIOEN=0	MISO receives; MOSI is not used

Note:

(1) BIDIMEN, BIDIOEN and RXONLY do not have the write protection control of SPIEN, but they need to be modified after one frame of data transmission is complete; otherwise, it will

cause partial interruption of communication and result in data transmission and reception errors.

(2) When the master communicates, if only receiving is required, the module will never stop receiving data, and BUSYFLG remains at 1. The received data needs to be processed in a timely manner. After communication is completed, first restore to full-duplex mode and then turn off SPIEN.

Figure 94 Connection in Full Duplex Mode

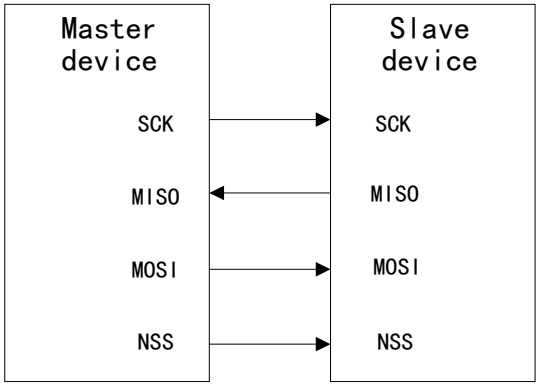


Figure 95 Connection in Half Duplex Mode

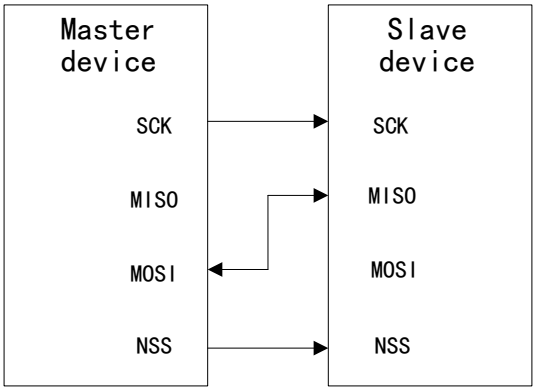


Figure 96 Connection in Simplex Mode (the master is used for receive and the slave is used for transmit)

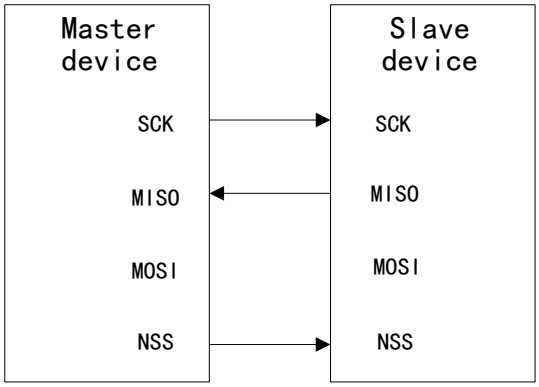
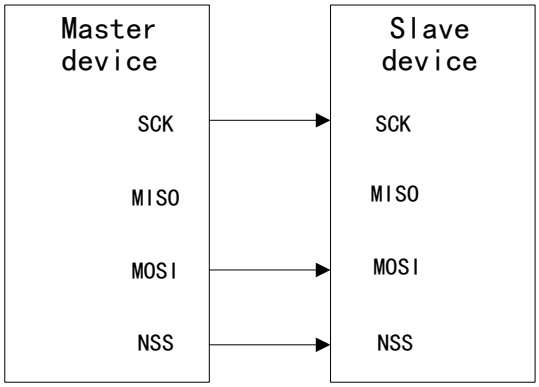


Figure 97 Connection in Simplex Mode(the master only transmit and the slave receive)



18.5.5.1 Transmitting and receiving of data

In order to prevent overrun when the data frame is short and ensure that SPI can work continuously, all SPI data need to pass through the 32-bit embedded FIFO. Each direction will have its own FIFO, TXFIFO and RXFIFO.

Handle FIFO according to duplex mode, data frame format, access size executed on FIFO data register and whether to use data package to process FIFO when accessing FIFO.

After read access to SPI_DR register, the earliest values that have not been read yet and are stored in RXFIFO will be returned. After write access to SPI_DR, the written data will be stored in TXFIFO at the end of the transmit queue. The TXFIFOVAL and RXFIFOVAL bits indicate the current occupancy levels of the two FIFO.

The read access to SPI_DR register must be managed by RXFIFONEFLG event. When the data are stored in RXFIFO and reach the threshold value (defined by previous bit), this event will be triggered; when RXFIFONEFLG is cleared, RXFIFO will be regarded to be empty, and in the similar way, the write access to the data frame to be transmitted is managed by TXFIFOEFLG event. When TXFIFO is less than or equal to half of its capacity, RXFIFONEFLG event will be triggered; otherwise, TXFIFOEFLG will be cleared, meanwhile, it will be regarded that there are data stored in TXFIFO. Therefore, when the data frame format is less than or equal to one byte, RXFIFO can store 4 data frames at most, and TXFIFO can store 3 data frames. When the software attempts to write more data to TXFIFO in 16-bit mode, this difference can prevent the three or 8-bit data frames that have been stored in TXFIFO from being damaged. TXFIFOEFLG and RXFIFONEFLG events can be polled or handled by interrupt.

18.5.5.2 Sequence processing

In transmitting data, multiple data can be formed into a sequence in order. When the transmission is started, TXFIFO will transmit continuously in order.

In single receive mode, in half-duplex mode, when SPI is enabled, the master

device will immediately receive the sequence until SPI is disabled or the single receive mode is disabled . When the data frame starts transmission, the slave cannot control the data sequence, so the slave must prepare the data before the transmission, to ensure there are data to be transmitted in TXFIFO.

When there are multiple slave devices, each sequence needs to correspond to different slave devices, so NSS pulse should be used to separate the sequence to ensure it is correct.

Note:

(1) Check whether the data transmission is completed according to TXFIFOVAL bit and BUSYFLG bit, and the clock output will stop when the transmission is completed.

(1) In packet mode, special attention should be paid to empty bytes when the data being transmitted are odd.

(2) In single receive mode of the master device, it is required to disable SPI or single receive mode to stop clock output.

(3) Master the correct receiving time to ensure the correct data transmission

(4) The action of disabling should be performed between the sampling time of first bit and the first bit of the next byte.

18.5.5.3 Data packing

If the data frame is less than or equal to one byte, when executing 16-bit read and write access to SPI_DR register, the data will be packed automatically and double data can be processed in parallel. After conducting write access to SPI_DR, 2-byte data will be transmitted; if the threshold value of RXFIFO is set to 16 bits, a receive RXFIFONEFLG event will be generated.

For a single RXFIFONEFLG event, the data receiver shall perform one read operation to SPI_DR, and only after that, can it obtain all data.

Note: The threshold value of RXFIFO should be consistent with the bit width of follow-up data access.

18.5.6 DMA Function

The request/response DMA mechanism in SPI facilitates high-speed data transmission, improves the system efficiency and enable to transfer data to SPI transmit buffer promptly, and the receive buffer can read the data in time to prevent overrun.

When SPI only transmits data, it is only needed to enable DMA transmission channel.

When SPI only receives data, it is only needed to enable DMA receiving channel.

DMA function of SPI mode can be enabled by configuring TXDMAEN and

RXDMAEN bits of SPI_CR2 register.

- When transmitting: When TXFIFOEFLG flag bit is set to 1, issue the DMA request, DMA controller writes data to SPI_DR, and then the TXFIFOEFLG flag bit will be cleared.
- When receiving: When setting RXFIFONEFLG flag bit to 1, issue the DMA request, DMA controller reads data from SPI_DR register, and then RXFIFONEFLG flag bit is cleared.

By monitoring BUSYFLG flag bit, confirm whether SPI communication is over after DMA has transferred all data to be transmitted in transmitting mode, which can avoid damaging the transmission of last data.

18.5.7 Communication rate

Due to the limitation of IO delay, SPI master/slave communication can support a maximum SPI clock frequency of 16MHz.

During the master communication process, SPI supports normal communication with a maximum of 2 divisions of the system clock. However, if DMA is used to send and receive a large amount of data, there may be discontinuity. If uninterrupted communication is required, it is recommended to configure the system clock to be 4 times the SPI clock. For example, when 16MHz communication is needed, the system clock can be configured to be 64MHz, and the BR bit of SPI_CR1 can be configured to work in 4 frequency divisions.

During the slave communication process, SPI supports normal communication with a maximum of 2 divisions of the system clock. However, when using DMA to send and receive large amounts of data, to avoid overflow of received data and underflow of transmit data, it is recommended to configure the system clock to four times the frequency of the external host's SPI clock. For example, when 16MHz communication from the external host is required, it is recommended to configure the system clock to 64MHz.

18.5.8 Disable SPI

After data transmission is over, end the communication by disabling SPI module.

When data are being transmitted or there are data in TXFIFO, it is not allowed to disable SPI by operating SPIEN bit in SPI_CR1 register. If SPIEN=0 is set, the clock signal will be transmitted continuously until the peripheral is enabled again. Certain steps are required to disable SPI to prevent the above situations.

Steps of disabling SPI

(1) Wait for TXFIFOEFLG to set to 1

(2) Wait for clearing RXFIFONEFLG flag bit to 0

(3) Wait for clearing BUSYFLG flag bit to 0

(4) Disable SPI (SPIEN=0)

18.5.9 SPI interrupt

An interrupt can be triggered by the following events during SPI operation:

- TXFIFO ready for loading
- RXFIFO receives data
- Receive the overflow

18.6 Register address mapping

Table 62 SPI Register Address Mapping

Register name	Description	Offset address
SPI_CR1	SPI control register 1	0x00
SPI_CR2	SPI control register 2	0x04
SPI_SR	SPI status register	0x08
SPI_DR	SPI data register	0x0C

18.7 Register functional description

These peripheral registers can be operated by half word (16 bits) or word (32 bits).

18.7.1 SPI control register 1 (SPI_CR1)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	BIDIMEN	R/W	Bidirectional Mode Enable 0: Double-line bidirectional mode 1: Single-line bidirectional mode Single-line bidirectional transmission means: transmission between MOSI pin of data master and MISO pin of slave.
14	BIDIOEN	R/W	Bidirectional Mode Output Enable 0: Disable (receive-only ode) 1: Enable (transmit-only mode) When BIDIMEN=1, namely in single-line bidirectional mode, this bit decides the transmission direction of transmission line.
13:12	Reserved		
11	DFF	R/W	Data Frame Length Format Select 0: 8-bit data frame format 1: 16-bit data frame format Note: For correct operation, write operations to this bit should only be performed when SPI is disabled (SPIEN = "0").

Field	Name	R/W	Description
10	RXONLY	R/W	Receive Only Mode Enable 0: Transmit and receive at the same time 1: Receive-only mode This bit only determines the direction of data transmission in the dual-line bidirectional mode (BIDIMEN=0)
9	SSM	R/W	Software Slave Device Enable 0: Disable 1: Enable When SSM is set, the level of NSS pin is determined by SSM.
8	SSI	R/W	Internal Slave Device Select Determines the level on the NSS pin, and I/O operations on the NSS pin are invalid. This bit can be set effectively only when SPI_CR1.SSM=1.
7	LSBFIRST	R/W	LSB First Transfer Select 0: First transmit the most significant bit (MSB) 1: First transmit the least significant bit (LSB)
6	SPIEN	R/W	SPI Device Enable 0: Disable 1: Enable Note: When SPI device is disabled, please operate according to the process of disabling SPI.
5:3	BR	R/W	Baud Rate Divider Factor Select 000: DIV=2 001: DIV=4 010: DIV=8 011: DIV=16 100: DIV=32 101: DIV=64 110: DIV=128 111: DIV=256 Baud rate=Fmaster/DIV Note: This bit cannot be modified during communication
2	MSTR	R/W	Master/Salve Mode Configure 0: Configure as slave mode 1: Configure as master mode Note: This bit cannot be modified during communication.
1	CPOL	R/W	Clock Polarity Configure The state maintained by SCK when SPI is in idle state. 0: SCK low 1: SCK high Note: This bit cannot be modified during communication.
0	CPHA	R/W	Clock Phase Configure This bit indicates on the edge of which clock to start sampling 0: On the edge of the first clock 1: On the edge of the second clock Note: This bit cannot be modified during communication.

18.7.2 SPI control register 2 (SPI_CR2)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	TXFIFOEIEN	R/W	Transmit FIFO Empty Interrupt Enable 0: Disable 1: Enable An interrupt occurs when there is no data in the transmitting FIFO
6	RXFIFONEIEN	R/W	Receive FIFO Not Empty Interrupt Enable 0: Disable 1: Allowed An interrupt is generated when there is data in the receiving FIFO
5	OVRIEN	R/W	Overflow interrupt enable 0: Disable 1: Enable When OVRFLG fag bit is set to 1, an interrupt request will be generated
4:2	Reserved		
1	TXDMAEN	R/W	Transmit Buffer DMA Enable When this bit is set, once TXFIFOEFLG flag is set, DMA request will be issued. 0: Disable 1: Enable
0	RXDMAEN	R/W	Receive Buffer DMA Enable When RXDMAEN=1, once RXFIFONEFLG flag is set, DMA request will be issued. 0: Disable 1: Enable

18.7.3 SPI status register (SPI_SR)

Offset address: 0x08

Reset value: 0x0000 0002

Field	Name	R/W	Description
31:13	Reserved		
12:11	TXFIFOVAL	R	FIFO Transmit Leve Select 00: FIFO is empty 01: FIFO/4 10: FIFO/2 11: FIFO is full (it can be considered as full when the threshold value of FIFO is greater than 1/2) Note: This bit is set to 1 or cleared to 0 by hardware.
10:9	RXFIFOVAL	R	FIFO Receive Leve Select 00: FIFO is empty 01: FIFO/4 10: FIFO/2

Field	Name	R/W	Description
			11: FIFO is full (it can be considered as full when the threshold value of FIFO is greater than 1/2) Note: This bit is set to 1 or cleared to 0 by hardware.
8	Reserved		
7	BUSYFLG	R	Busy Flag This bit indicates the work state of SPI 0: SPI is idle 1: SPI is communicating This bit can be set or reset by hardware
6	OVRFLG	R	Overrun Occur Flag This bit indicates whether overrun occurs or not 0: Not occurred 1: Occurred This bit can be set by hardware and reset by software.
5:2	Reserved		
1	TXFIFOEFLG	R	Transmit Buffer Empty Flag This bit indicates that the transmit buffer is empty or not 0: Not empty 1: Empty
0	RXFIFONEFLG	R	Receive Buffer Not Empty Flag This bit indicates that the receive buffer is empty or not 0: Empty 1: Not empty

18.7.4 SPI data register (SPI_DR)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
13:16	Reserved		
15:0	DATA	R/W	Transmit Receive Data register Store the data to be transmitted or received. The size of the FIFO is consistent with the length of the data frame, that is, for 8-bit data, DATA[7:0] will be used when transmitting and receiving data, and DATA[15:8] is invalid; for 16-bit data, DATA[15:0] will be used when transmitting and receiving data. The data register is used to connect the RX and TX FIFOs. Reading the data register accesses the RXFIFO; writing to the data register accesses the TX FIFO. Note: This register only supports 32-bit access.

19 Analog-to-digital converter (ADC)

19.1 Introduction

ADC with 12-bit precision has 8+2 external channels (2 channels for internal and external multiplexing) and 5 internal channels. It supports single, continuous or intermittent A/D conversion modes for each channel. ADC conversion results can be left-aligned or right-aligned and stored in 16-bit data register.

19.2 Main characteristics

(1) 12-bit resolution

(2) Up to 8+2 external channels and 5 internal channels

(3) Sampling intervals can be programmed per channel

(4) Data alignment with built-in data consistency

(5) Maximum conversion rate of 1.78 Msps at 12-bit resolution

(6) Trigger mode:
- On-chip timer signal trigger
- Software trigger

(7) Supports regular sequence, single, and continuous sampling modes

(8) Supports scan mode for single or continuous/discontinuous sequences; each ADC can convert multiple channels or scan a sequence of channels

(9) DMA request supporting regular data conversion
- DMA request will be generated after the conversion of regular channels is completed; the converted data result can be transmitted to the memory from the ADC_DR register

(10) ADC internal channel supports:
- TS
- VBG
- 1/2VDD
- OPAMP0
- OPAMP1

(11) Supports segmented sampling, with independent result registers for each conversion channel

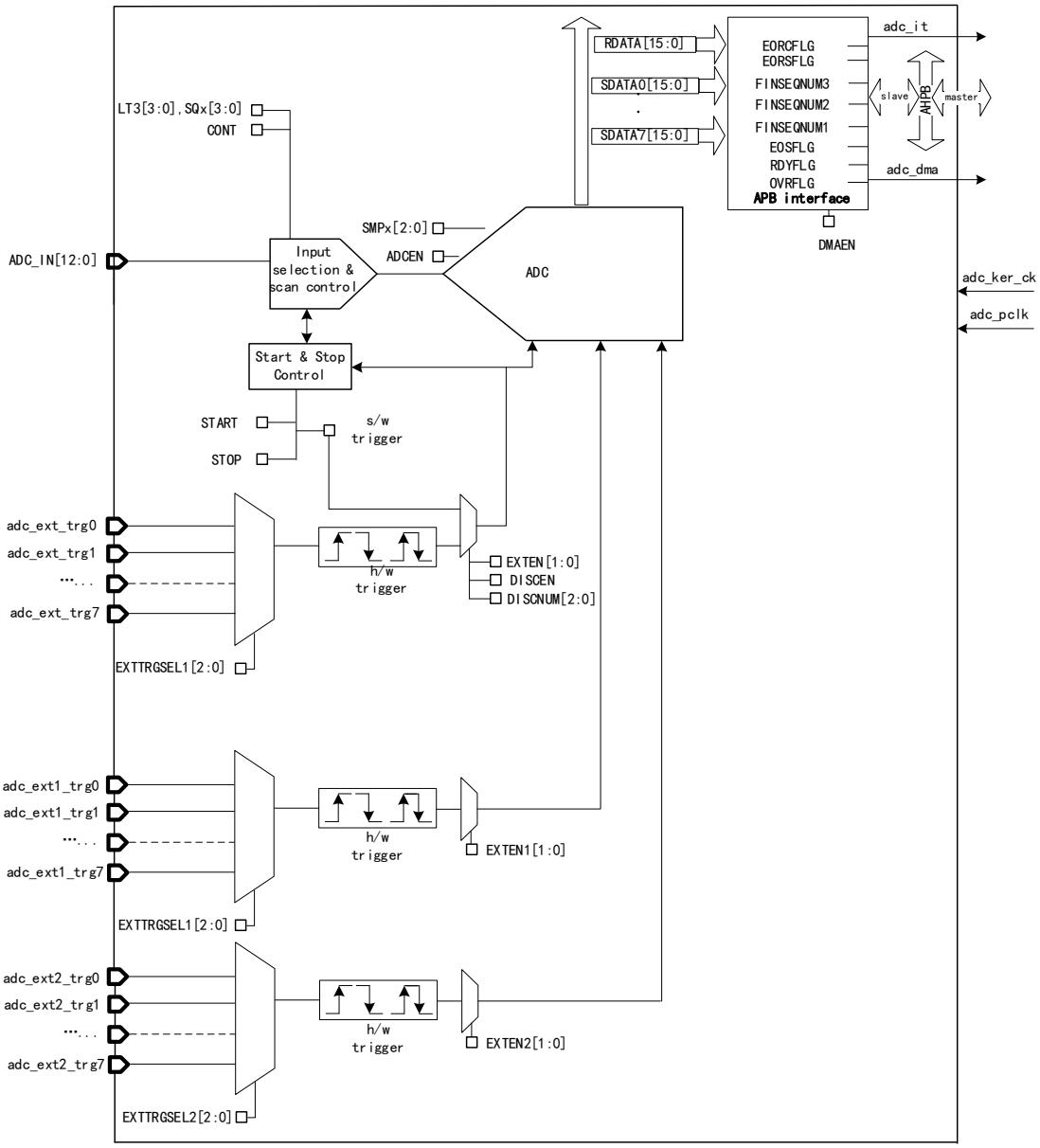
(12) Supports low-power modes, including automatic delay mode and automatic shutdown mode

(13) Interrupt

- End of conversion interrupt
- End of sequence conversion interrupt
- End of sampling phase interrupt
- ADC ready interrupt
- Overrun interrupt

19.3 Structure block diagram

Figure 98 Structure Block Diagram



19.4 ADC Functional Description

19.4.1 Clock

The dual-clock domain architecture means that the ADC clock is independent of the APB bus clock.

32-bit register read and write access via the APB bus; ADC sampling, conversion process, and data processing use an independent ADC clock (2, 4, 8 and 16 frequency division clocks can be configured based on the system clock).

19.4.2 Slave device APB interface

The ADC uses the APB slave port for control/status register access and data access. The characteristics of the APB interface are as follows:

- Word (32-bit) access

The APB slave interface does not support split/retry requests and never generates APB errors.

19.4.3 Single-ended channel

In single-ended input mode, the conversion voltage is the difference between the current input channel voltage and external V_{IN}-0.

19.4.4 ADC switch control

Setting the ADCEN bit in the ADCx_CR register to 1 enables the ADC. Conversion can be started when the RDYFLG flag of the ADC is set to 1.

Subsequently, regular conversion can be started by setting START to 1, or by an external trigger event if the trigger is enabled.

Procedure to disable the ADC by software:

(1) Check if START is 0 to ensure no conversion is currently being performed. If necessary, set STOP to 1, then wait for STOP=0 to stop any ongoing regular conversions.

(2) If required by the application, clear ADCEN to 0 to disable the ADC.

19.4.5 Restrictions when writing to ADC control bits

When the ADC is disabled, the software can configure and enable the ADC clock by writing to the RCU control bits. Software is only allowed to write to the START and STOP bits of the ADC_CR register when the ADC is enabled.

For all other control bits in the ADC_CFG1, ADC_CFG2, ADC_CFG3, ADC_SMP1, ADC_SMP2, ADC_SQ1, ADC_SQ2, ADC_SQ3, ADC_SEQNUM, and ADC_IER registers, software is only allowed to write to these bits when the ADC is enabled and no regular conversion is in progress.

19.4.6 Channel selection (ADC_SQx)

A regular conversion group consists of a maximum of 16 conversions. The regular channels and their order in the conversion sequence must be selected in the ADC_SQ1, ADC_SQ2, and ADC_SQ3 registers. The total number of conversions in the regular conversion group must be written to the LT3[3:0] bits in the ADCx_SQ1 register.

The ADC_SQ1, ADC_SQ2, and ADC_SQ3 registers cannot be modified when a regular conversion may be in progress. Therefore, STOP=1 must be written first to stop ADC regular conversion.

It should be noted that:

- When the ADC selects channel 1100
  - If it is necessary to measure the output signal of OPAMP0 and at the same time, the ADC needs to collect the OPAMP signal, then configure the OPA0EN, OPA0INSEL, and OPA0OUTSEL of the OPA_CR register of the OPAMP module to all be 1 to ensure the normal operation of OPAMP0.
  - If it is not necessary to measure the output signal of OPAMP0 and only the ADC needs to collect the OPAMP signal, then configure the OPA_CR registers of the OPAMP module, with OPA0EN and OPA0INSEL both set to 1 and OPA0OUTSEL set to 0, to ensure the normal operation of OPAMP0.
  - If only the signal at the PA0 pin is measured, configure the OPA_CR register of the OPAMP module to set OPA0EN to 0 and OPA0OUTSEL to 1, so that OPAMP0 is turned off and the PA0 signal is available for ADC detection.
- When the ADC selects channel 1101
  - If it is necessary to measure the output signal of OPAMP1 and at the same time, the ADC needs to collect the OPAMP signal, then configure the OPA1EN, OPA1INSEL, and OPA1OUTSEL of the OPA_CR register of the OPAMP module to all be 1 to ensure the normal operation of OPAMP1.
  - If it is not necessary to measure the output signal of OPAMP1 and only the ADC needs to collect the OPAMP signal, then configure the OPA_CR registers of the OPAMP module, with OPA1EN and OPA1INSEL both set to 1 and OPA1OUTSEL set to 0, to ensure the normal operation of OPAMP1.
  - If only the signal at the PA5 pin is measured, configure the OPA_CR register of the OPAMP module to set OPA1EN to 0 and OPA1OUTSEL to 1, so that OPAMP1 is turned off and the PA5 signal is available for ADC detection.

19.4.7 Configurable sampling time per channel (ADC_SMPx)

Before starting conversion, the ADC must establish a direct connection between

the voltage source to be measured and the ADC's built-in sampling capacitor. The sampling time must be sufficient for the input voltage source to charge the embedded capacitor to the input voltage level.

Different sampling times can be used during sampling, which can be programmed via the SMPx[2:0] bits in the ADC_SMPx register, allowing independent sampling times for each channel.

The formula for total conversion time is as follows:

$$T_{CONV} = \text{Sampling time} + 14 \text{ ADC clock cycles}$$

19.4.8 Single conversion mode (CONT=0)

In single conversion mode, the ADC performs all conversions on the channel once. When the CONT bit is 0, this mode can be started in the following ways:

- Set the START bit in the ADC_CR register to 1 (for regular channels)
- External hardware trigger event (for regular channels)

In a regular sequence, after each conversion is completed:

- The conversion data is stored in the ADC_DR register
- The EORCFLG (end of regular conversion) flag is set to 1
- An interrupt is generated when EORCIEN is set to 1

After the regular sequence is completed:

- The EORSFLG (end of regular sequence) flag is set to 1
- An interrupt is generated when EORSIEN is set to 1

Subsequently, the ADC stops working until a new external regular trigger occurs, or START is set to 1 again.

Note: To convert a single channel, program the sequence length to 1.

19.4.9 Continuous conversion mode (CONT=1)

This mode is only applicable to regular channels.

In continuous conversion mode, if a software or hardware regular trigger event occurs, the ADC performs all regular conversions on the channel once, then automatically restarts and continues to perform each conversion in the sequence. When the CONT bit is 1, this mode can be started by an external trigger or by setting the START bit in the ADC_CR register to 1.

In a regular sequence, after each conversion is completed:

- The conversion data is stored in the ADC_DR register
- The EORCFLG (end of conversion) flag is set to 1
- An interrupt is generated when EORCIEN is set to 1

After the conversion sequence is completed:

- The EORSFLG (end of sequence) flag is set to 1
- An interrupt is generated when EORSIEN is set to 1

Subsequently, a new sequence is restarted immediately, and the ADC continues to repeat the conversion sequence.

Note: To convert a single channel, program the sequence length to 1.

Discontinuous mode and continuous mode cannot be enabled simultaneously: DISCEN and CONT must not be set to 1 at the same time.

19.4.10 Start conversion (START)

Software starts ADC regular conversion by setting START to 1.

After START is set to 1, conversion starts:

- Immediately: when EXTEN=00 (software trigger)
- At the next valid edge of the selected regular hardware trigger: when EXTEN is not equal to 00
- In single mode using software regular trigger
  - Reset to zero as soon as the regular conversion sequence ends (EORSFLG is set to 1) or if DISCEN=1, the subgroup processing is completed
- In all cases
  - Cleared after executing the STOP program called by software

Note: In continuous mode, since the sequence restarts automatically, the START bit will not be cleared by hardware when EORSFLG is set to 1.

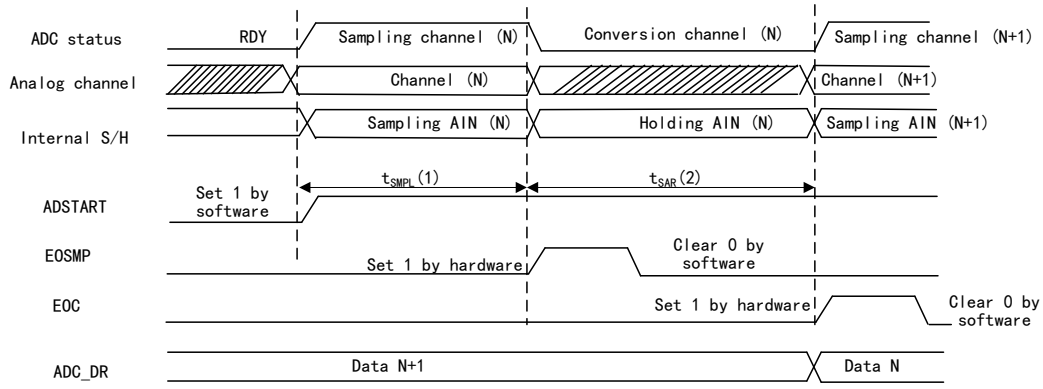
If hardware triggering is selected in single mode (CONT=0 and EXTEN is not equal to 00), the START bit will not be cleared by hardware when EORSFLG is set to 1. This allows the software to avoid resetting START for the next hardware trigger event. This ensures that no subsequent hardware triggers are missed.

Note: When software triggering is selected, the START bit should not be set to 1 if the EORCFLG flag is still high.

19.4.11 ADC timing

The time elapsed from the start to the end of conversion is the sum of the configured sampling time and conversion time.

Figure 99 Analog-to-Digital Conversion Time



Note:

(1) t_{SMPL} depends on SMPx[2:0]

(2) t_{SAR} is the conversion time: 14 ADC_CLK cycles

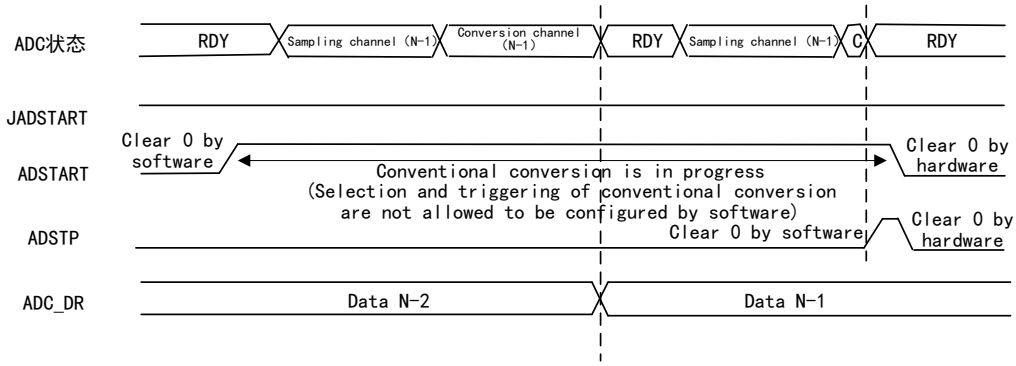
19.4.12 Stop ongoing conversion (STOP)

The software decides whether to stop the conversion. To stop an ongoing regular conversion, STOP should be set to 1. Stopping the conversion will reset the ongoing ADC operation. The ADC can then be reconfigured to prepare for new operations.

If the STOP bit is set to 1 by software, all ongoing regular conversions will be aborted, and some conversion results will be discarded.

After the execution of this program, the STOP/START bit will be cleared by hardware. The software must poll START until it is reset to determine that the ADC has completely stopped running.

Figure 100 Stop Ongoing Regular Conversion



19.4.13 External trigger conversion and trigger polarity (EXTTRGSEL, EXTEN)

Conversions or conversion sequences can be triggered by software or external events. If the EXTEN control bit is not equal to "00", external events can trigger conversions.

After the software sets the START bit to 1, the regular trigger selection becomes valid. Hardware triggers occurring during conversion will be ignored. If the START bit is 0, any regular hardware triggers that occur will be ignored.

The following table provides the correspondence between EXTEN values and trigger polarities.

Table 63 Configure Trigger Polarity for Regular External Triggers

EXTEN	Source
00	Disable hardware trigger detection, enable software trigger detection
01	Hardware trigger detection on rising edge
10	Hardware trigger detection on falling edge

EXTEN	Source
11	Perform hardware trigger detection on falling edges and falling edges

The EXTTRGSEL control bit is used to select the trigger for regular group conversion from 8 possible events.

Note: The regular trigger selection cannot be changed in real-time.

The following table lists all possible external triggers of the ADC for regular conversions.

Table 64 External Trigger of Regular Channel

Name	Source	Type	EXTTRGSEL[2:0]
adc_ext_trg0	Atimer TRG0	Internal signal from on-chip timers	000
adc_ext_trg1	Atimer TRG1	Internal signal from on-chip timers	001
adc_ext_trg2	Atimer TRG2	Internal signal from on-chip timers	010
adc_ext_trg3	Gtimer TRG0	Internal signal from on-chip timers	011
adc_ext_trg4	-	Reserved	100
adc_ext_trg5	-	Reserved	101
adc_ext_trg6	-	Reserved	110
adc_ext_trg7	-	Reserved	111

19.4.14 Discontinuous mode (DISCEN, DISCNUM)

Regular group mode

This mode can be enabled by setting the DISCEN bit in the ADC_CFG register to 1.

This mode is used to convert short sequences containing n (n ≤ 8) conversions, which are part of the conversion sequence selected in the ADC_SQx register. The value of n can be specified by writing to the DISCNUM[2:0] bits in the ADC_CFG register.

When an external trigger occurs, the next n conversions selected in the ADC_SQx register will start until all conversions in the sequence are completed. The total sequence length is defined by the LT[3:0] bits in the ADC_SQ1 register.

Each conversion generates an EORCFLG event, and the last conversion also generates an EORSFLG event. All subsequent trigger events will restart the entire sequence.

Note: When converting a regular group in discontinuous mode, there is no reverse (the number of conversions in the last subgroup of the sequence is less than n).

After converting all subgroups, the next trigger signal will start the conversion of the first subgroup.

Discontinuous mode and continuous mode cannot be enabled simultaneously. If both modes are enabled simultaneously, the ADC will recognize that the continuous mode is disabled and continue to perform related operations.

19.4.15 End of conversion (EORCFLG)

The ADC notifies the application each time a regular conversion end event occurs. The ADC immediately sets the EORCFLG flag to 1 when new regular conversion data appears in the ADC_DR register. An interrupt can be generated if the EORCIEN bit is set to 1. The EORCFLG flag can be cleared by software writing 1 to it or reading the ADC_DR.

19.4.16 End of conversion sequence (EORSFLG)

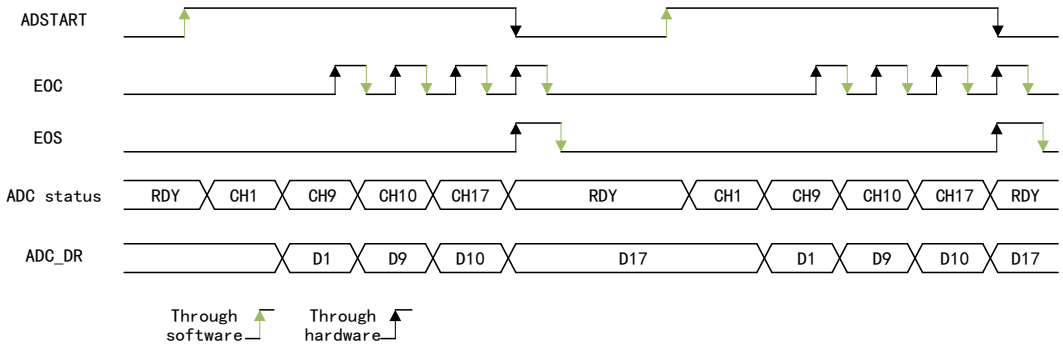
The ADC notifies the application each time a regular sequence end event occurs.

The ADC immediately sets the EORSFLG flag to 1 when the last data of the regular conversion sequence appears in the ADCx_DR register. An interrupt can be generated if the EORSIEN bit is set to 1. The EORSFLG flag can be cleared by software writing 1 to it.

19.4.17 Example timing diagrams (single mode/continuous mode, hardware/software trigger)

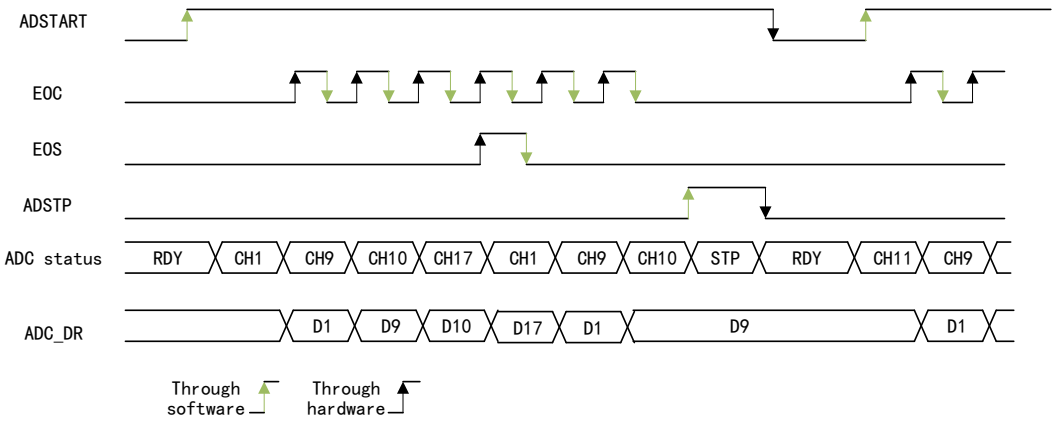
When EXTEN=00, CONT=0, and the selected channels=1, 9, 10, 17, the timing diagram is as follows:

Figure 101 Single sequence conversion, software trigger



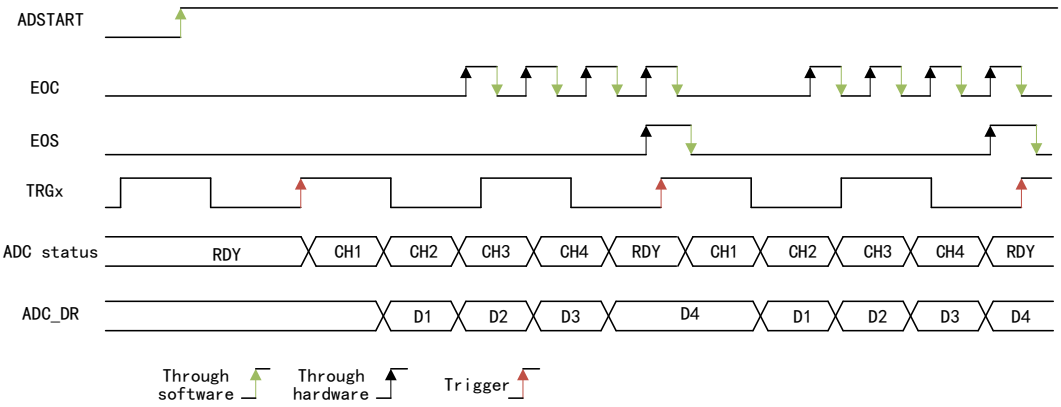
When EXTEN=00, CONT=1, and the selected channels=1, 9, 10, 17, the timing diagram is as follows:

Figure 102 Continuous sequence conversion, software trigger



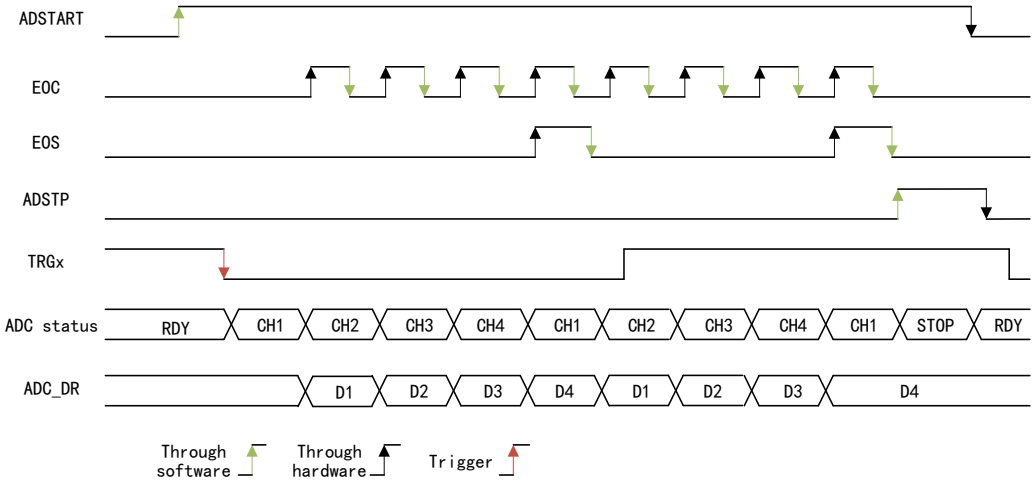
When TRGx (overfrequency) is selected as the trigger source, EXTEN=01, CONT=0, and the selected channels=1, 2, 3, 4, the timing diagram is as follows:

Figure 103 Single sequence conversion, hardware trigger



When TRGx (overfrequency) is selected as the trigger source, EXTEN=01, CONT=1, and the selected channels=1, 2, 3, 4, the timing diagram is as follows:

Figure 104 Continuous sequence conversion, hardware trigger



19.4.18 Data alignment

The ALIGN bit in the ADC_CFG register selects the alignment of the converted data storage. Data can be left-aligned or right-aligned, as shown in the following figure.

For regular group channels, there is no need to subtract an offset value, so only 12 bits are valid.

Figure 105 Data right alignment

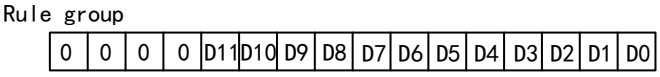
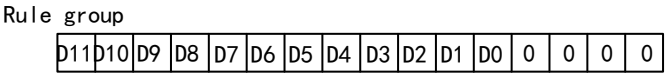


Figure 106 Data left alignment



The data of the sequential segmented sampling mode is stored in ADC_DR(0-7) register and can be left or right-aligned.

19.4.19 DMA request

Since the converted values of regular channels are stored in a single data register, DMA needs to be used when converting multiple regular channels to avoid losing data already stored in the ADC_DR register.

A DMA request is generated only at the end of the conversion of a regular channel, and the converted data is transferred from the ADC_DR register to the user-specified destination address.

Depending on the application, two different DMA modes are recommended, and the corresponding mode can be configured using the DMACFG bit of the ADCx_CFGR register:

(3) DMA single mode (DMACFG=0): This mode should be selected if DMA is programmed to transfer a fixed number of data.

- DMA programming is used to transmit the fixed-length data
- In this mode, ADC will generate a DMA request every time it converts data effectively. When ADC conversion is restarted, ADC will stop generating DMA request
- When the number of ADC conversions reaches the length of DMA, software is required to configure the STOP bit to stop ADC

(4) DMA circular mode (DMACFG=1): This mode should be selected if DMA is programmed in circular mode.

- DMA programming is in circular mode or double-buffer mode
- In this mode, when ADC conversion is started again and the converted data is valid, a DMA request will be generated

19.4.20 ADC overrun

ADC overrun means when the converted data is not read by DMA or CPU on time, another converted data will take effect.

When EORCFLG bit is 1 but another new conversion has been completed, an overrun event will occur, and OVRFLG bit of register ADC_SR will be set to 1; if ORIEN bit is set to 1, an overrun interrupt will be generated.

It is determined by OVRMOD bit of configuration register ADC_CFG1 that the data in the ADC data register are held or overwritten when an overrun event occurs:

- OVRMOD is 0: When an overrun event is detected, old data will be held in ADC_DR register
- OVRMOD is set to 1: When an overrun event is detected, ADC_DR register will overwrite the data by the last converted data

19.4.21 ADC (low-power mode)

Automatic delay conversion mode

The ADC executes the automatic delay conversion mode controlled by the DLYCM configuration bit. Automatic delay conversion can be used to simplify software and optimize the performance of applications using low-frequency clocks (which may be at risk of ADC overflow).

When DLYCM=1, a new conversion can only start when all previous data in the same group has been processed:

- For regular conversion: When the ADC_DR register has been read.

In this way, the speed of the ADC can be automatically adjusted to match the speed at which the system reads data.

Note: The sequential segmented sampling mode does not support.

Automatic shutdown mode

This mode can greatly reduce the power consumption of application, and is suitable for applications with relatively few conversions or long conversion request time interval. Automatic shutdown mode can be used in combination with automatic delay conversion mode in low-frequency application.

Automatic shutdown mode can be enabled by setting AUTOFF bit of configuration register ADC_CFG1 to 1. When AUTOFF bit is set to 1 and there is no ADC conversion, it will be powered off automatically, and when the conversion is started, ADC will be woken up automatically.

Note: The sequential segmented sampling mode does not support. This mode is applicable to the intermittent sampling mode (DISCEN is 1). During the intermittent sampling process, if the current sub-segment conversion is completed and there are no new trigger events, the hardware will shut down the analog ADC and enter a low-power mode, waiting for a new trigger event to automatically wake it up. After a new trigger event occurs, the analog ADC requires approximately 1.9μs of setup time. Only after this can the subsequent sampling conversion be carried out.

19.4.22 ADC interrupt

Interrupts are generated in the following cases:

- At the end of any conversion of the regular group (EORCFLG flag)
- At the end of the conversion sequence of the regular group (EORSFLG flag)
- Interrupt at the end of the first segment of sequence segmented sampling (FINSEQNUM1 flag)
- Interrupt at the end of the second segment of sequence segmented sampling (FINSEQNUM2 flag)
- Interrupt at the end of the third segment of sequence segmented sampling (FINSEQNUM3 flag)
- Overrun interrupt (OVRFLG flag)
- Interrupt at the end of regular conversion sampling (EOSFLG flag)
- Enable ADC ready interrupt (RDYFLG flag)

Separate interrupt enable bits can be used for flexibility.

Table 65 ADC Interrupt

Interrupt event	Event flag	Enable control bit
End of conversion of regular group	EORCFLG	EORCIEN
End of conversion sequence of regular group	EORSFLG	EORSIEN
End of the first segment of sequential segmented sampling	FINSEQNUM1	FINSEQNUM1IEN
End of the second segment of sequential segmented sampling	FINSEQNUM2	FINSEQNUM2IEN
End of the third segment of sequential segmented sampling	FINSEQNUM3	FINSEQNUM3IEN

Data overrun of regular group	OVRFLG	ORIEN
End of regular conversion sampling	EOSFLG	EOSFIEN
Enable ADC ready	RDYFLG	RDYIEN

19.4.23 Temperature calculation

The conversion formula between the sampling result Dsample of the temperature sensor and the current temperature T (unit: ℃) is as follows:

$$T = 25 + \frac{\text{Dsample} - \text{Tssample}}{\text{slope}}$$

Note:

(1) T= Current temperature

(2) Dsample= Tensor voltage at the actual temperature

(3) Tssample= For the 25℃ voltage, please refer to the value at address 0x0010 1C14 for details

(4) slope= Average slope，slope=3.802 mV/ºC

19.4.24 Direction for use

The process of enabling ADC through software:

(1) Write "1" to the ADRDY bit in the ADC_SR register to clear it to zero.

(2) Set ADCEN to 1.

(3) Wait until ADRDY=1 (ADRDY will be set by 1 after the ADC start time). This can be achieved by using an associated interrupt (setting ADRDYIE to 1).

(4) Write "1" to the ADRDY bit in the ADC_SR register to zero it (optional).

The process of disabling ADC through software:

(1) Check whether both START are 0 to ensure that no conversion is currently being performed. If necessary, set STOP to 1, and then wait until STOP=0 to stop any ongoing regular conversions.

(2) Clear ADCEN to 0

In the single-conversion mode, the ADC will perform all the conversions of the channel at once. When the CONT bit is 0, this mode can be initiated in the following way:

- set START position 1 in the ADC_CR register (for regular channels)
- External hardware trigger event (applicable to regular channels)

In a regular sequence, after each conversion is completed:

- The conversion data is stored in the 16-bit ADC_DR register

- Set the EORCFLG (End of Regular Conversion) flag to 1
- An interruption will occur when EORCIEN is at position 1

After the regular sequence is completed:

- Set the EORSFLG (End of Regular Sequence) flag to 1
- An interrupt will occur when EORSIEN is at position 1

Subsequently, the ADC will cease operation until a new external routine trigger occurs, or the START bit is reset to 1 again.

Note: To convert a single channel, you can program the sequence length to 1.

In continuous conversion mode, if a regular software or hardware trigger event occurs, the ADC will execute all the regular conversions of the channel once, and then automatically restart and continuously execute each conversion of the sequence. When the CONT bit is 1, this mode can be initiated by external trigger or by setting the START position 1 in the ADCx_CR register.

In a regular sequence, after each conversion is completed:

- The conversion data is stored in the 16-bit ADCx_DR register
- Set the EORCFLG (Conversion Completed) flag to 1
- An interruption will occur when EORCIEN is at position 1

After the conversion sequence is completed:

- Set the EORSFLG (sequence End) flag to 1
- An interruption will occur when EORSIEN is at position 1

Subsequently, the new sequence will be restarted immediately, and the ADC will continue to repeat the conversion sequence.

19.5 Register address mapping

Table 66 ADC Register Address Mapping Table

Register name	Description	Offset address
ADC_SR	ADC status register	0x00
ADC_IER	ADC interrupt enable register	0x04
ADC_CR	ADC control register	0x08
ADC_CFG1	ADC configuration register 1	0x0C
ADC_CFG2	ADC configuration register 2	0x10
ADC_SMP1	ADC sampling configuration register 1	0x14
ADC_SMP2	ADC sampling configuration register 2	0x18
ADC_CFG3	ADC configuration register 3	0x1C
ADC_SQ1	ADC sequence configuration register 1	0x2C
ADC_SQ2	ADC sequence configuration register 2	0x30

Register name	Description	Offset address
ADC_SQ3	ADC sequence configuration register 3	0x34
ADC_DR	ADC regular conversion data register	0x38
ADC_DRx	ADC data register x	0x3C+0x04*x (x=0~7)
ADC_SEQNUM	ADC sequential segmented sampling control register	0x5C

19.6 Register functional description

19.6.1 ADC status register (ADC_SR)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:11	Reserved		
10	FINSEQNUM3	R/W	NUM.3 of Sequential Section Sampling Finish 0: Not occurred 1: Occurred This bit can be cleared to 0 by writing 1 via software.
9	FINSEQNUM2	R/W	NUM.2 of Sequential Section Sampling Finish 0: Not occurred 1: Occurred This bit can be cleared to 0 by writing 1 via software.
8	FINSEQNUM1	R/W	NUM.1 of Sequential Section Sampling Finish 0: Not occurred 1: Occurred This bit can be cleared to 0 by writing 1 via software.
7:5	Reserved		
4	OVRFLG	R/W	ADC overrun This bit is set to 1 by hardware when an overrun event occurs on a regular channel, which means a new conversion has been completed when the EORCFLG flag is already set to 1. This bit can be cleared by software writing 1 to it. 0: No overrun event has occurred (or the flag event has been confirmed and cleared by software) 1: An overrun has occurred
3	EORSFLG	R/W	End of regular sequence flag Hardware sets this bit to 1 after the conversion of the regular channel sequence ends. This bit can be cleared by software writing 1 to it. 0: The regular conversion sequence is not completed (or the flag event has been confirmed and cleared by software) 1: The regular conversion sequence is completed
2	EORCFLG	R/W	End of conversion flag This bit is set to 1 by hardware each time a regular conversion of a channel ends and new data appears in the ADC_DR

Field	Name	R/W	Description
			register. This bit can be cleared by software writing 1 to it or reading the ADC_DR register. 0: The regular channel conversion is not completed (or the flag event has been confirmed and cleared by software) 1: The regular channel conversion is completed
1	EOSFLG	R/W	End of sampling flag This bit is set to 1 by hardware when the sampling phase ends during the conversion of any channel (regular channels only). 0: The sampling phase is not ended (or the flag event has been confirmed and cleared by software) 1: The sampling phase is ended
0	RDYFLG	R/W	ADC Ready Flag This bit can be cleared by software writing 1 to it. 0: ADC is not ready 1: ADC is ready to start conversion

19.6.2 ADC interrupt enable register (ADC_IER)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:11	Reserved		
10	FINSEQNUM3IEN	R/W	NUM.3 of Sequential Section Sampling Finish Interrupt Enable 0: Disable 1: Enable
9	FINSEQNUM2IEN	R/W	NUM.2 of Sequential Section Sampling Finish Interrupt Enable 0: Disable 1: Enable
8	FINSEQNUM1IEN	R/W	NUM.1 of Sequential Section Sampling Finish Interrupt Enable 0: Disable 1: Enable
7:5	Reserved		
4	ORIEN	R/W	Overrun interrupt enable This bit is set to 1 and cleared to 0 by software to enable/disable the overrun interrupt of regular conversions. 0: Overrun interrupt disabled 1: Overrun interrupt enabled. An interrupt is generated when the OVRFLG bit is set to 1. Note: Software is only allowed to write to this bit when START=0 (this ensures that no regular conversion is currently in progress).
3	EORSIEN	R/W	End of regular sequence of conversions interrupt enable This bit is set to 1 and cleared to 0 by software to enable/disable the interrupt at the end of the regular conversion sequence.

Field	Name	R/W	Description
			0: EOS interrupt disabled 1: EOS interrupt enabled. An interrupt is generated when the EORSFLG bit is set to 1. Note: Software is only allowed to write to this bit when START=0 (this ensures that no regular conversion is currently in progress).
2	EORCIEN	R/W	End of regular conversion interrupt enable This bit is set to 1 and cleared to 0 by software to enable/disable the interrupt at the end of regular conversion. 0: EOC interrupt disabled 1: EOC interrupt enabled. An interrupt is generated when the EORCFLG bit is set to 1. Note: Software is only allowed to write to this bit when START=0 (this ensures that no regular conversion is currently in progress).
1	EOSFIEN	R/W	End of sampling flag interrupt enable for regular conversions This bit is set to 1 and cleared to 0 by software to enable/disable the interrupt at the end of the sampling phase of regular conversion. 0: EOSF interrupt disabled 1: EOSF interrupt enabled. An interrupt is generated when the EOSFLG bit is set to 1. Note: Software is only allowed to write to this bit when START=0 (this ensures that no regular conversion is currently in progress).
0	RDYIEN	R/W	ADC Ready Interrupt Enable 0: Disable 1: Enable

19.6.3 ADC control register (ADC_CR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:30	MODESEL	R/W	ADC working mode select 00: Standard voltage, normal mode (5.5V to 2.7V) 01: Standard voltage, low power consumption mode (5.5V to 2.7V) 10: Low voltage, normal mode (2.7V to 2V) 11: Ultra-low voltage, normal mode (2V to 1.75V)
29:18	Reserved		
17	VREFSEL	R/W	VREF connect select 0: VREF connected to VDD5 1: VREF connected to IO input
16	TSENP	RW	TS enable signal High level is effective. 0: Disable 1: Enable
15:5	Reserved		

Field	Name	R/W	Description
4	STOP	R/W	ADC stop of regular conversion command This bit is set to 1 by software to stop and discard the ongoing regular conversion (STOP command). The bit is cleared by hardware when the conversion has been effectively discarded and the ADC regular sequence and trigger can be reconfigured. Subsequently, the ADC will be ready to receive a new command to start regular conversion (START command). 0: No ADC stop regular conversion command is currently being executed. 1: Writing 1 stops the ongoing regular conversion. A read value of 1 indicates that the STOP command is being executed. Note: Software is only allowed to set STOP to 1 when START=1 (the ADC is enabled, will eventually perform regular conversion, and there are no pending requests to disable the ADC).
3	Reserved		
2	START	R/W	ADC start of regular conversion This bit is set to 1 by software to start the regular channel conversion of the ADC. Depending on the value of the EXTEN configuration bit, conversion can start immediately (software trigger configuration) or after a regular hardware trigger event occurs (hardware trigger configuration). The bit is cleared by hardware: - In the single mode of using the software's regular trigger (CONT=0, EXTEN=0x0), the subgroup is reset to zero as soon as the regular conversion sequence ends (EORSFLG is set to 1) or if DISCEN=1. - In all cases: cleared by hardware when the STOP bit is cleared after the execution of the STOP command. 0: No ADC regular conversion is currently in progress. 1: Writing 1 starts regular conversion. A read value of 1 indicates that the ADC is running and will eventually convert regular channels. Note: Software is only allowed to set START to 1 when ADCEN=1 (the ADC is enabled, and there are no pending requests to disable the ADC).
1	Reserved		
0	ADCEN	R/W	ADC enable This bit is set to 1 and cleared by software to enable the ADC. 0: Disable ADC (OFF state) 1: Enable ADC

19.6.4 ADC configuration register 1 (ADC_CFG1)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:20	Reserved		
19:17	DISCNUM	R/W	Discontinuous mode channel count: These bits are written by software to define the number of regular channels converted in discontinuous sampling mode after receiving an external trigger. 000: 1 channel

Field	Name	R/W	Description
			001: 2 channels ... 110: 7 channels 111: 8 channels Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
16	DISCEN	R/W	Discontinuous mode for regular channels This bit is set to 1 and cleared to 0 by software to enable/disable discontinuous mode for regular channels. 0: Discontinuous mode for regular channels disabled 1: Discontinuous mode for regular channels enabled Note: Discontinuous mode and continuous mode cannot be enabled simultaneously: DISCEN and CONT must not be set to 1 at the same time. Software is only allowed to write to this bit when START=0 (to ensure no regular conversion is currently in progress).
15	AUTOFF	R/W	Auto-Off Mode Enable 0: Disabled 1: Enabled Note: Software is only allowed to write to this bit when START=0 (to ensure no conversion is currently in progress).
14	DLYCM	R/W	Delayed conversion mode This bit is set to 1 and cleared to 0 by software to enable/disable automatic delayed conversion mode. 0: Automatic delayed conversion mode disabled 1: Automatic delayed conversion mode enabled Note: Software is only allowed to write to this bit when START=0 (to ensure no conversion is currently in progress).
13	CONT	R/W	Single/continuous conversion mode for regular conversions This bit is set to 1 and cleared to 0 by software. When this bit is set to 1, regular conversions will continue until the bit is cleared. 0: Single conversion mode 1: Continuous conversion mode Note: Discontinuous mode and continuous mode cannot be enabled simultaneously: DISCEN and CONT must not be set to 1 at the same time. Software is only allowed to write to this bit when START=0 (to ensure no regular conversion is currently in progress).
12	OVRMOD	R/W	Overrun Mode This bit is set to 1 and cleared by software to configure the management method of data overrun. 0: If an overrun is detected, the ADCx_DR register retains the original data. 1: If an overrun is detected, the ADCx_DR register is overwritten by the previous conversion result. Note: Software is only allowed to write to this bit when START=0 (this ensures that no regular conversion is currently in progress).
11:10	EXTEN1	R/W	External trigger enable and polarity selection for regular channels

Field	Name	R/W	Description
			These bits are set to 1 and cleared by software to select the external trigger polarity and enable triggering of the regular group. 00: Disable hardware trigger detection (conversion can be started by software) 01: Perform hardware trigger detection on rising edge 10: Perform hardware trigger detection on falling edge 11: Perform hardware trigger detection on rising and falling edges Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
9	Reserved		
8:6	EXTTRGSEL1	R/W	External trigger selection for regular group These bits select the external event used to trigger conversion of the regular group. 000: ATMR TRG0 001: ATMR TRG1 010: ATMR TRG2 011: GTMR TRG0 1xx: Reserved Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
5	ALIGN	R/W	Data alignment This bit is set and cleared by software. 0: Right-aligned 1: Left-aligned Software is only allowed to write to these bits when START=0(to ensure no conversion is currently in progress).
4:2	Reserved		
1	DMACFG	R/W	Direct memory access configuration This bit is set to 1 and cleared to 0 by software to select between the two working modes of DMA, and is only valid when DMAEN=1. 0: Select DMA single mode 1: Select DMA circular mode Note: Software is only allowed to write to this bit when START=0 (to ensure no regular conversion is currently in progress).
0	DMAEN	R/W	Direct memory access enable This bit is set to 1 and cleared to 0 by software to enable the generation of DMA requests. 0: DMA disabled 1: DMA enabled Note: Software is only allowed to write to this bit when START=0 (to ensure no regular conversion is currently in progress).

19.6.5 ADC configuration register 2 (ADC_CFG2)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:12	Reserved		
11:8	TGAP	R/W	Time of The Gap between Conversions in Sequential Section Configure This bit configures the time interval (in ADC_CLK units) between the completion of the last conversion and the start of the next sampling in each segment of the sequence. 0x0: No gap 0x1: 2 ADC_CLK cycles gap 0x2: 4 ADC_CLK cycles gap ..... 0xF: 2^15 ADC_CLK cycles gap
7:1	Reserved		
0	SEQEN	R/W	Sequential Section Sampling Enable 0: Disable 1: Enable

19.6.6 ADC sampling configuration register 1 (ADC_SMP1)

Offset address: 0x14

Reset value: 0x1111 1111

Field	Name	R/W	Description
31	Reserved		
30:28	SMP7	R/W	Channel 7(SQx=1000) sampling time selection Refer to SMP0 bit.
27	Reserved		
26:24	SMP6	R/W	Channel 6(SQx=0111) sampling time selection Refer to SMP0 bit.
23	Reserved		
22:20	SMP5	R/W	Channel 5(SQx=0110) sampling time selection Refer to SMP0 bit.
19	Reserved		
18:16	SMP4	R/W	Channel 4(SQx=0101) sampling time selection Refer to SMP0 bit.
15	Reserved		
14:12	SMP3	R/W	Channel 3(SQx=0100) sampling time selection Refer to SMP0 bit.
11	Reserved		
10:8	SMP2	R/W	Channel 2(SQx=0011) sampling time selection Refer to SMP0 bit.
7	Reserved		
6:4	SMP1	R/W	Channel 1(SQx=0010) sampling time selection Refer to SMP0 bit.
3	Reserved		

Field	Name	R/W	Description
2:0	SMP0	R/W	Channel 0(SQx=0001) sampling time selection These bits are written by software to select the sampling time for each channel. During the sampling period, the channel selection bits must remain unchanged. 000: 2 ADC clock cycles 001: 4 ADC clock cycles 010: 8 ADC clock cycles 011: 16 ADC clock cycles 100: 32 ADC clock cycles 101: 64 ADC clock cycles 110: 128 ADC clock cycles 111: 256 ADC clock cycles Note: Software is only allowed to write to these bits when START=0 (to ensure no conversion is currently in progress).

19.6.7 ADC sampling configuration register 2 (ADC_SMP2)

Offset address: 0x18

Reset value: 0x1111 1111

Field	Name	R/W	Description
31:19	Reserved		
18:16	SMP12	R/W	Channel 12(SQx=1101) sampling time selection Refer to SMP0 bit.
15	Reserved		
14:12	SMP11	R/W	Channel 11(SQx=1100) sampling time selection Refer to SMP0 bit.
11	Reserved		
10:8	SMP10	R/W	Channel 10(SQx=1011) sampling time selection Refer to SMP0 bit.
7	Reserved		
6:4	SMP9	R/W	Channel 9(SQx=1010) sampling time selection Refer to SMP0 bit.
3	Reserved		
2:0	SMP8	R/W	Channel 8(SQx=1001) sampling time selection These bits are written by software to select the sampling time for each channel. During the sampling period, the channel selection bits must remain unchanged. 000: 2 ADC clock cycles 001: 4 ADC clock cycles 010: 8 ADC clock cycles 011: 16 ADC clock cycles 100: 32 ADC clock cycles 101: 64 ADC clock cycles 110: 128 ADC clock cycles 111: 256 ADC clock cycles

Field	Name	R/W	Description
			Note: Software is only allowed to write to these bits when START=0 (to ensure no conversion is currently in progress).

19.6.8 ADC configuration register 3 (ADC_CFG3)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
12:10	EXTTRGSEL3	R/W	External trigger selection for regular group 3 These bits select the external event used to trigger conversion of the regular group. 000: ATMR TRG0 001: ATMR TRG1 010: ATMR TRG2 011: GTMR TRG0 1xx: Reserved Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
9:8	EXTEN3	R/W	External trigger enable and polarity selection for regular channels 3 These bits are set to 1 and cleared by software to select the external trigger polarity and enable triggering of the regular group. 00: Disable hardware trigger detection (this bit is not allowed to be 00 in segmented sampling function) 01: Perform hardware trigger detection on rising edge 10: Perform hardware trigger detection on falling edge 11: Perform hardware trigger detection on rising and falling edges Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
7:5	Reserved		
4:2	EXTTRGSEL2	R/W	External trigger selection for regular group 2 These bits select the external event used to trigger conversion of the regular group. 000: ATMR TRG0 001: ATMR TRG1 010: ATMR TRG2 011: GTMR TRG0 1xx: Reserved Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
1:0	EXTEN2	R/W	External trigger enable and polarity selection for regular channels 2 These bits are set to 1 and cleared by software to select the external trigger polarity and enable triggering of the regular group. 00: Disable hardware trigger detection (this bit is not allowed to be 00 in segmented sampling function) 01: Perform hardware trigger detection on rising edge 10: Perform hardware trigger detection on falling edge

Field	Name	R/W	Description
			11: Perform hardware trigger detection on rising and falling edges Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).

19.6.9 ADC sequence configuration register 1 (ADC_SQ1)

Offset address: 0x2C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:28	SQ7	R/W	7th conversion in regular sequence These bits are written by software to assign the channel number as the seventh conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
27:24	SQ6	R/W	6th conversion in regular sequence These bits are written by software to assign the channel number as the sixth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
23:20	SQ5	R/W	5th conversion in regular sequence These bits are written by software to assign the channel number as the fifth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
19:16	SQ4	R/W	4th conversion in regular sequence These bits are written by software to assign the channel number as the fourth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
15:12	SQ3	R/W	3rd conversion in regular sequence These bits are written by software to assign the channel number as the third conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
11:8	SQ2	R/W	2nd conversion in regular sequence These bits are written by software to assign the channel number as the second conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
7:4	SQ1	R/W	1st conversion in regular sequence These bits are written by software to assign the channel number as the first conversion in the regular conversion sequence. 0000: Close all channels 0001: Open external channel 0- Measure external signal -PA7 0010: Open external Channel 1- Measure external signal -PA8 0011: Open external Channel 2- Measure external signal -PA9 0100: Open external channel 3- Measure external signal -PB0 0101: Open external channel 4- Measure external signal - PB1 0110: Open external channel 5- Measure external signal - PB12

Field	Name	R/W	Description
			0111: Open external channel 6- Measure external signal - PB5 1000: Open external channel 7- Measure the external signal - PB6 1001: Open the TS channel - Measure the temperature sensor signal 1010: Open the PMU channel - Measure the VBG voltage signal 1011: Open the PMU channel - Measure the VDD-DIV2 signal 1100: Open the OPAMP0 channel - Measure the OPAMP0 signal 1101: Open the OPAMP1 channel - Measure the OPAMP1 signal Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress). • When the ADC selects channel 1100 - If it is necessary to measure the output signal of OPAMP0 and at the same time, the ADC needs to collect the OPAMP signal, then configure the OPA0EN, OPA0INSEL, and OPA0OUTSEL of the OPA_CR register of the OPAMP module to all be 1 to ensure the normal operation of OPAMP0. - If it is not necessary to measure the output signal of OPAMP0 and only the ADC needs to collect the OPAMP signal, then configure the OPA_CR registers of the OPAMP module, with OPA0EN and OPA0INSEL both set to 1 and OPA0OUTSEL set to 0, to ensure the normal operation of OPAMP0. - If only the signal at the PA0 pin is measured, configure the OPA_CR register of the OPAMP module to set OPA0EN to 0 and OPA0OUTSEL to 1, so that OPAMP0 is turned off and the PA0 signal is available for ADC detection. • When the ADC selects channel 1101 - If it is necessary to measure the output signal of OPAMP1 and at the same time, the ADC needs to collect the OPAMP signal, then configure the OPA1EN, OPA1INSEL, and OPA1OUTSEL of the OPA_CR register of the OPAMP module to all be 1 to ensure the normal operation of OPAMP1. - If it is not necessary to measure the output signal of OPAMP1 and only the ADC needs to collect the OPAMP signal, then configure the OPA_CR registers of the OPAMP module, with OPA1EN and OPA1INSEL both set to 1 and OPA1OUTSEL set to 0, to ensure the normal operation of OPAMP1. - If only the signal at the PA5 pin is measured, configure the OPA_CR register of the OPAMP module to set OPA1EN to 0 and OPA1OUTSEL to 1, so that OPAMP1 is turned off and the PA5 signal is available for ADC detection.
3:0	LT3	R/W	Regular channel sequence length These bits are written by software to define the total number of conversions in the regular channel conversion sequence. 0000: 1 conversion 0001: 2 conversions ... 1111: 16 conversions Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).

19.6.10 ADC sequence configuration register 2 (ADC_SQ2)

Offset address: 0x30

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:28	Reserved		
27:24	SQ14	R/W	14th conversion in regular sequence These bits are written by software to assign the channel number as the fourteenth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
23:20	SQ13	R/W	13th conversion in regular sequence These bits are written by software to assign the channel number as the thirteenth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
19:16	SQ12	R/W	12th conversion in regular sequence These bits are written by software to assign the channel number as the twelfth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
15:12	SQ11	R/W	11th conversion in regular sequence These bits are written by software to assign the channel number as the eleventh conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
11:8	SQ10	R/W	10th conversion in regular sequence These bits are written by software to assign the channel number as the tenth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
7:4	SQ9	R/W	9th conversion in regular sequence These bits are written by software to assign the channel number as the ninth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).
3:0	SQ8	R/W	8th conversion in regular sequence These bits are written by software to assign the channel number as the eighth conversion in the regular conversion sequence Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).

19.6.11 ADC sequence configuration register 3 (ADC_SQ3)

Offset address: 0x34

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7:4	SQ16	R/W	16th conversion in regular sequence These bits are written by software to assign the channel number as the sixteenth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).

Field	Name	R/W	Description
3:0	SQ15	R/W	15th conversion in regular sequence These bits are written by software to assign the channel number as the fifteenth conversion in the regular conversion sequence. Note: Software is only allowed to write to these bits when START=0 (to ensure no regular conversion is currently in progress).

19.6.12 ADC regular conversion data register (ADC_DR)

Offset address: 0x38

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA	R	Regular Data converted These bits are read-only. They contain the conversion result of the last converted regular channel.

19.6.13 ADC data register 0 (ADC_DR0)

Offset address: 0x3C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA0	R	Sequence segmented sampling result data 0

19.6.14 ADC data register 1 (ADC_DR1)

Offset address: 0x40

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA1	R	Sequence segmented sampling result data 1

19.6.15 ADC data register 2 (ADC_DR2)

Offset address: 0x44

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA2	R	Sequence segmented sampling result data 2

19.6.16 ADC data register 3 (ADC_DR3)

Offset address: 0x48

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA3	R	Sequence segmented sampling result data 3

19.6.17 ADC data register 4 (ADC_DR4)

Offset address: 0x4C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA4	R	Sequence segmented sampling result data 4

19.6.18 ADC data register 5 (ADC_DR5)

Offset address: 0x50

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA5	R	Sequence segmented sampling result data 5

19.6.19 ADC data register 6 (ADC_DR6)

Offset address: 0x54

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA6	R	Sequence segmented sampling result data 6

19.6.20 ADC data register 7 (ADC_DR7)

Offset address: 0x58

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	SDATA7	R	Sequence segmented sampling result data 7

19.6.21 ADC sequential segmented sampling control register (ADC_SEQNUM)

Offset address: 0x5C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:16	SGNUM	R/W	The Number of Sequencial Section 00: 1 section 01: 2 sections 10: 3 sections 11: Invalid
15:11	Reserved		
10:8	SEQNUM3	R/W	Section3 Transmission Time Setup 000: Transmit 1 time

Field	Name	R/W	Description
			001: Transmit 2 times ..... 101: Transmit 6 times Others: Invalid
7	Reserved		
6;4	SEQNUM2	R/W	Section2 Transmission Time Setup 000: Transmit 1 time 001: Transmit 2 times ..... 110: Transmit 7 times Others: Invalid
3	Reserved		
2:0	SEQNUM1	R/W	Section1 Transmission Time Setup 000: Transmit 1 time 001: Transmit 2 times ..... 111: Transmit 8 times

20 Comparator (COMP)

20.1 Full Name and Abbreviation Description of Terms

Table 67 Full Name and Abbreviation Description of Terms

Full name in English	English abbreviation
Comparator	COMP
Invert	INV
Hysteresis	HYS
Input Plus	INP
Input Minus	INM

20.2 Introduction

Four general-purpose comparators (COMP0 and COMP1/2/3) are embedded in MCU, and they can be used in combination with the timer.

20.3 Main characteristics

(1) COMP0 has configurable positive and negative inputs for flexible voltage selection
- Positive input: 2 IOs, BG, 1/2BG
- Negative input: 2 IOs, 2 amplifier outputs

(2) The positive end of COMP1/2/3 shares one IO and a virtual center point, while the negative end is connected to two IOs each

(3) Programmable hysteresis
- Positive hysteresis
- Hysteresis voltage: no hysteresis, 20mV, 40mV, 80mV

(4) Digital filtering

(5) Output phase-inverting

(6) Comparator interrupt

(7) Output as PWM brake trigger signal

20.4 Structure block diagram

Figure 107 COMP0 Structure Block Diagram

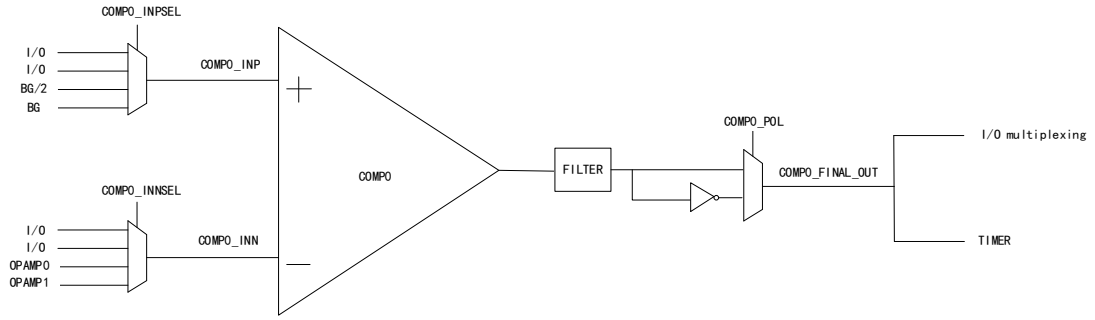
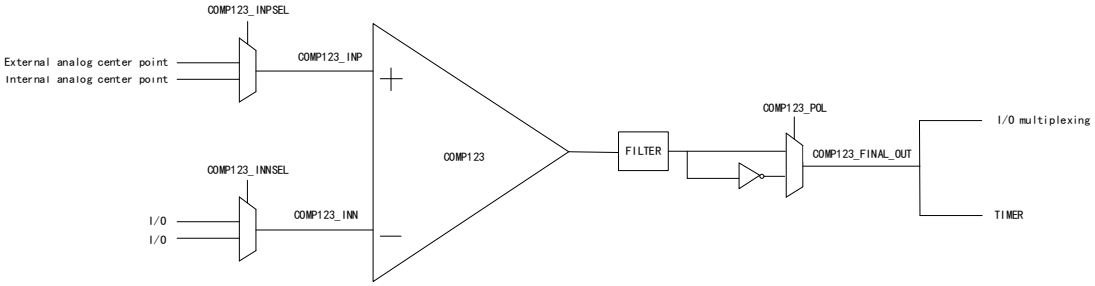


Figure 108 COMP1/2/3 Structure Block Diagram



20.5 Functional description

20.5.1 COMP hysteresis

The comparator includes a programmable hysteresis function to avoid false output transitions when the input signal has large noise. This hysteresis function is asymmetric and only acts on the falling edge of the comparator output. The internal hysteresis function can be disabled, allowing the hysteresis amount to be set by external components.

20.5.2 COMP digital filter

PCLK is used as the filtering clock, supporting 8 levels of clock division and 10 levels of frequency division clock count for high and low level filtering. The filtered COMP output can be output to IO or TIMER.

20.5.3 COMP interrupt

The comparator output internally generates an interrupt event.

The COMP interrupt can be enabled by following these steps:

(1) Configure and enable the interrupt mode corresponding to the COMP output event, and select triggering on rising edge, falling edge, or both edges

(2) Configure and enable the mapping to the corresponding NVIC IRQ channel

(3) Enable COMP

20.5.4 Direction for use

(1) Configure the COMP_CRx register and set the LOCK value, CFG value, COMPEN value, VPSEL[2:0], VNSEL[2:0], and HYSEN[1:0];

(2) Configure the COMP_CRx register and set the POL value, REN value, FEN value, RFEN value, and SWEG value.

20.6 Register address mapping

COMP0 base address: 0x4000 5C00

COMP1/2/3 base address: 0x4000 6000Table 68 COMP Register Address Mapping

Register name	Description	Offset address
COMP_CR0	Control status register 0	0x00
COMP_ISR0	Status register 0	0x04
COMP_CR1	Control status register 1	0x00
COMP_CR2	Control status register 2	0x04
COMP_CR3	Control status register 3	0x08
COMP_ISR1	Status register 1	0x0C
COMP_ISR2	Status register 2	0x10
COMP_ISR3	Status register 3	0x14

20.7 Register functional description

20.7.1 Control status register 0 (COMP_CR0)

Offset address: 0x00

Reset value: 0x0003 C0000

Field	Name	R/W	Description
31	LOCK	R/W	COMP_CRx register lock This bit is set by software and cleared by hardware system reset. It locks all contents of the comparator x control register COMP_CRx [28:0]. When locked, all control bits and flag bits are read-only. When unlocked, control bits can also be written by software. 0: Unlock 1: Locked
30	VAL	R	Comparator output flag

Field	Name	R/W	Description
			This read-only flag reflects the level value of the comparator output before the polarity selector and masking.
29	Reserved		
28:26	PSC	R/W	Digital filtering clock frequency division configure 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128
25:22	CFG	R/W	Digital filtering counting cycle configuration 0000: 1 0001: 2 0010: 4 0011: 8 0100: 16 0101: 32 0110: 64 0111: 128 1000: 256 1001: 512 1010~1111: Reserved
21	SWEG	R/W	Software interrupt Set to 1 by software, write 1 and set to 0 on PEND; When this bit is 0, writing 1 will set PEND and generate an interrupt. 0: No effect 1: Software generates an interrupt
20	RFEN	R/W	Comparator interrupt enables both the rising and falling edges simultaneously 0: Disable 1: Enable
19	FEN	R/W	Comparator interrupts the falling edge enable 0: Disable 1: Enable
18	REN	R/W	Comparator interrupts the rising edge enable 0: Disable 1: Enable
17:16	HYSPEN	R/W	Comparator hysteresis enable Positive hysteresis, default value 11. 00: Disable 01: 1st level hysteresis (20mV) 10: 2nd level hysteresis (40mV)

Field	Name	R/W	Description
			11: 3rd level hysteresis (80mV)
15:14	HYSNEN	R/W	Comparator hysteresis enable Negative hysteresis, default value is 11. 00: Disable 01: 1st level hysteresis (20mV) 10: 2nd level hysteresis (40mV) 11: 3rd level hysteresis (80mV)
13	POL	R/W	Comparator polarity This bit is controlled by software to select the output polarity of comparator x: 0: Phase not reversed 1: Phase reversed.
12:9	Reserved		
8:7	VPSEL	R/W	Comparator positive input voltage select When COMPEN=0, all channels are disconnected. 2 IO, 1/2, 1 of BG voltage. 00：PA7 01：PB9 10：BG/2 11：BG
6	Reserved		
5:4	VNSEL	R/W	Comparator negative terminal input voltage select When COMPEN=0, all channels are disconnected. 2 IOs, 2 amplifier outputs. 00：PA8 01：PA9 10：OPAMP0_OUT 11：OPAMP1_OUT
3:1	Reserved		
0	COMPEN	R/W	Comparator enable 0: Disable 1: Enable

20.7.2 Status Register 0 (COMP_ISR0)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	IFLG	RW1C	Interrupt status Set to 1 by hardware when an edge trigger request corresponding to REN/FEN/RFEN occurs, When a request is triggered by SWEG software, the hardware is set to 1. and can be cleared to 0 by writing 1 to this bit.

Field	Name	R/W	Description
			This flag reflects the level value of the comparator output before the polarity selector and masking.

20.7.3 Control status register x (COMP_CRx)

Offset address: 0x00 +0x04*(x-1)(x=1~3)

Reset value: 0x0003 C0000

Field	Name	R/W	Description
31	LOCK	R/W	COMP_CRx register lock This bit is set by software and cleared by hardware system reset. It locks all contents of the comparator x control register COMP_CRx [28:0]. When locked, all control bits and flag bits are read-only. When unlocked, control bits can also be written by software. 0: Unlock 1: Locked
30	VAL	R	Comparator output flag This read-only flag reflects the level value of the comparator output before the polarity selector and masking.
29	Reserved		
28:26	PSC	R/W	Digital filtering clock frequency division configure 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128
25:22	CFG	R/W	Digital filtering counting cycle configuration 0000: 1 0001: 2 0010: 4 0011: 8 0100: 16 0101: 32 0110: 64 0111: 128 1000: 256 1001: 512 1010~1111: Reserved
21	SWEG	R/W	Software interrupt Set to 1 by software, write 1 and set to 0 on IFLG; When this bit is 0, writing 1 will set IFLG and generate an interrupt. 0: No effect 1: Software generates an interrupt

Field	Name	R/W	Description
20	RFEN	R/W	Comparator interrupt enables both the rising and falling edges simultaneously 0: Disable 1: Enable
19	FEN	R/W	Comparator interrupts the falling edge enable 0: Disable 1: Enable
18	REN	R/W	Comparator interrupts the rising edge enable 0: Disable 1: Enable
17:16	HYSPEN	R/W	Comparator hysteresis enable Positive hysteresis, default value 11. 00: Disable 01: 1st level hysteresis (20mV) 10: 2nd level hysteresis (40mV) 11: 3rd level hysteresis (80mV)
15:14	HYSNEN	R/W	Comparator hysteresis enable Negative hysteresis, default value is 11. 00: Disable 01: 1st level hysteresis (20mV) 10: 2nd level hysteresis (40mV) 11: 3rd level hysteresis (80mV)
13	POL	R/W	Comparator polarity This bit is controlled by software to select the output polarity of comparator x: 0: Phase not reversed 1: Phase reversed.
12:8	Reserved		
7	VPSEL	R/W	Comparator positive input voltage select When COMPEN=0, all channels are disconnected. Select the path 0: External analog center point -PB6 1: Internal analog center point Note: This bit is shared by Comp1/2/3 and configured in the COMP1 register.
6:5	Reserved		
4	VNSEL	R/W	Comparator negative terminal input voltage select When COMPEN=0, all channels are disconnected. COMP1： 0：PA1 1：PB1 COMP2： 0：PA2 1：PB2 COMP3： 0：PA0

Field	Name	R/W	Description
			1：PB5
3:1	Reserved		
0	COMPEN	R/W	Comparator enable 0: Disable 1: Enable

20.7.4 Status Register x (COMP_ISRx)

Offset address: 0x0C +0x04*x (x=1~3)

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	IFLG	RW1C	Interrupt status Set to 1 by hardware when an edge trigger request corresponding to REN/FEN/RFEN occurs, When a request is triggered by SWEG software, the hardware is set to 1. and can be cleared to 0 by writing 1 to this bit. This flag reflects the level value of the comparator output before the polarity selector and masking.

21 Operational Amplifier (OPAMP)

21.1 Introduction

The MCU is embedded with two independent operational amplifiers (OPAMP0 and OPAMP1), which can be used in combination with COMP and ADC.

21.2 Main Characteristics

(1) Supports adjustable internal gain, with adjustable ranges (x1,4,6,8,10,12,16)

(2) Supports external gain up to 16 times

(3) The output of OPAMP can be used as the input of ADC

(4) The output of OPAMP can be used as the negative input of COMP

(5) Supports four selectable built-in bias Settings: 1/2VDDA, 1/4VDDA, BG, and 1/4BG

21.3 Structure Block Diagram

Figure 109 OPAMP Structure Block Diagram

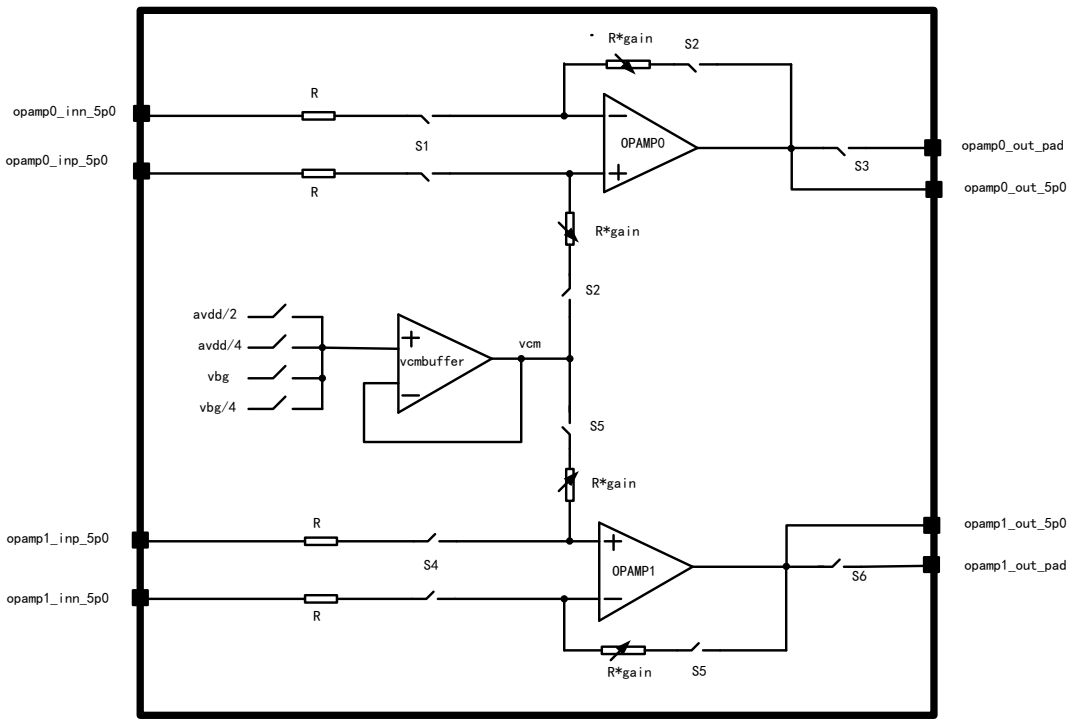


Figure 110 Internal gain application diagram

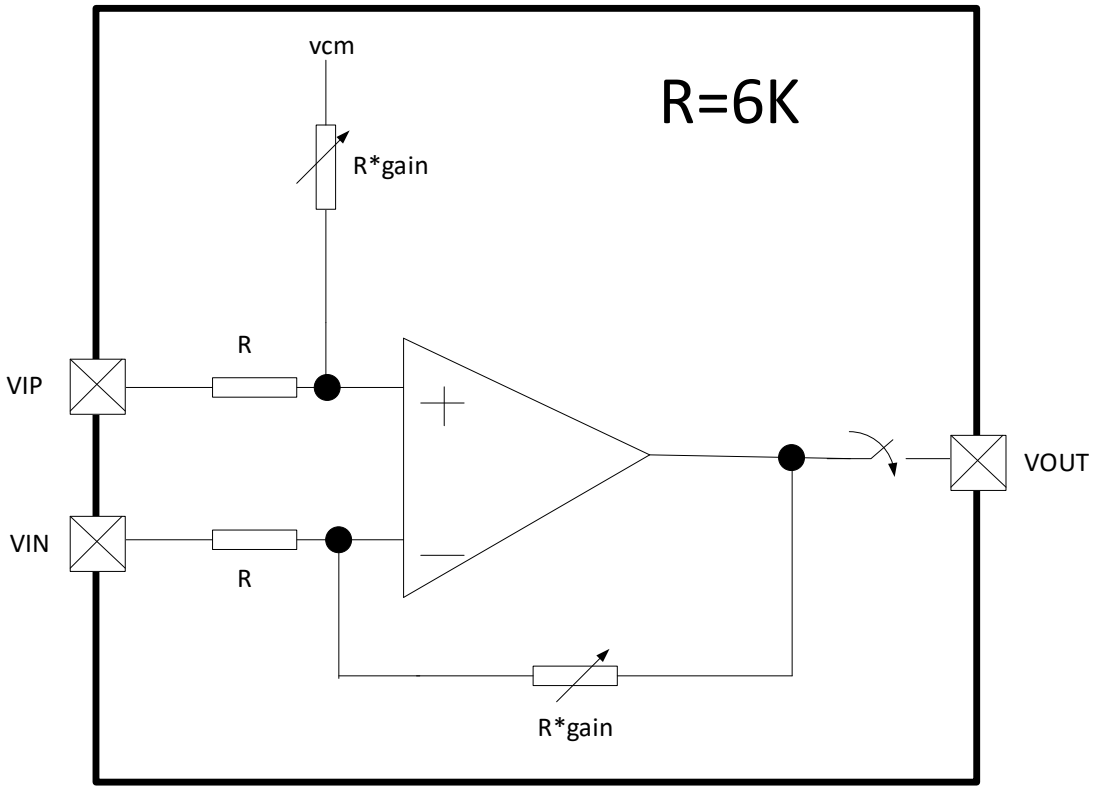
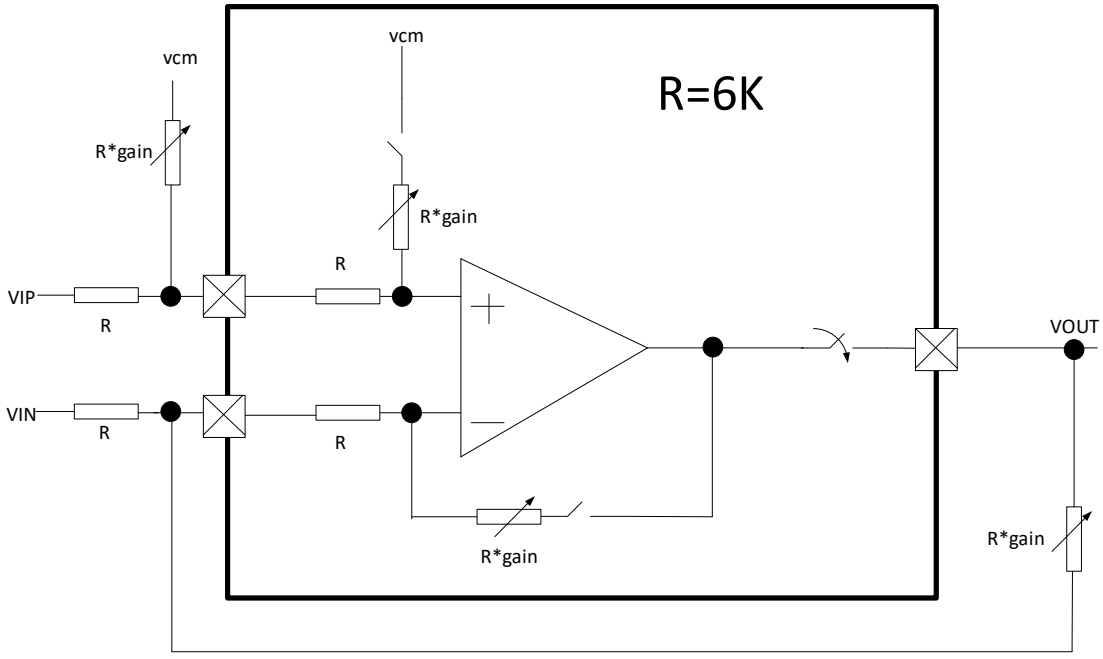


Figure 111 External gain application diagram



21.4 Register address mapping

Table 69 OPAMP Register Address Mapping Table

Register name	Description	Offset address
OPA_CR	OPAMP control register	0x00

21.5 Register functional description

21.5.1 OPAMP control register (OPA_CR)

Offset address: 0x00

Reset value: 0x0004 0000

Field	Name	R/W	Description
31:19	Reserved		
18:16	OPAVCMSEL	R/W	Operational amplifier built-in bias voltage select signal 0xx: No VCM input voltage selected 100: VCM = 0.5*AVDD 101: VCM = 0.25*AVDD 110: VCM = VBG 111: VCM = 0.25*VBG
15:14	Reserved		
13	OPA1OUTSEL	R/W	Operational amplifier 1 output IO control signal 0: The OPAMP1 output is not connected to the output PAD 1: The OPAMP1 output is connected to the output PAD
12	OPA1INSEL	R/W	Operational amplifier 1 input IO control signal 0: The input PAD is not connected to the OPAMP1 input 1: Connect the input PAD to the OPAMP1 input
11:9	OPA1GAINSEL	R/W	Operational amplifier 1 gain select signal 000: Gain is determined by external resistors, no internal resistors used 001: Use internal resistors, gain is 1 010: Use internal resistors, gain is 2 011: Use internal resistors, gain is 4 100: Use internal resistors, gain is 6 101: Use internal resistors, gain is 8 110: Use internal resistors, gain is 12 111: Use internal resistors, gain is 16
8	OPA1EN	R/W	Operational Amplifier 1 enable 0: Disable 1: Enable
7:6	Reserved		
5	OPA0OUTSEL	R/W	Operational amplifier 0 output IO control signal 0: The OPAMP0 output is not connected to the output PAD 1: The OPAMP0 output is connected to the output PAD
4	OPA0INSEL	R/W	Operational amplifier 0 input IO control signal 0: The input PAD is not connected to the OPAMP0 input 1: Connect the input PAD to the OPAMP0 input
3:1	OPA0GAINSEL	R/W	Operational amplifier 0 gain select signal 000: Gain is determined by external resistors, no internal resistors used 001: Use internal resistors, gain is 1 010: Use internal resistors, gain is 4

Field	Name	R/W	Description
			011: Use internal resistors, gain is 6 100: Use internal resistors, gain is 8 101: Use internal resistors, gain is 10 110: Use internal resistors, gain is 12 111: Use internal resistors, gain is 16
0	OPA0EN	R/W	Operational Amplifier 0 enable 0: Disable 1: Enable

22 Cyclic redundancy check computing unit (CRC)

22.1 Introduction

The cyclic redundancy check (CRC) computing unit can get 16/32-bit CRC computing result by calculating the input data through a fixed generator polynomial, which is mainly used to detect or verify the correctness and integrity of the data after transmission or saving.

22.2 Main characteristics

(1) Process 16-bit and 32-bit data

(2) Supports CRC16-CCITT and CRC32

(3) Programmable CRC initial value

(4) 32-bit data register

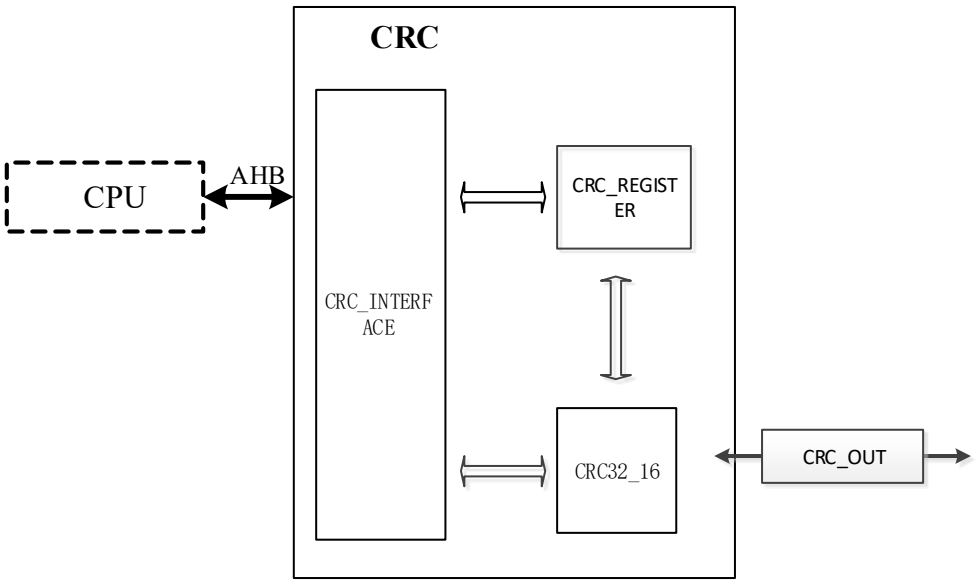
(5) After the CRC is paused, the calculation can continue, that is, the CRC result is not reset to zero and can be calculated based on the previous result

(6) The high and low bits of input data can be inverted in order to adapt to different data storage methods (byte, half word or word, little-endian and big-endian system)

(7) The output data is reversed, that is, the high and low bits are inverted

22.3 Structure block diagram

Figure 112 Structure Block Diagram



22.4 Functional description

22.4.1 Calculation method

Use CRC-32 (Ethernet) polynomial: 0x4C11DB7

Polynomial formula:$(X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X+1)$

CRC16-CCITT polynomial: 0x1021

Polynomial formula: $(X^{16}+X^{12}+X^{5}+1)$

22.4.2 Calculating Time

- When processing 32-bit data, the calculation time is 4 AHB clock cycles
- When processing 16-bit data, the calculation time is 2 AHB clock cycles

22.4.3 Functional characteristics

CRC unit contains a 32-bit read/write register CRC_DR, used to write new data and give CRC computing results. Every time a new data is written, the result will be a combination of the last calculation result and the new calculation result. (Execute operation for the whole word). CRC_Data can access word or right-aligned half word or right-aligned bytes, while other registers can only access 32 bits.

It can perform CRC calculation on 16-bit, 32-bit data, CRC32 and CRC16 are optional, CRC initial value can be configured, input and output data can be reversed.

Input data is not reversed, in bytes, half-words, or words. Output data can be reversed or not reversed.

22.4.4 Direction for use

CRC initialization steps

(1) Turn on the CRC clock

(2) Configure the CRC_CR register, set CRC32 or CRC16, input data flip form and output data

(3) Configure the CRC_INIT register and set the initial value of CRC

(4) Configure the CRC_DR register and set the data to be verified

(5) Read CRC_DR to obtain the calculation result of CRC

22.5 Register address mapping

Table 70 CRC Register Address Mapping

Register name	Description	Offset address
CRC_DR	Data register	0x00
CRC_CR	Control register	0x04
CRC_INIT	Initial value register	0x08

22.6 Register functional description

22.6.1 Data register (CRC_DR)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	DATA	R/W	32bit Data As an input register: Store the new data of CRC calculator when writing. As an output register: Return the results of CRC computing when reading.

22.6.2 Control register (CRC_CR)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:4	Reserved		
3	OUTFLIP	R/W	Output Data Reverse 0: Not reverse 1: Reverse
2:1	INFLIP	R/W	Input Data Reverse Reverse the input data in different units. 00: Not reverse 01: In byte 10: In unit 11: In word
0	CRCSEL	R/W	CRC select 0：32 bit 1：16 bit

22.6.3 Initial value register (CRC_INIT)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	INIT	R/W	Initial CRC Value The CRC initial value is programmable, and this bit is used to set the initial value of CRC.

23 Divider (DIV)

23.1 Introduction

The hardware divider can automatically perform signed or unsigned 32-bit integer division operations. The shift-subtract divider algorithm is a division calculation process based on basic long division.

23.2 Main characteristics

(1) 32-bit divisor and dividend, output 32-bit quotient and remainder

(2) Completes one division operation in 8 HCLK cycles

(3) If the divisor is zero, the overrun status flag bit will be set, and no new data will be calculated

(4) Writing to the divisor register automatically performs division operation

(5) The quotient and remainder registers will retain the results of the previous calculation

23.3 Register address mapping

Table 71 DIV Register Address Mapping

Register name	Description	Offset address
DIV_DVDR	Dividend register	0x00
DIV_DVSR	Divisor register	0x04
DIV_QUOTR	Quotient register	0x08
DIV_RMDR	Remainder register	0x0C
DIV_SR	Status register	0x10
DIV_CR	Control register	0x14

23.4 Register functional description

23.4.1 Dividend Register (DIV_DVDR)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	DIVIDEND	R/W	Dividend data If the division operation is written when it is busy, it will take effect next time.

23.4.2 Divisor Register (DIV_DVSR)

Offset address: 0x04

Reset value: 0x0000 0001

Field	Name	R/W	Description
31:0	DIVISOR	R/W	Divisor data After this register is written, the division operation will be automatically triggered. If the division operation is busy, writing a non-zero divisor will cause the division to be recalculated. If the written divisor is 0, it does not affect the continuation of the division operation. The quotient and remainder remain the results calculated from the last non-zero divisor. Meanwhile, DIV_SR.OVFLG=1 indicates that the divisor for this configuration is 0.

23.4.3 Quotient Register (DIV_QUOTR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	QUOTIENT	R	Quotient data If the divisor is 0, the register retains the previous result.

23.4.4 Remainder Register (DIV_RMDR)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	REMAINDER	R/W	Remainder data If the divisor is 0, the register retains the previous result.

23.4.5 Status Register (DIV_SR)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	OVFLG	R/W1C	Divide the zero overflow status flag Before the next division operation, the software writes 1 to clear. 0: The divisor of the current operation is not zero 1: The divisor of the current operation is zero

23.4.6 Control Register (DIV_CR)

Offset address: 0x14

Reset value: 0x0000 0001

Field	Name	R/W	Description
31:1	Reserved		
0	USIGNEN	R/W	Unsigned enable 0: Signed division 1: Unsigned division

24 Chip electronic signature

24.1 Product Identity Identifier (UID)

The unique identity identifier (UID) can be used as the serial number of a product, facilitating tracking, asset management and anti-counterfeiting. It can be used as a password and combined with security units (such as secure boot, cluster key, signature, etc.) to achieve application scenarios such as device authentication, firmware integrity verification, key distribution, and software encryption and decryption.

The 96-bit product unique identity identifier UID is unique to any MCU under any circumstances. Under no circumstances can the user read this identity identifier and cannot modify it.

The storage space address of UID0 is: 0x00101E88

Field	Name	R/W	Description
31:0	UID0	R	The [31:0] bit position of the unique identity marker of the product is assigned at the time of factory shipment.

The storage space address of UID1 is: 0x00101E8C

Field	Name	R/W	Description
31:0	UID1	R	The [63:32] bit position of the unique identity marker of the product is assigned at the time of factory shipment.

The storage space address of UID2 is: 0x00101E90

Field	Name	R/W	Description
31:0	UID2	R	The [95:64] bit position of the unique identity marker of the product is assigned at the time of factory shipment.

24.2 Product Model Identification (PID)

The product model identification (PID) is used to uniquely identify a specific combination of information such as the chip model and version, facilitating hardware design, production tracking, and firmware adaptation.

The storage space address of PID is: 0x00101D80

Field	Name	R/W	Description
31:16	Reserved		
15:0	CHIPID	R	chip version:0x4402

25 Gate Driver (Only M3122/M3114/M3115 Supported)

25.1 Introduction for M3114

The G32M3114 is embedded with a three-phase medium-voltage high-speed gate driver IC, which is specially designed for driving double-N-channel VDMOS power transistor or IGBT in bridge circuits, and is suitable for application schemes for battery-powered DC brushless motors. The embedded typical dead time is 500ns. When the dead time of the MCU output signal is less than the embedded dead time, the actual dead time is the embedded dead time. On the contrary, when the dead time of the MCU output signal is greater than the embedded dead time, the actual dead time is the output dead time of MCU. The embedded VCC and VBS undervoltage protection functions can prevent the system from turning on the external power transistors at low driving voltage. The output of the high-side driving circuit and the output of the low-side driving circuit are controlled through input signals. The built-in LDO of the driver supports power supply for control chips such as the MCU, and this LDO supports a 5V voltage output.

25.2 Main characteristics

- Operating supply voltage range: 5~20V
- Floating offset voltage: +200V
- Embedded VCC and VBS undervoltage protection
- Embedded straight-through prevention function
- Embedded input pull-down resistor
- Embedded output pull-down resistor
- Matching the transmission time of high and low-end channels
- High dv/dt noise suppression capability
- Input and output in-phase
- Compatible with 3.3V/5V logic input
- Peak input current 1.1A@15V, 3.3nF load fall time 40ns
- Peak output current 0.9A@15V, 3.3nF load rise time 65ns
- LDO load capacity 60mA@15V
- Overtemperature protection threshold 151℃ /131℃

25.3 Functional description

25.3.1 Logic truth value of the drive

(1) VBS undervoltage will only set the HO output low.

(2) VCC undervoltage will set both LO and HO outputs low.

(3) The VG signal is pulled up to high by default, and the LDO output is turned off when the external pull-down is low.

(4) After the over temperature protection function is triggered, both the drive output and LDO output are turned off.

Table 72 Logic truth value of the drive

OTP	VG	VCCUV	VBSUV	LIN	HIN	LO	HO	LDO
normal	normal	normal	normal	L	H	L	H	5V
normal	normal	normal	normal	H	L	H	L	5V
normal	normal	normal	normal	L	L	L	L	5V
normal	normal	normal	normal	H	H	L	L	5V
normal	normal	normal	UV	H&L	H&L	H&L	L	5V
normal	normal	UV	normal	H&L	H&L	L	L	5V
normal	L	normal	normal	H&L	H&L	H&L	H&L	0V
OVER	normal	normal	normal	H&L	H&L	L	L	0V

25.4 Introduction for M3115

The G32M3115 is embedded with a three-phase medium-voltage high-speed gate driver IC, which is specially designed for driving double-N-channel VDMOS power transistor or IGBT in bridge circuits, and is suitable for application schemes for battery-powered DC brushless motors. The embedded typical dead time is 220ns. When the dead time of the MCU output signal is less than the embedded dead time, the actual dead time is the embedded dead time. On the contrary, when the dead time of the MCU output signal is greater than the embedded dead time, the actual dead time is the output dead time of MCU. The embedded VCC and VBS undervoltage protection functions can prevent the system from turning on the external power transistors at low driving voltage. The output of the high-side driving circuit and the output of the low-side driving circuit are controlled through input signals.

25.5 Main characteristics

- Operating supply voltage range: 5.5~20V
- Floating offset voltage: +200V
- Embedded minimum dead time:220ns
- Embedded VCC and VBS undervoltage protection
- Embedded straight-through prevention function
- Embedded input pull-down resistor
- Embedded output pull-down resistor
- Matching the transmission time of high and low-end channels

- High dv/dt noise suppression capability
- Input and output in-phase
- Compatible with 3.3V/5V logic input
- Peak output current 2.0A @15V, 3.3 nF load fall time 20 ns
- Peak output current 2.7A@15V, 3.3 nF load rise time 35 ns

25.6 Functional description

25.6.1 Logic truth value of the drive

Table 73 Logic truth value of the drive

VCCUV	VBSUV	LIN	HIN	LO	HO
normal	normal	L	H	L	H
normal	normal	H	L	H	L
normal	normal	L	L	L	L
normal	normal	H	H	L	L
under vccuv	normal	L	H	L	L
under vccuv	normal	H	L	L	L
under vccuv	normal	L	L	L	L
under vccuv	normal	H	H	L	L
normal	under vbsuv	L	H	L	L
normal	under vbsuv	H	L	H	L
normal	under vbsuv	L	L	L	L
normal	under vbsuv	H	H	L	L

25.7 Introduction for M3122

The G32M3122 integrates a three-phase 40V gate driver integrated circuit for driving P/NMOS power transistors. and is suitable for DC brushless motor applications such as low-voltage fans and water pumps. This chip integrates 5V LDO, has a current-carrying capacity of 60mA, and can provide power for internal logic circuits and external MCU. Multiple protection functions are built in G32M3122, including undervoltage protection, input straight-through protection, and startup protection. It has a dead time of 300ns to effectively ensure normal operation of the system. At the same time, it provides an output driving voltage of 11V, has excellent sink current capability, and can effectively drive the external P/NMOS power transistors.

25.8 Main characteristics

- Three-phase gate driver for P/NMOS power transistors

- Recommended range of supply voltage: 5.5V~30V
- Built-in 5V/60mA LDO
- HO output current+260mA/-80mA@V_{sup}=24V
- LO output current+50mA/-240mA@V_{sup}=24V
- Low standby power consumption
- 3.3V/5V input logic compatibility
- LDO overload start protection
- Built-in power undervoltage protection UVLO
- Built-in straight-through prevention function
- Built-in 300ns dead time
- Matching of high and low-side channels

25.9 Functional description

25.9.1 Logic truth value of the drive

Table 74 Logic truth value of the drive

LIN	HIN	LO	HO
0	0	OFF	OFF
0	1	OFF	ON
1	0	ON	OFF
1	1	OFF	OFF
Suspension	Suspension	OFF	OFF

Note:

（1） 1: Logic high level, 0: Logic low level.

（2） ON: V_{HO}=V_{SUP1}-11V, V_{LO}=11V OFF: V_{HO}=V_{SUP1}, V_{LO}=0V.

（3） It is recommended to maintain a strong input state at the input port to avoid unstable states such as high resistance and suspension.

26 Revision history

Table 75 Document Revision History

Date	Version	Revision History
September,2025	1.0	• Initial version
May, 2026	1.1	• Add the sealing model information
May, 2026	1.2	• Modify the bit6:4 description of FMC_SR
July, 2026	1.3	• Modify the USART_CR1 register • Modify the levitation offset voltage of M3115

Statement

This manual is formulated and published by Zhuhai Geehy Semiconductor Co., Ltd. (hereinafter referred to as "Geehy"). The contents in this manual are protected by laws and regulations of trademark, copyright and software copyright. Geehy reserves the right to correct and modify this manual at any time. Please read this manual carefully before using the product. Once you use the product, it means that you (hereinafter referred to as the "users") have known and accepted all the contents of this manual. Users shall use the product in accordance with relevant laws and regulations and the requirements of this manual.

1. Ownership of rights

This manual can only be used in combination with chip products and software products of corresponding models provided by Geehy. Without the prior permission of Geehy, no unit or individual may copy, transcribe, modify, edit or disseminate all or part of the contents of this manual for any reason or in any form.

The "Geehy" or "Geehy" words or graphics with "®" or "TM" in this manual are trademarks of Geehy. Other product or service names displayed on Geehy products are the property of their respective owners.

2. No intellectual property license

Geehy owns all rights, ownership and intellectual property rights involved in this manual.

Geehy shall not be deemed to grant the license or right of any intellectual property to users explicitly or implicitly due to the sale and distribution of Geehy products and this manual.

If any third party’s products, services or intellectual property are involved in this manual, Geehy shall not be deemed to authorize users to use the aforesaid third party’s products, services or intellectual property, nor shall it be deemed to provide any form of guarantee for third-party products, services, or intellectual property, including but not limited to any non-infringement guarantee for third-party intellectual property, unless otherwise agreed in sales order or sales contract of Geehy.

3. Version update

Users can obtain the latest manual of the corresponding products when ordering Geehy products.

If the contents in this manual are inconsistent with Geehy products, the agreement in Geehy sales order or sales contract shall prevail.

4. Information reliability

The relevant data in this manual are obtained from batch test by Geehy Laboratory or cooperative third-party testing organization. However, clerical errors in correction or errors caused by differences in testing environment are unavoidable. Therefore, users should understand that Geehy does not bear any responsibility for such errors that may occur in this manual. The relevant data in this manual are only used to guide users as performance parameter reference and do not constitute Geehy's guarantee for any product performance.

Users shall select appropriate Geehy products according to their own needs, and effectively verify and test the applicability of Geehy products to confirm that Geehy products meet their own needs, corresponding standards, safety or other reliability requirements. If losses are caused to users due to the user's failure to fully verify and test Geehy products, Geehy will not bear any responsibility.

5. Compliance requirements

Users shall abide by all applicable local laws and regulations when using this manual and the matching Geehy products. Users shall understand that the products may be restricted by the export, re-export or other laws of the countries of the product suppliers, Geehy, Geehy distributors and users. Users (on behalf of itself, subsidiaries and affiliated enterprises) shall agree and undertake to abide by all applicable laws and regulations on the export and re-export of Geehy products and/or technologies and direct products.

6. Disclaimer

This manual is provided by Geehy on an "as is" basis. To the extent permitted by applicable laws, Geehy does not provide any form of express or implied warranty, including without limitation the warranty of product merchantability and applicability of specific purposes.

Geehy products are not designed, authorized, or guaranteed to be suitable for use as critical components in military, life support, pollution control, or hazardous substance management systems, nor are they designed, authorized, or guaranteed to be suitable for applications that may cause injury, death, property, or environmental damage in case of product failure or malfunction.

If the product is not labeled as "Automotive grade", it means it is not suitable for automotive applications. If the user's application of the product is beyond the specifications, application fields, and standards provided by Geehy, Geehy will assume no responsibility.

Users shall ensure that their application of the product complies with relevant standards, and the requirements of functional safety, information security, and environmental standards. Users are fully responsible for their selection and use of Geehy products. Geehy will bear no responsibility for any disputes arising from the subsequent design and use of Geehy products by users.

7. Limitation of liability

In any case, unless required by applicable laws or agreed in writing, Geehy and/or any third party providing this manual and the products on an "as is" basis shall not be liable for damages, including any general or special direct, indirect or collateral damages arising from the use or no use of this manual and the products (including without limitation data loss or inaccuracy, or losses suffered by users or third parties), which cover damage to personal safety, property, or environment, for which Geehy will not be responsible.

8. Scope of application

The information in this manual replaces the information provided in all previous versions of the manual.