User Manual

G32M3101x8

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

Version: V 1.2

© Geehy Semiconductor Co., Ltd.

Contents

1 Introduction and Document Description Rules 6

1.1 Introduction 6

1.2 Document description rules 6

2 System Architecture 10

2.1 Full Name and Abbreviation Description of Terms 10

2.2 System architecture block diagram 10

2.3 Memory mapping 12

2.4 Start the boot mode 15

3 Flash Memory 16

3.1 Full Name and Abbreviation Description of Terms 16

3.2 Main characteristics 16

3.3 Functional description 17

3.4 Register address mapping 22

3.5 Register functional description 22

4 System configuration controller (SCU) 26

4.1 Register address mapping 26

4.2 Register functional description 26

5 Reset and Clock Control (RCC) 38

5.1 Full Name and Abbreviation Description of Terms 38

5.2 Reset functional description 38

5.3 Functional description of clock management 40

6 Power management unit (PMU) 43

6.1 Full Name and Abbreviation Description of Terms 43

6.2 Introduction 43

6.3 Structure block diagram 43

6.4 Functional description 44

7 Nested Vector Interrupt Controller (NVIC) 48

7.1 Full Name and Abbreviation Description of Terms 48

7.2 Introduction 48

7.3 Main characteristics 48

7.4 Interrupt and exception vector table 48

8 External interrupt and event controller (EINT) 50

8.1 Introduction 50

8.2 Main characteristics 50

8.3 Structure block diagram 51

8.4 Functional description 51
8.5 Register address mapping 53
8.6 Register functional description 54
9 Direct memory access (DMA) 58
9.1 Full Name and Abbreviation Description of Terms 58
9.2 Introduction 58
9.3 Main characteristics 59
9.4 Structure block diagram 60
9.5 Functional description 60
9.6 Register address mapping 65
9.7 Register functional description 65
10 General-Purpose Input/Output Pin (GPIO) 73
10.1 Full Name and Abbreviation Description of Terms 73
10.2 Introduction 73
10.3 Main characteristics 73
10.4 Structure block diagram 74
10.5 Functional description 74
10.6 Register address mapping 80
10.7 Register functional description 81
11 Advanced Timer (ATIMER) 93
11.1 Introduction 93
11.2 Main characteristics 93
11.3 Structure block diagram 94
11.4 Functional description 94
11.5 Register address mapping 107
11.6 Register functional description 108
12 General Timer (GTIMER) 129
12.1 Introduction 129
12.2 Main characteristics 129
12.3 Structure block diagram 130
12.4 Functional description 130
12.5 Register address mapping 143
12.6 Register functional description 144
13 Basic Timer (BTIMER) 158
13.1 Introduction 158
13.2 Main characteristics 158
13.3 Structure block diagram 159

13.4 Functional description 159
13.5 Register address mapping 167
13.6 Register functional description 168
14 Low-power timer (LPTIMER) 177
14.1 Introduction 177
14.2 Main characteristics 177
14.3 Structure block diagram 177
14.4 Functional description 177
14.5 Register address mapping 178
14.6 Register functional description 178
15 Watchdog timer (WDT) 180
15.1 Independent watchdog 180
15.2 Window watchdog 181
15.3 IWDT register address mapping 184
15.4 IWDT register functional description 184
15.5 WWDT register address mapping 185
15.6 WWDT register functional description 185
16 Universal asynchronous receiver/transmitter (UART) 188
16.1 Full Name and Abbreviation Description of Terms 188
16.2 Introduction 188
16.3 Main characteristics 188
16.4 Structure block diagram 190
16.5 Functional description 190
16.6 Register address mapping 201
16.7 Register functional description 201
17 Serial peripheral interface (SPI) 208
17.1 Full Name and Abbreviation Description of Terms 208
17.2 Introduction 208
17.3 Main characteristics 208
17.4 functional description 209
17.5 Register address mapping 216
17.6 Register functional description 217
18 Analog-to-digital converter (ADC) 221
18.1 Introduction 221
18.2 Main characteristics 221
18.3 Structure block diagram 222
18.4 ADC Functional Description 223

18.5 Register address mapping 235
18.6 Register functional description 236
19 Comparator (COMP) 250
19.1 Full Name and Abbreviation Description of Terms 250
19.2 Introduction 250
19.3 Main characteristics 250
19.4 Structure block diagram 251
19.5 Functional description 251
19.6 Register address mapping 252
19.7 Register functional description 252
20 Operational Amplifier (OPAMP) 256
20.1 Introduction 256
20.2 Main Characteristics 256
20.3 Structure Block Diagram 256
20.4 Register functional description 257
21 Cyclic redundancy check computing unit (CRC) 258
21.1 Introduction 258
21.2 Main characteristics 258
21.3 Structure block diagram 258
21.4 Functional description 258
21.5 Register address mapping 259
21.6 Register functional description 260
22 Divider (DIV) 261
22.1 Introduction 261
22.2 Main characteristics 261
22.3 Register address mapping 261
22.4 Register functional description 261
23 Chip electronic signature 263
23.1 Product Identity Identifier (UID) 263
23.2 Product Model Identification (PID) 263
24 Gate driver 265
24.1 Introduction 265
25 Revision history 266

1 Introduction and Document Description Rules

1.1 Introduction

This reference manual provides application developers with all the information about how to use SoC 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 SoC 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
Control	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	UART
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

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 (3 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

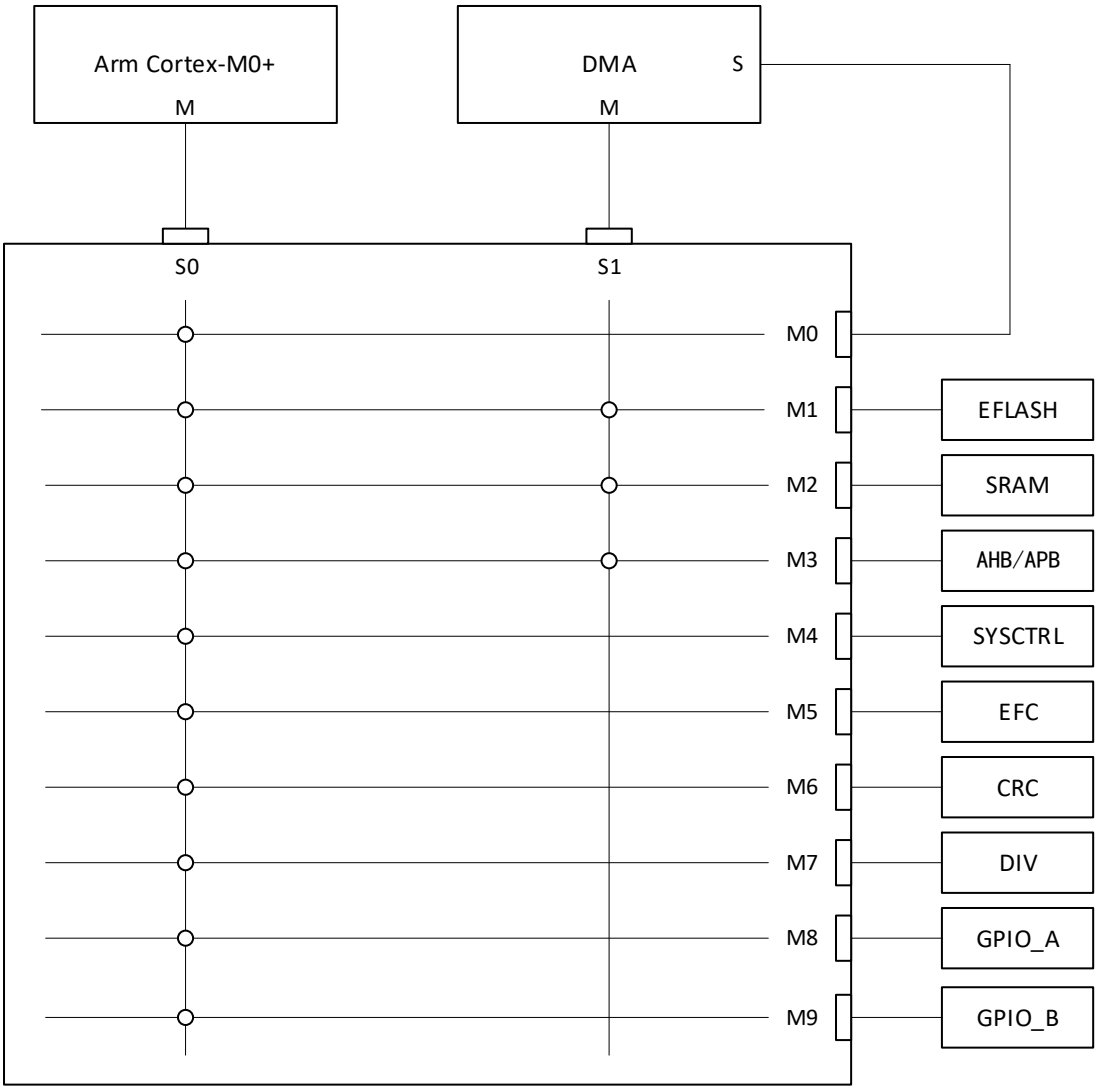


Figure 2 System Architecture Block Diagram

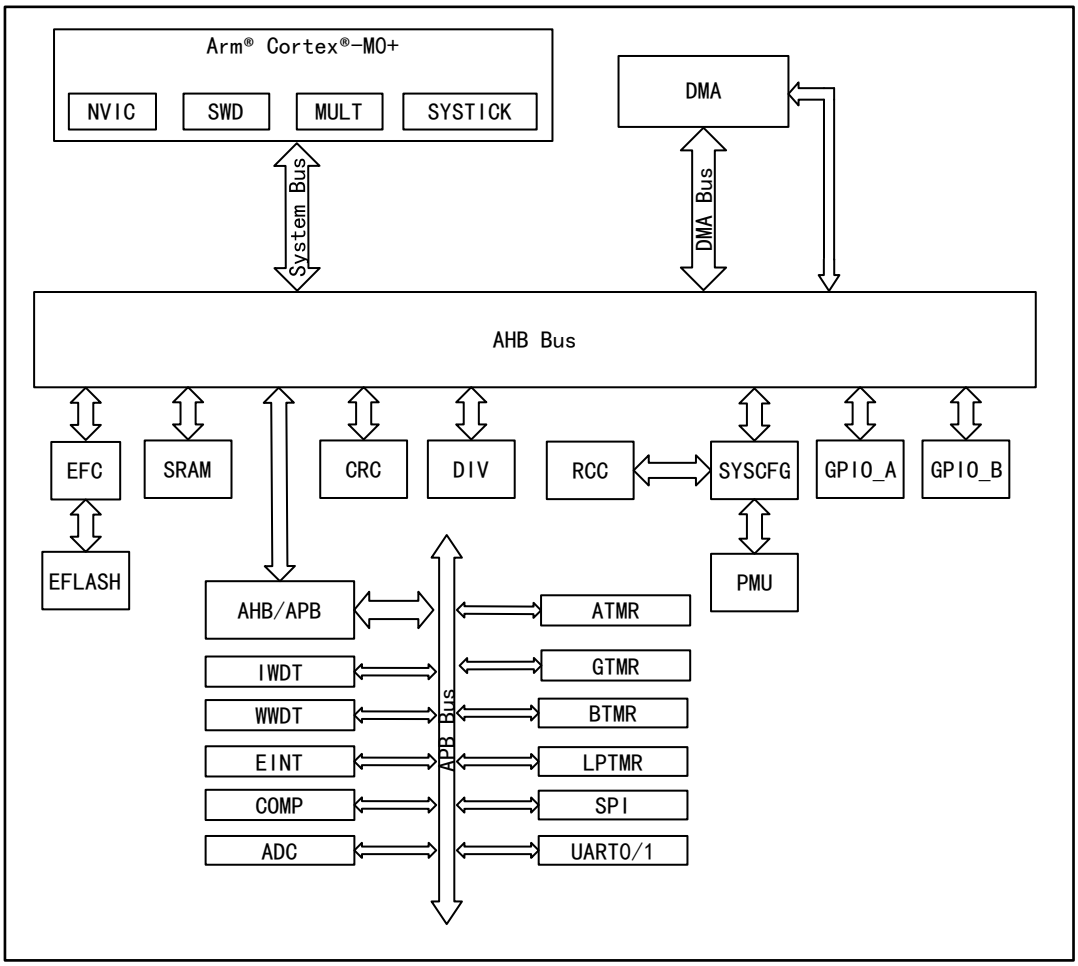


Table 5 Bus Name

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 EFC, 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.

2.3 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 3 Address Mapping

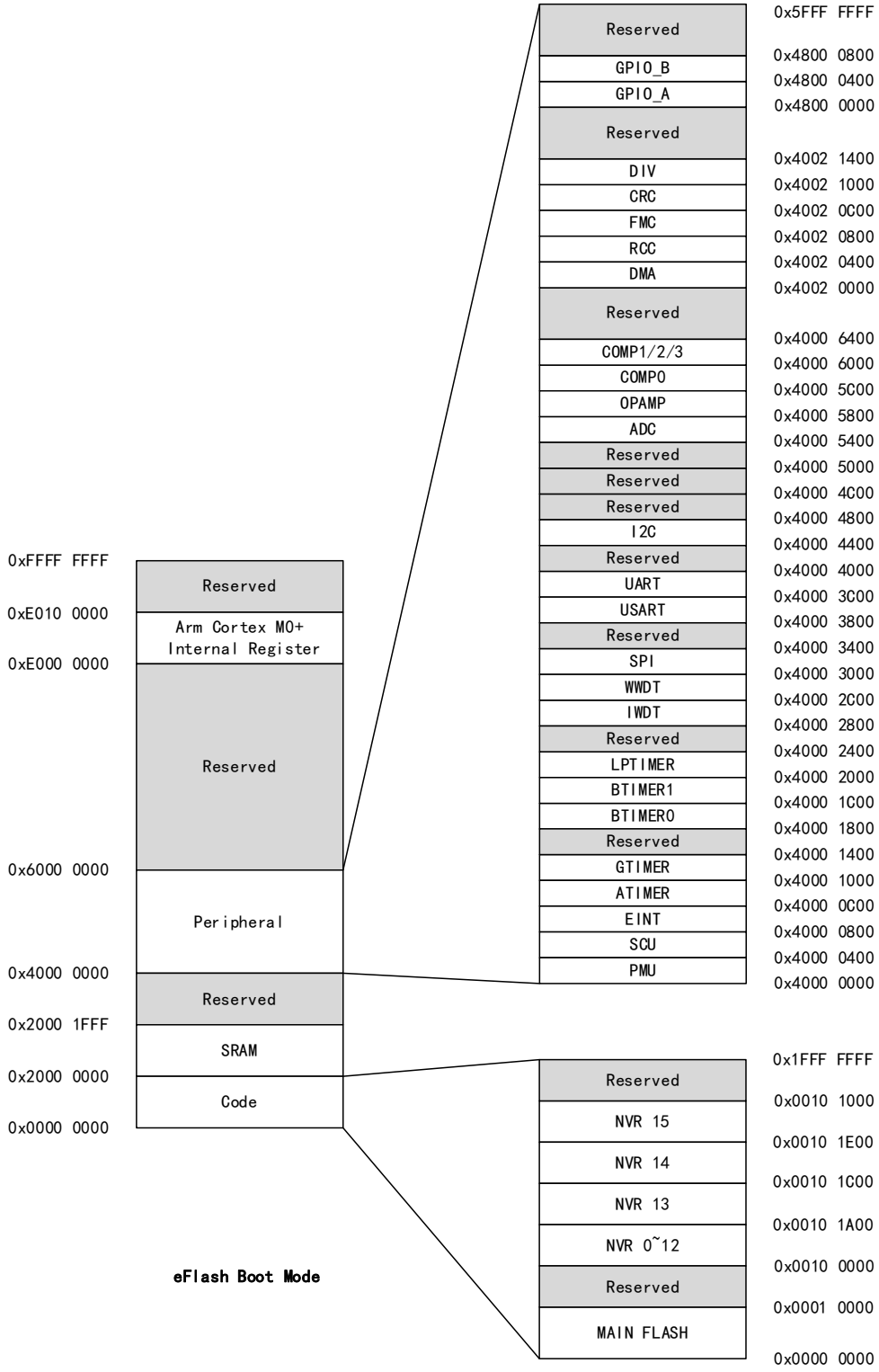


Table 6 Memory address allocation

Module	Name	Address range	Size (bytes)
Reserve	-	0x20002000 - 0x3FFFFFFF	512 MB
SRAM	-	0x20000000 - 0x20001FFF	8 KB
Reserve	-	0x00101000 - 0x1FFFFFFF	512MB

Module	Name	Address range	Size (bytes)
NVR 7 (Manufacturer Area)	-	0x00100E00 - 0x00100FFF	512 B
NVR 6 (Manufacturer Area)	-	0x00100C00 – 0x00100DFF	512 B
NVR 5(User Option Byte Area)	-	0x00100A00 – 0x00100BFF	512 B
NVR 0~4 （Bootloader Area）	-	0x00100000 – 0x001009FF	2.5 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	0xE0000000 - 0xE00FFFFF	1MB
-	Reserve	0x40021C00 - 0x5FFFFFFF	511MB
AHB	GPIOB	0x40021800 - 0x40021BFF	1 KB
	GPIOA	0x40021400 - 0x400217FF	1 KB
	DIV	0x40021000 - 0x400213FF	1 KB
	CRC	0x40020C00 - 0x40020FFF	1 KB
	FLASHCTRL	0x40020800 - 0x40020BFF	1 KB
	SYSCTRL	0x40020400 - 0x400207FF	1 KB
	DMA	0x40020000 - 0x400203FF	1 KB
-	Reserve	0x4000C000 - 0x4001FFFF	80 KB
APB	ADC	0x4000B000 - 0x4000BFFF	4 KB
	LPTIMER	0x4000A000 - 0x4000 AFFF	4 KB
	EINT	0x40009000 - 0x40009FFF	4 KB
	IWDT	0x40008000 - 0x40008FFF	4 KB
	WWDT	0x40007000 - 0x40007FFF	4 KB
	COMP	0x40006000 - 0x40006FFF	4 KB
	UART1	0x40005000 - 0x40005FFF	4 KB

Bus	Peripheral module	Address range	Size (bytes)
	UART0	0x40004000 - 0x40004FFF	4 KB
	SPI	0x40003000 - 0x40003FFF	4 KB
	BTIMER	0x40002000 - 0x40002FFF	4 KB
	GTIMER	0x40001000 - 0x40001FFF	4 KB
	ATIMER	0x40000000 - 0x40000FFF	4 KB

2.3.1 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.

2.4 Start the boot mode

2.4.1 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 G32M3101 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
Embedded Flash Controller	EFC

3.2 Main characteristics

(1) Main area 64KB, NVR 4KB

(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

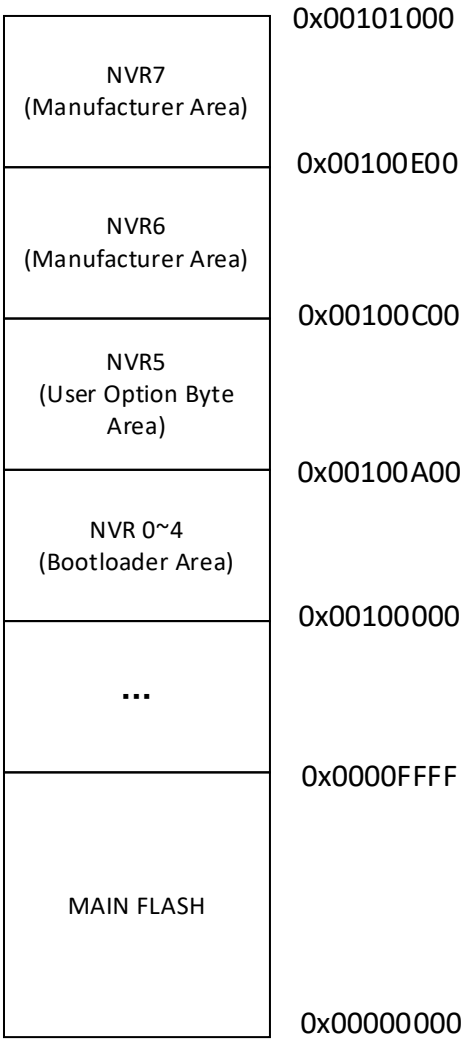
(6) Software triggers reloading

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

3.3 Functional description

3.3.1 Storage area address mapping

Figure 4 Storage area address mapping



3.3.2 User Option Byte Area (NVR5)

The addresses 0x00100A00 to 0x00100BFF are the NVR5 user option byte area. Some of these addresses have special functions, as shown in the following table:

Table 9 User Option Byte Area (NVR5)

Address	Name	Bit width	R/W	Description
0x00100BE8	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
0x00100BEC	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
0x00100BF4	FLASH_CRYPT_EN	32-bit	R/W	flash encryption and decryption switch 0x55AAAA55: Encryption and decryption enable Other values: No encryption or decryption mode
0x00100BF8	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
0x00100BFC	OTP5_EN	32-bit	R/W	OTP configuration enable 0x55AAAA55: Only read allowed Other values: Can be accessed freely

3.3.3 Manufacturer Area (NVR6/NVR7)

The addresses 0x00100C00 to 0x00100DFF are the NVR6 vendor area. Some of these addresses have special functions, as shown in the following table:

Table 10 Manufacturer Area (NVR6)

Address	Name	Bit width	R/W	Description
0x00100C08	Measured value of VBB1.5V	32-bit	R	VBB1.5V measured value (Unit : mV)
0x00100C0C	Measured value of HSI	32-bit	R	HSI 64MHz measured value (Unit : Hz)
0x00100C10	Measured value of LSI	32-bit	R	LSI 32.768K measured value (Unit : Hz)
0x00100C14	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

Address	Name	Bit width	R/W	Description
0x00100C1C	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)
0x00100D80	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 "23.2 Product Model Identification"

Addresses 0x00100E00 to 0x00100FFF are the NVR7 vendor area. Some of these addresses have special functions, as shown in the following table:

Table 11 Manufacturer Area (NVR7)

Address	Name	Bit width	R/W	Description
0x00100E88	UID0	32-bit	R	For detailed information, please refer to the content of the "23.1 Product Identity Marking" section
0x00100E8C	UID1	32-bit	R	For detailed information, please refer to the content of the "23.1 Product Identity Marking" section
0x00100E90	UID2	32-bit	R	For detailed information, please refer to the content of the "23.1 Product Identity Marking" section
0x00100E94	UID3	32-bit	R	For detailed information, please refer to the content of the "23.1 Product Identity Marking" section

3.3.4 Data encryption function

Flash IP provides encryption functions for the boot area and main area. Whether to encrypt is determined by the FLASH_CRYPT_EN enable signal loaded by trim load and the 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.

3.3.5 Direction for use

3.3.5.1 EFLASH unlock

Table 12 unlock code

	MAIN FLASH	user area（NVR5）
KEY	0xABCD6789	0x33AADD55

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.

3.3.5.2 EFLASH erasure

Table 13 The erasure methods for different areas

MAIN FLASH	user area（NVR5）
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 FLASH_KEY to unlock the main area of flash

(2) Set FLASH_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 FLASH_SR->BUSY bit is reset to zero, or FLASH_SR->OPENDR 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 FLASH_KEY to unlock the main area of flash

(2) Set FLASH_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 FLASH_SR->BUSY bit is reset to zero, or FLASH_SR->OPENDR is set to 1

NVR5 area sector erased

This operation is only for the NVR5 area. The steps are as follows:

(1) Write the unlock sequence to FLASH_NVRKEY to unlock the NVR5 area of flash

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

(3) Write 0xA5A5 (any value is acceptable) to any address in the FLASH-NVR5 sector area.

(4) Detect that the FLASH_SR->BUSY bit is reset to zero, or FLASH_SR->OPENDR is set to 1

3.3.5.3 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 FLASH_KEY to unlock the main area of flash

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

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

(4) Detect that the FLASH_SR->BUSY bit is reset to zero, or FLASH_SR->OPENDR is set to 1

NVR5 area write programe

This operation is only for the NVR5 area. The steps are as follows:

(1) Write the unlock sequence to FLASH_NVRKEY to unlock the NVR5 area of flash

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

(3) Write data to any address in the FLASH-NVR5 area.

(4) Detect that the FLASH_SR->BUSY bit is reset to zero, or FLASH_SR->OPENDR is set to 1

3.4 Register address mapping

Table 14 FLASH Register Address Mapping

Register name	Description	Offset address
FLASH_RKEY	Write protect register	0x00
FLASH_MKEY	Key register for unlocking the MAIN area	0x04
FLASH_NVR5KEY	Key register for unlocking the NVR5 area	0x0C
FLASH_CR	Flash control register	0x14
FLASH_ER	Flash enable register	0x18
FLASH_SR	Flash status register	0x1C

3.5 Register functional description

3.5.1 Write protect register (FLASH_RKEY)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	RKEY	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 (FLASH_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 NVR5KEY area (FLASH_NVR5KEY)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		

Field	Name	R/W	Description
0	NVR5KEY	R/W	NVR5 area Key input register Write 0x33AADD55 to unlock the NVR5 area. If OPT5_EN encrypts this area, the unlock is invalid, and the readback is not unlocked. Write any value to disable unlock, and write 0 to lock. 0: Not unlocked 1: Unlock

3.5.4 Control register (FLASH_CR)

Offset address: 0x14

Reset value: 0x0000 0300

Field	Name	R/W	Description
31:16	Reserved		
15	FORCE_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
14:10	Reserved		
9:8	LATENCY	R/W	Access delay cycle The default power-on value is 2 at 64MHz. When the system clock is greater than 32 MHz, LATENCY needs to be configured to be greater than or equal to 1. 0-16MHz: 0 wait cycle 16-32MHz: 1 wait cycle 32-64MHz: 2 wait cycles
7:5	Reserved		
4	READONLY	R/W	Flash ip Read-only control Cannot be erased/written. 0: Invalid 1: Valid
3:2	Reserved		
1:0	OPERATE	R/W	Flash operation types configure 00: Read operation 01: Sector erase 10: Chip erase 11: Write When the BUSY bit of FLASH_SR is set, this register cannot be written.

3.5.5 Enable register (FLASH_ER)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	LSIIE	R/W	LSI erase error interrupt enable 0: Disable 1: Enable
6	RPTIE	R/W	FLASH read/write protection interrupt enable 0: Disable 1: Enable
5	KEYIE	R/W	Flash KEY error interrupt enable 0: Disable 1: Enable
4	OPEIE	R/W	FLASH erase end interrupt enable 0: Disable 1: Enable
3:0	Reserved		

3.5.6 Status register (FLASH_SR)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
7	LSIERR	R/W	LSI erase error interrupt Erasing and writing to flash is invalid when the system clock is switched to LSI. The LSI clock cannot be switched when the Flash is busy, otherwise the system will crash. Set to 1 by hardware, and write 1 by software to clear to 0. 0: No interrupt 1: Interrupt occurs
6	RPTERR	R/W	FLASH read write protection interrupt 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 1 by software to clear to 0. 0: No interrupt 1: Interrupt occurs
5	KEYERR	R/W	Flash KEY error interrupt 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 1 by software to clear to 0. 0: No interrupt 1: Interrupt occurs
4	OPENDR	R/W	FLASH erase end interrupt Set to 1 by hardware, and write 1 by software to clear to 0.

Field	Name	R/W	Description
			0: No interrupt 1: Interrupt occurs
3:1	Reserved		
0	BUSY	R	FLASH erase busy 0: Idle 1: Busy

4 System configuration controller (SCU)

The configuration registers of the system module are all in the SCU module, configured using the AHB interface.

4.1 Register address mapping

Table 15 SYSCTRL Register Address Mapping

Register name	Description	Offset address
SYS_KEY	System register protection register	0x00
SYS_RCCR	Clock control register	0x04
SYS_SCCR	System clock control register	0x08
SYS_CIER	Clock interrupt enable register	0x0C
SYS_CICR	Clock interrupt flag register	0x10
SYS_AHBRST	AHB peripheral reset register	0x14
SYS_APBRST	APB peripheral reset register	0x18
SYS_AHBCG	AHB peripheral clock enable register	0x1C
SYS_APBCG	APB peripheral clock enable register	0x20
SYS_RSTCSR	Reset control/status register	0x28
SYS_ADCCR	ADC system control register	0x2C
SYS_OPAMPCR	OPAMP control register	0x30
SYS_PVDCSR	PVD control/status register	0x3C
SYS_PREDRVCR	Gate driver control register	0x40
SYS_LVDCSR	Undervoltage control/status register	0x44

4.2 Register functional description

4.2.1 System register protection register (SYS_KEY)

Offset address: 0x00

Reset value: 0x0000 00000

Field	Name	R/W	Description
31:17	Reserved		
16	KEYST	R/W	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 KEYST 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.

Field	Name	R/W	Description
15:0	LOCKKEY[15:0]	W	Password protection configuration for system register write operations Writing 0x87E4 to unlock, and KEYST is automatically set 1 at the same time.

4.2.2 Clock control register (SYS_RCCR)

Offset address: 0x04

Reset value: 0x0000 0003

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

4.2.3 System clock control register (SYS_SCCR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:23	Reserved		
22:20	CLKOUT_DIV[2:0]	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	CLKOUT_EN	R/W	Clock output enable 0: Disable

Field	Name	R/W	Description
			1: Enable
18	Reserved		
17:16	CLKOUT_SEL[1:0]	R/W	Clock output select 00: System clock 01: HSI clock 10: LSI clock 11: CPU HCLK clock
15:11	Reserved		
10:8	PDIV[2:0]	R/W	APB clock division 000: HCLK 001: HCLK 2 frequency division 010: HCLK 4 frequency division 011: HCLK 8 frequency division 100: HCLK 16 frequency division 101: HCLK 32 frequency division 110: HCLK 64 frequency division 111: HCLK 128 frequency division
7	Reserved		
6:4	HDIV[2:0]	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	SWST	R	Current system clock flag 0: HSI is the system clock 1: LSI is the system clock
0	SW	R/W	System clock select 0: HSI 1: LSI

4.2.4 Clock interrupt enable register (SYS_CIER)

Offset address: 0x0C

Reset value: 0x0000 0000

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

Field	Name	R/W	Description
1	Reserved		
0	LSI_RDY_IE	R/W	LSI clock stabilization interrupt enable 0: Disable 1: Enable

4.2.5 Clock interrupt flag register (SYS_CICR)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:3	Reserved		
2	HSI_RDYF	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	LSI_RDYF	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.

4.2.6 AHB peripheral reset register (SYS_AHBRST)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:9	Reserved		
8	FLASH_RST	R/W	FLASH controller reset 0: Not reset 1: Reset
7	Reserved		
6	DIV_SHIFT_RST	R/W	Divider reset 0: Not reset 1: Reset
5	CRC_RST	R/W	CRC reset 0: Not reset 1: Reset
4	DMA_RST	R/W	DMA reset 0: Not reset 1: Reset
3:2	Reserved		

Field	Name	R/W	Description
1	GPIOB_RST	R/W	GPIOB reset 0: Not reset 1: Reset
0	GPIOA_RST	R/W	GPIOA reset 0: Not reset 1: Reset

4.2.7 APB peripheral reset register (SYS_APBRST)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	LPTMR_RST	R/W	LPTMR reset 0: Not reset 1: Reset
14	ADC_RST	R/W	ADC reset 0: Not reset 1: Reset
13	COMP_RST	R/W	COMP reset 0: Not reset 1: Reset
12:11	Reserved		
10	EINT_RST	R/W	EINT reset 0: Not reset 1: Reset
9:7	Reserved		
6	UART1_RST	R/W	UART1 reset 0: Not reset 1: Reset
5	UART0_RST	R/W	UART0 reset 0: Not reset 1: Reset
4	SPI_RST	R/W	SPI reset 0: Not reset 1: Reset
3	Reserved		
2	BTMR_RST	R/W	BTMR reset 0: Not reset 1: Reset
1	GTMR_RST	R/W	GTMR reset 0: Not reset 1: Reset

Field	Name	R/W	Description
0	ATMR_RST	R/W	ATMR reset 0: Not reset 1: Reset

4.2.8 AHB peripheral clock enable register (SYS_AHBCG)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:7	Reserved		
6	DIV_SHIFT_EN	R/W	Divider clock enable 0: Disable 1: Enable
5	CRC_EN	R/W	CRC clock enable 0: Disable 1: Enable
4	DMA_EN	R/W	DMA clock enable 0: Disable 1: Enable
3:2	Reserved		
1	GPIOB_EN	R/W	GPIOB clock enable 0: Disable 1: Enable
0	GPIOA_EN	R/W	GPIOA clock enable 0: Disable 1: Enable

4.2.9 APB peripheral clock enable register (SYS_APBCG)

Offset address: 0x20

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15	LPTMR_EN	R/W	LPTMR clock enable 0: Disable 1: Enable
14	ADC_EN	R/W	ADC clock enable 0: Disable 1: Enable
13	COMP_EN	R/W	COMP clock enable 0: Disable 1: Enable
12:11	Reserved		
10	EINT_EN	R/W	EINT clock enable 0: Disable

Field	Name	R/W	Description
			1: Enable
9	Reserved		
8	WWDT_EN	R/W	WWDT clock enable 0: Disable 1: Enable The software can only write 1. Writing 0 is invalid.
7	Reserved		
6	UART1_EN	R/W	UART1 clock enable 0: Disable 1: Enable
5	UART0_EN	R/W	UART0 clock enable 0: Disable 1: Enable
4	SPI_EN	R/W	SPI clock enable 0: Disable 1: Enable
3	Reserved		
2	BTMR_EN	R/W	BTMR clock enable 0: Disable 1: Enable
1	GTMR_EN	R/W	GTMR clock enable 0: Disable 1: Enable
0	ATMR_EN	R/W	ATMR clock enable 0: Disable 1: Enable

4.2.10 Reset control/status register (SYS_RSTCSR)

Offset address: 0x28

Reset value: 0x0000 0080

Reset range: Only POR reset

Field	Name	R/W	Description
31:16	Reserved		
15	LOCKUP_RST_EN	R/W	LOCKUP system reset enable 0: Disable LOCKUP system reset 1: When CPU generates LOCKUP, the chip performs a system reset
14	PVD_RST_EN	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
13:8	Reserved		

Field	Name	R/W	Description
7	POR_RSTF	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	LOCKUP_RSTF	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.
5	WWDT_RSTF	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	IWDT_RSTF	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	SFT_RSTF	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	PVD_RSTF	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	NRST_RSTF	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	OP_RSTF	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.

4.2.11 ADC system control register (SYS_ADCCR)

Offset address: 0x2C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:2	Reserved		

Field	Name	R/W	Description
1:0	ADC_CLK_DIV	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

4.2.12 OPAMP control register (SYS_OPAMPCR)

Offset address: 0x30

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:19	Reserved		
18:16	OPAMP_SEL_MUX	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	OPAMP1_OUT_CTRL	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	OPAMP1_IN_CTRL	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	OPAMP1_SEL_GAIN	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	OPAMP1_EN	R/W	Operational Amplifier 1 enable 0: Disable 1: Enable
7:6	Reserved		
5	OPAMP0_OUT_CTRL	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	OPAMP0_IN_CTRL	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

Field	Name	R/W	Description
3:1	OPAMP0_SEL_GAIN	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 4 011: 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	OPAMP0_EN	R/W	Operational Amplifier 0 enable 0: Disable 1: Enable

4.2.13 PVD control/status register (SYS_PVDCSR)

Offset address: 0x3C

Reset value: 0x0000 0008

Field	Name	R/W	Description
31:16	Reserved		
15	PVDO	R	PVD monitoring results output 0: VBB is lower than PVD threshold voltage 1: VBB is higher than PVD threshold voltage Note: When PVD_EN is not enabled or within the analog setup start time, this bit remains 0.
14	PVDF	R/W	PVD interrupt event flag
13:6	Reserved		
5	PVD_HT	R/W	PVD events above the threshold voltage monitoring Note: After PVD_EN is enabled, this bit cannot be modified.
4	PVD_LT	R/W	PVD events below the threshold voltage monitoring Note: After PVD_EN is enabled, this bit cannot be modified.
3:1	PVDTHSEL[2:0]	R/W	PVD threshold voltage configure 000: 2.6V/2.7V 001: 2.9V/3.0V 010: 3.2V/3.3V 011: 3.5V/3.6V 100: 3.8V/3.9V (default) 101: 4.1V/4.2V 110: 4.4V/4.5V 111: 4.7V/4.8V Note: After PVD_EN is enabled, this setting cannot be modified.
0	PVD_EN	R/W	PVD enable 0: Disable 1: Enable

4.2.14 Gate driver control register (SYS_PREDRVCR)

Offset address: 0x40

Reset value: 0x0000 0AA0

Field	Name	R/W	Description
31:12	Reserved		
11:10	PREDRV_HO_RISE_SEL	R/W	High-side drive output slope adjustment (HO rising edge time) 00：140ns 01：90ns 10：55ns(default) 11：45ns Note: When PREDRV_EN=1, this bit cannot be modified. Load capacitance: 470pF
9:8	PREDRV_HO_FALL_SEL	R/W	High-side drive output slope adjustment (HO falling edge time) 00：630ns 01：350ns 10：200ns(default) 11：170ns Note: When PREDRV_EN=1, this bit cannot be modified. Load capacitance: 470pF
7:6	PREDRV_LO_RISE_SEL	R/W	Low-side drive output slope adjustment (LO rising edge time) 00：870ns 01：450ns 10：290ns(default) 11：250ns Note: When PREDRV_EN=1, this bit cannot be modified. Load capacitance: 470pF
5:4	PREDRV_LO_FALL_SEL	R/W	Low-side drive output slope adjustment (LO falling edge time) 00：160ns 01：110ns 10：80ns(default) 11：60ns Note: When PREDRV_EN=1, this bit cannot be modified. Load capacitance: 470pF
3:2	Reserved		
1	PREDRV_SW	R/W	Gate Driver switch When VSUP_F is 1, this bit is hardware protected to 0.
0	PREDRV_EN	R/W	Gate Driver switch enable When PREDRV_SW is 0 or VSUP_F is 1, this bit is hardware protected to 0.

4.2.15 Undervoltage control/status register (SYS_LVDCSR)

Offset address: 0x44

Reset value: 0x0000 0AA0

Field	Name	R/W	Description
31:5	Reserved		

Field	Name	R/W	Description
4	LVD_IE	R/W	LVD Interrupt Enable 0: Disable 1: Enable
3:1	Reserved		
0	LVDF	R	VSUP Under_Voltage Flag 0: No under-voltage occurred 1: The chip is currently in a state of under-voltage Note: A voltage lower than 3.6V during the VSUP ascent is in an under-voltage state, and a voltage lower than 3.4V during the VSUP descent is also in an under-voltage state

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 two forms, namely, system reset and power reset.

5.2.1 System reset

5.2.1.1 "System reset" reset source

The reset source is divided into external reset source and internal reset source.

External reset source:

- Low level on NRST pin.

Internal 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 SYS_RSTCSR (reset control/status register).

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

following:

- System reset control register SYS_RSTCSR
- PVD control register SYS_PVDCSR

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 FORCE_OPTLOAD bit in FLASH_CR register which is controlled by software.

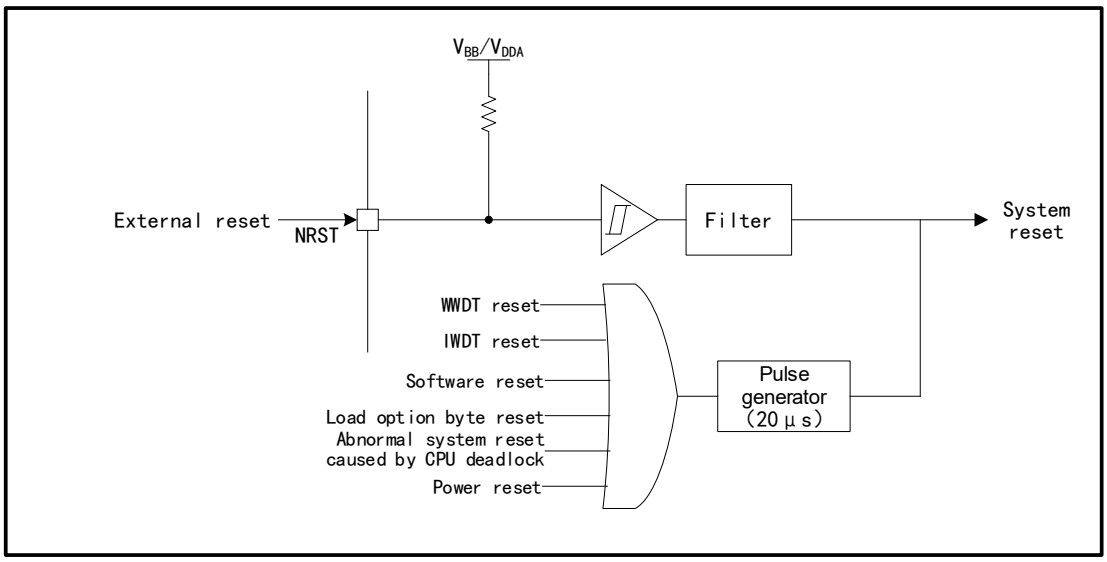
5.2.1.2 "System reset" reset circuit

When the external reset NRST generates a reset signal, it will first pass through an analog filter of approximately 50ns before acting 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 5 "System Reset" Reset Circuit



5.2.2 Power reset

"Power reset" reset source is as follows:

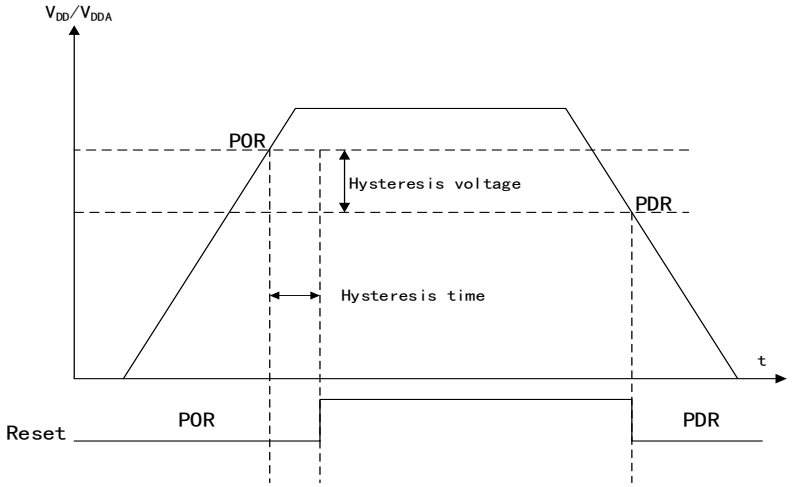
- Power-on reset (POR reset)
- Power-down reset (PDR reset)
- Chip temperature higher than 175℃

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

reset all registers.

5.2.3 Power-on/power-off reset timing

Figure 6 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 HSI_RDY bit of the SYS_RCCR register indicates whether HSI is stable. HSI clock will be input to the system only when the hardware sets the HSI_RDY bit to 1. If the HSI_RDY_IE bit of the SYS_CIER register is enabled, the corresponding interrupt will be generated. HSI can be enabled and disabled through the HSI_ON bit of the SYS_RCCR register. When the HSI_ON bit is configured to 0, the HSI_RDY flag will be cleared after 4 HSI clock cycles.

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 mode. The LSI_RDY bit of the SYS_RCCR register indicates whether LSI is stable. LSI clock will be input to the system only when the hardware sets the LSI_RDY bit to 1. If the LSI_RDY_IE bit of the SYS_CIER register is enabled, the corresponding interrupt will be generated. LSI can be enabled and disabled

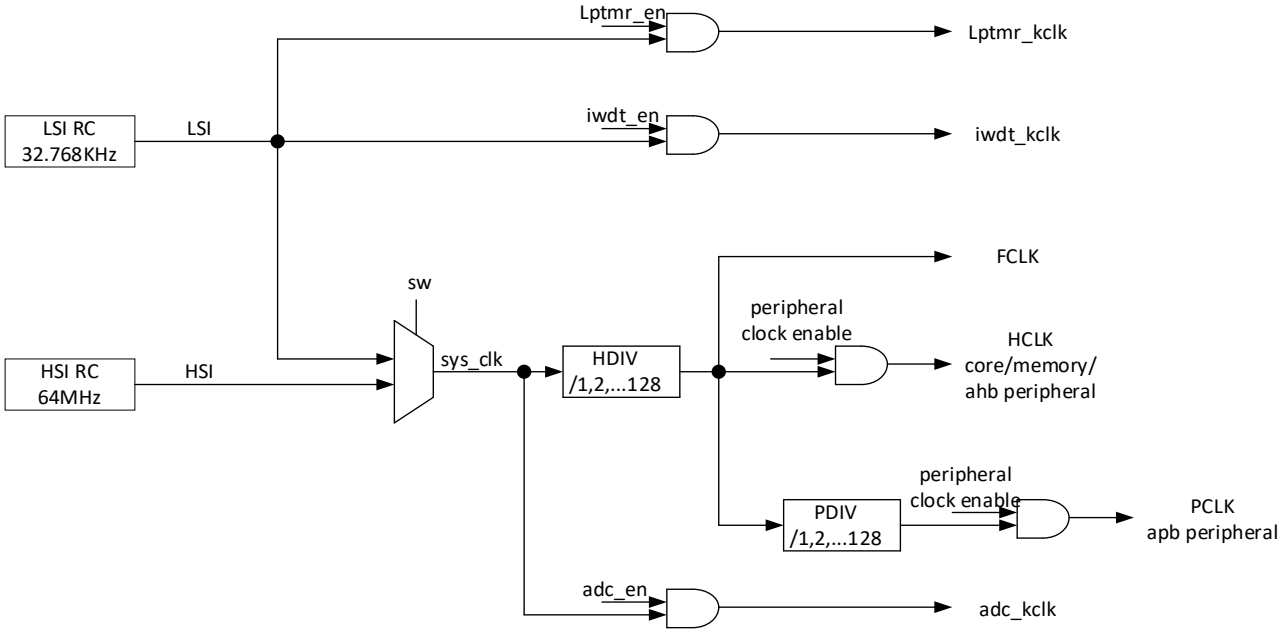
through the LSI_ON bit of the SYS_RCCR register. When the LSI_ON bit is configured to 0, the LSI_RDY flag will be cleared after 2 LSI clock cycles.

To ensure the normal operation of IWDT, when the user turns off LSI_ON, the LSI_RDY flag will be cleared by hardware along with LSI_ON, but IWDT can continue counting, and the CLKOUT function can still output this clock.

5.3.3 Clock tree

Clock tree of G32M3101 is shown in the figure below:

Figure 7 G32M3101 Clock Tree



5.3.4 System Clock and Switching

HSI or LSI can be selected as the system clock. 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 SW in SYS_SCCR to the corresponding configuration

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

The clock source used as the system clock is automatically protected by hardware, and software cannot turn it off. When switching the system clock, if the target clock is not yet turned on or stable, the system will maintain the current clock source and automatically wait for the target clock to stabilize before switching. During the switching process, the hardware automatically protects and prohibits turning off the clock source. After the switch is completed, the original clock can be turned off as needed.

5.3.5 Bus clock

Built-in AHB and APB buses. The clock source of AHB is the system clock, which is divided according to the configuration bit HDIV[2:0], with a maximum frequency of 64MHz. The clock source of APB is the AHB clock, which is divided according to the configuration bit PDIV[2:0], with a maximum frequency of 64MHz.

5.3.6 CLKOUT

The internal clock can be output to GPIO by configuring the CLKOUT_DIV, CLKOUT_EN, and CLKOUT_SEL bits of the SYS_SCCR register. Clock output sources include the following:

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

Note: When this function is enabled or the output source is switched, clock glitches may occur.

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 LSI_ON, but IWDT can still count normally after the software turns off LSI_ON.

5.3.8 Clock source selection of LPTMR

The working clock source of LPTMR is the LSI clock. It is necessary to turn on LSI_ON, confirm that the LSI_RDY clock is stable, and then turn on the LPTMR module. When the user turns off LSI_ON, LPTMR will stop counting.

5.3.9 Clock source selection of ADC

The working clock source of ADC is the system clock, and the division configuration is controlled by ADC_CLK_DIV[1:0], with divider factors of 2, 4, 8, and 16 respectively. When the system is in low power mode, ADC does not work because there is no system clock.

6 Power management unit (PMU)

6.1 Full Name and Abbreviation Description of Terms

Table 17 Full Name and Abbreviation Description of Terms

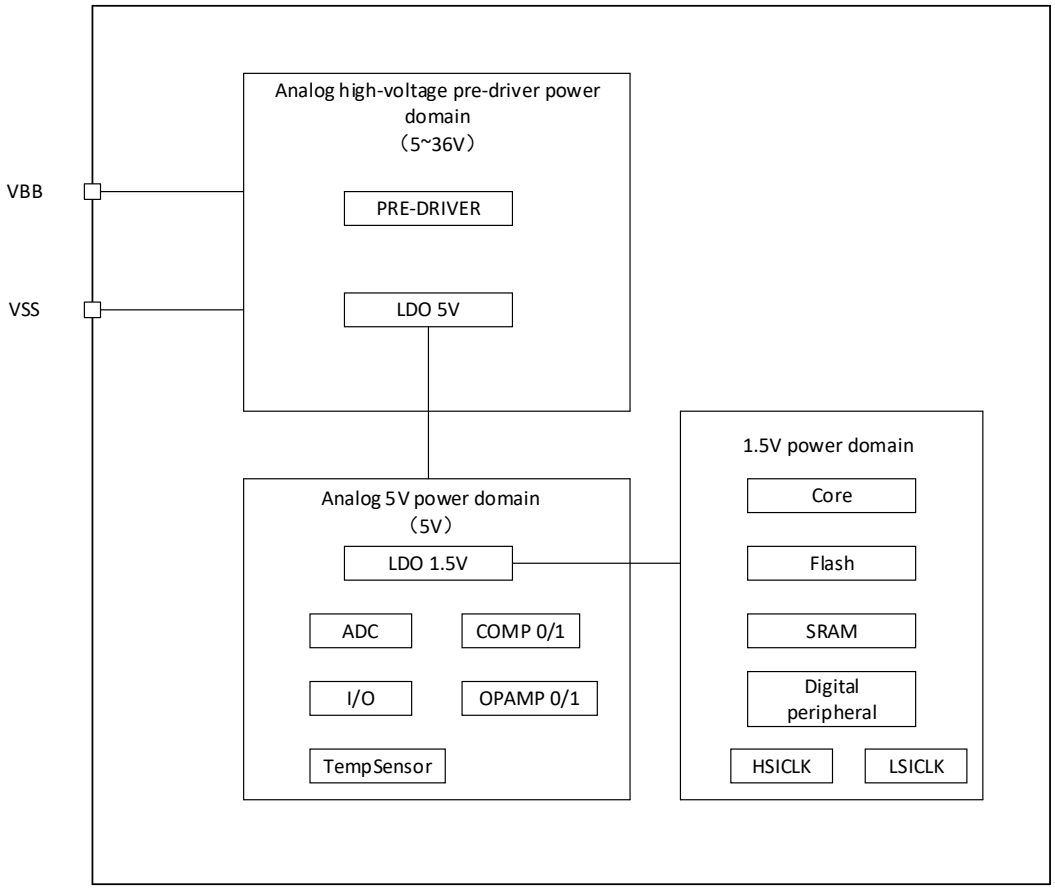
Full name in English	English abbreviation
Power Management Unit	PMU
Power On Reset	POR
Power Down Reset	PDR
Power Voltage Detector	PVD

6.2 Introduction

The power supply is the foundation for stable operation of a system, with an operating voltage of 5 ~ 36V, and 1.5V power supply can be provided by the built-in voltage regulator.

6.3 Structure block diagram

Figure 8 Power Supply Control Structure Block Diagram



6.4 Functional description

6.4.1 Power domain

The power domain of the product includes: analog high-voltage Gate Driver power domain, analog 5V power domain, and 1.5V power domain.

Table 18 Power Domains

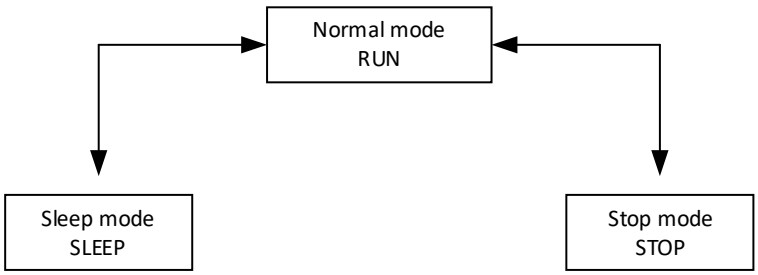
Name	Voltage range (V)	Description
V_{BB}	5～36V	Power the I/O (for specific I/O, see the pin distribution diagram) and the internal voltage regulator through the VBB pins.
V_{DD5}	5V	The LDO reduces the input power supply voltage to 5V (VDD5) to supply power to all modules except the Gate Driver.

6.4.2 Power control

6.4.2.1 Reduce the power in low-power mode

There are two low-power modes: sleep mode and stop 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 9 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 is shown in the table.

Table 19 Peripheral Working Status in STOP Mode

	STOP	
	state (1)	wake-up source
CPU Core	-	-
FLASH	-	-
SRAM	-(2)	-
HSI	-	-
LSI	O	-
POR/PDR	Y	Y
SYSTICK	-	-
PVD	O	O

	STOP	
DIV_SHIFT	-	-
CRC	-	-
GPIO	O	O (4)
DMA	-	-
LPTMR	O	O
ADC	-	-
COMP	-	-
OPAMP	-	-
IWDT	O	O
WWDT	-	-
UART1	-	-
UART0	-	-
SPI	-	-
BTMR	-	-
GTMR	-	-
ATMR	-	-

Note:

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

(2) SRAM retains data in STOP mode.

(3) GPIO pins can be woken up by configuring the EINT module in STOP mode.

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, with a frequency of 64MHz.

Sleep mode

The characteristics of sleep mode are shown in the table below

Table 20 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	Wake up immediately

Characteristics	Description
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 21 Characteristics of Stop Mode

Characteristics	Description
Enter	Set the SLEEPDEEP bit of the core register to 1 Set the LPM bit of SYS_LPMCR in the SYSCTRL 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	Wake up through relevant interrupts or events configured by EINT
Wake up delay	HSI analog clock establishment and stabilization time + FLASH exit DEEP STANDBY time (typical: 29 μ s)
After wake-up	The system clock automatically switches to HSI clock 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 stop mode, and the system re-executes the program after waking up.

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 ($_{VPOR}$/PDR), the system remains in the reset state.

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

Function	Description
PVD status	The PVDO flag in SYS_PVDCSR can indicate the current VBB state. To avoid false triggering, when the PVD function is turned off or just turned on for about 20us (during the function establishment time), this flag remains 0.
PVD interrupt	PVD_HT and PVD_LT in SYS_PVDCSR can monitor above/below the threshold. If a monitoring event is triggered, PVDF 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 SoC can be set to a safe state through the interrupt service program.

Function	Description
PVD system reset	Configure PVD_RST_EN in SYS_RSTCSR to enable the PVD system reset function. When VBB is lower than the threshold, the system enters system reset until VBB returns to above the threshold voltage, then it resumes normal mode and re-executes the program.

7 Nested Vector Interrupt Controller (NVIC)

7.1 Full Name and Abbreviation Description of Terms

Table 22 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 23 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
LVD	2	Can set	0x0000_0048	LVD detection interrupt
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_13	7	Can set	0x0000_005C	EINT line [13:4] 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
DMA_CH2	11	Can set	0x0000_006C	DMA channel 2 interrupt
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
BTMR	17	Can set	0x0000_0084	Btimer interrupt
-	18	-	0x0000_0088	Reserved
LPTMR	19	Can set	0x0000_008C	LPtimer interrupt
-	20	-	0x0000_0090	Reserved
COMP	21	Can set	0x0000_0094	COMP interrupt
-	22	-	0x0000_0098	Reserved
-	23	-	0x0000_009C	Reserved
-	24	-	0x0000_00A0	Reserved
SPI	25	Can set	0x0000_00A4	SPI interrupt
-	26	-	0x0000_00A8	Reserved
UART0	27	Can set	0x0000_00AC	UART0 interrupt
UART1	28	Can set	0x0000_00B0	UART1 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 16 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

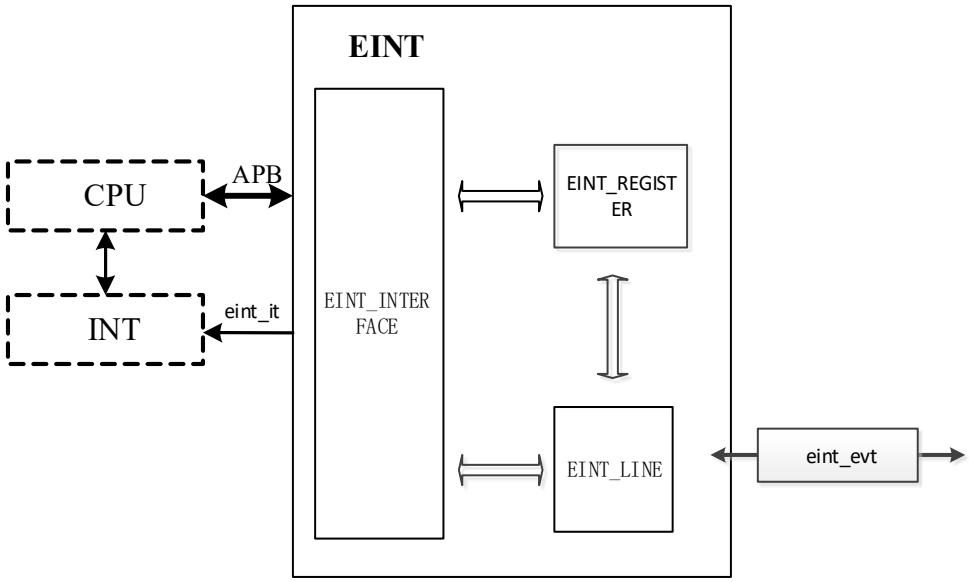
(5) Each external interrupt line has dedicated status bit

(6) Simulate all external event interrupts

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

8.3 Structure block diagram

Figure 10 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 24 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 2 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.

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 25 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
…	…
PA9/PB9	EINT 9
PB10	EINT 10
PB11	EINT 11
PB12	EINT 12
PB13	EINT 13
Reserved	EINT 14

External Interrupt and Event Channel Name	External Interrupt and Event Line No.
Reserved	EINT 15
PVD output	EINT 16
LPTMR	EINT 17
Reserved	EINT 18
Reserved	EINT 19
Reserved	EINT 20
Reserved	EINT 21
Reserved	EINT 22
Reserved	EINT 23
Reserved	EINT 24
Reserved	EINT 25
Reserved	EINT 26
Reserved	EINT 27
Reserved	EINT 28
Reserved	EINT 29
Reserved	EINT 30
Reserved	EINT 31

8.4.6 Direction for use

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

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

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

(4) 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

8.5 Register address mapping

Table 26 External Interrupt/Event Controller Register Address Mapping

Register name	Description	Offset address
EINT_IMASK	Interrupt mask register	0x00
EINT_EMASK	Event mask register	0x04

Register name	Description	Offset address
EINT_RTEN	Enable the rising edge to trigger the register	0x08
EINT_FTEN	Enable the falling edge to trigger the register	0x0C
EINT_SWINTE	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		
17:16	IMASKx	R/W	Interrupt Request Mask on Line x (x=16~17) 0: Mask 1: Open
15:14	Reserved		
13:0	IMASKx	R/W	Interrupt Request Mask on Line x (x=0~13) 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		
17:16	EMASKx	R/W	Event Request Mask on Line x (x=16~17) 0: Mask 1: Open
15:14	Reserved		
13:0	EMASKx	R/W	Event Request Mask on Line x (x=0~13) 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		

Field	Name	R/W	Description
17:16	RTENx	R/W	Rising Trigger Event Enable and Interrupt of Line x (x=16~17) 0: Disable 1: Enable
15:14	Reserved		
13:0	RTENx	R/W	Rising Trigger Event Enable and Interrupt of Line x (x=0~13) 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		
17:16	FTENx	R/W	Falling Trigger Event Enable and Interrupt of Line x (x=16-17) 0: Disable 1: Enable
15:14	Reserved		
13:0	FTENx	R/W	Falling Trigger Event Enable and Interrupt of Line x (x=0-13) 0: Disable 1: Enable

8.6.5 Software interrupt event register (EINT_SWINTE)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:16	SWINTEx	R/W	Software Interrupt Event on Line x (x=16~17) Set 1 by software, write 1 or clear 0 for the corresponding bit of EINT_IPEND. 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)
15:14	Reserved		
13:0	SWINTEx	R/W	Software Interrupt Event on Line x (x=0~13) Set 1 by software, write 1 or clear 0 for the corresponding bit of EINT_IPEND. 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		
17:16	IPENDx	RC_W1	Interrupt Pending Occur of Line x Flag (x=16~17) 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.
15:14	Reserved		
13:0	IPENDx	RC_W1	Interrupt Pending Occur of Line x Flag (x=0~13) 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
31:28	CFG_EINT7	R/W	EINT7 pin Configure 00: PA7 01: PB7
27:24	CFG_EINT6	R/W	EINT6 pin Configure 00: PA6 01: PB6
23:20	CFG_EINT5	R/W	EINT5 pin Configure 00: PA5 01: PB5
19:16	CFG_EINT4	R/W	EINT4 pin Configure 00: PA4 01: PB4
15:12	CFG_EINT3	R/W	EINT3 pin Configure 00: PA3 01: PB3
11:8	CFG_EINT2	R/W	EINT2 pin Configure 00: PA2 01: PB2

Field	Name	R/W	Description
7:4	CFG_EINT1	R/W	EINT1 pin Configure 00: PA1 01: PB1
3:0	CFG_EINT0	R/W	EINT0 pin Configure 00: PA0 01: PB0

8.6.8 Configuration register 1 (EINT_CFG1)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:24	Reserved		
23:20	CFG_EINT13	R/W	EINT13 pin Configure 00: not have 01: PB13
19:16	CFG_EINT12	R/W	EINT12 pin Configure 00: not have 01: PB12
15:12	CFG_EINT11	R/W	EINT11 pin Configure 00: not have 01: PB11
11:8	CFG_EINT10	R/W	EINT10 pin Configure 00: not have 01: PB10
7:4	CFG_EINT9	R/W	EINT9 pin Configure 00: PA9 01: PB9
3:0	CFG_EINT8	R/W	EINT8 pin Configure 00: PA8 01: PB8

9 Direct memory access (DMA)

9.1 Full Name and Abbreviation Description of Terms

Table 27 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 3 channels in total. Each channel of DMA has exactly the same function. The request selection module selects among 8 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 3 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 3 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

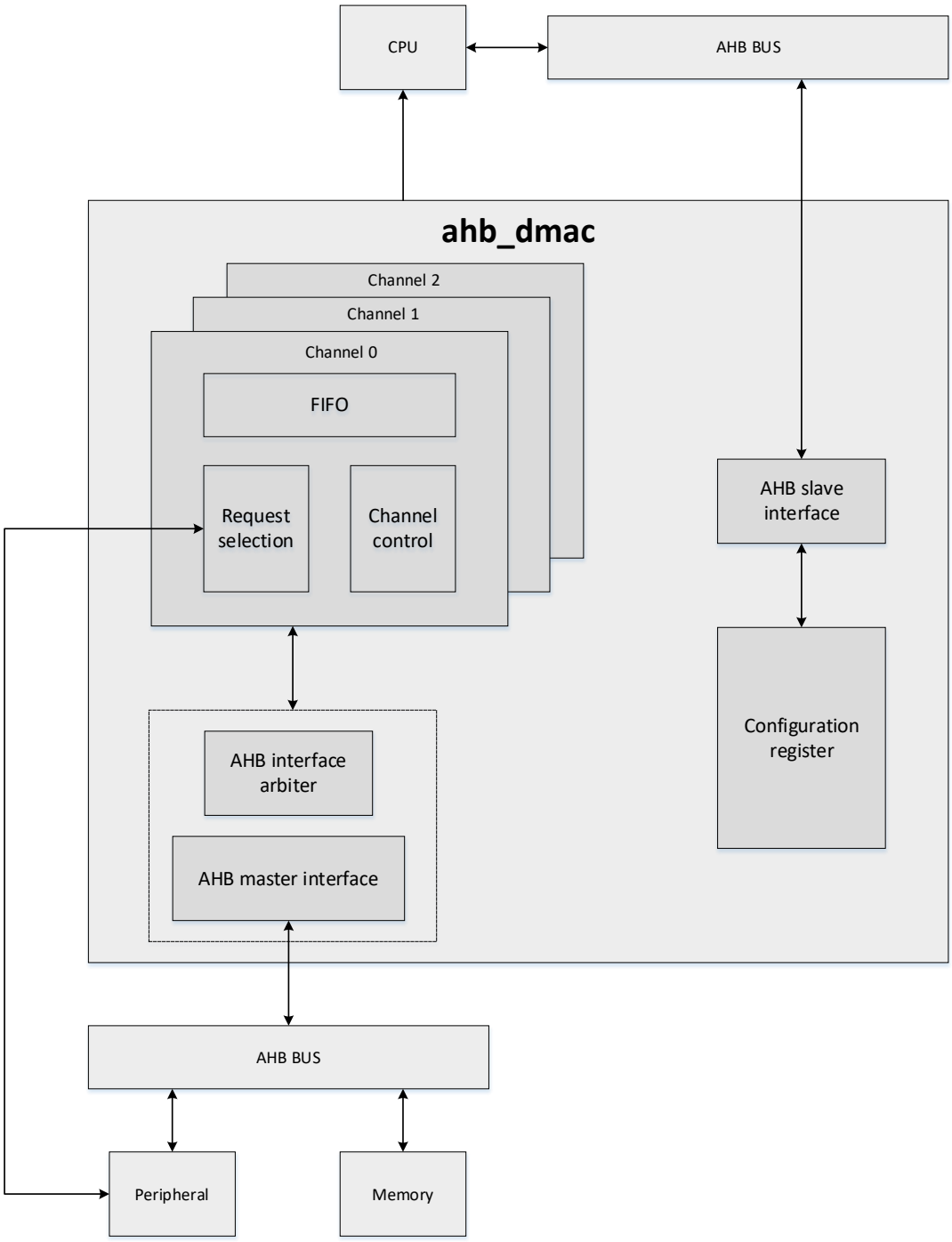
(9) Support source and target incremental modes

(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 PERSIZECFG bits

9.4 Structure block diagram

Figure 11 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 3 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 28 DMA Request Mapping Table

Peripheral	Channel 0	Channel 1	Channel 2
Peripheral 0	GTMR_CH0	GTMR_CH1	SPI_RX
Peripheral 1	GTMR_CH2	GTMR_UP	-
Peripheral 2	GTMR_CH3	-	-
Peripheral 3	ADC	ADC	ADC
Peripheral 4	SPI_RX	SPI_TX	SPI_TX
Peripheral 5	UART0_TX	UART0_RX	UART0_RX
Peripheral 6	UART1_RX	UART1_TX	UART1_TX
Peripheral 7	-	-	-

9.5.2 Arbiter

When multiple DMA channel requests 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 three priorities of high, medium 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 performs data transmission with memory and peripherals through the memory port and peripheral port. Both the memory port and peripheral port of DMA are connected to the AHB matrix bus. DMA memory and peripherals can access internal Flash, internal SRAM, AHB peripherals, and APB peripherals.

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; in direct mode, for memory-to-peripheral transmission, triggered when the memory-side bus access permission is not obtained until before the peripheral request; for peripheral-to-memory transmission, triggered when there is no response after more than 4 peripheral requests.
- Direct mode error interrupt: In direct mode, for peripheral-to-memory transmission, if MEMIM 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 PERIM and MEMIM 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 PERSIZECFG and MENSIZECFG bits of DMA_SCFG register. The PERSIZECFG and MENSIZECFG 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 PERSIZECFG and MENSIZECFG 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 29 FIFO Threshold Configuration

MENSIZECFG	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	
	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		

MENSIZECFG	FIFO threshold	MBCFG=01	MBCFG=10	MBCFG=11
	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 EN. If EN=1, first configure EN 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 EN to 1 to enable the channel

9.6 Register address mapping

Table 30 DMA Register Address Mapping Table

Register name	Description	Offset address
DMA_LINTSTS	DMA interrupt status register	0x00
DMA_LIFCLR	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_FCTRLx	DMA channel x FIFO control register	0x24+0x18 x

9.7 Register functional description

9.7.1 DMA interrupt status register (DMA_LINTSTS)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:22	Reserved		
21	TXCIFLG2	R	Channel 2 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_LIFCLR register. 0: No transmission completion event 1: Transmission completion event is generated.
20	HTXIFLG2	R	Channel 2 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_LIFCLR register. 0: No half-transmission event 1: Half-transmission event occurs
19	TXEIFLG2	R	Channel 2 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_LIFCLR register. 0: No transmission error 1: Transmission error occurs
18	DMEIFLG2	R	Channel 2 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_LIFCLR register. 0: No direct mode error 1: Direct mode error occurs
17	Reserved		

Field	Name	R/W	Description
16	FEIFLG2	R	Channel 2 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_LIFCLR register. 0: No FIFO error event 1: FIFO error event occurs
15:12	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_LIFCLR 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_LIFCLR 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_LIFCLR 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_LIFCLR register. 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_LIFCLR 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_LIFCLR 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_LIFCLR 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_LIFCLR register. 0: No transmission error

Field	Name	R/W	Description
			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_LIFCLR 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_LIFCLR register. 0: No FIFO error event 1: FIFO error event occurs

9.7.2 DMA interrupt flag clear register (DMA_LIFCLR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:22	Reserved		
21	CTXCIFLG2	W	Channel 2 Clear Transfer Complete Flag 0: Invalid 1: The TXCIFLG2 flag in DMA_LINTSTS register is cleared to 0
20	CHTXIFLG2	W	Channel 2 Clear Half Transfer Interrupt Flag 0: Invalid 1: The HTXIFLG2 flag in DMA_LINTSTS register is cleared to 0
19	CTXEIFLG2	W	Channel 2 Clear Transfer Error Interrupt Flag 0: Invalid 1: The TXEIFLG2 flag in DMA_LINTSTS register is cleared to 0
18	CDMEIFLG2	W	Channel 2 Clear Direct Mode Error Interrupt Flag 0: Invalid 1: The DMEIFLG2 flag in DMA_LINTSTS register is cleared to 0
17	Reserved		
16	CFEIFLG2	W	Channel 2 Clear FIFO Error Interrupt Flag 0: Invalid 1: The FEIFLG2 flag in DMA_LINTSTS register is set to 0
15:12	Reserved		
11	CTXCIFLG1	W	Channel 1 Clear Transfer Complete Interrupt Flag 0: Invalid 1: The TXCIFLG1 flag in DMA_LINTSTS register is cleared to 0
10	CHTXIFLG1	W	Channel 1 Clear Half Transfer Interrupt Flag 0: Invalid 1: The HTXIFLG1 flag in DMA_LINTSTS register is cleared to 0
9	CTXEIFLG1	W	Channel 1 Clear Transfer Error Interrupt Flag 0: Invalid

Field	Name	R/W	Description
			1: The TXEIFLG1 flag in DMA_LINTSTS register is cleared to 0
8	CDMEIFLG1	W	Channel 1 Clear Direct Mode Error Interrupt Flag 0: Invalid 1: The DMEIFLG1 flag in DMA_LINTSTS 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_LINTSTS register is set to 0
5	CTXCIFLG0	W	Channel 0 Clear Transfer Complete Interrupt Flag 0: Invalid 1: The TXCIFLG0 flag in DMA_LINTSTS register is cleared to 0
4	CHTXIFLG0	W	Channel 0 Clear Half Transfer Interrupt Flag 0: Invalid 1: The HTXIFLG0 flag in DMA_LINTSTS register is cleared to 0
3	CTXEIFLG0	W	Channel 0 Clear Transfer Error Interrupt Flag 0: Invalid 1: The TXEIFLG0 flag in DMA_LINTSTS register is cleared to 0
2	CDMEIFLG0	W	Channel 0 Clear Direct Mode Error Interrupt Flag 0: Invalid 1: The DMEIFLG0 flag in DMA_LINTSTS 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_LINTSTS register is set to 0

9.7.3 DMA channel x configuration register (DMA_SCFGx) (x=0,1,2)

Offset address: 0x10+0x18 x

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:28	Reserved		
27:25	PSEL	R/W	Peripheral Selection 000: Select Peripheral 0 001: Select Peripheral 1 010: Select Peripheral 2 011: Select Peripheral 3 100: Select Peripheral 4 101: Select Peripheral 5 110: Select Peripheral 6 111: Select Peripheral 7 These bits can be written only when EN 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)

Field	Name	R/W	Description
			11: INCR16 (16-tick increment burst transmission) This bit can be written only when EN 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 EN 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 EN 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 EN bit is 0.
17:16	PRILCFG	R/W	Priority Level Configure 00: Low 01: Medium 10: High 11: Reserved These bits can be written only when EN bit is 0.
15	PERIOSIZE	R/W	Peripheral increment offset size 0: The offset used to calculate the peripheral address is related to PERSIZECFG 1: The offset used to calculate the peripheral address is fixed to be 4 If PERIM bit is 0, this bit is meaningless, and it can be written only when EN bit is 0. 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	MEMSIZECFG	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 EN bit is 0. In direct mode, when EN bit is 1, MEMSIZECFG bit will be forced to be of the same value as that of PERSIZECFG bit.
12:11	PERSIZECFG	R/W	Peripheral Data Size Configure 00: Byte (8 bits)

Field	Name	R/W	Description
			01: Half word (16 bits) 10: Word (32 bits) 11: Reserved These bits can be written only when EN bit is 0.
10	MEMIM	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 EN bit is 0.
9	PERIM	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 EN 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 EN 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 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	EN	R/W	Channel Enable 0: Disable 1: Enable This bit shall be set to 0 by hardware in any of the following situations:

Field	Name	R/W	Description
			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=0,1,2)

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. This register is 0 after completion of transmission, and the initial value will be automatically reloaded in any of the following circumstances: Configure the channel in circular mode Re-enable channel

9.7.5 DMA channel x peripheral address register (DMA_PADDR) (x=0,1,2)

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 EN bit is 0.

9.7.6 DMA channel x memory 0 address register (DMA_M0ADDR) (x=0,1,2)

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=0,1,2)

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_FCTRL) (x=0,1,2)

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 EN bit is 0; when the memory-to-memory mode is selected and EN bit is 1, this bit will be set to 1 by hardware.
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 EN bit is 0.

10 General-Purpose Input/Output Pin (GPIO)

10.1 Full Name and Abbreviation Description of Terms

Table 31 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 24 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
- Configurable maximum output rate

(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) Support Schmitt input and non-Schmitt input

(7) Output rate: up to 20MHz

(8) Configurable output drive capability in two levels: 20mA/10mA

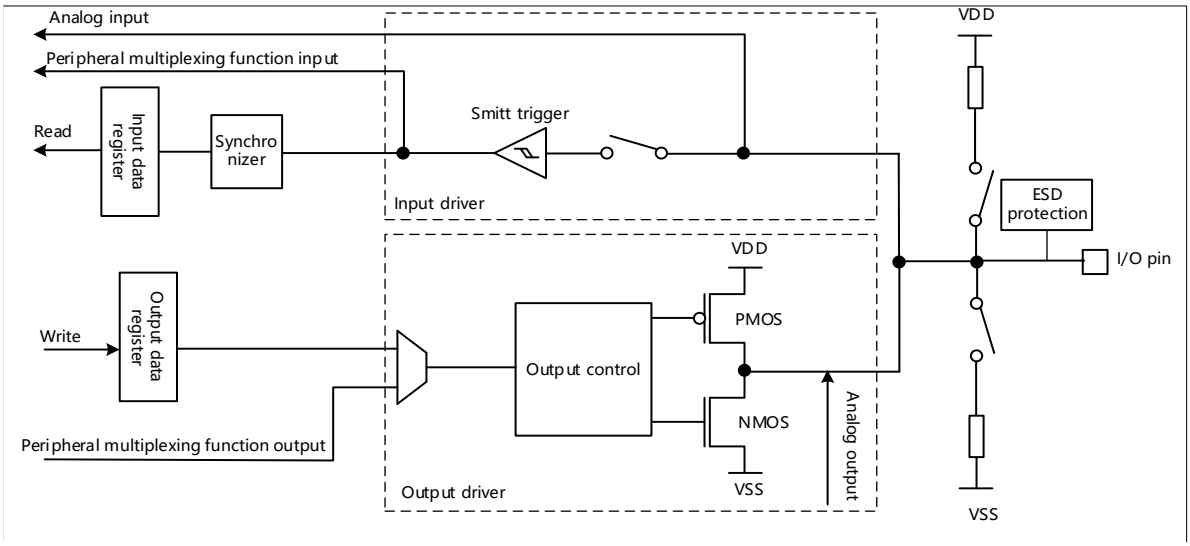
(9) Configurable latch protection: GPIO-related configuration registers have latch protection/unlock functions

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

(11) Input filter configurable: some IOs have configurable analog filter function (50ns)

10.4 Structure block diagram

Figure 12 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

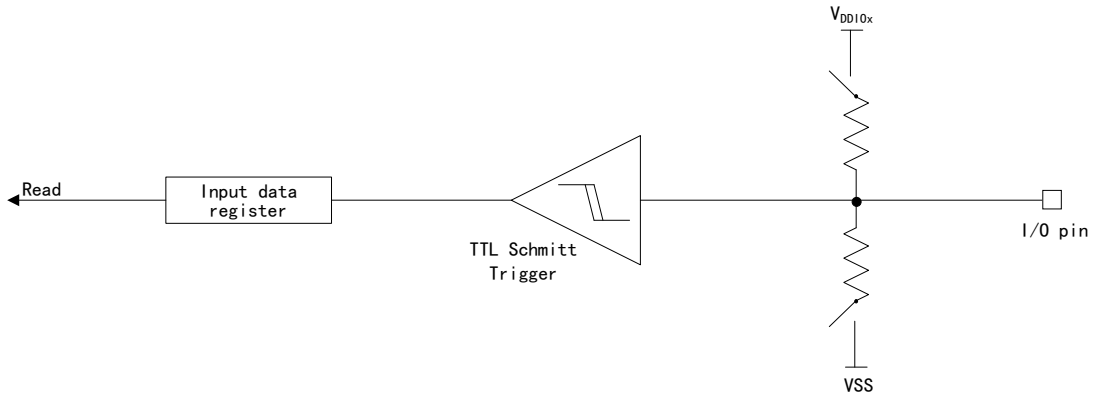
In (pull-up, pull-down, floating) input mode

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

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 13 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.

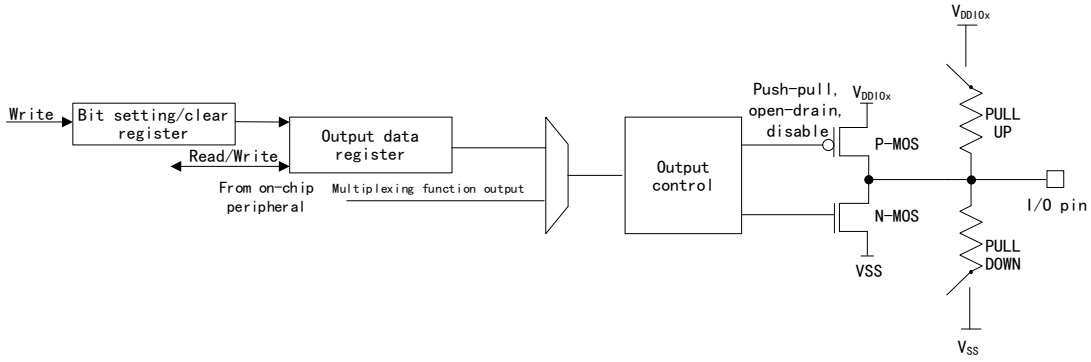
When GPIO is configured as the output pin, the output speed of the port can be configured and the output drive mode (push-pull/open-drain) can be selected.

In output mode

- Schmitt trigger is opened
- Activate the output buffer
- By configuring the pull-up/pull-down register GPIOx_PUPD, 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_ODATA
- 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_IDATA captures the data on I/O pin in each AHB clock cycle
- Read the actual I/O state through the input data register GPIOx_IDATA

Figure 14 Output Mode I/O Structure



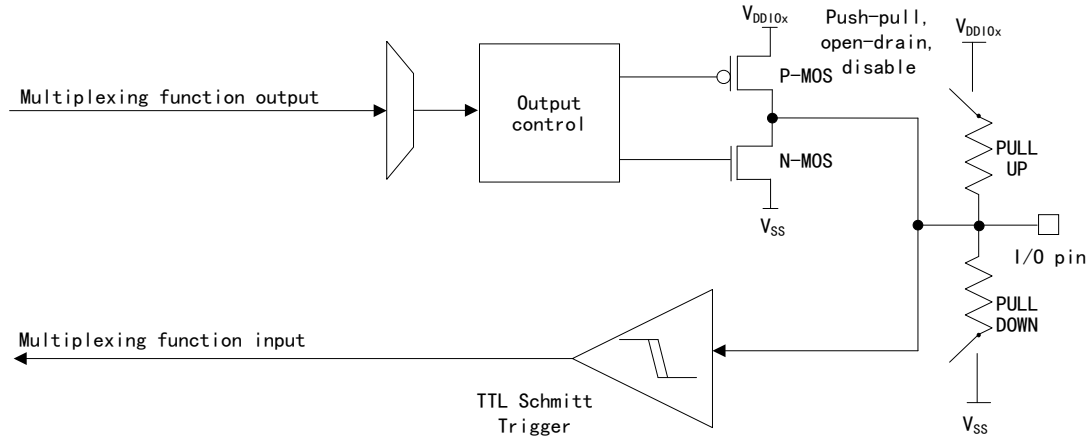
10.5.4 Multiplexing mode

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_PUPD, 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_IDATA

Figure 15 Multiplexing Mode I/O Structure

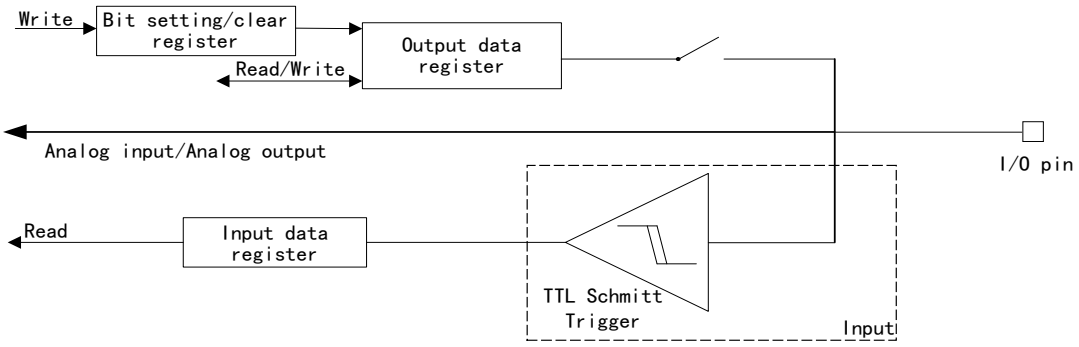


10.5.5 Analog mode

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 16 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_BSC) allows set/reset operation for each bit of the output data register (GPIOx_ODATA). The valid data width of the set/reset register is double the valid data width of GPIOx_ODATA.

Writing 0 to any bit in GPIOx_BSC will not affect the value of the GPIOx_ODATA register. BSy and BCy bits of GPIOx_BSC are set to 1 for the same time, operation of BSy bit has the priority. GPIOx_ BSC register can change the corresponding bit of the GPIOx_ODATA register, and GPIOx_ODATA bit can be accessed directly from GPIOx_BSC register.

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

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. The multiplexer has up to 16 multiplexing function inputs, but in fact it uses up to 8 (AF0-AF7), which are

configured by GPIOx_ALFSEL0 and GPIOx_ ALFSEL1 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:

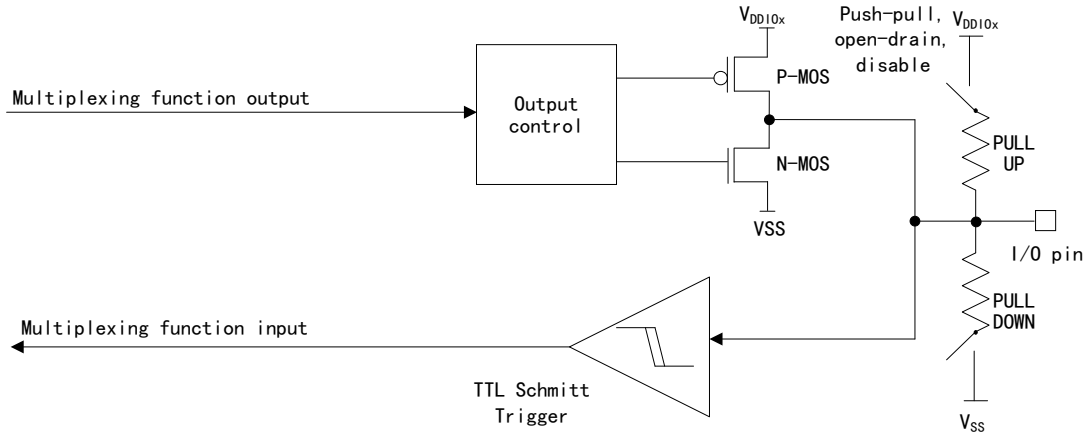
- After reset, the pin is configured with multiplexing function
- I/O port is configured as input, output or analog input
- The I/O port is connected to the defined AFx
- Configure pin pull-up/pull-down and output speed
- Configure I/O as multiplexing function in GPIOx_MODE

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_PUPD, 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_IDATA

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

Figure 17 Multiplexing Mode I/O Structure



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_MODE to 01 input mode

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

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

10.5.10.2 Output configuration

The configuration process is as follows:

(1) Unlock by configuring 0xA5A5_5A5A through GPIO_LOCK

(2) Configure GPIO_MODE to 10 output mode

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

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

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

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

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_MODE to 11 multiplex mode

(3) Configure GPIO_ALFSEL for multiplexing selection

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

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

(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_DCFG

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

10.5.10.4 Analog configuration

The configuration process is as follows:

(1) Unlock by configuring 0xA5A5_5A5A through GPIO_LOCK

(2) Configure GPIO_MODE 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:0x4002 1400

GPIOB base address:0x4002 1800

Table 32 GPIO Register Address Mapping

Register name	Description	Offset address
GPIOA_MODE	GPIOA mode register	0x00

GPIOA_IEN	GPIOA input enable register	0x04
GPIOA_ITYPCFG	GPIOA input type register	0x08
GPIOA_PUPD	GPIOA pull-up/down register	0x0C
GPIOA_OTYPCFG	GPIOA output type register	0x10
GPIOA_DCFG	GPIOA driving configuration register	0x14
GPIOA_IDATA	GPIOA input register	0x18
GPIOA_ODATA	GPIOA output data register	0x1C
GPIOA_BSC	GPIOA set/reset register	0x20
GPIOA_RST	GPIOA reset register	0x24
GPIOA_LOCK	GPIOA lock register	0x28
GPIOA_ALFSEL	GPIOA multiplex select register	0x2C
GPIOA_FLT	GPIOA filtering configuration register	0x34
GPIOB_MODE	GPIOB mode register	0x00
GPIOB_IEN	GPIOB input enable register	0x04
GPIOB_ITYPCFG	GPIOB input type register	0x08
GPIOB_PUPD	GPIOB pull-up/down register	0x0C
GPIOB_OTYPCFG	GPIOB output type register	0x10
GPIOB_DCFG	GPIOB driving configuration register	0x14
GPIOB_IDATA	GPIOB input register	0x18
GPIOB_ODATA	GPIOB output data register	0x1C
GPIOB_BSC	GPIOB set/reset register	0x20
GPIOB_RST	GPIOB reset register	0x24
GPIOB_LOCK	GPIOB lock register	0x28
GPIOB_ ALFSEL0	GPIOB low bit multiplex select register	0x2C
GPIOB_ ALFSEL1	GPIOB high bit multiplex select register	0x30

10.7 Register functional description

10.7.1 GPIOA Mode Register (GPIOA_MODE)

Offset address: 0x00

Reset value: 0x0005 7555

Field	Name	R/W	Description
31:20	Reserved		
19:18	MODE9	R/W	Work mode configure 00: Analog mode 01: General digital input mode

Field	Name	R/W	Description
			10: General digital output mode 11: Multiplexing function mode
17:16	MODE8	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
15:14	MODE7	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
13:12	MODE6	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
11:10	MODE5	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
9:8	MODE4	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
7:6	MODE3	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
5:4	MODE2	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
3:2	MODE1	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode 11: Multiplexing function mode
1:0	MODE0	R/W	Work mode configure 00: Analog mode 01: General digital input mode 10: General digital output mode

Field	Name	R/W	Description
			11: Multiplexing function mode

10.7.2 GPIOA Input Enable Register (GPIOA_IEN)

Offset address: 0x04

Reset value: 0x0000 0040

Field	Name	R/W	Description
31:10	Reserved		
x	IENx	R/W	Port enable (x=0~9) 0: Disable 1: Enable Note: Effective in non-analog mode.

10.7.3 GPIOA Input Type Register (GPIOA_ITYPCFG)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:10	Reserved		
x	ITYPCFGX	R/W	Input type configure (x=0~9) 0: Schmitt input 1: CMOS input

10.7.4 GPIOA Pull-Up/Pull-Down Register (GPIOA_PUPD)

Offset address: 0x0C

Reset value: 0x0000 1000

Field	Name	R/W	Description
31:20	Reserved		
2x+1	PUPDx	R/W	Up and down pull select (x=0~9) It takes effect after pull up and down is enabled. 0: Pull-up 1: Pull-down
2x	PUPDENx	R/W	Up and down pull enable (x=0~9) 0: Disable 1: Enable

10.7.5 GPIOA Output Type Register (GPIOA_OTYPCFG)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:10	Reserved		
x	OTYPCFGx	R/W	Output type configure (x=0~9) 0: Push-pull 1: Open-drain

10.7.6 GPIOA Drive Configuration Register (GPIOA_DCFG)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:10	Reserved		
x	DCFGx	R/W	Output driving capability Configure (x=0~9) 0: Low drive 1: High drive

10.7.7 GPIOA input data register (GPIOA_IDATA)

Offset address: 0x18

Reset value: 0x0000 0040

Field	Name	R/W	Description
31:10	Reserved		
x	IDATAX	R	Input data (x=0~9)

10.7.8 Port output data register (GPIOx_ODATA) (x=A…E, H)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:10	Reserved		
x	ODATAX	R/W	Output data (x=0~9)

10.7.9 GPIOA Set/Reset Register (GPIOA_BSC)

Offset address: 0x20

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:26	Reserved		
x+16	BCx	W	ODATAx Reset (x=0~9) Write-only, reading these bits returns 0. 0: No operation on the corresponding ODATAx bit 1: Reset the corresponding ODATAx bit Note: If BCx and PAx_BS are set at the same time, PAx_BS has higher priority.
15:10	Reserved		
x	BSx	W	ODATAx set (x=0~9) Write-only, reading these bits returns 0. 0: No operation on the corresponding ODATAx bit 1: Set the corresponding ODATAx bit to 1

10.7.10 GPIOA Reset Register (GPIOA_RST)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:10	Reserved		
x	BCx	W	ODATAx Reset (x=0~9) Write-only, reading these bits returns 0. 0: No operation on the corresponding ODATAx bit 1: Reset the corresponding ODATAx bit

10.7.11 GPIOA Lock Register (GPIOA_LOCK)

Offset address: 0x28

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	LOCKKEY	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.

10.7.12 GPIOA Multiplex Select Register (GPIOA_ALFSEL)

Offset address: 0x2C

Reset value: 0x001C 0000

Field	Name	R/W	Description
31:30	Reserved		
3x+2:3x	ALFSELx	R/W	Multiplex select (x=6~9) 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
17:0	Reserved		

10.7.13 GPIOA Filter Configuration Register (GPIOA_FLT)

Offset address: 0x34

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:7	Reserved		
6	PA6FLTEN	R/W	PA6 analog filtering enable 0: Disable 1: Enable
5:0	Reserved		

10.7.14 GPIOB Mode Register (GPIOB_MODE)

Offset address: 0x00

Reset value: 0x0D55 57D5

Field	Name	R/W	Description
31:28	Reserved		
2x+1:2x	MODEx	R/W	Work mode configure (x=0~13) 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_IEN)

Offset address: 0x04

Reset value: 0x0000 0018

Field	Name	R/W	Description
31:14	Reserved		
x	IENx	R/W	Port enable (x=0~13) 0: Disable 1: Enable Note: Effective in non-analog mode.

10.7.16 GPIOB Input Type Register (GPIOB_ITYPCFG)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:14	Reserved		
x	ITYPCFGX	R/W	Input type configure (x=0~13) 0: Schmitt input 1: CMOS input

10.7.17 GPIOB Pull-Up/Pull-Down Register (GPIOB_PUPD)

Offset address: 0x0C

Reset value: 0x0000 01C0

Field	Name	R/W	Description
31:28	Reserved		
2x+1	PUPDx	R/W	Up and down pull select (x=0~13) It takes effect after pull up and down is enabled. 0: Pull-up 1: Pull-down
2x	PUPDENx	R/W	Up and down pull enable (x=0~13) 0: Disable 1: Enable

10.7.18 GPIOB Output Type Register (GPIOB_OTYPCFG)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:14	Reserved		
x	OTYPCFGx	R/W	Output type configure (x=0~13) 0: Push-pull 1: Open-drain

10.7.19 GPIOB Drive Configuration Register (GPIOB_DCFG)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:14	Reserved		
x	DCFGx	R/W	Output driving capability configure (x=0~13) 0: Low drive 1: High drive

10.7.20 GPIOB Input Register (GPIOB_IDATA)

Offset address: 0x18

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:14	Reserved		
x	IDATAx	R	Input data (x=0~13)

10.7.21 GPIOB output data register (GPIOB_ODATA)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:14	Reserved		
x	ODATAx	R/W	Output data (x=0~13)

10.7.22 GPIOB Set/Reset Register (GPIOB_BSC)

Offset address: 0x20

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:30	Reserved		
x+16	BCx	W	ODATAx Reset (x=0~13) Write-only, reading these bits returns 0. 0: No operation on the corresponding ODATAx bit 1: Reset the corresponding ODATAx bit Note: If BCx and BSx are set at the same time, BSx has higher priority.
15:14	Reserved		

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

10.7.23 GPIOB Reset Register (GPIOB_RST)

Offset address: 0x24

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:14	Reserved		
x	BCx	W	ODATAx Reset (x=0~13) Write-only, reading these bits returns 0. 0: No operation on the corresponding ODATAx bit 1: Reset the corresponding ODATAx bit

10.7.24 GPIOB Lock Register (GPIOB_LOCK)

Offset address: 0x28

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	LOCKKEY	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.

10.7.25 GPIOB Low Bit Multiplex Select Register (GPIOB_ALFSEL0)

Offset address: 0x2C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:24	Reserved		
23:21	ALFSEL7	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
20:18	ALFSEL6	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
17:15	ALFSEL5	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
14:12	ALFSEL4	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
11:9	ALFSEL3	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
8:6	ALFSEL2	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
5:3	ALFSEL1	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
2:0	ALFSEL0	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7

10.7.26 GPIOB High Bit Multiplex Select Register (GPIOB_ALFSEL1)

Offset address: 0x30

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:15	ALFSEL13	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
14:12	ALFSEL12	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
11:9	ALFSEL11	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
8:6	ALFSEL10	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
5:3	ALFSEL9	R/W	Multiplex select 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7
2:0	ALFSEL8	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

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) Single-pulse mode

(8) Complementary output and dead zone insertion

(9) Timing function

(10) Braking function

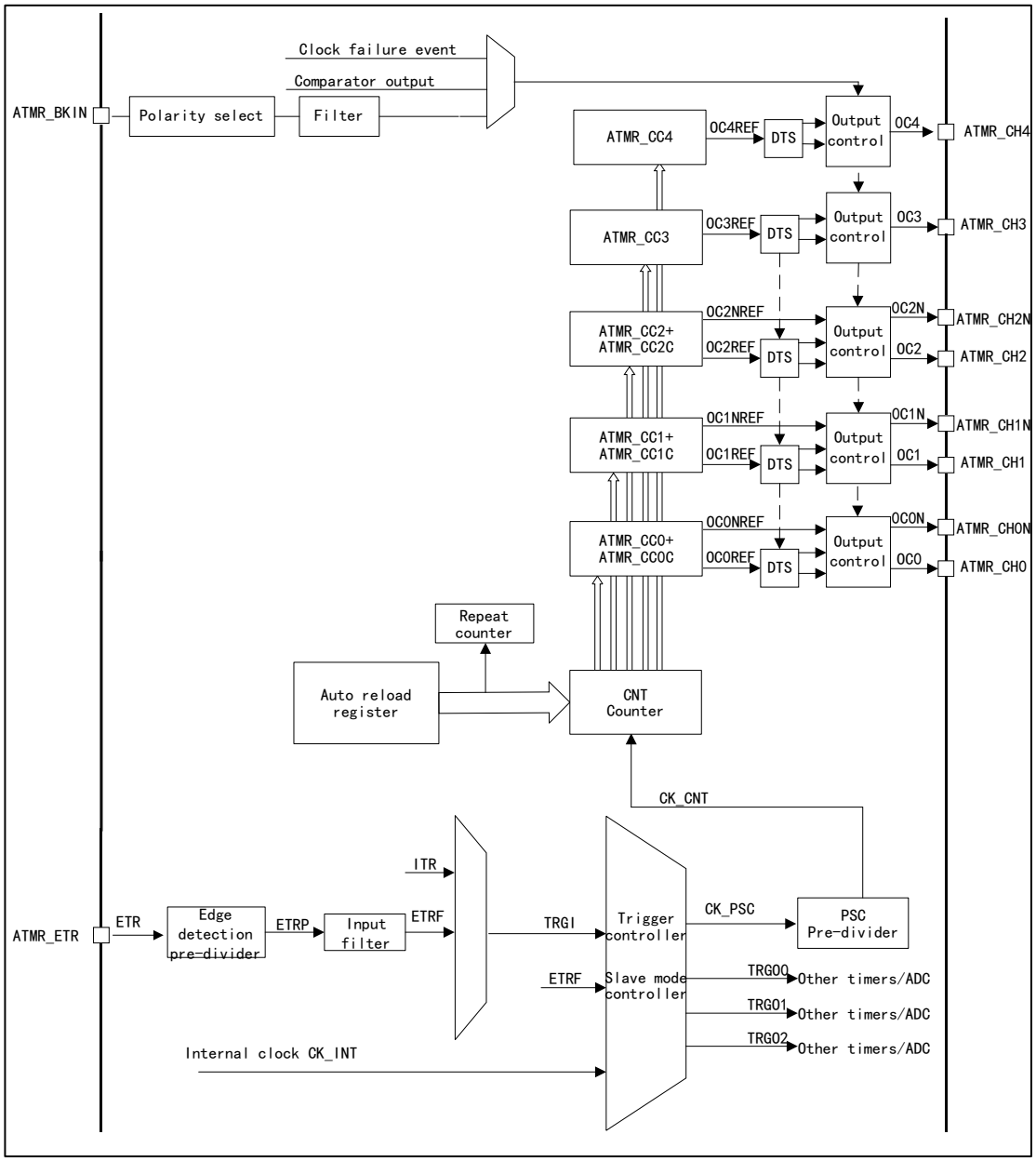
(11) Master/Slave mode controller of timer
- Timers can be synchronized and cascaded
- Support multiple slave modes and synchronization signals

(12) 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

(13) 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 18 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 2

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_CTRL1).

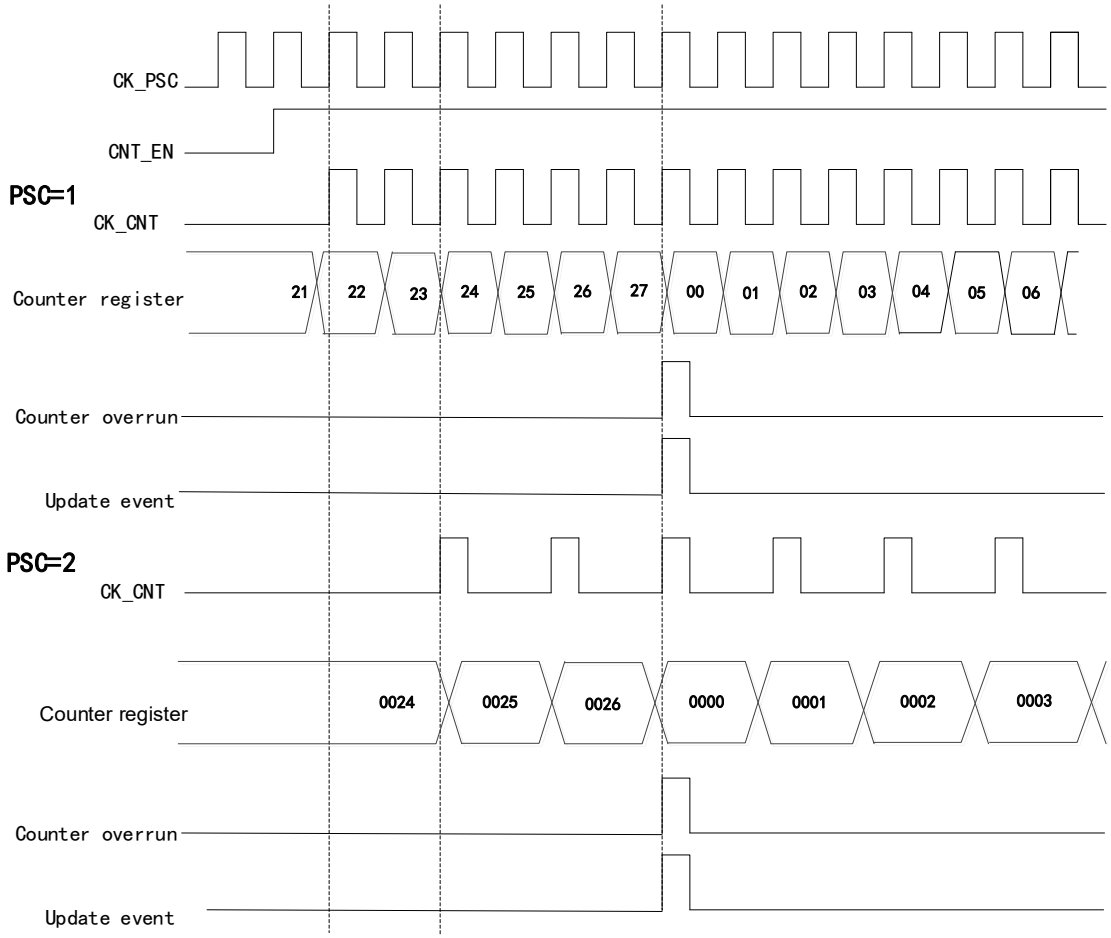
When the counter is in count-up mode, the counter will count up from 0; every time a pulse is generated, the counter will increase by 1 and when the value of the counter (ATMR_CNT) is equal to the value of the auto reload (ATMR_AUTORLD), the counter will start to count from 0 again, a count-up overrun 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 UD bit of control register ATMR_CTRL1.

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

Figure 19 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_CTRL1).

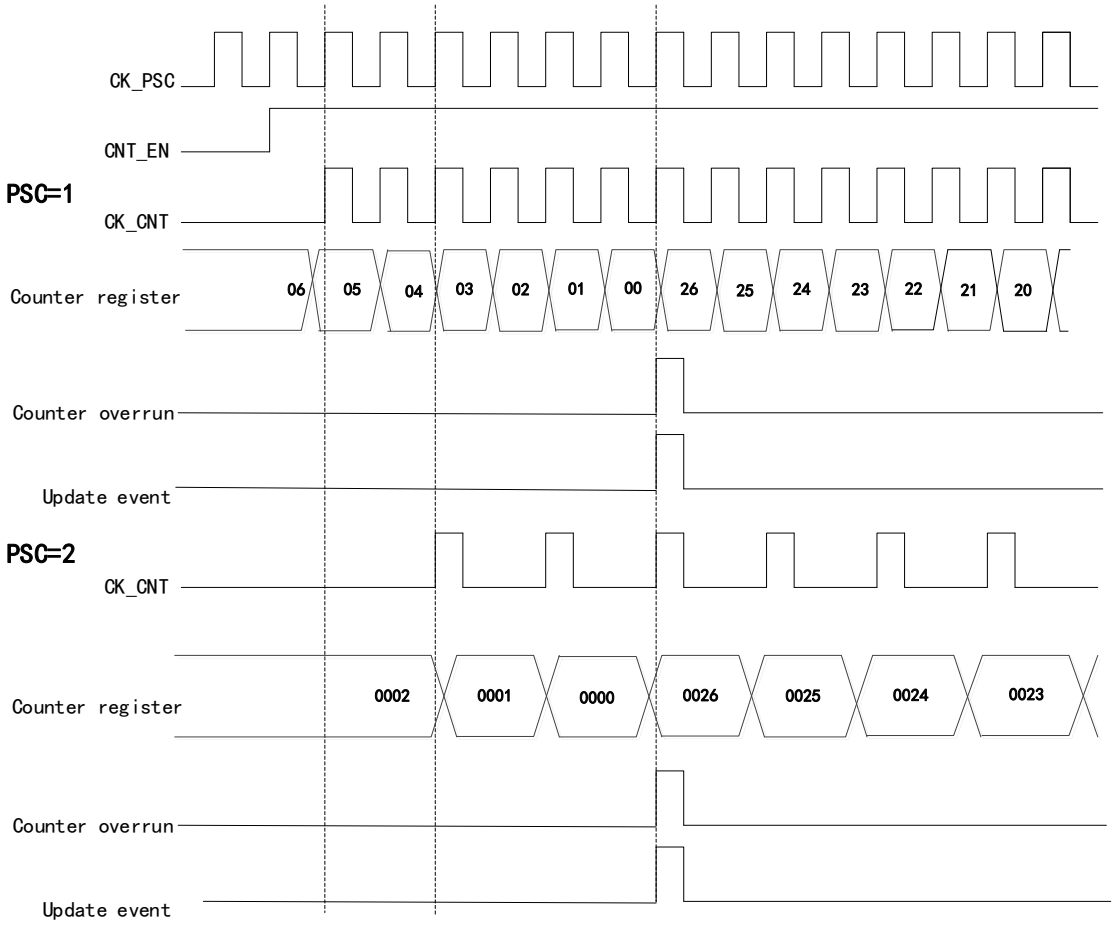
When the counter is in count-down mode, it will start to count down from the value of the auto reload (ATMR_AUTORLD); every time a pulse is generated, the counter will decrease by 1 and when it becomes 0, the counter will start to count again from (ATMR_AUTORLD), meanwhile, a count-down overrun event 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 UD bit of the ATMR_CTRL1 register.

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

Figure 20 Timing Diagram of Count-down Mode when Division Factor is 1 or 2



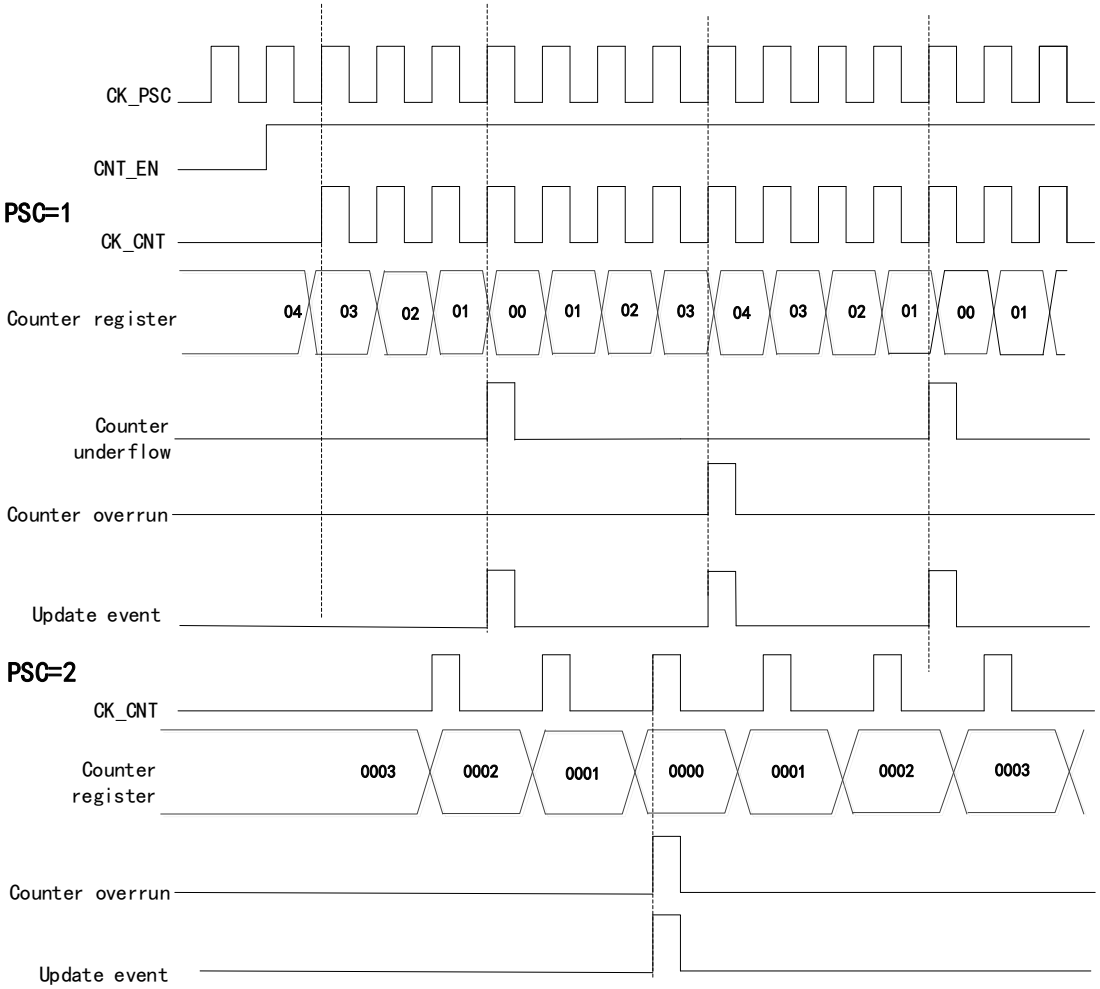
Central alignment mode

Set to the central alignment mode by configuring CAMSEL bit of control register (ATMR_CTRL1).

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



Repeat counter REPCNT

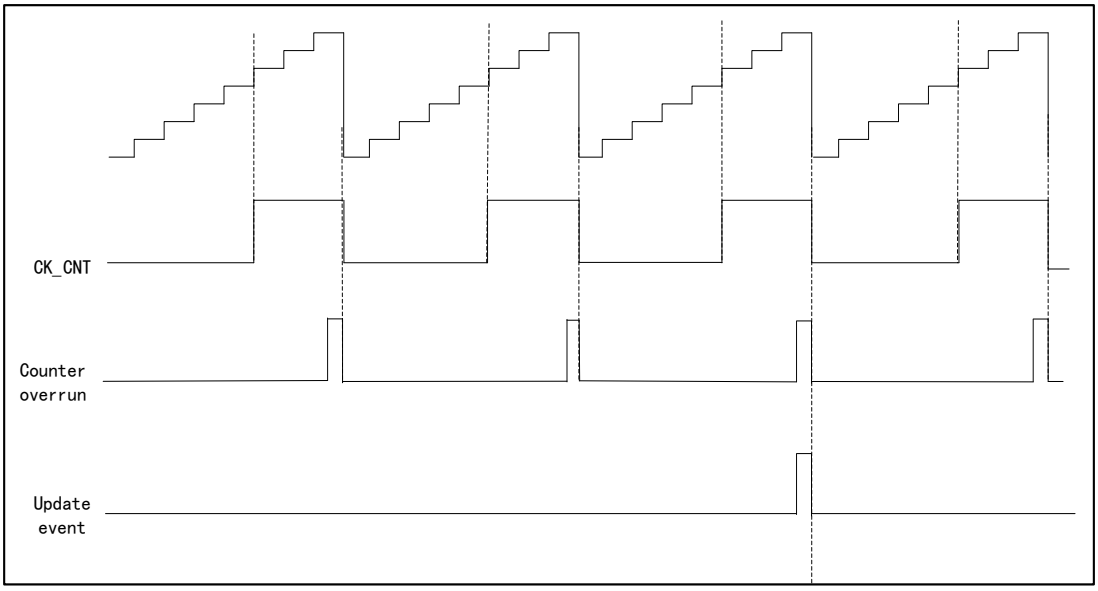
There is no repeat counter REPCNT in the basic/general-purpose timer, which means that when an overrun event or underrun event occurs in the basic/general-purpose timer, an update event will be generated directly; while in the advanced timer, because of the existence of the repeat counter, when an overrun/unerrrun 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 22 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_STS register is 1, an interrupt occurs if CCxIEN in the ATMR_DIEN 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 23 Timing Diagram of PWM1 Count-up Mode

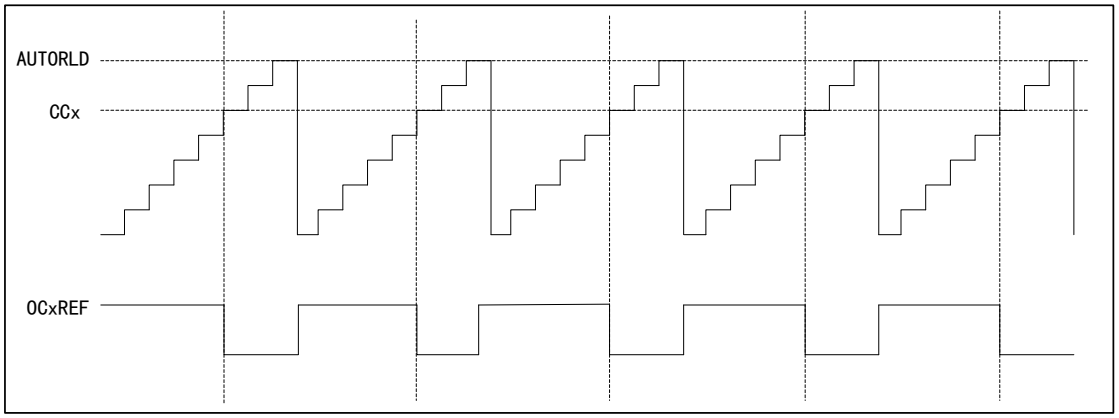


Figure 24 Timing Diagram of PWM1 Count-down Mode

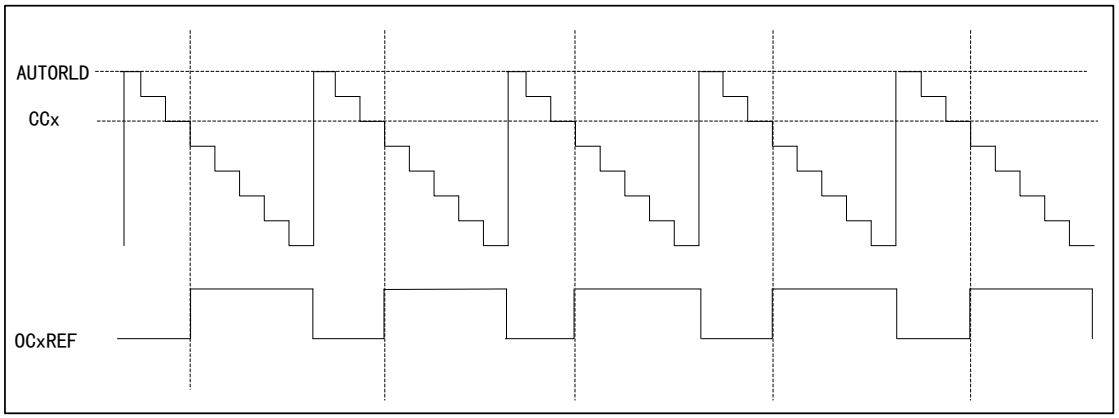
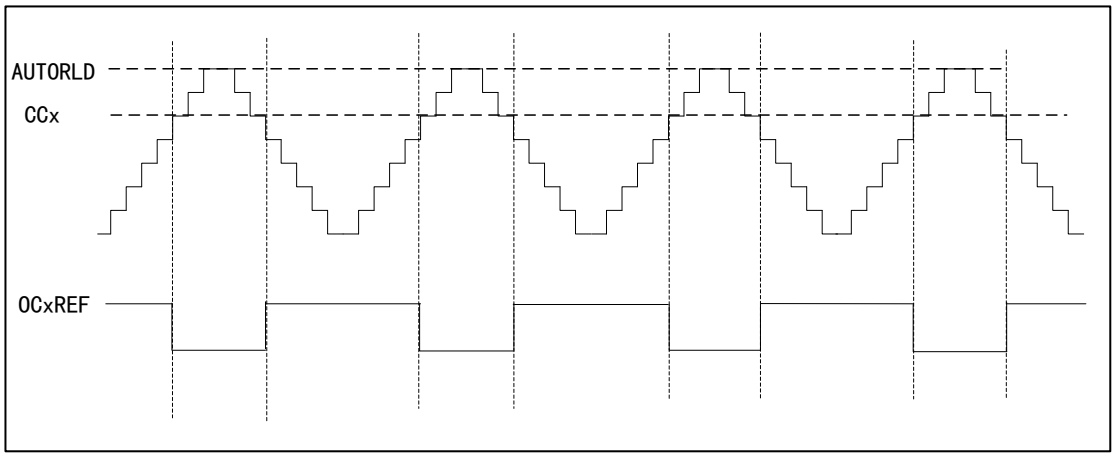


Figure 25 Timing Diagram of PWM1 Central alignment Mode



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 26 Timing Diagram of PWM2 Count-up Mode

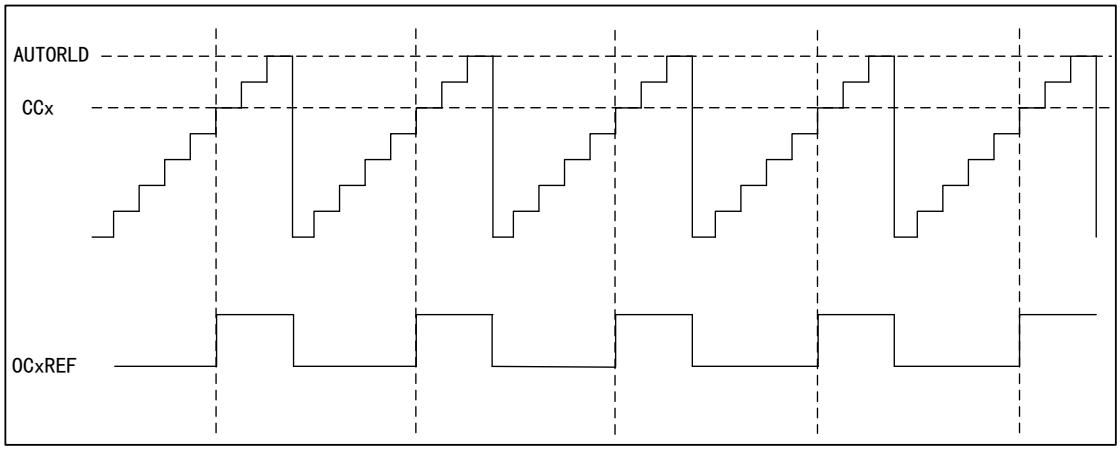


Figure 27 Timing Diagram of PWM2 Count-down Mode

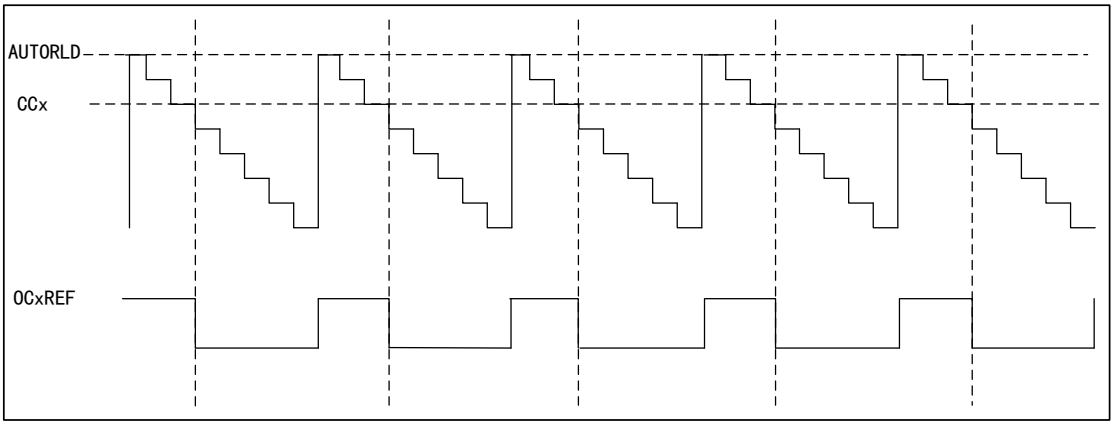
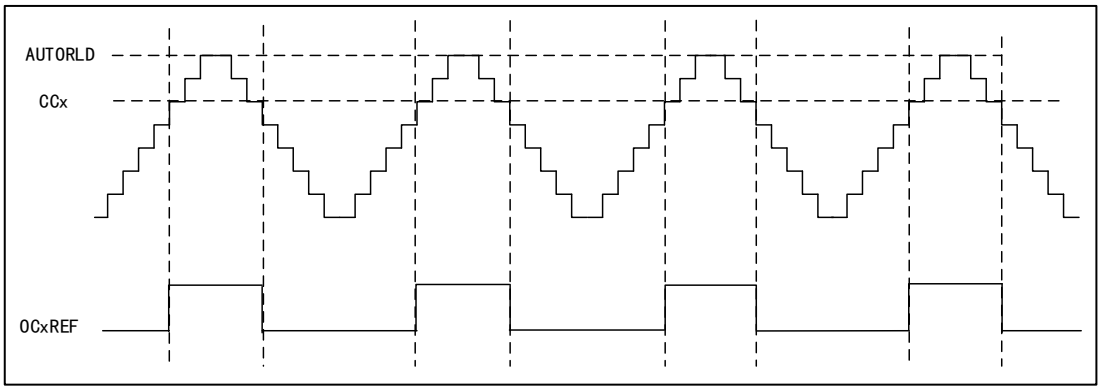


Figure 28 Timing Diagram of PWM2 Central alignment Mode



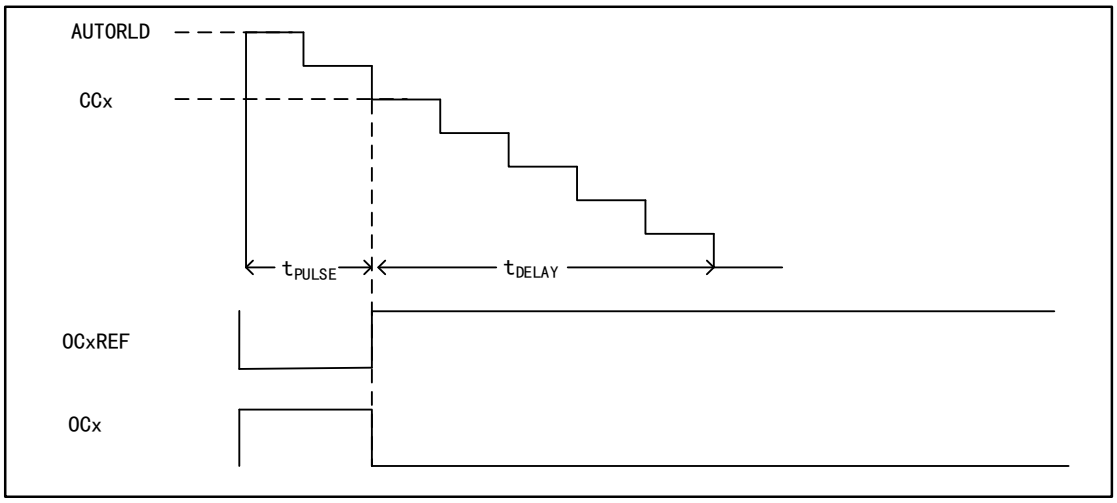
11.4.5 Single-pulse mode

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_CTRL1 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 29 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_CTRL2 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

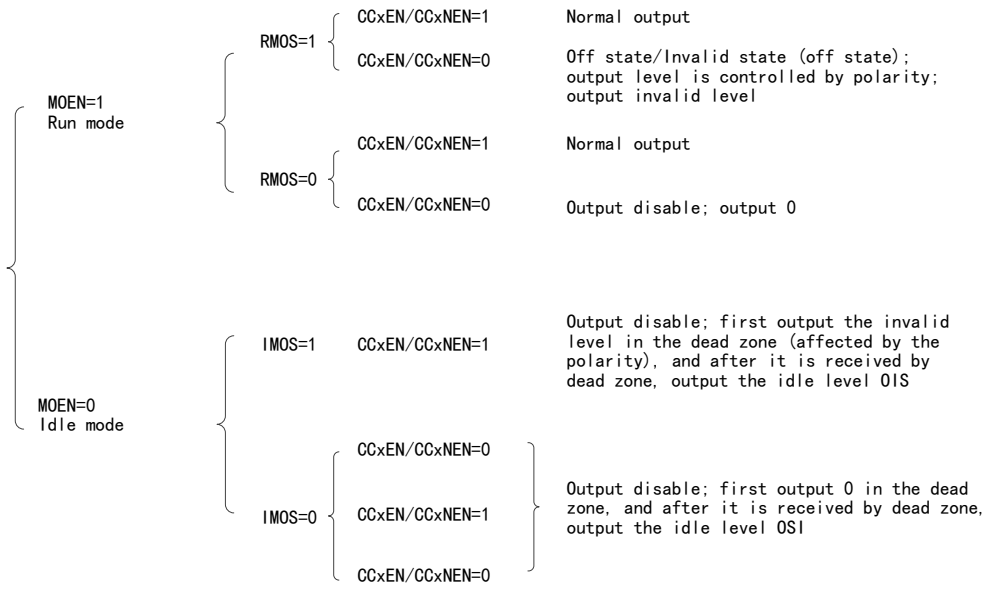
- Application environment of IMOS: In idle mode of corresponding complementary channel and timer (MOEN=0), the timer is not working (CCxEN=0, CCxNEN=0) or is working (CCxEN=1, CCxNEN=1)

(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 30 Register Structural Relationship Affecting Output Waveform



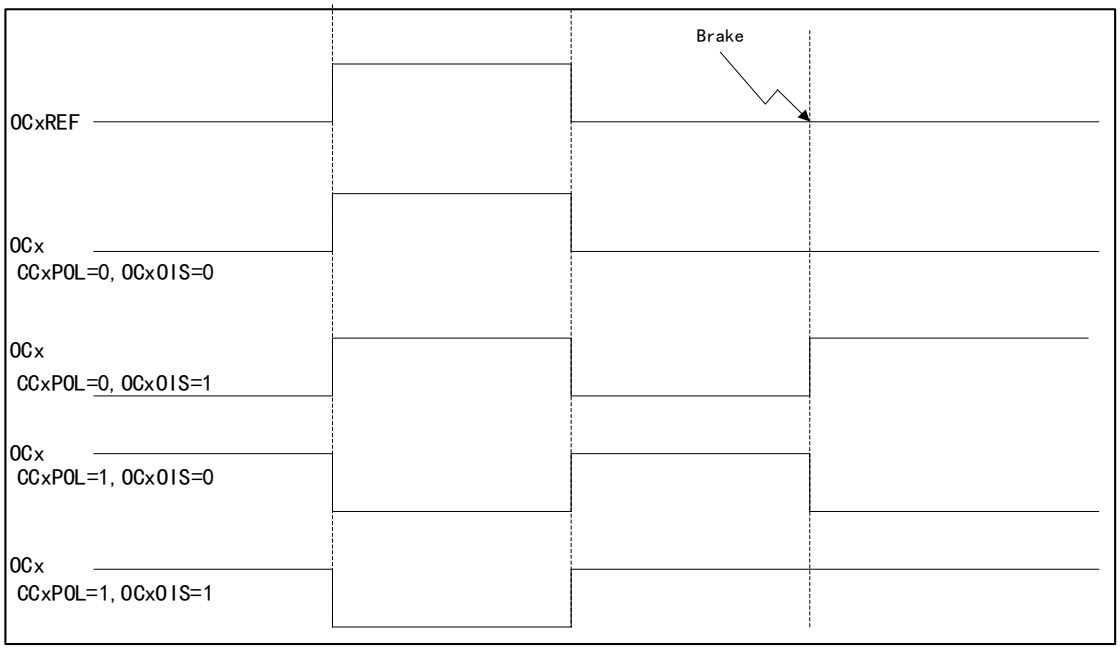
11.4.7 Braking function

The signal source of braking is clock fault event and external input interface.

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.

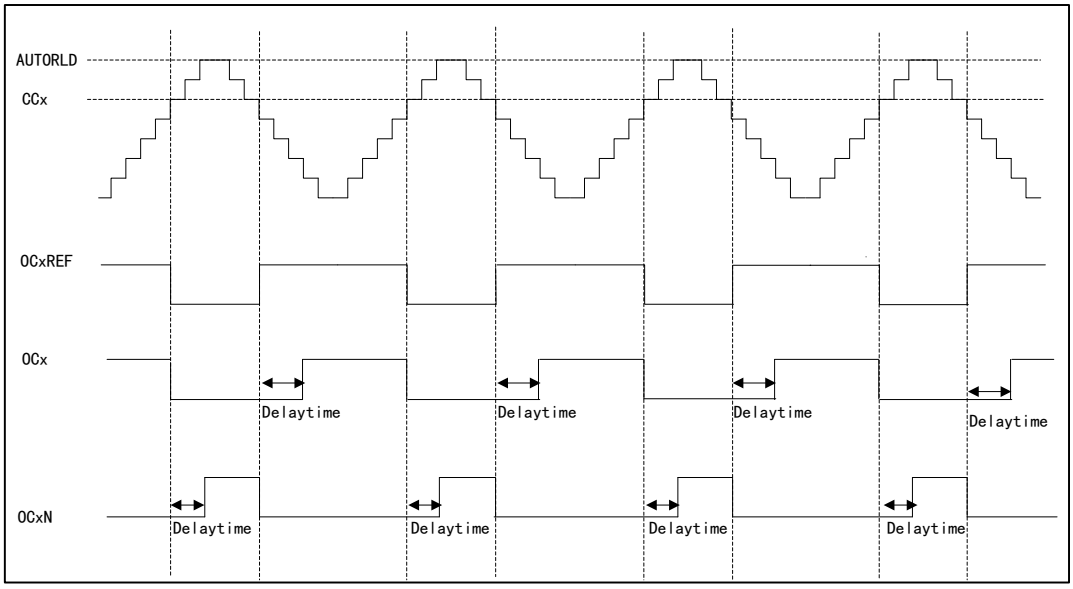
Figure 31 Braking Event Timing Diagram



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 DTS bit of the ATMR_BDT register can control the duration of the dead time.

Figure 32 Complementary Output with 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

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

11.4.10 Slave mode

ATMR timer can synchronize external trigger

- Reset mode
- Gated mode
- Trigger mode

SMFSEL bit in ATMR_SMCTRL 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.

11.4.11 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

11.4.12 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

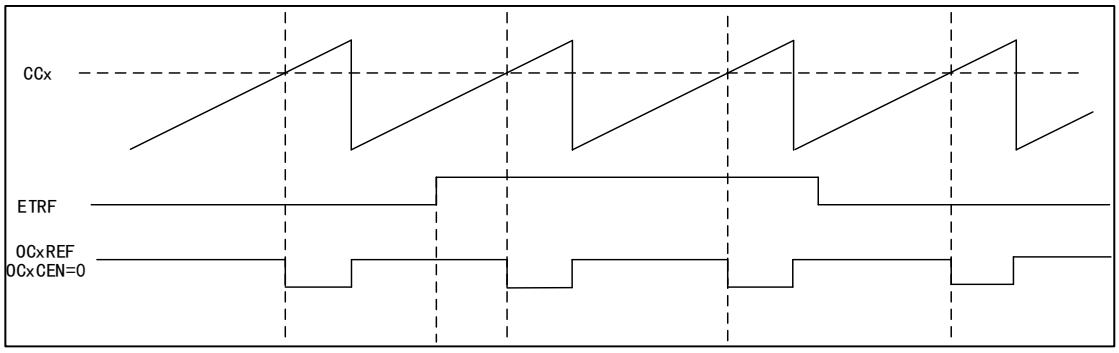
11.4.13 Clear OCxREF signal when an external event occurs

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 ATMR_CCMx is set to 1, and OCxREF signal will remain low until the next update event occurs.

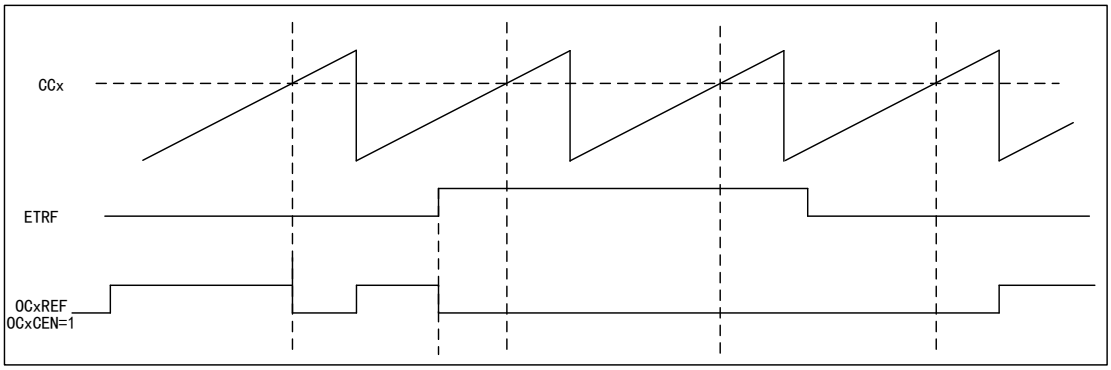
Set ATMR to PWM mode, disable the external trigger prescaler, and disable the external trigger mode 2; when ETRF input is high, set OCxCEN=0, and the output OCxREF signal is shown in the figure below.

Figure 33 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 34 OCxREF Timing Diagram



11.4.14 Timer internal trigger connection

Table 33 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 34 ATIMER Register Address Mapping

Register name	Description	Offset address
ATMR_CTRL1	Control register 1	0x00
ATMR_CTRL2	Control register 2	0x04
ATMR_SMCTRL	Slave mode control register	0x08

Register name	Description	Offset address
ATMR_DIEN	DMA/Interrupt enable register	0x0C
ATMR_STS	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
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_OUTCTRL1	Output control register 1	0x48
ATMR_OUTCTRL2	Output control register 2	0x4C
ATMR_CH4CFG	Channel 4 configuration register	0x50
ATMR_TRGOCR	TRGO control register	0x54
ATMR_BREAK	Break filter register	0x58
ATMR_CC4	Channel 4 compare register	0x5C
ATMR_OCxAEN	Lower comparison register control register	0x60
ATMR_CC0C	Channel 0 compare register	0x64
ATMR_CC1C	Channel 1 compare register	0x68
ATMR_CC2C	Channel 2 compare register	0x6C

11.6 Register functional description

11.6.1 Control register 1 (ATMR_CTRL1)

Offset address: 0x00

Reset value: 0x0000

Field	Name	R/W	Description
15:10	Reserved		
9:8	CLKDIV	R/W	Clock Division

Field	Name	R/W	Description
			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	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	UD	R/W	Update Disable

Field	Name	R/W	Description
			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, 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_CTRL2)

Offset address: 0x04

Reset value: 0x0000

Field	Name	R/W	Description
15	Reserved		
14	OC3OIS	R/W	Configure OC3 output idle state. Refer to OC0OIS bit
13	OC2NOIS	R/W	Configure OC2N output idle state. Refer to OC0NOIS bit
12	OC2OIS	R/W	Configure OC2 output idle state. Refer to OC0OIS bit
11	OC1NOIS	R/W	Configure OC1N output idle state. Refer to OC0NOIS bit
10	OC1OIS	R/W	Configure OC1 output idle state. Refer to OC0OIS bit
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 (CCxIFLG=1), a pulse signal is output for TRGO0 0100: Compare mode 1; OC0REF is used to trigger TRGO0 0101: Compare mode 2; OC1REF is used to trigger TRGO0 0110: Compare mode 3; OC2REF is used to trigger TRGO0 0111: Compare mode 4; OC3REF is used to trigger TRGO0 1000: TRGO0 is not generated (TRGO0 will be generated only according to MMSZE and MMSPE) 1010: OC3REF rising and falling edges generate TRGO0 1011: OC4REF rising and falling edges generate TRGO0 1100: OC3REF rising edge and OC4REF rising edge generate TRGO0 1101: OC3REF falling edge and OC4REF falling edge generate TRGO0 1110: OC3REF rising edge and OC4REF falling edge generate TRGO0 1111: OC3REF falling edge and OC4REF 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_SMCTRL)

Offset address: 0x08

Reset value: 0x0000

Field	Name	R/W	Description
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. 0: Internal trigger ITR0

Field	Name	R/W	Description
			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 CTRL1_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_DIEN)

Offset address: 0x0C

Reset value: 0x0000

Field	Name	R/W	Description
15:8	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_STS)

Offset address: 0x10

Reset value: 0x0000

Field	Name	R/W	Description
15	Reserved		
14	CC4IFLG	RC_W0	Compare Channel 4 Interrupt Flag Refer to STS_CC0IFLG
13:8	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 STS_CC0IFLG
3	CC2IFLG	RC_W0	Compare Channel 2 Interrupt Flag Refer to STS_CC0IFLG
2	CC1IFLG	RC_W0	Compare Channel 1 Interrupt Flag Refer to STS_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
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) UD=0 on ATMR_CTRL1 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 UD=0 on ATMR_CTRL1 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 UD=0 on ATMR_CTRL1 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

Field	Name	R/W	Description
15:8	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 corresponding interrupt and DMA request will be generated; at this time, if CC0IFLG=1, it is required to configure CC0RCFLG=1.

Field	Name	R/W	Description
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

Field	Name	R/W	Description
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 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.

Field	Name	R/W	Description
			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

Field	Name	R/W	Description
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

Field	Name	R/W	Description
15:14	Reserved		
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 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

Field	Name	R/W	Description
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

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

11.6.11 Prescaler register (ATMR_PSC)

Offset address: 0x28

Reset value: 0x0000

Field	Name	R/W	Description
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: 0xFFFF

Field	Name	R/W	Description
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

Field	Name	R/W	Description
15:8	Reserved		

Field	Name	R/W	Description
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

Field	Name	R/W	Description
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

Field	Name	R/W	Description
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

Field	Name	R/W	Description
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

Field	Name	R/W	Description
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

Note: According to the lock setting, AOEN, BRKPOL, BRKEN, IMOS, RMOS and DTS[7:0] 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
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.
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_CTRL2 register. 10: Lock write protection level 2

Field	Name	R/W	Description
			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	DTS	R/W	Dead Time Setup DT is the dead zone duration, and the relationship between DT and register DTS is as follows: DTS[7:5]=0xx=>DT=DTS[7:0]×T_{dts}, T_{dts}=T_{DTS}; DTS[7:5]=10x=>DT=(64+DTS[5:0])×T_{dts}, T_{dts}=2×T_{DTS}; DTS[7:5]=110=>DT=(32+DTS[4:0])×T_{dts}, T_{dts}=8×T_{DTS}; DTS[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 1 μs, the dead time can be set from 32 μs to 63 μs; If the step time is 2 μs, the dead time can be set from 64 μs to 126 μs. 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_OUTPUTTCTRL1)

Offset address: 0x48

Reset value: 0x0000

Field	Name	R/W	Description
31:9	Reserved		
8	OUTPUTCTRL_BUF	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	CH4_FORCE_EN	R/W	Channel 4 Outputs the Control Enable Register 0: Channel 4 outputs PWM waveform 1: The output of channel 4 is controlled by the corresponding bit of output control register 2
6	CH3_FORCE_EN	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	CH2N_FORCE_EN	R/W	Complementary Channel Output Control Enable Register for Channel 2 0: Complementary channel of channel 2 outputs PWM waveform

Field	Name	R/W	Description
			1: The output of complementary channel of channel 2 is controlled by the corresponding bit of output control register 2
4	CH2_FORCE_EN	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	CH1N_FORCE_EN	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	CH1_FORCE_EN	R/W	Channel 1 Outputs the Control Enable Register 0: Channel 1 outputs PWM waveform 1: The output of channel 1 is controlled by the corresponding bit of output control register 2
1	CH0N_FORCE_EN	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	CH0_FORCE_EN	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_OUTPUTCTRL2)

Offset address: 0x4C

Reset value: 0x0000

Field	Name	R/W	Description
31:8	Reserved		
7	CH4_FORCE_VALUE	R/W	Channel 4 Output Level Register 0: Channel 4 outputs low level 1: Channel 4 outputs high level
6	CH3_FORCE_VALUE	R/W	Channel 3 Output Level Register 0: Channel 3 outputs low level 1: Channel 3 outputs high level
5	CH2N_FORCE_VALUE	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

Field	Name	R/W	Description
4	CH2_FORCE_VALUE	R/W	Channel 2 Output Level Register 0: Channel 2 outputs low level 1: Channel 2 outputs high level
3	CH1N_FORCE_VALUE	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	CH1_FORCE_VALUE	R/W	Channel 1 Output Level Register 0: Channel 1 outputs low level 1: Channel 1 outputs high level
1	CH0N_FORCE_VALUE	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	CH0_FORCE_VALUE	R/W	Channel 0 Output Level Register 0: Channel 0 outputs low level 1: Channel 0 outputs high level

Note: ATMR_OUTPUTCTRL2 must be configured first before ATMR_OUTPUTCTRL1. If it is necessary to change the output control buffer function and modify the values of other control bits in ATMR_OUTPUTCTRLx: First, write the OUTPUTCTRL_BUF of ATMR_OUTPUTCTRL1, and then write the ATMR_OUTPUTCTRL2 register and the ATMR_OUTPUTCTRL1 register in sequence.

11.6.21 Channel 4 Configuration Register (ATMR_CH4CFG)

Offset address: 0x50

Reset value: 0x0000

Field	Name	R/W	Description
31:18	Reserved		
17	CC4G	R/W	Compare 4 generation This bit is set to '1' by software to generate a comparison event, and automatically cleared to '0' by hardware. 0: No action 1: Generate a comparison event on channel CC4 If channel CC4 is configured as output: Set CC4IFLG=1, and if the corresponding interrupt is enabled, the corresponding interrupt is generated.
16	OC4PE	R/W	Output compare 4 preload enable 0: Disable the preload function of the ATMR_CC4 register. The ATMR_CC4 register can be written at any time, and the newly written value takes effect immediately. 1: Enable the preload function of the ATMR_CC4 register. Read and write operations only operate on the preload register. The preload value of ATMR_CC4 is transferred to the current register when an update event occurs.

Field	Name	R/W	Description
			Note 1: Once the LOCK level is set to 3 (LOCKCFG bit in the ATMR_BDT register) and CC0SEL='00' (the channel is configured as output), this bit cannot be modified. Note 2: Only in single-pulse mode (SPMEN='1' in the ATMR_CTRL1 register), the PWM mode can be used without confirming the preload register; otherwise, its action is uncertain.
15	Reserved		
13	CC4P	R/W	Compare 4 output polarity When CC4 channel is configured as output: 0: OC4 is active high 1: OC4 is active low
12:10	OC4M	R/W	Output Compare 4 mode Defines the action of the output reference signal OC4REF, and OC4REF determines the value of OC4. OC4REF is active high, and the active level of OC4 depends on the CC4P bit. 000: Freeze The comparison between the output compare register ATMR_CC4 and the counter ATMR_CNT has no effect on OC4REF; 001: Set channel 4 to active level when matching. When the value of the counter ATMR_CNT is equal to that of the compare register 4 (ATMR_CC4), force OC4REF to high. 010: Set channel 4 to inactive level when matching. When the value of the counter ATMR_CNT is equal to that of the capture/compare register 4 (ATMR_CC4), force OC4REF to low. 011: Reverse. When ATMR_CC4=ATMR_CNT, reverse the level of OC4REF. 100: Force to inactive level. Force OC4REF to be low 101: Force to active level. Force OC4REF to be high 110: PWM mode 1 – In up-counting, channel 4 is at active level once ATMR_CNT<ATMR_CC4, otherwise inactive level; in down-counting, channel 4 is at inactive level (OC4REF=0) once ATMR_CNT>ATMR_CC4, otherwise active level (OC4REF=1). 111: PWM mode 2 – In up-counting, channel 4 is at inactive level once ATMR_CNT<ATMR_CC4, otherwise active level; in down-counting, channel 4 is at active level once ATMR_CNT>ATMR_CC4, otherwise inactive level. Note: Once the LOCK level is set to 3 (LOCKCFG bit in the ATMR_BDT register) and CC4SEL=00 (the channel is configured as output), this bit cannot be modified. In PWM mode 1 or PWM mode 2, the OC4REF level changes only when the comparison result changes or when switching from freeze mode to PWM mode in output compare mode.
9	OC4CE	R/W	Output Compare 4 clear enable 0: OC4REF is unaffected by ETRF input. 1: Once high level of ETRF input is detected, clear OC4REF=0
8	CC4E	R/W	CC4E: Compare 4 output enable When CC4 channel is configured as output: 0: Off – OC4 output is disabled, so the output level of OC4 depends on the values of MOEN, IMOS, RMOS, OCxOIS and OCxNOIS bits.

Field	Name	R/W	Description
			1: On – The OC0 signal is output to the corresponding output pin, and its output level depends on the values of the MOEN, IMOS, RMOS, OCxOIS and OCxNOIS bits.
7:0	Reserved		

11.6.22 TRGO Control Register (ATMR_TRGOCR)

Offset address: 0x54

Reset value: 0x0000

Field	Name	R/W	Description
15:8	Reserved		
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_SMCTRL 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. 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: Compare – The OC4REF signal is used as the trigger output (TRGO2) 1001: Reserved 1010: OC3REF rising edge and falling edge generate TRGO2 1011: OC4REF rising edge and falling edge generate TRGO2 1100: OC3REF rising edge and OC4REF rising edge generate TRGO2 1101: OC3REF falling edge and OC4REF falling edge generate TRGO2 1110: OC3REF rising edge and OC4REF falling edge generate TRGO2 1111: OC3REF falling edge and OC4REF rising edge generate TRGO2
3:0	MMS1	R/W	Master mode 1 selection

Field	Name	R/W	Description
			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_SMCTRL 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) 0111: Compare – The OC3REF signal is used as the trigger output (TRGO1) 1000: Compare – The OC4REF signal is used as the trigger output (TRGO1) 1001: Reserved 1010: OC3REF rising and falling edges generate TRGO1 1011: OC4REF rising and falling edges generate TRGO1 1100: OC3REF rising edge and OC4REF rising edge generate TRGO1 1101: OC3REF falling edge and OC4REF falling edge generate TRGO1 1110: OC3REF rising edge and OC4REF falling edge generate TRGO1 1111: OC3REF falling edge and OC4REF rising edge generate TRGO1

11.6.23 Break Filter Register (ATIMER_BREAK)

Offset address: 0x58

Reset value: 0x0000

Field	Name	R/W	Description
15:8	Reserved		
7:2	BRK_FILT	R/W	Filter coefficient Based on the APB clock. BRK_FILT*APB_CLK
1	BRK_FILT_EN	R/W	Break_in input filtering function enable 0: Disable 1: Enable

Field	Name	R/W	Description
0	ANA_BRK_FILT_EN	R/W	Analog Break_in input filtering function enable 0: Disable 1: Enable

11.6.24 Channel 4 Compare Register (ATMR_CC4)

Offset address: 0x5C

Reset value: 0x0000

Field	Name	R/W	Description
15:0	CC4	R/W	Compare 4 value Refer to ATMR_CC0.

11.6.25 Lower Compare Register Control Register (ATMR_OCXAEN)

Offset address: 0x60

Reset value: 0x0000

Field	Name	R/W	Description
15:3	Reserved		
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.

11.6.26 Channel 0 Lower Compare Register (ATMR_CC0C)

Offset address: 0x64

Reset value: 0x0000

Field	Name	R/W	Description
15:0	CC0C	R/W	Channel 0 Compare Register Complementary Register When the compare channel 0 is configured as PWM output mode, the counter is in the center alignment mode, and the asymmetric PWM output mode is enabled (OC0AEN of ATMR_OCxAEN register is 1), CC0C 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 CC0C 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

Field	Name	R/W	Description
			in the ATMR_CCM1 register), the written value will affect the output compare result when an update event occurs.

11.6.27 Channel 1 Lower Compare Register (ATMR_CC1C)

Offset address: 0x68

Reset value: 0x0000

Field	Name	R/W	Description
15:0	CC1C	R/W	Channel 1 Compare Register Complementary Register Refer to ATMR_CC0C.

11.6.28 Channel 2 Lower Compare Register (ATMR_CC2C)

Offset address: 0x6C

Reset value: 0x0000

Field	Name	R/W	Description
15:0	CC2C	R/W	Channel 2 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

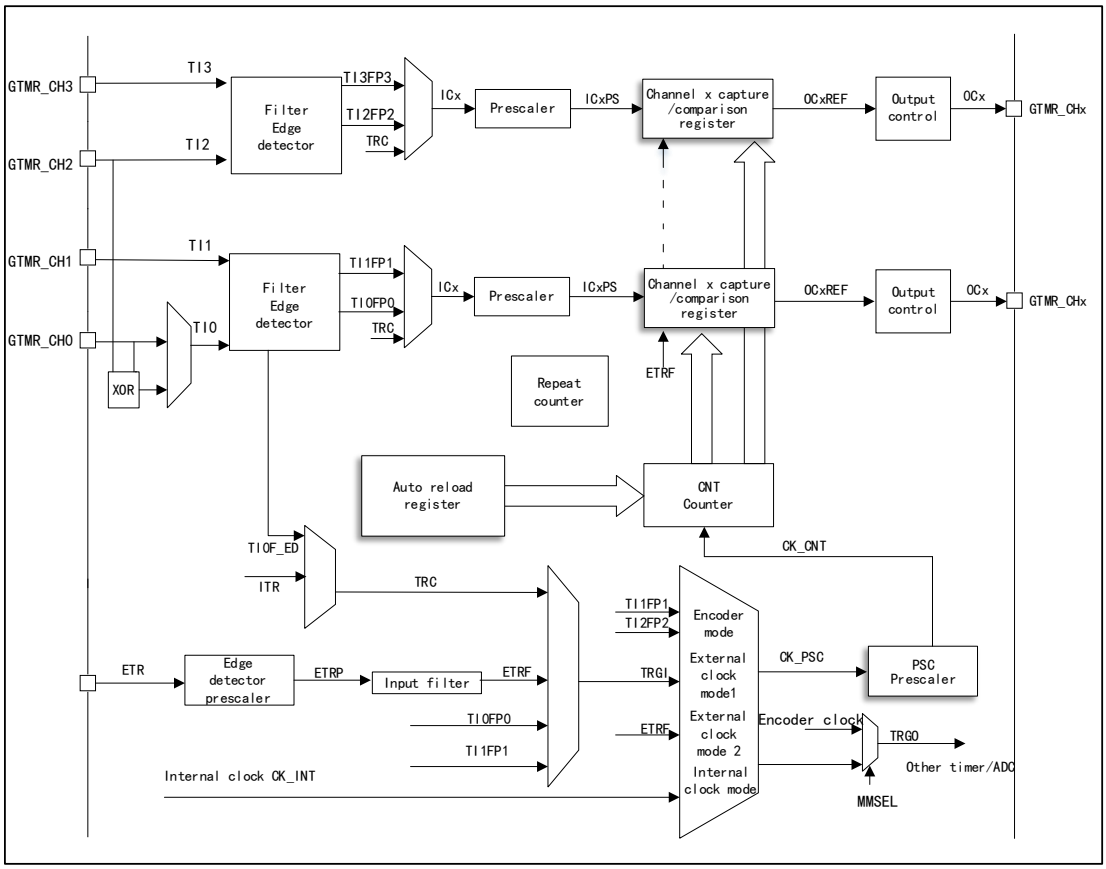
(14) The timer has an independent DMA request mechanism for generation

(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 35 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 2

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

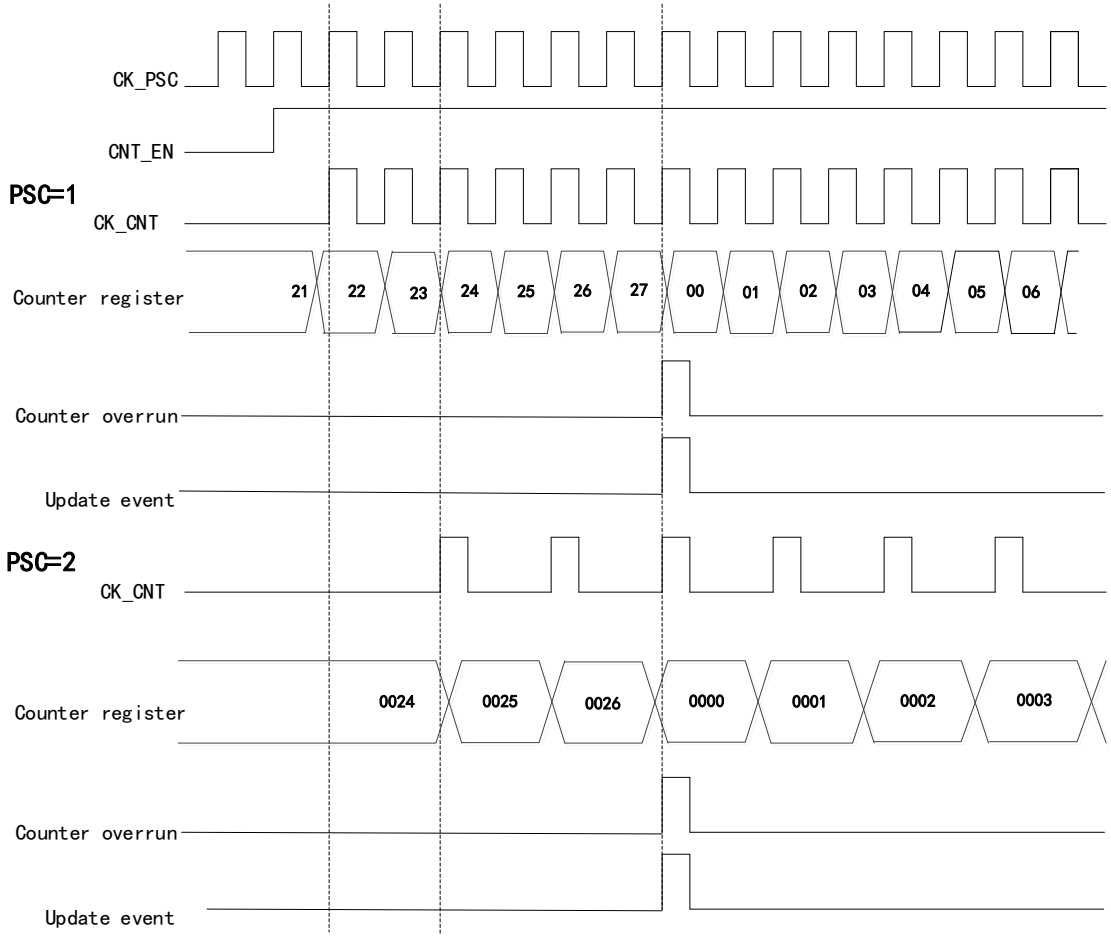
Count-up mode

Set to the count-up mode by configuring CNTDIR bit of control register (GTMR_CTRL1).

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 UD bit of control register GTMR_CTRL1.

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

Figure 36 Timing Diagram of Count-up Mode when Division Factor is 1 or 2



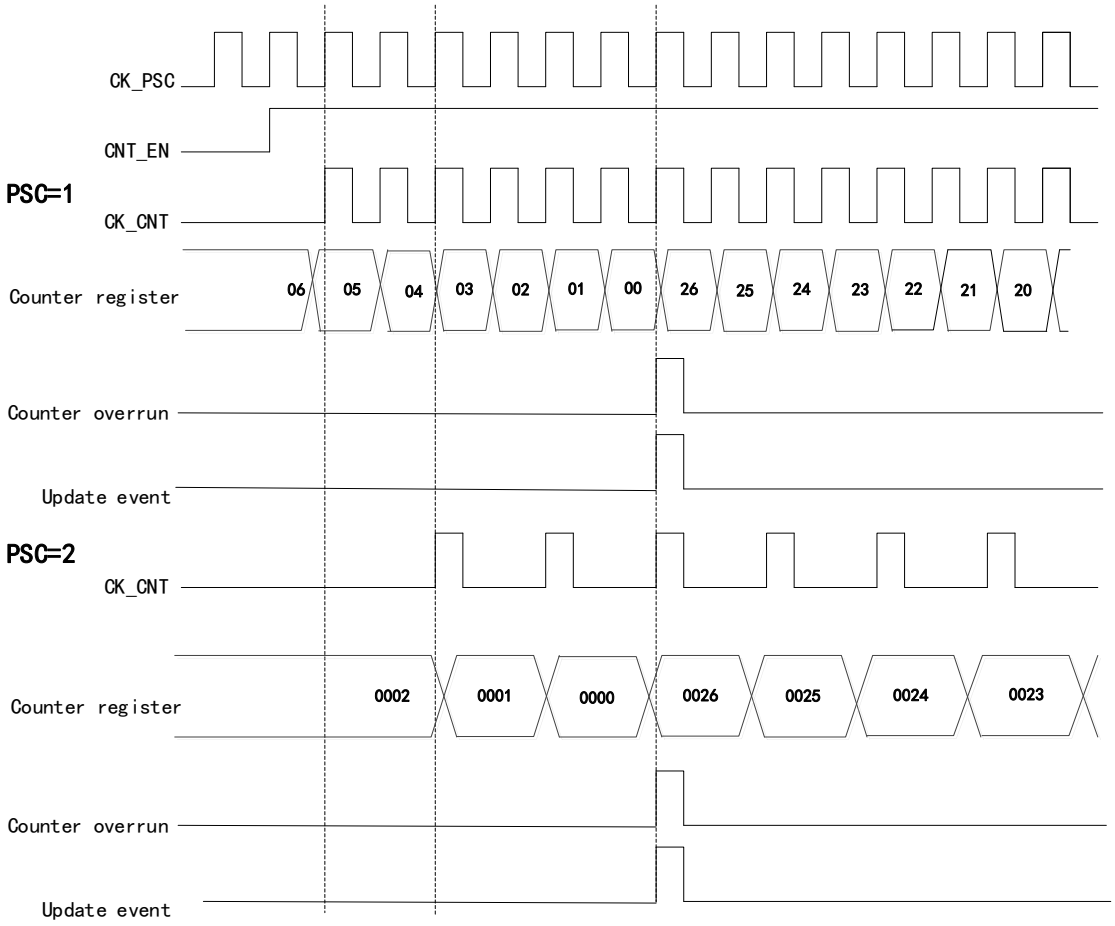
Count-down mode

Set to the count-down mode by configuring CNTDIR bit of control register (GTMR_CTRL1).

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 UD bit of the GTMR_CTRL1 register.

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

Figure 37 Timing Diagram of Count-down Mode when Division Factor is 1 or 2



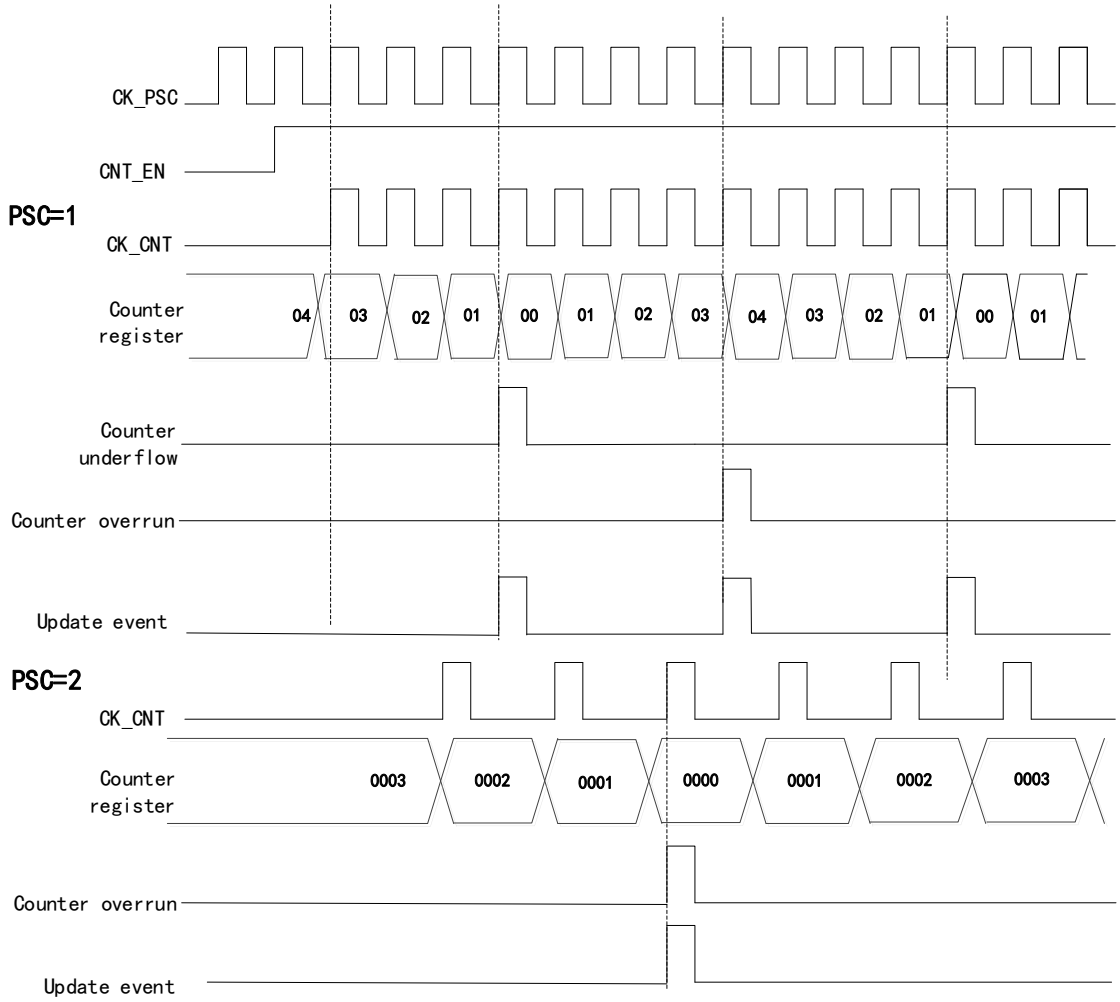
Central alignment mode

Set to the central alignment mode by configuring CAMSEL bit of control register (GTMR_CTRL1).

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 38 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_STS 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_STS register, if CCxIEN=1 in GTMR_DIEN register, an interrupt will be generated; if CCDSEL=1 in GTMR_CTRL2 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 39 Timing Diagram of PWM1 Count-up Mode

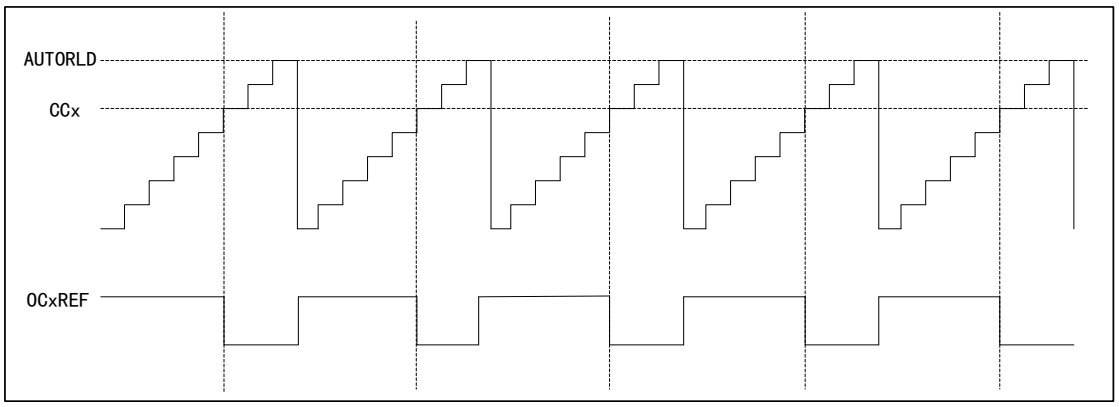


Figure 40 Timing Diagram of PWM1 Count-down Mode

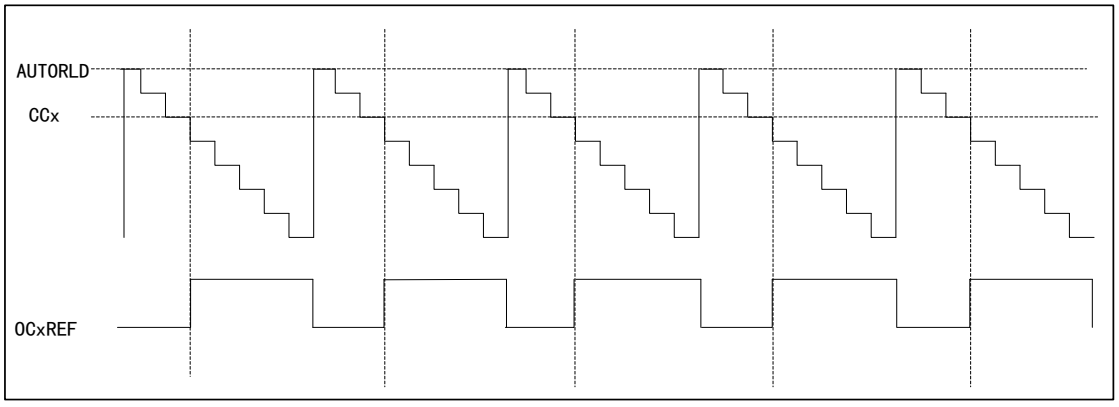
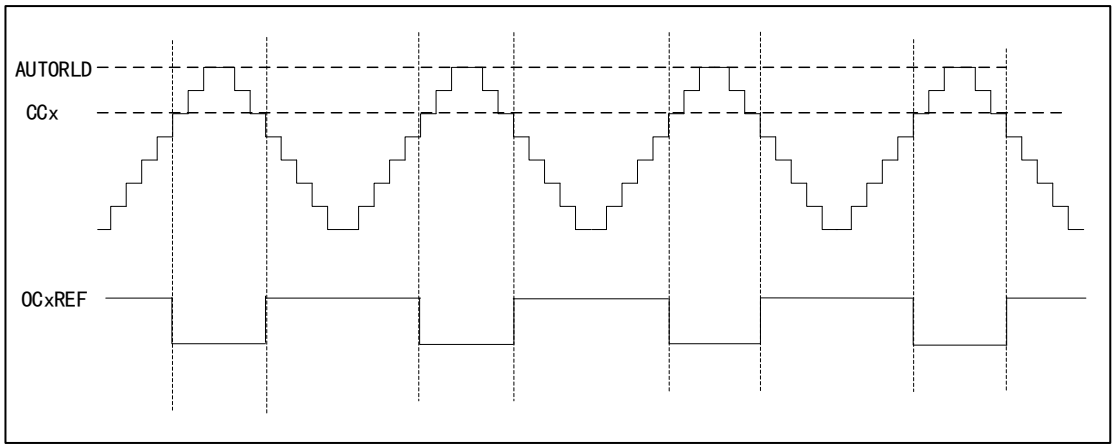


Figure 41 Timing Diagram of PWM1 Central alignment Mode



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 42 Timing Diagram of PWM2 Count-up Mode

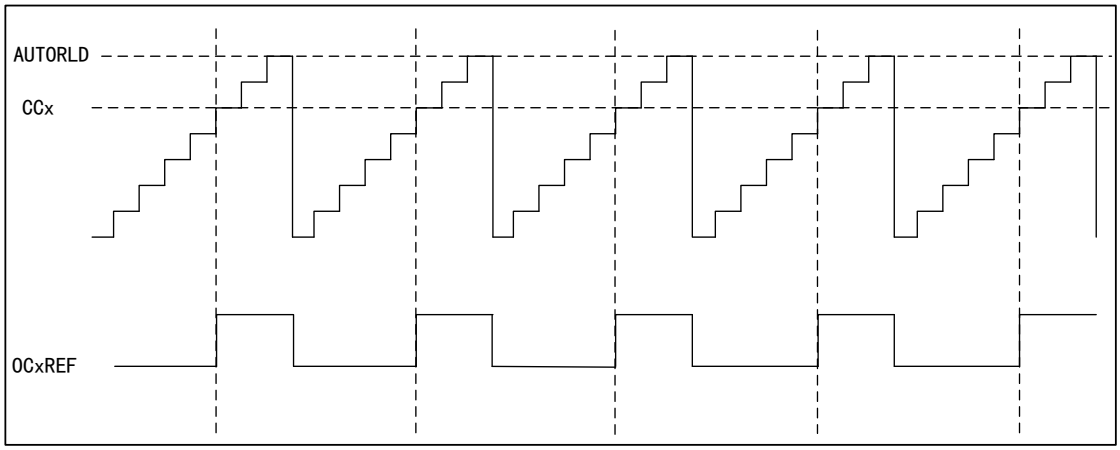


Figure 43 Timing Diagram of PWM2 Count-down Mode

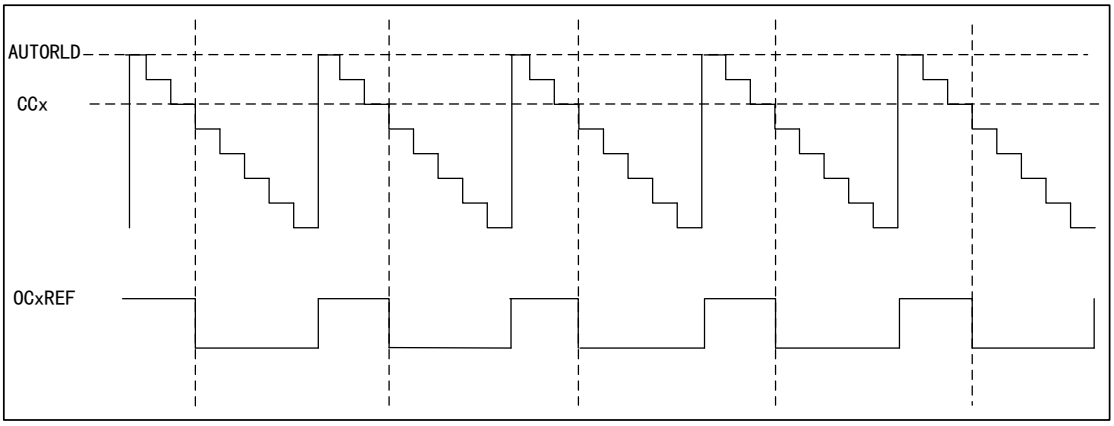
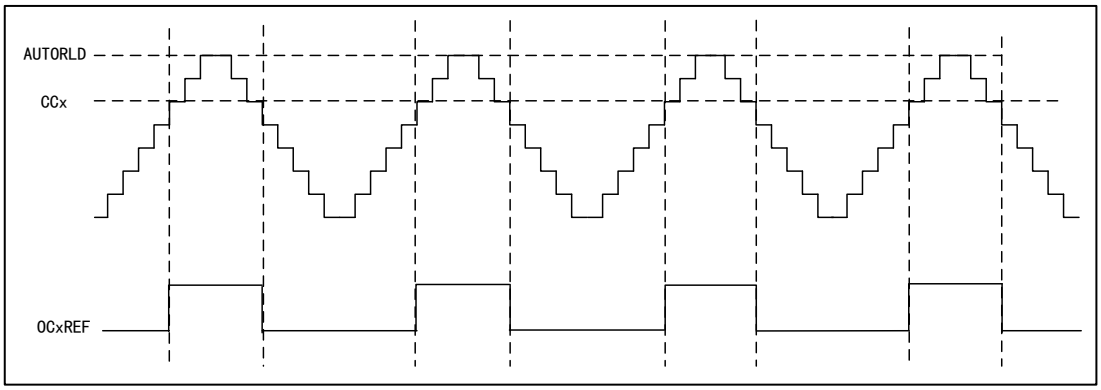


Figure 44 Timing Diagram of PWM2 Central alignment Mode



12.4.6 PWM input mode

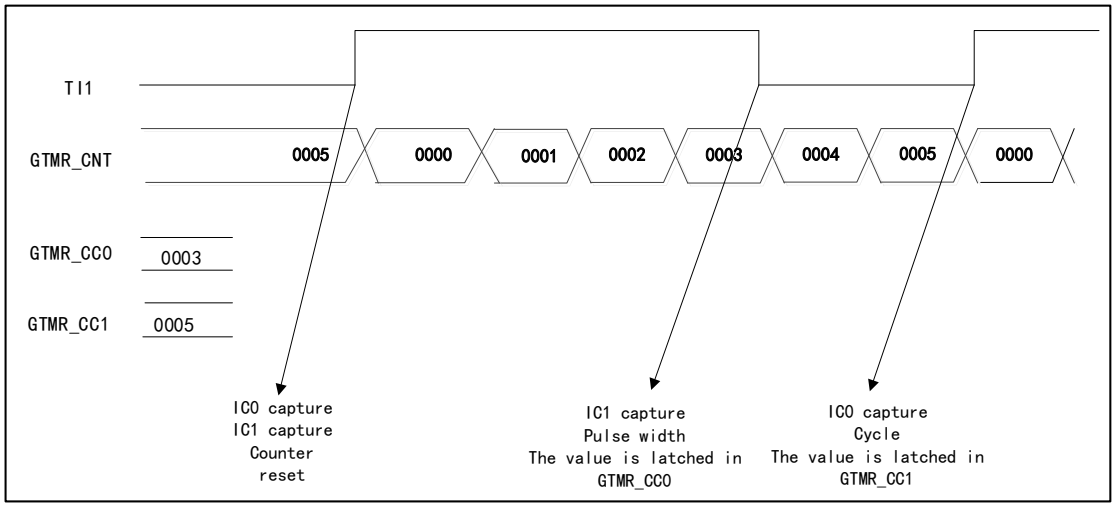
PWM input mode is a particular case of input capture.

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_SMCTRL register).

Figure 45 Timing Diagram in PWM Input Mode



12.4.7 Single-pulse mode

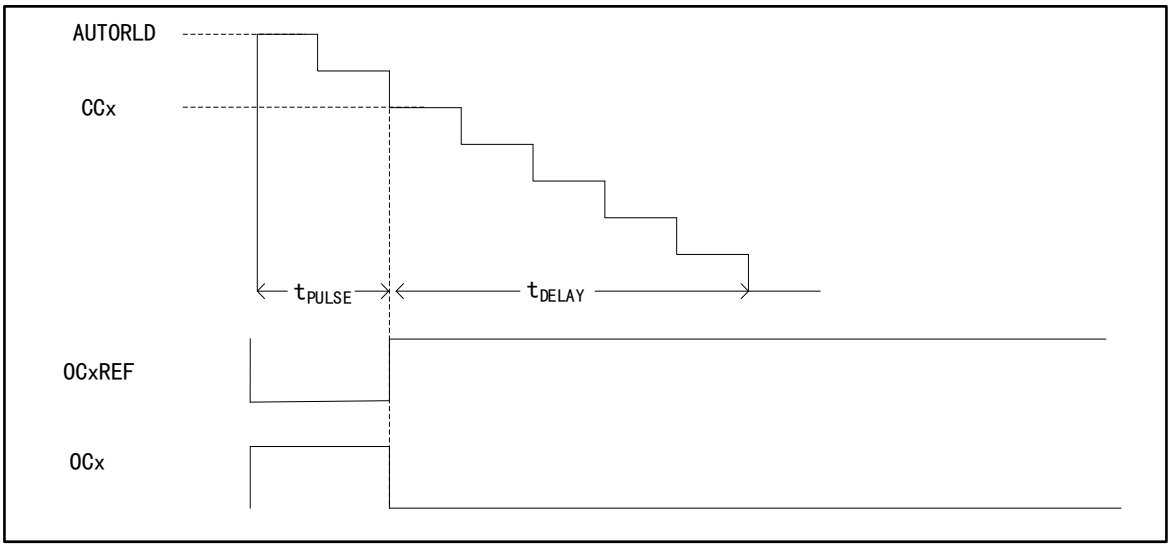
The single-pulse mode is a special case of timer compare output, and is also a

special case of PWM output mode.

Set SPMEN bit of GTMR_CTRL1 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 46 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_SMCTRL 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_CTRL1 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 35 Relationship between Count Direction and Encoder

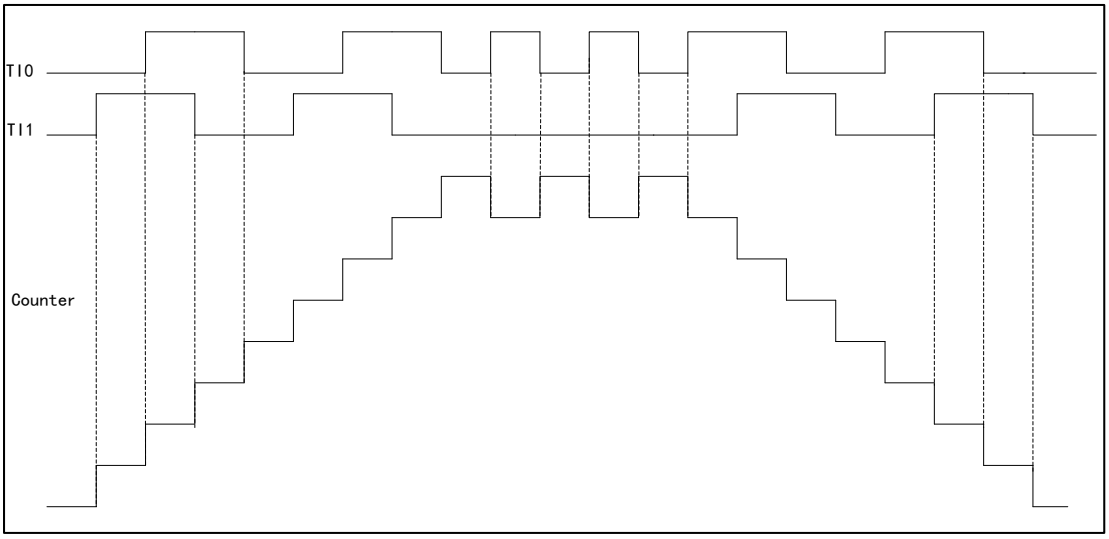
Effective edge		**Count only in TI0**		**Count only in TI1**		**Count in both TI0 and TI1**	
Level of relative signal		High	Low	High	Low	High	Low
TI0FP0	Rising Edge	Count down	Count up	—		Count down	Count up
	Falling Edge	Count up	Count down			Count up	Count down
TI1FP1	Rising Edge	—		Count up	Count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

The external incremental encoder can be directly connected with SoC, 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,

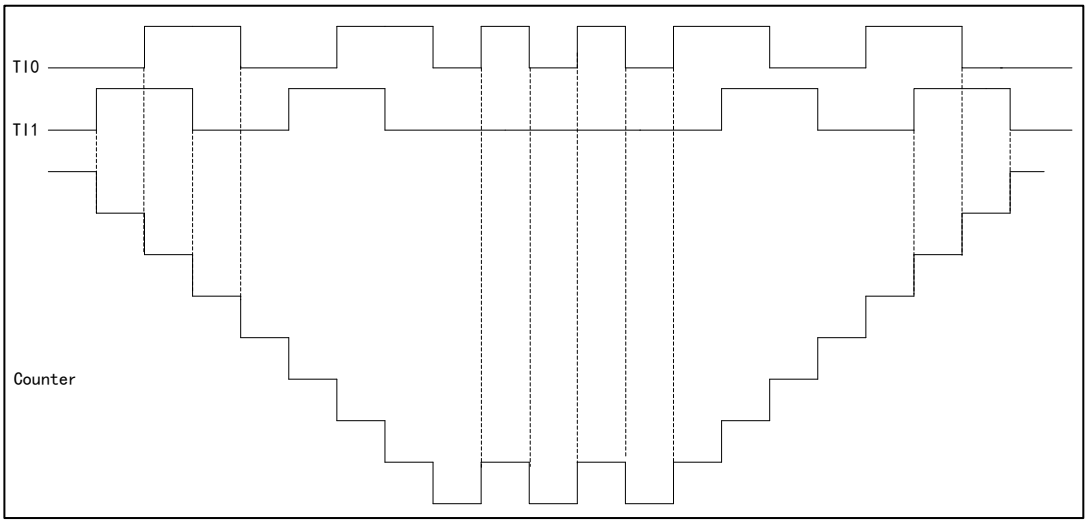
- IC0FP0 is mapped to TI0
- IC1FP1 is mapped to TI1
- Neither IC0FP0 nor IC1FP1 is phase-inverting
- The input signal is valid at the rising edge and falling edge
- Enable the counter

Figure 47 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 48 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_SMCTRL 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

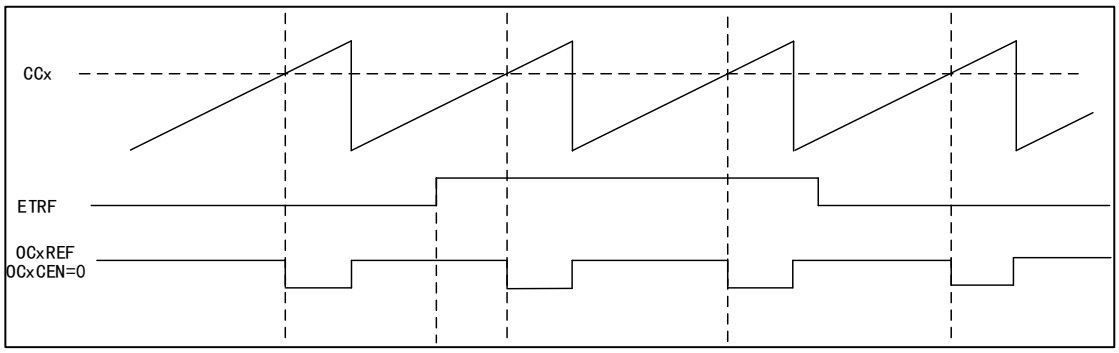
This function is used for output compare and PWM mode.

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

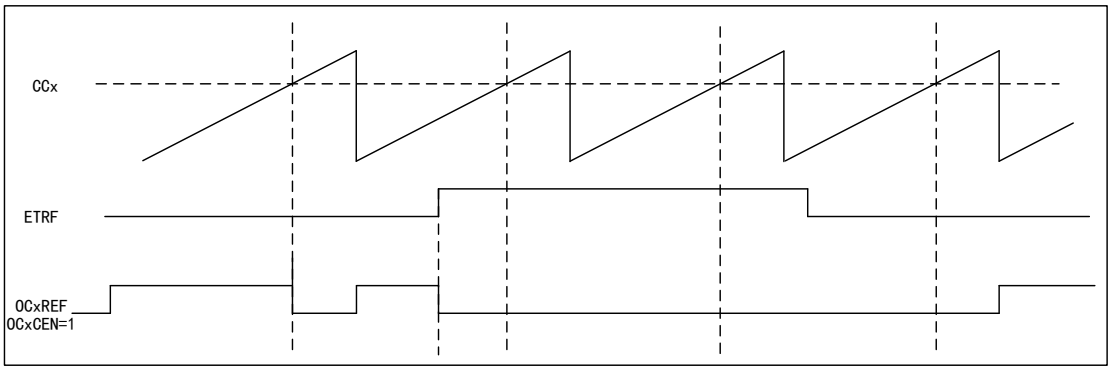
external trigger mode 2; when ETRF input is high, set OCxCEN=0, and the output OCxREF signal is shown in the figure below.

Figure 49 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 50 OCxREF Timing Diagram



12.4.14 Timer internal trigger connection

Table 36 Timer Internal Trigger Connection

Slave timer	ITR0
ATMR	GTMR
GTMR	ATMR

12.5 Register address mapping

Table 37 GTIMER Register Address Mapping

Register name	Description	Offset address
GTMR_CTRL1	Control register 1	0x00
GTMR_CTRL2	Control register 2	0x04
GTMR_SMCTRL	Slave mode control register	0x08

Register name	Description	Offset address
GTMR_DIEN	DMA/Interrupt enable register	0x0C
GTMR_STS	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 capture/compare register	0x3C
GTMR_CC3	Channel 3 capture/compare register	0x40

12.6 Register functional description

12.6.1 Control register 1 (GTMR_CTRL1)

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	UD	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_CTRL2)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		
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 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 (CCxIFLG=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_SMCTRL)

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

Field	Name	R/W	Description
			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 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: Internal trigger ITR0 001: Internal trigger ITR1 010: Internal trigger ITR2 011: Internal trigger ITR3 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 CTRL1_CNTEN=1, the prescaler is directly driven by the internal clock.

Field	Name	R/W	Description
			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. 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_DIEN)

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

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

12.6.5 Status register (GTMR_STS)

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 STS_CC0RCFLG
11	CC2RCFLG	RC_W0	Capture/Compare Channel 2 Repetition Capture Flag Refer to STS_CC0RCFLG
10	CC1RCFLG	RC_W0	Capture/Compare Channel 1 Repetition Capture Flag Refer to STS_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		

Field	Name	R/W	Description
4	CC3IFLG	RC_W0	Captuer/Compare Channel 3 Interrupt Flag Refer to STS_CC0IFLG
3	CC2IFLG	RC_W0	Captuer/Compare Channel 2 Interrupt Flag Refer to STS_CC0IFLG
2	CC1IFLG	RC_W0	Capture/Compare Channel 1 Interrupt Flag Refer to STS_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 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) UD=0 on GTMR_CTRL1 register, and when the value of the repeat counter overruns/underruns, an update event will be generated; (2) URSSEL=0 and UD=0 on GTMR_CTRL1 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 UD=0 on GTMR_CTRL1 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

Field	Name	R/W	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: 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

Field	Name	R/W	Description
			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 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

Field	Name	R/W	Description
			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
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 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

Field	Name	R/W	Description
			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
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

Field	Name	R/W	Description
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 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

Field	Name	R/W	Description
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: 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 Prescaler register (GTMR_PSC)

Offset address: 0x28

Reset value: 0x0000

Field	Name	R/W	Description
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; 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

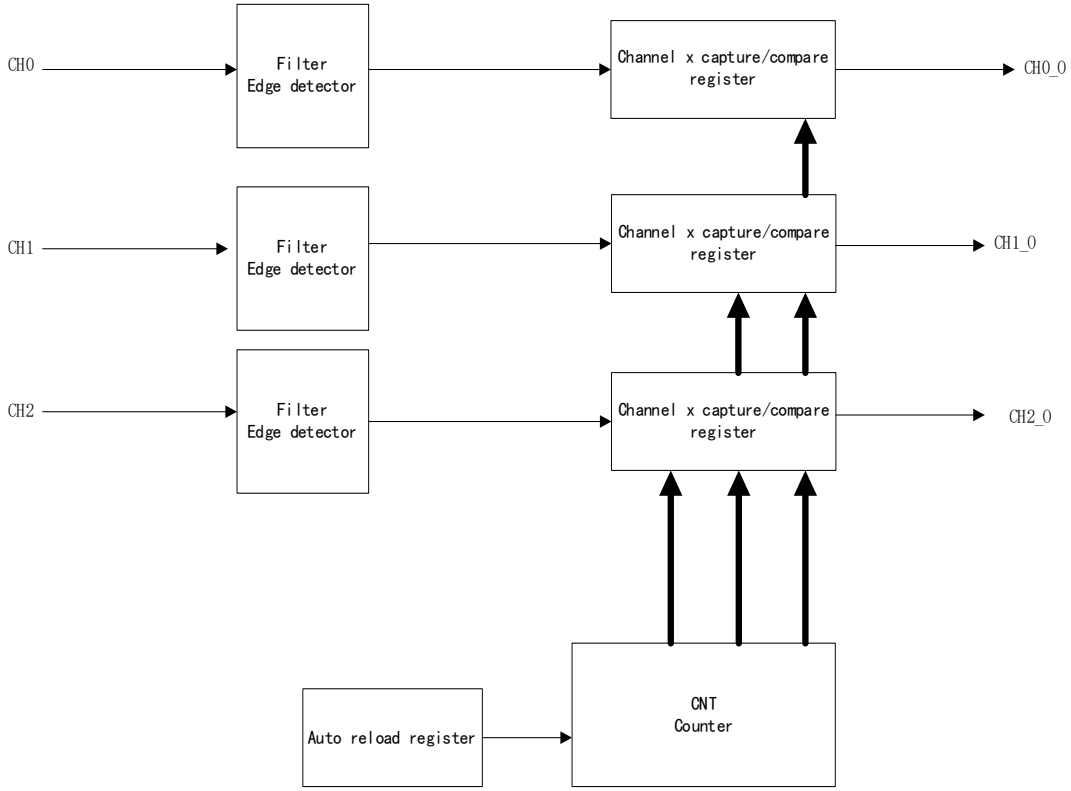
(9) Single-pulse mode

(10) Timing function

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

13.3 Structure block diagram

Figure 51 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_CTRL1 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

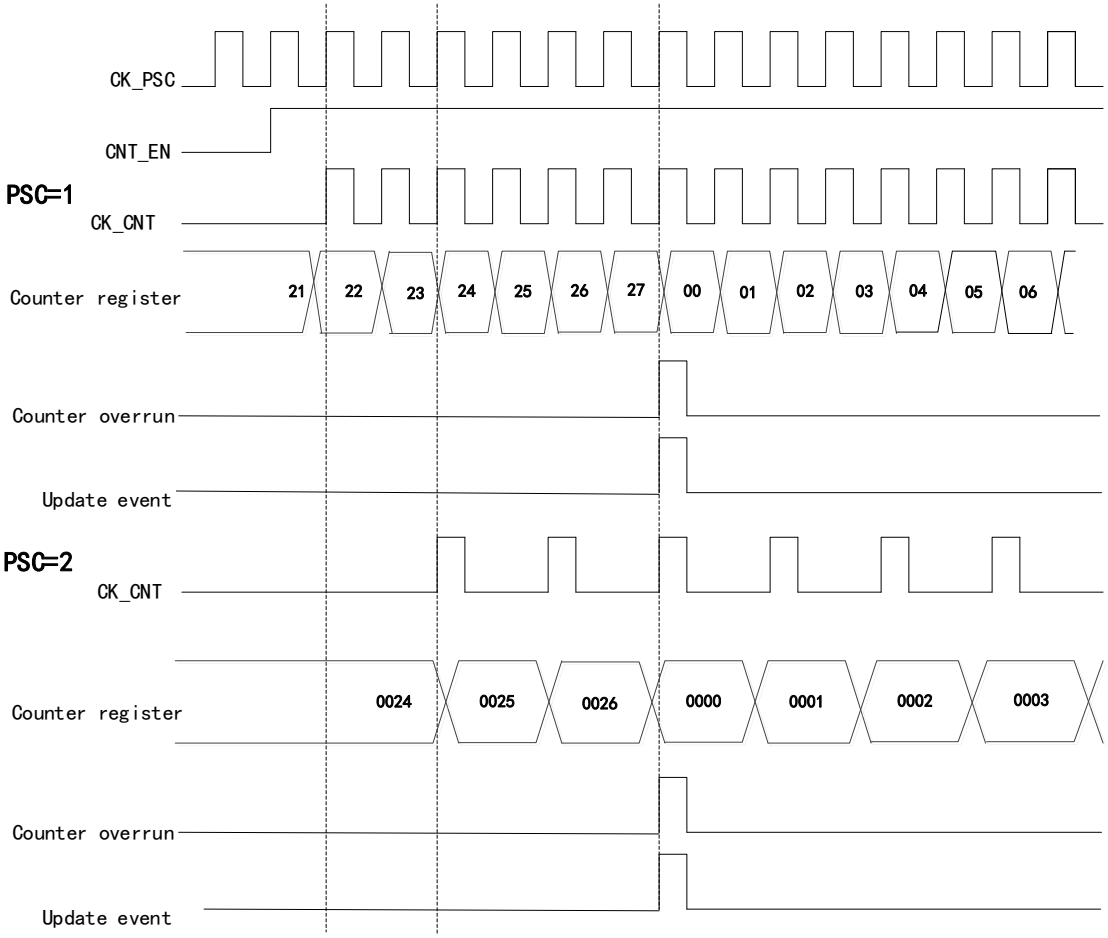
Count-up mode

Set to the count-up mode by configuring CNTDIR bit of control register (BTMR_CTRL1).

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



Count-down mode

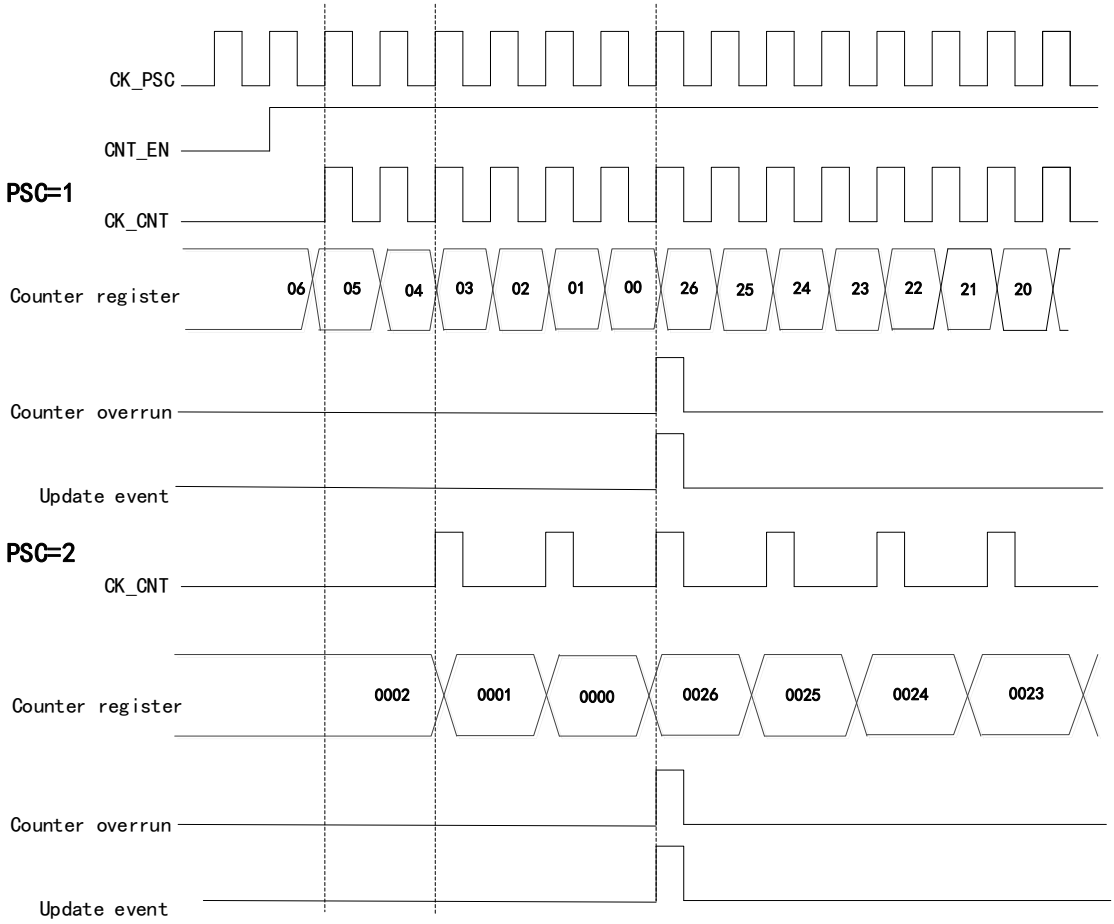
Set to the count-down mode by configuring CNTDIR bit of control register (BTMR_CTRL1).

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



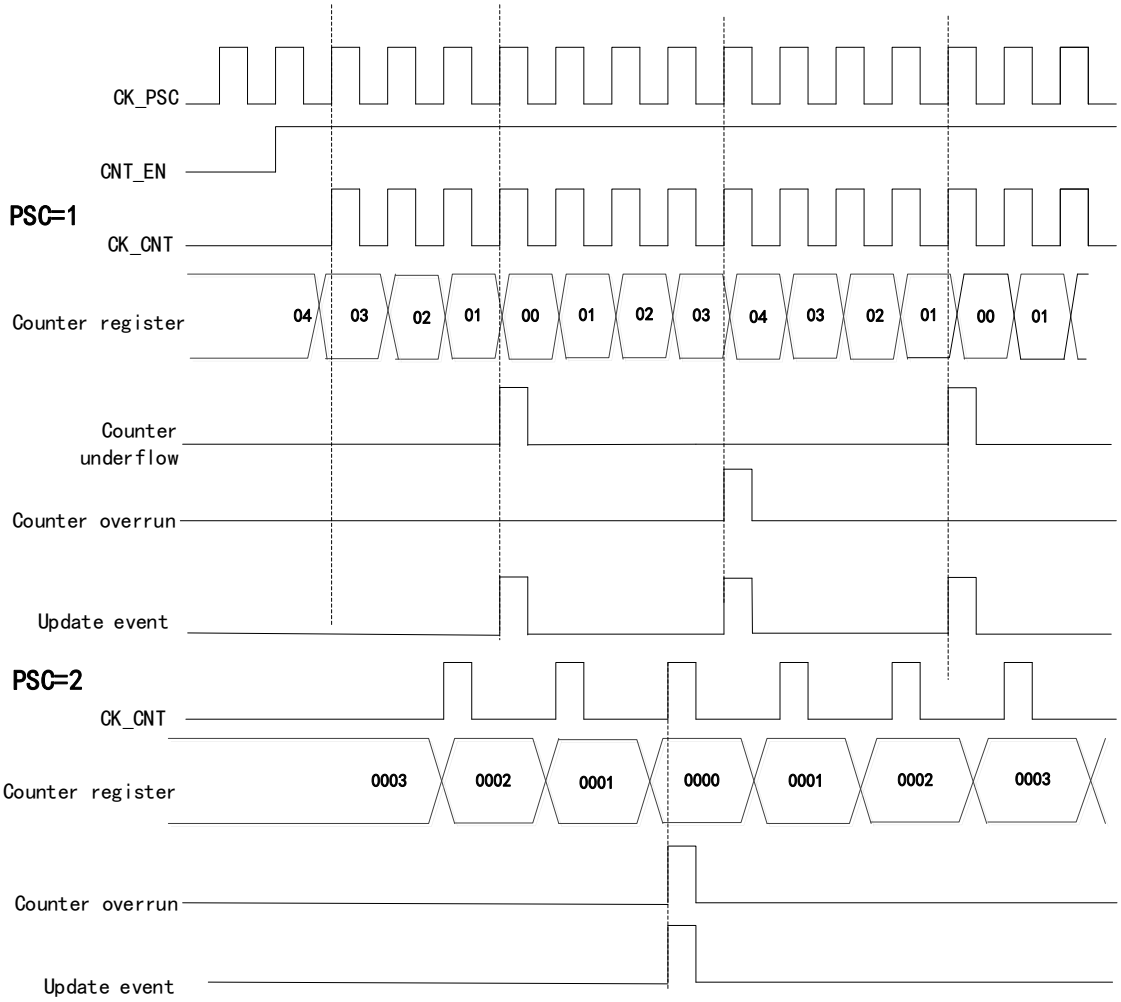
Central alignment mode

Set to the central alignment mode by configuring CAMSEL bit of control register (BTMR_CTRL1).

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 54 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 three 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 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_STS 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_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 BTMR_CCMx 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 55 Timing Diagram of PWM1 Count-up Mode

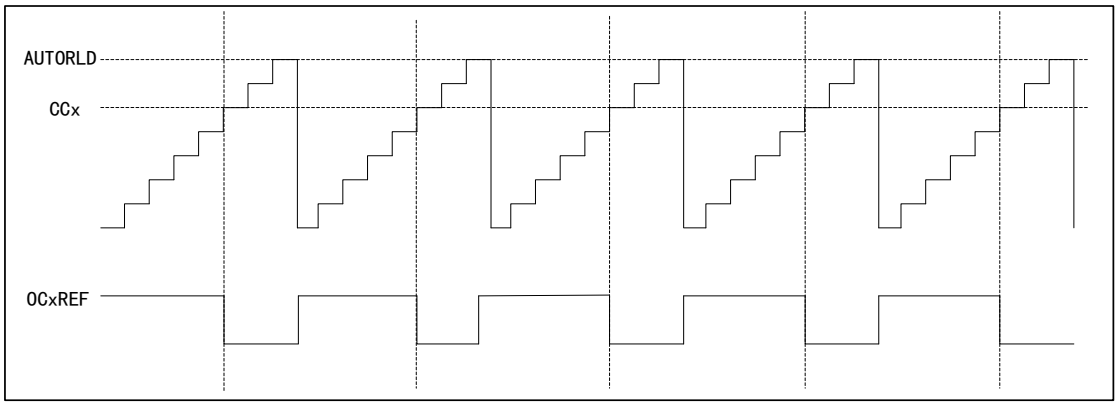


Figure 56 Timing Diagram of PWM1 Count-down Mode

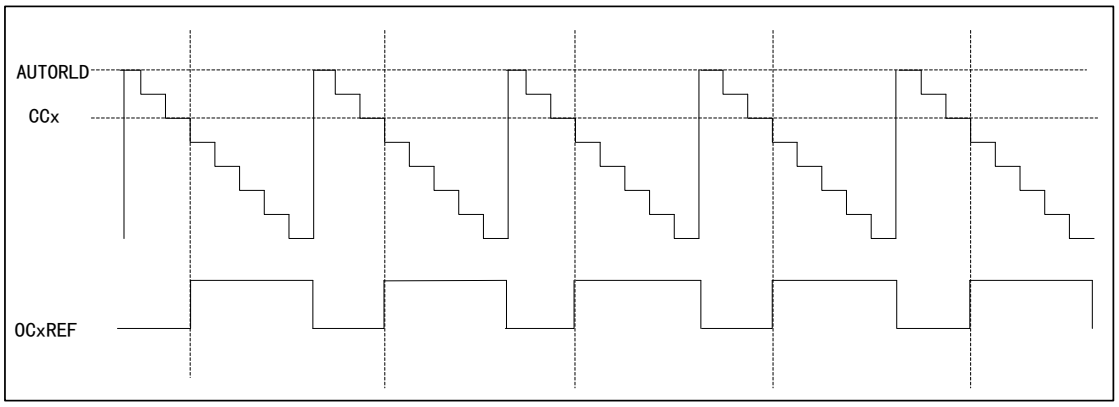
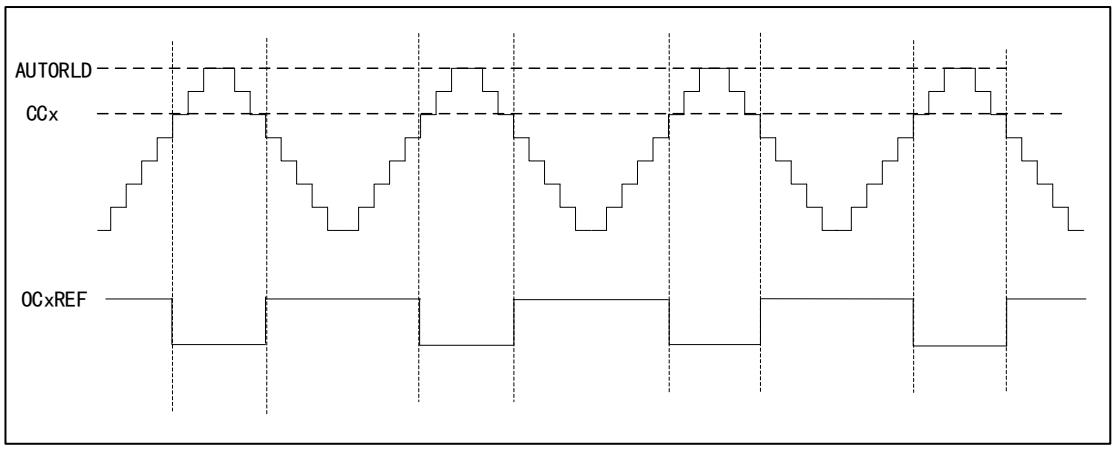


Figure 57 Timing Diagram of PWM1 Central alignment Mode



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 58 Timing Diagram of PWM2 Count-up Mode

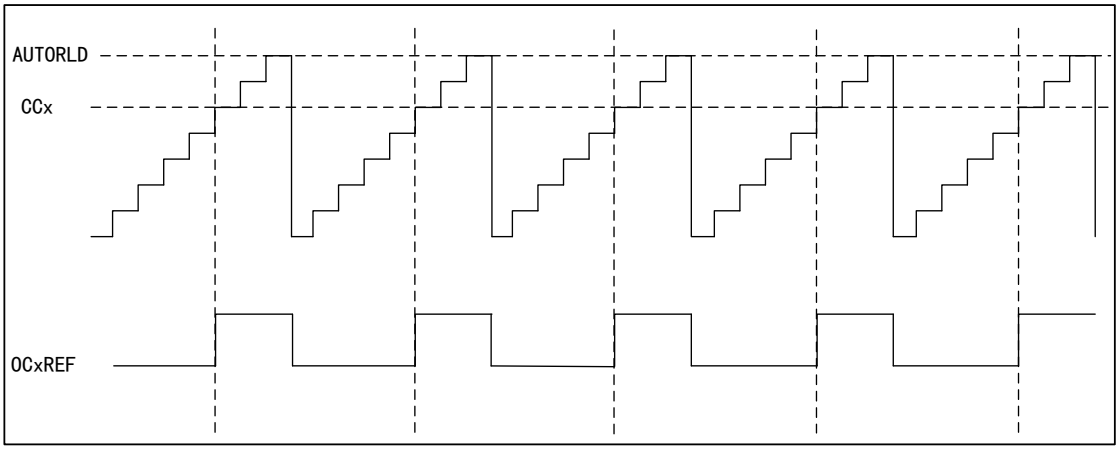


Figure 59 Timing Diagram of PWM2 Count-down Mode

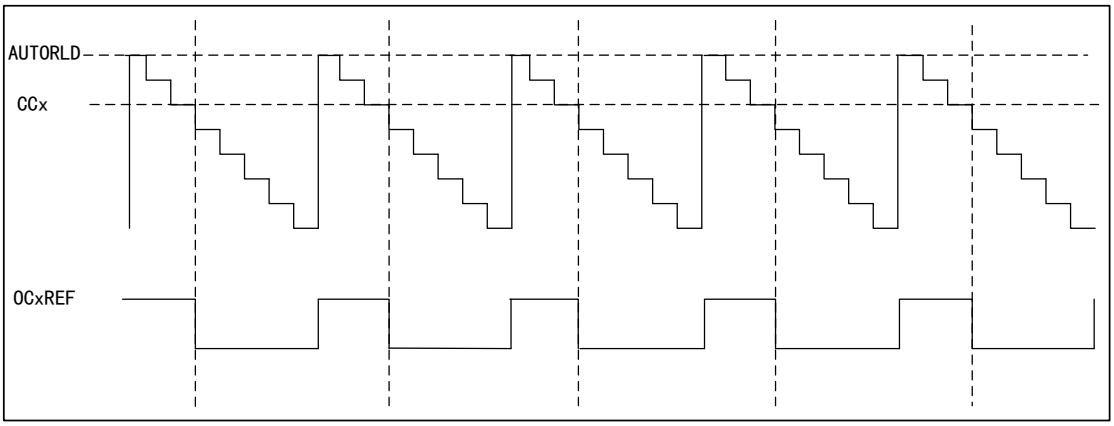
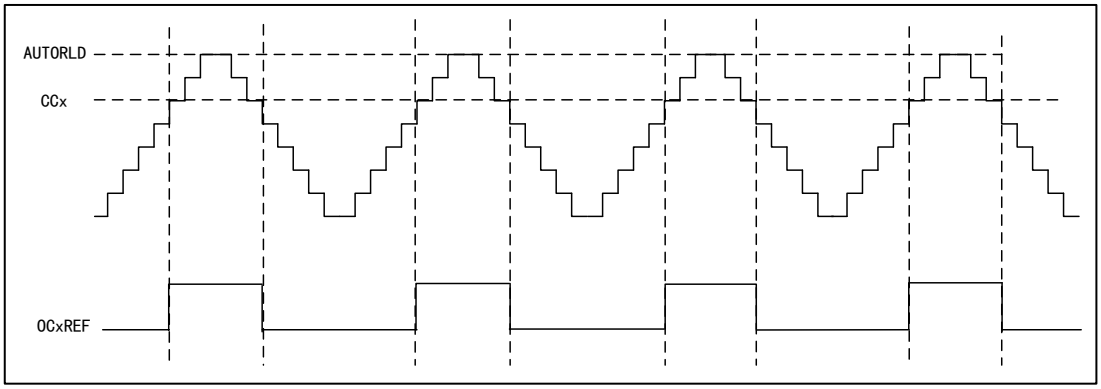


Figure 60 Timing Diagram of PWM2 Central alignment Mode



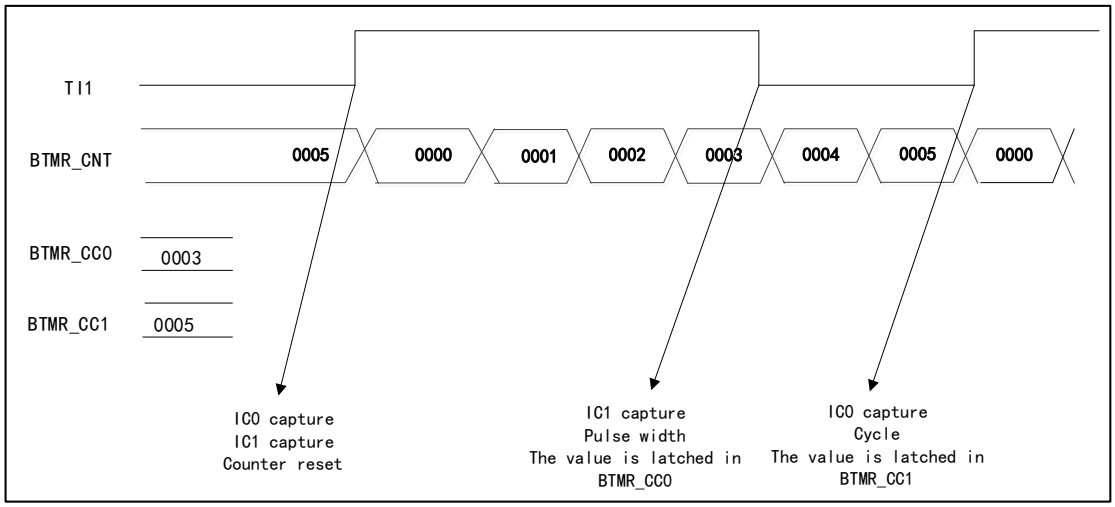
13.4.6 PWM input mode

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 61 Timing Diagram in PWM Input Mode



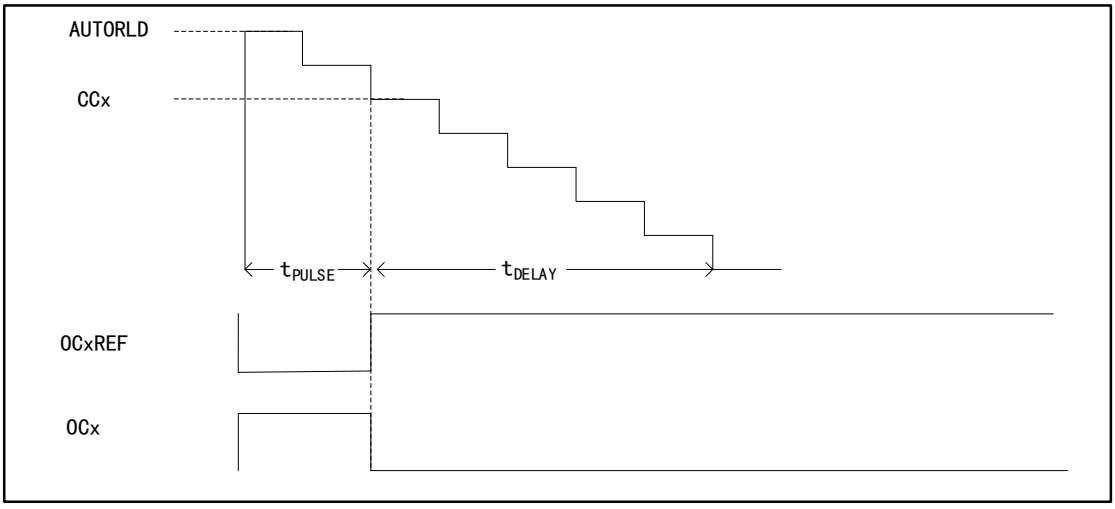
13.4.7 Single-pulse mode

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_CTRL1 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-CCx; in the count-down mode, the delay time is AUTORLD-CCx and the pulse width is CCx.

Figure 62 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.

- CCx_IOSEL=1 for BTMR_CCM2 register, set CCx channel as output
- OCxMOD=100/101 for BTMR_CCM1 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 38 BTIMER Register Address Mapping

Register name	Description	Offset address
BTMR_CTRL1	Control register 1	0x00
BTMR_CCM1	Capture/Compare mode register 1	0x04
BTMR_CCM2	Capture/Compare mode register 2	0x08
BTMR_CEG	Control event generation register	0x0C
BTMR_IEN	Interrupt enable register	0x10
BTMR_STS	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
BTMR_CC1	Channel 1 capture/compare register	0x28
BTMR_CC2	Channel 2 capture/compare register	0x2C

13.6 Register functional description

13.6.1 Control register 1 (BTMR_CTRL1)

Offset address: 0x00

Reset value: 0x0000

Field	Name	R/W	Description
31:8	Reserved		
7	PSC_BUFF_EN	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	ALD_BUFF_EN	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	UPEN	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:

Field	Name	R/W	Description
			The counter overruns/underruns; Set UEG bit; 1: Disable update event
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 Capture/Compare mode register 1 (BTMR_CCM1)

Offset address: 0x04

Reset value: 0x0000

Field	Name	R/W	Description
31:30	Reserved		
29:27	OC2MOD	R/W	Output Compare Channel 2 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.
26:24	IC2F	R/W	Input filtering coefficient Based on TIMER

Field	Name	R/W	Description
			000: bypass 001: 1 PCLK 010: 2 PCLK 011: 3 PCLK 100: 4 PCLK 101: 5 PCLK 110: 6 PCLK 111: 7 PCLK
23:22	CC2_EDGE_SEL	R/W	Capture effective edge 00: Rising edge is effective 01: Falling edge is effective 10: Both edges are effective 11: Rising edge active
21	CC2POL	R/W	Capture/Compare Channel 2 Output Polarity Configure When CC2 channel is configured as output: 0: Phase not reversed 1: Phase reversed. When CC2 channel is configured as input: 0: Phase not reversed 1: Phase reversed.
20	CC2EN	R/W	CC2 channel input and output enable 0: Not enable 1: Enable
19:17	OC1MOD	R/W	Output Compare Channel 1 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.
16:14	IC1F	R/W	Input filtering coefficient Based on TIMER 000: bypass

Field	Name	R/W	Description
			001: 1 PCLK 010: 2 PCLK 011: 3 PCLK 100: 4 PCLK 101: 5 PCLK 110: 6 PCLK 111: 7 PCLK
13:12	CC1_EDGE_SEL	R/W	Capture effective edge 00: Rising edge is effective 01: Falling edge is effective 10: Both edges are effective 11: Rising edge active
11	CC1POL	R/W	Capture/Compare Channel 1 Output Polarity Configure When CC1 channel is configured as output: 0: Phase not reversed 1: Phase reversed. When CC1 channel is configured as input: 0: Phase not reversed 1: Phase reversed.
10	CC1EN	R/W	CC1 channel input and output enable 0: Not enable 1: Enable
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

Field	Name	R/W	Description
			010: 2 PCLK 011: 3 PCLK 100: 4 PCLK 101: 5 PCLK 110: 6 PCLK 111: 7 PCLK
3:2	CC0_EDGE_SEL	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 Capture/Compare mode register 2 (BTMR_CCM2)

Offset address: 0x08

Reset value: 0x0000

Field	Name	R/W	Description
31:6	Reserved		
5	CC2_IOSEL	R/W	Channel 2 input and output select 0: Analog input mode 1: Output mode
4	CC1_IOSEL	R/W	Channel 1 input and output select 0: Analog input mode 1: Output mode
3	CC0_IOSEL	R/W	Channel 0 input and output select 0: Analog input mode 1: Output mode
2	CC2_BUFF_EN	R/W	CC2 register auto- reloading buffer enable When the buffer is disabled, program modifications will immediately change the value loaded into CC2; when the buffer is enabled, program modifications to CC2 will change the value loaded into the counter at the next update event. 0: Disable 1: Enable
1	CC1_BUFF_EN	R/W	CC1 register auto- reloading buffer enable When the buffer is disabled, program modifications will immediately change the value loaded into CC1; when the buffer is

Field	Name	R/W	Description
			enabled, program modifications to CC1 will change the value loaded into the counter at the next update event. 0: Disable 1: Enable
0	CC0_BUFF_EN	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

Field	Name	R/W	Description
15:4	Reserved		
3	CC2EG	W	Capture/Compare Channel 2 Event Generation Refer to CC0EG description
2	CC1EG	W	Capture/Compare Channel 1 Event Generation Refer to CC0EG 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_IEN)

Offset address: 0x10

Reset value: 0x0000

Field	Name	R/W	Description
15:4	Reserved		
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 1: Enable
0	UIEN	R/W	Update Interrupt Enable 0: Disable 1: Enable

13.6.6 Status register (BTMR_STS)

Offset address: 0x14

Reset value: 0x0000

Field	Name	R/W	Description
15:7	Reserved		
6	CC2RCFLG	RC_W0	Capture/compare Channel 2 Repetition Capture Flag Refer to STS_CC0RCFLG
5	CC1RCFLG	RC_W0	Capture/Compare Channel 1 Repetition Capture Flag Refer to STS_CC0RCFLG
4	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.
3	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.
2	CC2IFLG	RC_W0	Capture/Compare Channel 2 Interrupt Flag Refer to STS_CC0IFLG
1	CC1IFLG	RC_W0	Capture/Compare Channel 1 Interrupt Flag Refer to STS_CC0IFLG
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 capture/compare channel 0 is configured as input:

Field	Name	R/W	Description
			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

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

13.6.8 Prescaler register (BTMR_PSC)

Offset address: 0x1C

Reset value: 0x0000

Field	Name	R/W	Description
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: 0xFFFF

Field	Name	R/W	Description
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

Field	Name	R/W	Description
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_CCM1 register), the written value will immediately affect the output comparison results; If the output compare preload is enabled (OC0PEN=1 for BTMR_CCM1 register), the written value will affect the output comparison result when an update event is generated.

13.6.11 Channel 1 capture/compare register (BTMR_CC1)

Offset address: 0x28

Reset value: 0x0000

Field	Name	R/W	Description
15:0	CC1	R/W	Capture/Compare Channel 1 Value Refer to BTMR_CC0

13.6.12 Channel 2 capture/compare register (BTMR_CC2)

Offset address: 0x2C

Reset value: 0x0000

Field	Name	R/W	Description
15:0	CC2	R/W	Capture/Compare Channel 2 Value Refer to BTMR_CC0

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

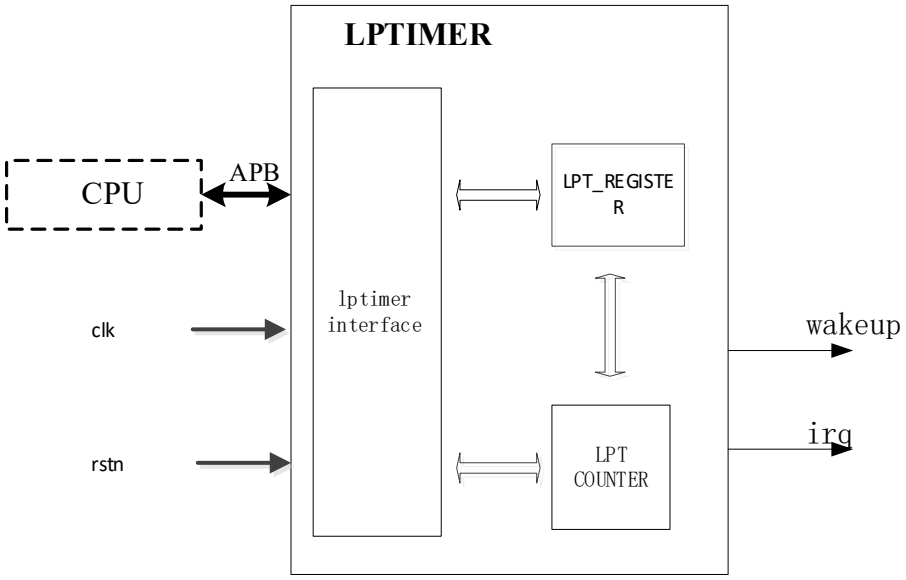
(2) Configurable clock frequency selection

(3) Low-frequency independent clock drive

(4) Wake-up output

14.3 Structure block diagram

Figure 63 Structure Block Diagram



14.4 Functional description

14.4.1 Direction for use

(1) Configure LPTMR_PSC register

(2) Configure LPTMR_WUPV register

(3) Configure LPTMR_CTRL to enable interrupts and the module

14.5 Register address mapping

Table 39 LPTIMER Register Address Mapping

Register name	Description	Offset address
LPTMR_CTRL	Control register	0x00
LPTMR_PSC	Prescale register	0x04
LPTMR_WUPV	Wake-up Value Register	0x08
LPTMR_STS	Status register	0x0C
LPTMR_CNT	Counter register	0x10

14.6 Register functional description

14.6.1 Control register (LPTMR_CTRL)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	EN	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 Value Register (LPTMR_WUPV)

Offset address: 0x08

Reset value: 0x0000 FFFF

Field	Name	R/W	Description
31:16	Reserved		

Field	Name	R/W	Description
15:0	WUPV	R/W	Wakeup value

14.6.4 Status register (LPTMR_STS)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:3	Reserved		
2	CNTUFLG	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 CNTUFLG bit is cleared to 0.
1	PSCUFLG	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 PSCUFLG bit is cleared to 0.
0	INTSTS	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

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_CNTRLD, key register IWDT_KEY and status register IWDT_STS.

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

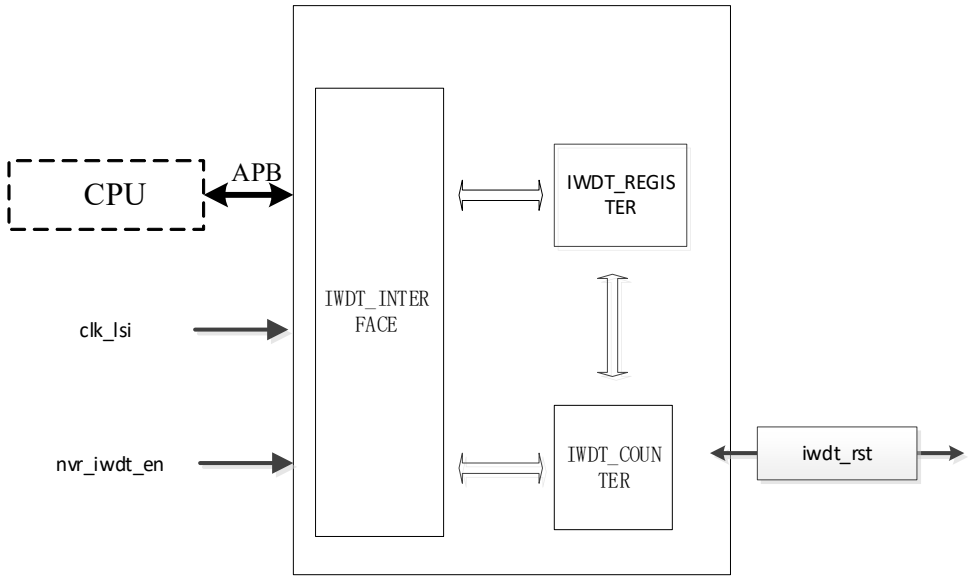
(1) Configurable 12-bit down counter

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

(3) Hardware boot

15.1.3 Structure block diagram

Figure 64 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_CNTRLD 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

(1) Configure KEY=16 'hCCCC to enable IWDT;

(2) Configure KEY=16'h5555, and set the IWDT_PSC register (PSC is set to 1, indicating that it is being updated. The next write operation can only be performed after zeroing);

(3) Configure KEY=16'h5555, and set the IWDT_CNTRLD register (PSC set to 1, indicating that it is being updated, and the next write operation can only be performed after zeroing);

(4) Set the KEY to 16 'hAAAA at regular intervals to feed the dog and reload the value of the IWDT counter to prevent a reset when it decays to zero.

15.2 Window watchdog

15.2.1 Introduction

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

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

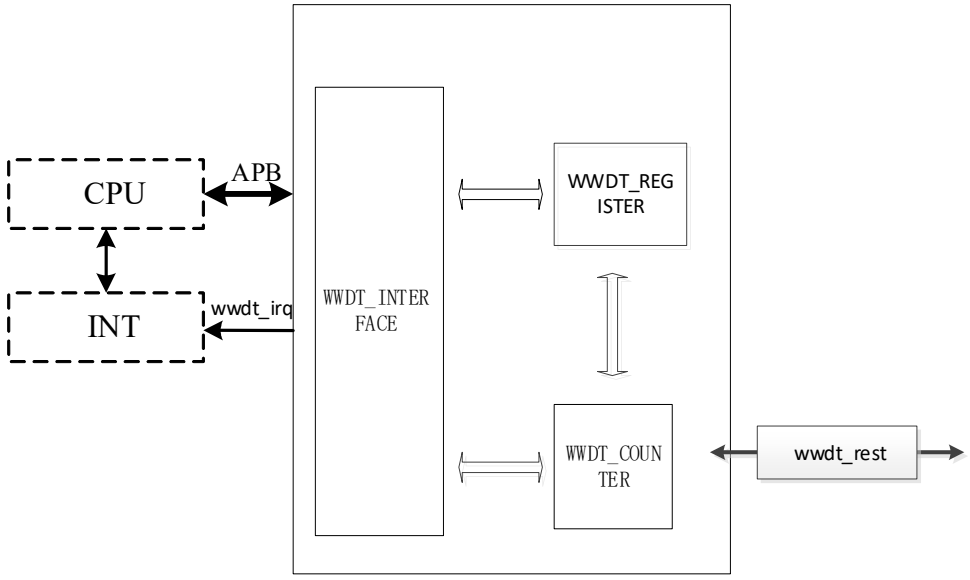
(1) Configurable 7-bit down counter

(2) Configurable window selection

(3) Configurable window interrupt

15.2.3 Structure block diagram

Figure 65 Window Watchdog Structure Block Diagram



15.2.4 Functional description

Enable window watchdog timer, and the reset conditions are:

- When the counter counts down to 0x3F, a reset will be generated.
- 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_CTRL 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.

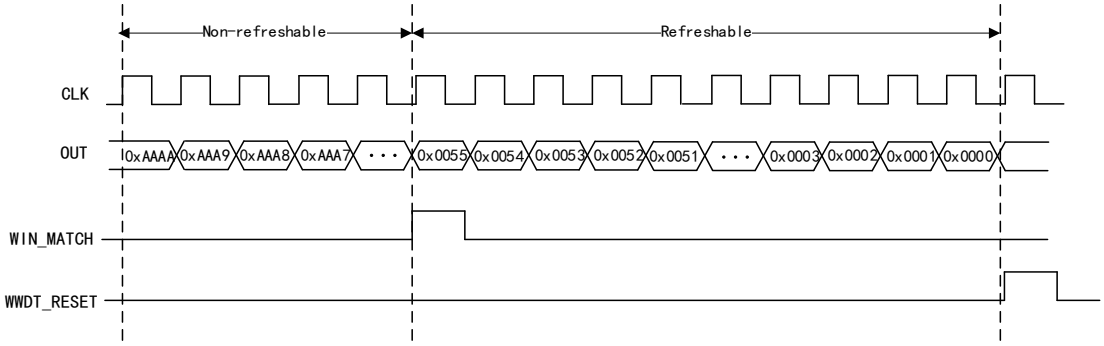
15.2.4.3 Window mode

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

15.2.4.4 Feed the dog

The WWDT feeding action is to directly load the count into the CNT bit of the WWDT_CTRL register, and the CNT restarts counting from the loaded value, as shown in the figure where the window value is configured as 0x0055.

Figure 66 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_STS register.

15.2.4.5 Direction for use

(1) Configure the WWDT_CFG register to set the prescaler value, window value, and interrupt enable bit.

(2) Configure the WWDT_CTRL register to set the watchdog enable and the down-counter count value.

(3) Periodically load the CNT bit of the WWDT_CTRL register to feed the watchdog. The feeding operation must be within the window; otherwise, a reset will occur.

(4) When the count reaches 0x40, the hardware will set the EWIFLG bit in the WWDT_STS register. If the interrupt enable is turned on, an interrupt will be generated to the CPU, the interrupt service routine will be called, and the counter value will be loaded to avoid reset; the software can write 0 to clear EWIFLG and the interrupt.

15.3 IWDT register address mapping

Table 40 IWDT Register Mapping

Register name	Description	Offset address
IWDT_KEY	Key register	0x00
IWDT_PSC	Prescale register	0x04
IWDT_CNTRLD	Counter reload register	0x08
IWDT_STS	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_CNTRLD 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). The read-out value is 0x0000.

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 PSCUFLG=0 for IWDT_STS register, can the prescaler factor be changed; in the process of reading this register, only when PSCUFLG=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

Field	Name	R/W	Description
			111: PSC=256

15.4.3 Counter reload register (IWDT_CNTRLD)

Offset address: 0x08

Reset value: 0x0000 0FFF

Field	Name	R/W	Description
31:12	Reserved		
11:0	CNTRLD	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 CNTUFLG=0. In the process of reading this register, only when CNTUFLG=0 in IWDT_STS 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_STS)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:2	Reserved		
1	CNTUFLG	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 CNTUFLG bit is cleared to 0.
0	PSCUFLG	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 PSCUFLG bit is cleared to 0.

15.5 WWDT register address mapping

Table 41 WWDT Register Address Mapping

Register name	Description	Offset address
WWDT_CTRL	Control register	0x00
WWDT_CFG	Configuration register	0x04
WWDT_STS	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_CTRL)

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	CNT	R/W	Down counter Configurable when enabled; when disabled, the count reverts to the default value 0x7F. A reset will occur when the count value is greater than WIN, and a reset is also triggered when the T6 bit of the count is 0. Note: When writing to the WWDT_CTRL register, always set the T6 bit to '1' to avoid an immediate reset.

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	TBPSC	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	WIN	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_STS)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		

Field	Name	R/W	Description
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 asynchronous receiver/transmitter (UART)

16.1 Full Name and Abbreviation Description of Terms

Table 42 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

UART (universal asynchronous receiver transmitter) 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. UART also provides a wide range of baud rate and supports multiprocessor communication.

UART 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.

UART 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: 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 16x oversampling rate

(7) Independent transmitter and receiver enable bit

(8) Independent signal polarity control transmitter and receiver

(9) Can switch TX/RX pins

(10) Programmable baud rate generator, with the baud rate up to 2.5Mbits

(11) Automatic baud rate detection

(12) Generation and detection of LIN break frame

(13) Receive frame error detection

(14) Hardware parity detection

(15) 1/16 bit noise detection

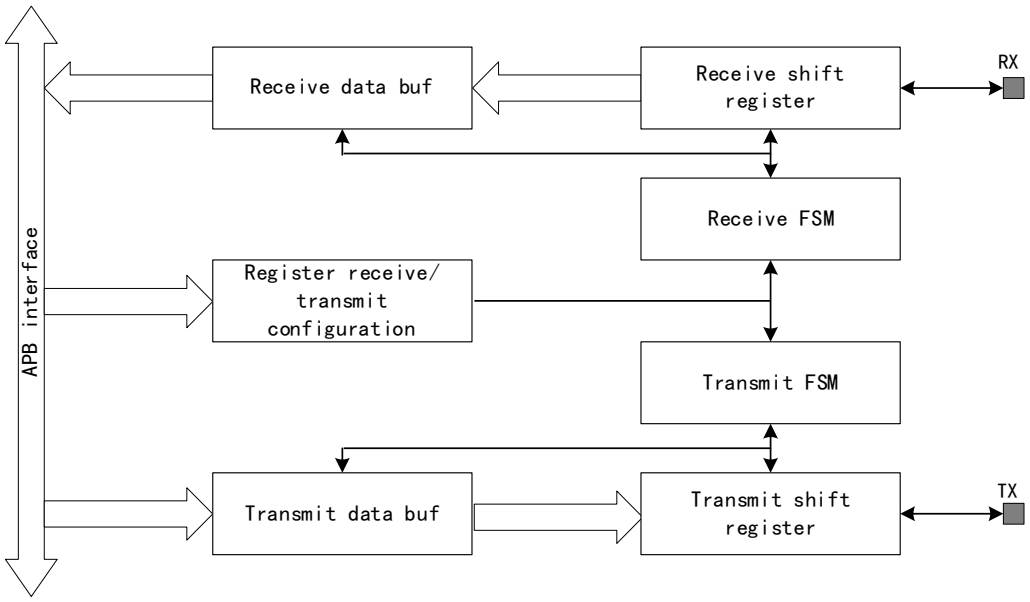
(16) DMA can be used for continuous communication

(17) 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

(18) Multiple interrupt sources:
- The transmit register is empty
- Transmission completed
- The receive register is not empty
- Parity check error
- LIN transmission collision detection
- LIN break detection
- Noise error
- Overrun error
- Frame error
- Failed to receive interrupt on time

16.4 Structure block diagram

Figure 67 Structure Block Diagram



16.5 Functional description

Table 43 UART Pin Description

Pin	Type	Description
UART_RX	Input	Data receiving
UART_TX	Output I/O (single-line mode/smart card mode)	Data transmission When the transmitter is enabled and does not transmit data, the default is high
UART_CK	Output	Clock output
UART_nRTS	Input	Request to send in hardware flow control mode
UART_nCTS	Output	Clear to send in hardware flow control mode
UART_DE	Input	Drive enable activating external transmitter/receiver

16.5.1 Single-line half-duplex communication

HDEN bit of UART_CTRL3 register determines whether to enter the single-line half-duplex mode.

When UART enters single-line half-duplex mode:

- The CLKEN bit and LINMEN bit in the UART_CTRL2 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 TXCFLG bit of UART_STS 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 UART_CTRL1 register

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

Table 44 UART Frame Format

DBLCFG bit	PCEN bit	UART data frame
0	0	Start bit+8-bit data+stop bit
0	1	Start bit+7-bit data+parity check bit+stop bit
1	0	Start bit+9-bit data+stop bit
1	1	Start bit+8-bit data+ parity check bit+stop bit

Configurable stop bit

2 different stop bits can be configured through STOPCFG bit of UART_CTRL2 register.

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

Check bit

PCFG bit of UART_CTRL1 determines the parity check bit; when PCFG=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 UART_CTRL1 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 UART_STS register will be set.

- If the check control is enabled, corresponding interrupt will be triggered.

16.5.3 Transmitter

When TXEN bit of the register UART_CTRL1 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 UART, the least significant bit of the data will be moved out by TX pin first. In this mode, UART_DATA 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 UART_CTRL1 register to enable UART

(2) Decide the word length by setting DBLCFG bit of UART_CTRL1 register

(3) Decide the number of stop bits by setting STOPCFG bit of UART_CTRL2 register

(4) If multi-buffer communication is selected, DMA should be enabled in UART_CTRL3 register

(5) Set the baud rate of communication in UART_BR register

(6) Enable TXEN bit in UART_CTRL1 register, and transmit an idle frame

(7) Write data to UART_DATA register (if DMA is not enabled, repeat step 7 for each byte to be transmitted)

(8) Wait for TXCFLG bit of UART_STS register to be set to 1, indicating transmission completion

Note: TXEN 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 UART_DATA 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 UART_CTRL1 register is set to 1, an interrupt will be generated.

(2) If UART 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 UART 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, TXCFLG bit will be set to 1; at this time if TXCIEN bit in UART_CTRL1 register is set to 1, an interrupt will be generated.

(5) After the last data is written to the UART_TXDATA register, before entering the low-power mode or before disabling the UART module, wait to set TXCFLG 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 TXBF bit of UART_CTRL1 register, and the length of the break frame is determined by the DBLCFG bit of UART_CTRL1 register. If the TXBF 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 TXBF bit is reset before transmission of the break frame starts, the break frame will not be transmitted. To transmit two consecutive break frames, the TXBF 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 TXEN bit of UART_CTRL1 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 UART, RX pin will first introduce the least significant bit of the data. In this mode, UART_DATA 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 UART_DATA.

Receiving configuration steps

(1) Set UEN bit of UART_CTRL1 register to enable UART

(2) Decide the word length by setting DBLCFG bit of UART_CTRL1 register

(3) Decide the number of stop bits by setting STOPCFG bit of UART_CTRL2 register

(4) If multi-buffer communication is selected, DMA should be enabled in UART_CTRL3 register

(5) Set the baud rate of communication in UART_BR register

(6) Set RXEN bit of UART_CTRL1 to enable receiving

Note:

(1) RXEN 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 UART_DATA register, the RXBNEFLG bit of UART_STS 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 UART_DATA register by software or by writing 0.

(6) In multi-buffer mode, after each byte is received, the RXBNEFLG bit of UART_STS 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, UART will handle it as receiving a frame error.

16.5.4.3 Idle frame

When the receiver receives an idle frame, UART will handle it as receiving an ordinary data frame; if IDLEIEN bit of UART_CTRL1 is set, an interrupt will be generated.

16.5.4.4 Select the clock source

The clock source must be selected by clock control system before UART is enabled

(1) The clock source shall be selected according to the transmission speed and the possibility of use of UART in low-power mode.

(2) The clock source frequency is f_{CK}.

- The range of communication speed is determined by the clock source. UART should be enabled before the clock source is selected.
- When UART adopts dual-clock domain or wakes up the stop mode, PCLK, HSICLK or SYSCLK can be used as the clock source; otherwise, the clock source is PCLK.
- If LSICLK is selected as the clock source, UART can receive data even in low-power mode. It can select according to the received data and wake-up mode, and wake up SoC when necessary, so that DMA can read the received data.
- The receiver realizes the data recovery of different oversampling technologies configured by users to distinguish valid incoming data and noises, which requires a trade-off between the maximum communication speed and noise/clock inaccuracy immunity.

16.5.4.5 Overrun error

When RXBNEFLG bit of UART_STS register is set to 1 and a new character is received at the same time, an overrun error will be caused. Only after RXEN 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 OVREFLG bit of UART_STS 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 ERRIEN bit of UART_CTRL1 is set, an interrupt will be generated
- When OVREFLG 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 OVREFLG bit can be reset by reading UART_STS and UART_DATA 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 UART_STS register
- Invalid data is transmitted from the shift register to UART_DATA register

- In single- byte communication, no interrupt will be generated, but in multi-buffer communication, an interrupt will be generated by setting the ERRIEN bit of UART_CTRL3 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 UART_STS register
- Invalid data is transmitted from the shift register to UART_DATA register
- In single- byte communication, no interrupt will be generated, but in multi-buffer communication, an interrupt will be generated by setting the ERRIEN bit of UART_CTRL3 register

16.5.5 TX and RX pin swap

The TX/RX pin functions can be swapped by enabling the SWAPEN bit in the UART_CTLR1 register. After enabling, the functions of the TX and RX pins are swapped

16.5.6 Baud rate generator

The baud rate divider factor (UARTDIV) is a 16-digit number consisting of 12-digit integer part and 4-digit decimal part. Its relationship with the system clock:

$$\text{Baud rate}=f_{CK}/(16\times\text{UARTDIV})$$

After writing to UART_BR, 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. UART must be enabled after the system clock is enabled in the clock control unit.

Table 45 Baud Rate Error Calculation (UART_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, UART can detect and automatically set the value of the UART_BR 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, an automatic baud rate detection error will also be generated.

ABRDEN bit of UART_CTRL2 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 ABRDFLG flag bit of UART_STS 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 ABRDEFLG 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 ABRDFLG flag (writing a 0).

(4) UART cannot be disabled during automatic baud rate detection; otherwise, the BR value may be damaged.

16.5.8 LIN mode

LINMEN bit of UART_CTRL2 register decides whether to enter LIN mode.

When entering LIN mode:

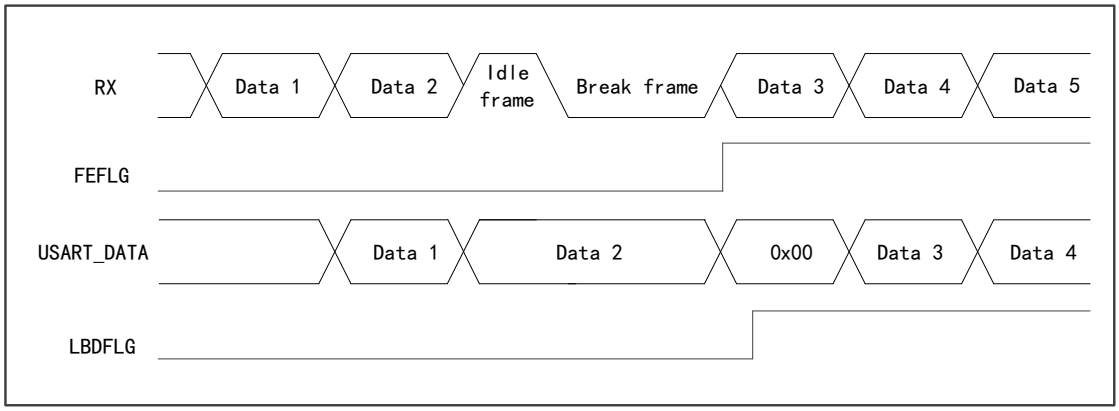
- Each data frame includes 8 data bits and 1 stop bit
- The CLKEN bit and STOPCF bit of UART_CTRL2 register and IREN bit, HDEN bit of UART_CTRL3 register need to be cleared to 0.

In LIN master mode, UART can generate break frame, and the detection length of break frame can be set to 10 or 11 bits through LBDLCFG bit of UART_CTRL2. The break frame detection circuit is independent of UART receiver, and whether in idle state or in data transmission state, RX pin can detect the break frame, and the LBDFLG bit of UART_STS register is set to 1; at this time, if the LBDIEN bit of UART_CTRL2 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.

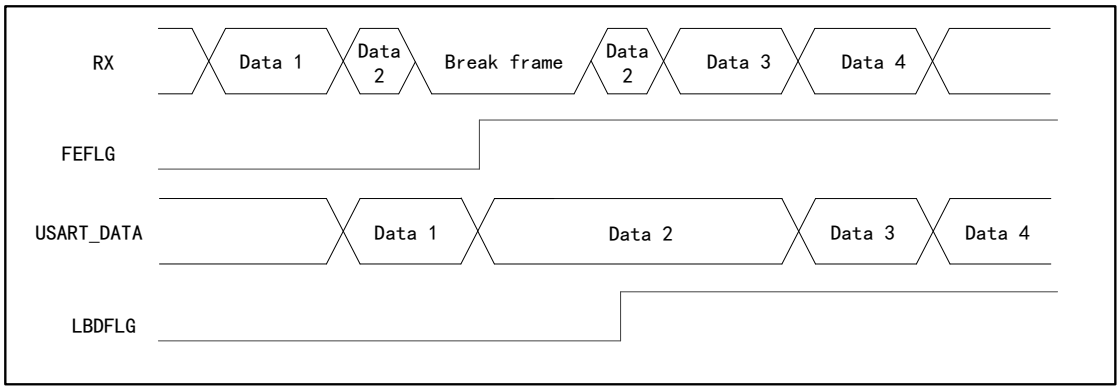
Figure 68 Break Frame Detection in Idle State



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 69 Break Frame Detection in Data Transmission State



16.5.9 DMA multi-buffer communication

To reduce the burden of processors, UART can access the data buffer in DMA mode.

Transmission in DMA mode

The DMATXEN bit of UART_CTRL3 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 TXCFLG flag bit of UART_STS register to 0

(2) Set the address of SRAM memory storing data as DMA source address

(3) Set the address of UART_DATA 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 TXCFLG bit of UART_STS register to be set to 1, indicating transmission completion

Receive by DMA

The DMARXEN bit of UART_CTRL3 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 UART_DATA 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

16.5.10 Interrupt request

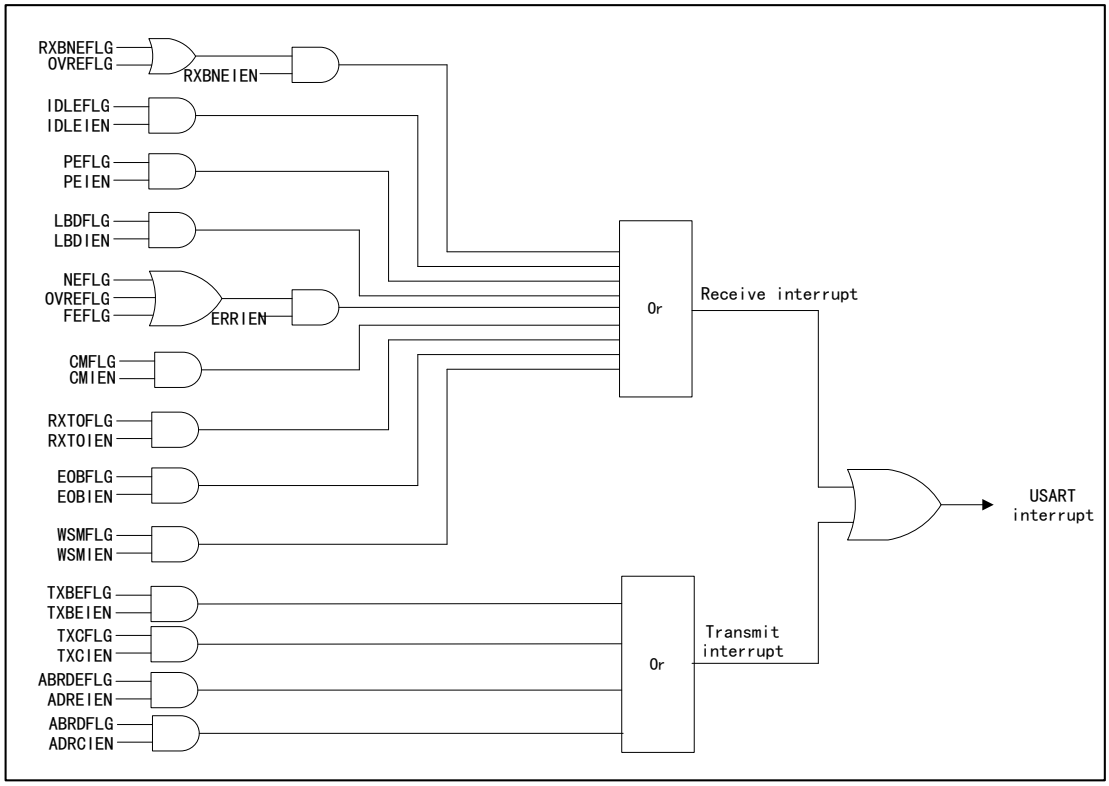
Table 46 UART Interrupt Request

Interrupt event	Event flag bit	Enable bit
The receive register is not empty	RXBNEFLG	RXBNEIEN
Overrun error	OVREFLG	
Idle line is detected	IDLEFLG	IDLEIEN

Interrupt event		Event flag bit	Enable bit
Parity check error		PEFLG	PEIEN
LIN break error		LBDFLG	LBDIEN
Receiving error in DMA mode	Noise error	NEFLG	ERRIEN
	Overrun error	OVREFLG	
	Frame error	FEFLG	
Error of failing to receive on time		RXTOFLG	RXTOIEN
Data transmit register is empty		TXBEFLG	TXBEIEN
Transmission completed		TXCFLG	TXCIEN
Automatic baud rate error flag		ABRDEFLG	ADREIEN
Automatic baud rate completion flag		ABRDFLG	ABRCIEN

All interrupt requests of UART are connected to the same interrupt controller, and the interrupt requests have logical or relation before they are transmitted to the interrupt controller.

Figure 70 UART Interrupt Mapping



16.5.11 Comparison of UART supporting functions

Table 47 Comparison of UART Supporting Functions

UART mode	UART1	UART2
Half duplex (single-line mode)	√	√
Mode supporting automatic baud rate detection	2.5	2.5
LIN	√	√
Multi-buffer communication (DMA)	√	√
Receiving timeout interrupt	√	√

Note:

(1) "√" means this function is supported, while "—" means that this function is not supported.

16.6 Register address mapping

Table 48 UART Register Address Mapping

Register name	Description	Offset address
UART_STS	Status register	0x00
UART_DATA	Data register	0x04
UART_BR	Baud rate register	0x08
UART_CTRL1	Control register 1	0x0C
UART_CTRL2	Control register 2	0x10
UART_CTRL3	Control register 3	0x14
UART_RXTO	Receive timeout register	0x1C

16.7 Register functional description

16.7.1 Interrupt and status register (UART_STS)

Offset address: 0x00

Reset value: 0x0000 00C0

Field	Name	R/W	Description
31:13	Reserved		
12	ABRDEFLG	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	ABRDFLG	RC_W0C	Auto Baud Rate Detection Flag 0: Automatic baud rate detection is not completed

Field	Name	R/W	Description
			1: Automatic baud rate detection is completed Set to 1 by hardware; cleared to 0 by software (writing 0 to this bit).
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	Reserved		
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 UART_DATA register.
6	TXCFLG	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 UART_STS register and then writing to the UART_DATA 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 UART_DATA 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 UART_STS register and then reading the UART_DATA register. This bit will not be set high again until RXBNEFLG is set.
3	OVREFLG	R	Overrun Error Occur Flag 0: No overrun error 1: Overrun error is detected

Field	Name	R/W	Description
			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 UART_STS register and then reading the UART_DATA 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 UART_STS register and then reading the UART_DATA 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 ERRIEN 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 UART_STS register and then reading the UART_DATA 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 OVREFLG flag bit. In multi-buffer communication mode, if the ERRIEN 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 UART_STS register and then reading the UART_DATA register. Software must wait for the RXNEFLG flag bit to be set to 1 before clearing PEFLG.

16.7.2 Data register (UART_DATA)

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. Note: This register only supports 32-bit access.

16.7.3 Baud rate register (UART_BR)

This register can be set only when UART 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 UART Baud Rate Divider Factor The integral part of UART baud rate divider factor is determined by these 12 bits. If TXEN or RXEN is disabled respectively, the baud rate counter stops counting.
3:0	FBR	R/W	Fraction of UART Baud Rate Divider Factor The decimal part of UART baud rate divider factor is determined by these four bits. If TXEN or RXEN is disabled respectively, the baud rate counter stops counting.

16.7.4 Control register 1 (UART_CTRL1)

Offset address: 0x0C

Reset value: 0x0000 0000

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 UART 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.
25:14	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	DBLCFG	R/W	Data Bits Length Configure 0: 1 start bit, 8 data bits, n stop bits

Field	Name	R/W	Description
			1: 1 start bit, 9 data bits, n stop bits 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	PCFG	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	TXCIEN	R/W	Transmit Complete Interrupt Enable 0: Disable 1: An interrupt will be generated when TXCFLG is set
5	RXNEIEN	R/W	Receive register Not Empty Interrupt Enable 0: Disable 1: An interrupt will be generated when OVREFLG or RXBNEFLG is set
4	IDLEIEN	R/W	IDLE Interrupt Enable 0: Disable 1: An interrupt will be generated when IDLEFLG is set
3	TXEN	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	RXEN	R/W	Receive Enable 0: Disable 1: Enable, and start to detect the start bit on RX pin
1	Reserved		
0	TXBF	R/W	Transmit Break Frame 0: Not transmit 1: Will transmit

Field	Name	R/W	Description
			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 (UART_CTRL2)

Offset address: 0x10

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:19	Reserved		
18	ADREIEN	R/W	Automatic baud rate detection error interrupt enable 0: Disable 1: Enable
17	ABRCIEN	R/W	Automatic baud rate completion interrupt enable 0: Disable 1: Enable
16	ABRDEN	R/W	Auto Baud Rate Detection Enable 0: Disable 1: Enable Set or cleared to 0 by software.
15	Reserved		
14	LINMEN	R/W	LIN Mode Enable 0: Disable 1: Enable This bit is set or cleared by software. In LIN mode, the TXBF bit in the UART_CTRL1 register can be used to send LIN synchronization break characters (13 low bits) and detect LIN synchronization break characters.
13	STOPCFG	R/W	STOP Bit Configure 0: 1 stop bit 1: 2 stop bits
12:7	Reserved		
6	LBDIEN	R/W	LIN Break Detection Interrupt Enable 0: Disable 1: Generate an interrupt when LBDFLG bit is set.
5	LBDLCFG	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
4:0	Reserved		

16.7.6 Control register 3 (UART_CTRL3)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:12	Reserved		

Field	Name	R/W	Description
11	SAMCFG	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 UART is not enabled.
10:8	Reserved		
7	DMATXEN	R/W	DMA Transmit Enable 0: Disable 1: Enable This bit is set or cleared by software.
6	DMARXEN	R/W	DMA Receive Enable 0: Disable 1: Enable This bit is set or cleared by software.
5:4	Reserved		
3	HDEN	R/W	Half-duplex Mode Enable 0: Disable 1: Enable
2:1	Reserved		
0	ERRIEN	R/W	Error interrupt Enable 0: Disable 1: Enable; when DMARXEN is set and one among FEFLG, OVREFLG or NEFLG is set, an interrupt will be generated.

16.7.7 Receive timeout register (UART_RXTO)

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 Serial peripheral interface (SPI)

17.1 Full Name and Abbreviation Description of Terms

Table 49 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

17.2 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.

17.3 Main characteristics

(1) Master and slave operation with 4-wire full duplex synchronous transmission and receiving

(2) Select 8-bit or 16-bit transmission frame format

(3) Support special transmission and receiving mark and can trigger interrupts

(4) Have SPI bus busy state flag

(5) Clock polarity and phase are programmable

(6) Independent transmit and receive FIFO

(7) Data sequence is programmable; select MSB or LSB in front

(8) With DMA transmission

17.4 functional description

17.4.1 Description of SPI signal line

Table 50 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
NSS	Software NSS mode: NSS pin can be used for other purposes. Hardware NSS mode of master device hardware: NSS outputs, in single-master mode, NSS OFF output: Operation of multiple master environments is allowed, Slave hardware NSS mode: The NSS signal is set to low as the chip selection signal of the slave

17.4.2 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.

17.4.2.1 Phase and polarity of clock signal

The clock polarity and clock phase are CPOL and CPHA bits of SPI_CTRL1 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 51 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.

17.4.2.2 Data frame format

Select LSB or MSB first by configuring LSBSEL bit of SPI_CTRL1 register. When accessing SPI_DATA 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.

17.4.3 NSS mode

Software NSS mode: Select to enable or disable this mode by configuring SSEN bit of SPI_CTRL1 register, and the internal NSS signal level is driven by ISSEL bit of SPI_CTRL1 register.

Hardware NSS mode:

- Enable NSS output: When SPI is in master mode, enable SSEN 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.

17.4.4 SPI mode

17.4.4.1 Initialization of SPI master mode

In master mode, serial clock is generated on SCK pin.

Configure master mode

- Configure MSMCFG=1 in SPI_CTRL1 register, and set to master mode

- Select the serial clock baud rate by configuring BRSEL bit in SPI_CTRL1 register
- Select the polarity and phase by configuring CPOL and CPHA bits in SPI_CTRL1 register
- Select the transmission mode by configuring BMOEN and BMEN bits in SPI_CTRL1 register
- Select data frame length by configuring DFLSEL bit of SPI_CTRL2 register
- Select LSB or MSB first by configuring LSBSEL in SPI_CTRL1 register
- NSS configuration: in hardware mode, it is required to connect NSS pin to high level during the entire data frame transmission; in software mode, it is required to set SSEN bit and ISSEL bit in SPI_CTRL1 register
- Configure SPIEN bit in SPI_CTRL1 register to enable SPI

In master mode: MOSI pin is data output, while MISO is data input.

17.4.4.2 Initialization of SPI slave mode

In slave mode, SCK pin receives the serial clock from the master device.

Configuration of slave mode

- Configure MSMCFG=0 in SPI_CTRL1 register, and set to slave mode
- Select the polarity and phase by configuring CPOL and CPHA bits in SPI_CTRL1 register
- Select the transmission mode by configuring BMOEN and BMEN bits in SPI_CTRL1 register
- Select data frame length by configuring DFLSEL bit of SPI_CTRL2 register
- Select LSB or MSB first by configuring LSBSEL in SPI_CTRL1 register
- NSS configuration:
  - In hardware mode: NSS pin must be low in the whole data frame transmission process
  - In software mode: Set SSEN bit in SPI_CTRL1 register and clear ISSEL bit
- Configure SPIEN bit in SPI_CTRL1 register to enable SPI

In slave mode: MOSI pin is data input, while MISO is data output.

17.4.4.3 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.

17.4.4.4 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.

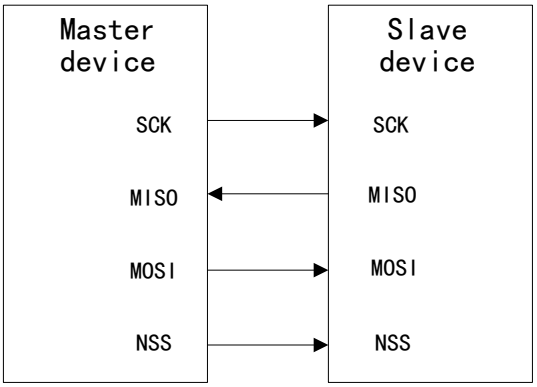
The master device decides using which slave device to transmit data by pulling down the NSS pin of the slave device.

17.4.5 Data transmission and receiving process in different modes of SPI

Table 52 Run Mode of SPI

Mode	Configuration	Data pin
Full-duplex mode of master device	BMEN=0, RXOMEN=0	MOSI transmits; MISO receives
Full-duplex mode of slave device	BMEN=0, RXOMEN=0	MOSI receives; MISO transmits

Figure 71 Connection in Full Duplex Mode



17.4.5.1 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_DATA register, the earliest values that have not been read yet and are stored in RXFIFO will be returned. After write access to SPI_DATA, the written data will be stored in TXFIFO at the end of the transmit queue. The FTLSEL and FRLSEL bits indicate the current occupancy levels of the two FIFO.

The read access to SPI_DATA register must be managed by RXBNEFLG event. When the data are stored in RXFIFO and reach the threshold value (defined by previous bit), this event will be triggered; when RXBNEFLG 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 TXBEFLG event. When TXFIFO is less than or equal to half of its capacity, RXBNEFLG event will be triggered; otherwise, TXBEFLG 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. TXBEFLG and RXBNEFLG events can be polled or handled by interrupt.

17.4.5.2 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 FTLSEL bit and BSYFLG 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.

17.4.5.3 Data packing

If the data frame is less than or equal to one byte, when executing 16-bit read and write access to SPI_DATA register, the data will be packed automatically and double data can be processed in parallel. After conducting write access to SPI_DATA, 2-byte data will be transmitted; if the threshold value of RXFIFO is

set to 16 bits, a receive RXBNEFLG event will be generated.

For a single RXBNEFLG event, the data receiver shall perform one read operation to SPI_DATA, 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.

17.4.6 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 TXDEN and RXDEN bits of SPI_CTRL2 register.

- When transmitting: When TXBEFLG flag bit is set to 1, issue the DMA request, DMA controller writes data to SPI_DATA, and then the TXBEFLG flag bit will be cleared.
- When receiving: When setting RXBNEFLG flag bit to 1, issue the DMA request, DMA controller reads data from SPI_DATA register, and then RXBNEFLG flag bit is cleared.

By monitoring BSYFLG 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.

17.4.7 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_CTRL1 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 clearing FTLSEL to 0

(2) Wait for clearing BSYFLG flag bit to 0

(3) Wait for clearing FRLSEL to 0

(4) Disable SPI (SPIEN=0)

Steps of disabling SPI in some single-receive mode

(1) Wait for clearing RXOMEN to 0 or setting BMOEN to 1

(2) Wait for clearing BSYFLG flag bit to 0

(3) Wait for clearing FRLSEL to 0

(4) Disable SPI (SPIEN=0)

17.4.8 SPI interrupt

An interrupt can be triggered by the following events during SPI operation:

- TXFIFO ready for loading
- RXFIFO receives data

17.4.8.1 Status flag bit

Transmit buffer idle flag TXBEFLG

TXBEFLG=1 means that TXFIFO has space to store the transmitted data; TXBEFLG flag bit is connected to TXFIFO bit, and in the process of storing data, if the storage content of TXFIFO is less than or equal to FIFO/2, TXBEFLG flag bit remains high. When the storage content of TXFIFO is greater than FIFO/2, TXBEFLG flag bit will be cleared to 0. If TXBEIEN bit in SPI_CTRL2 register is set, an interrupt will be generated.

Receive buffer non-empty flag RXBNEFLG

RXBNEFLG=1 indicates that the RXFIFO is full of data. If RXBNEIEN bit in SPI_CTRL2 register is set, an interrupt will be generated.

Busy flag BSYFLG

BSYFLG flag is set and cleared by hardware, which can indicate the state of SPI communication layer. When BSYFLG=1, it indicates SPI is communicating. BSYFLG flag can be used to detect whether transmission is over to avoid destroying the last transmitted data.

BSYFLG flag will be cleared to 0 in the following situations

- End the transmission in master mode
- Master mode fault
- In slave mode, there is at least one SPI cycle between two data transmissions
- Disable SPI

During continuous communication:

- In master mode: BSYFLG=1 in the whole transmission process

- In save mode: BSYFLG is kept low within one SCK clock cycle between transmission of data

Note: It is best to use TXBEFLG and RXBNEFLG flags to process the transmitting and receiving of each data item

17.4.8.2 Error flag bit

Overrun error OVRFLG

An overrun error will be generated when the following events occur

- When RXBNEFLG flag bit is still 1 after the master device has transmitted data
- When the space in RXFIFO cannot store the data to be received when receiving data
- When the software or DMA cannot read the data in RXFIFO in time

When an overrun error occurs: OVRFLG bit is set to 1; if ERRIEN bit is also set, an interrupt will be generated.

After an overrun error occurs, the data in the receiving buffer are not the data transmitted by the master device, and by reading SPI_DATA value, the data are the data not read before, and the subsequent data will be discarded.

OVRFLG flag can be cleared by reading SPI_DATA register and SPI_STS register according to the sequence.

Table 53 SPI Interrupt Request

Interrupt flag	Interrupt event	Enable control bit	Clearing method
TXBEFLG	Transmit buffer empty flag	TXBEIEN	Write SPI_DATA register
RXBNEFLG	Receive buffer non-empty flag	RXBNEIEN	Read SPI_DATA register
OVRFLG	Overrun error	OVRIEN	Read SPI_DATA register, and then read SPI_STS register

17.5 Register address mapping

Table 54 SPI Register Address Mapping

Register name	Description	Offset address
SPI_CTRL1	SPI control register 1	0x00
SPI_CTRL2	SPI control register 2	0x04
SPI_STS	SPI status register	0x08
SPI_DATA	SPI data register	0x0C

17.6 Register functional description

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

17.6.1 SPI control register 1 (SPI_CTRL1)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
12	BMEN	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.
11	BMOEN	R/W	Bidirectional Mode Output Enable 0: Disable (receive-only ode) 1: Enable (transmit-only mode) When BMEN=1, namely in single-line bidirectional mode, this bit decides the transmission direction of transmission line.
10	DFLSEL	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").
9	SSEN	R/W	Software Slave Device Enable 0: Disable 1: Enable When SSEN is set, the level of NSS pin is determined by SSEN.
8	ISSEL	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 CTRL1_SSEN=1.
7	LSBSEL	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.

Field	Name	R/W	Description
5:3	BRSEL	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	MSMCFG	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.

17.6.2 SPI control register 2 (SPI_CTRL2)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:5	Reserved		
4	OVRIEN	R/W	Overflow interrupt enable 0: Disable 1: Enable When OVRFLG fag bit is set to 1, an interrupt request will be generated
3	TXBEIEN	R/W	Transmit Buffer Empty Interrupt Enable 0: Disable 1: Enable When TXBEFLG fag bit is set to 1, an interrupt request will be generated
2	RXBNEIEN	R/W	Receive Buffer Not Empty Interrupt Enable 0: Disable 1: Allowed When RXBNEFLG flag bit is set to 1, an interrupt request will be generated

Field	Name	R/W	Description
1	TXDEN	R/W	Transmit Buffer DMA Enable When this bit is set, once TXBEFLG flag is set, DMA request will be issued. 0: Disable 1: Enable
0	RXDEN	R/W	Receive Buffer DMA Enable When RXDEN=1, once RXBNEFLG flag is set, DMA request will be issued. 0: Disable 1: Enable

17.6.3 SPI status register (SPI_STS)

Offset address: 0x08

Reset value: 0x0000 0002

Field	Name	R/W	Description
31:8	Reserved		
7	BSYFLG	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.
5:4	FTLSEL	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.
3:2	FRLSEL	R	FIFO Receive Leve Select 00: FIFO is empty 01: FIFO/4 10: FIFO/2 11: FIFO is full Note: This bit is set to 1 or cleared to 0 by hardware.
1	TXBEFLG	R	Transmit Buffer Empty Flag This bit indicates that the transmit buffer is empty or not 0: Not empty 1: Empty
0	RXBNEFLG	R	Receive Buffer Not Empty Flag This bit indicates that the receive buffer is empty or not 0: Empty

Field	Name	R/W	Description
			1: Not empty

17.6.4 SPI data register (SPI_DATA)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	DATA	R/W	Transmit Receive Data register Store the data to be transmitted or received. When writing this register, the data will be written to the transmit buffer; when reading this register, the data in receive buffer will be read. The size of the buffer 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.

18 Analog-to-digital converter (ADC)

18.1 Introduction

ADC with 12-bit precision has 14 channels, including 8+2 external channels (2 channels for internal and external multiplexing) and 6 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.

18.2 Main characteristics

(1) 12-bit resolution

(2) Up to 14 sampling channels

(3) Sampling intervals can be programmed per channel

(4) Data alignment with built-in data consistency

(5) Maximum conversion rate of 2 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_DATA register

(10) ADC internal channel supports:
- TS
- VBG
- 1/12 VBB
- VDD5
- OPAMP0
- OPAMP1

(11) Supports segmented sampling, with independent result registers for each conversion channel

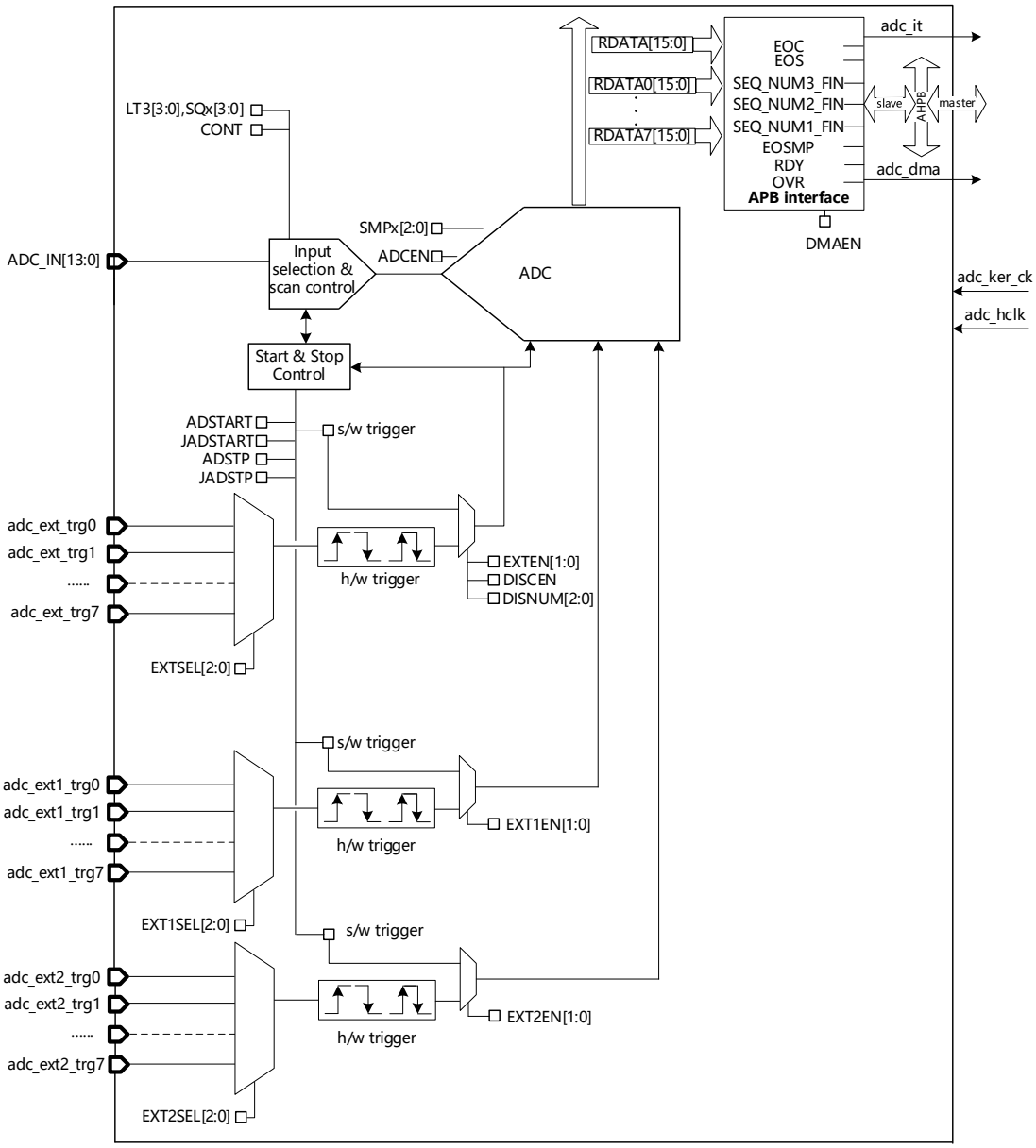
(12) Supports low-power modes, including automatic delay mode and automatic shutdown mode

(13) Interrupt

- End of conversion interrupt
- End of sequence conversion interrupt
- End of sampling phase interrupt
- ADC ready interrupt
- Overrun interrupt

18.3 Structure block diagram

Figure 72 Structure Block Diagram



18.4 ADC Functional Description

18.4.1 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).

18.4.2 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.

18.4.3 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.

18.4.4 ADC switch control

Setting the ADEN bit in the ADCx_CR register to 1 enables the ADC. Conversion can be started when the RDY flag of the ADC is set to 1.

Subsequently, regular conversion can be started by setting ADSTART to 1, or by an external trigger event if the trigger is enabled.

Procedure to disable the ADC by software:

(1) Check if ADSTART is 0 to ensure no conversion is currently being performed. If necessary, set ADSTP to 1, then wait for ADSTP=0 to stop any ongoing regular and injected conversions.

(2) If required by the application, clear ADEN to 0 to disable the ADC.

18.4.5 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 ADSTART and ADSTP bits of the ADC_CR register when the ADC is enabled.

For all other control bits in the ADC_CFGR1, ADC_CFGR2, ADC_CFGR3, ADC_SMPR1, ADC_SQR1, ADC_SQR2, ADC_SQR3, ADC_SEQ_NUM, 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.

18.4.6 Channel selection (ADC_SQRx)

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_SQR1, ADC_SQR2, and ADC_SQR3 registers. The total number of conversions in the regular conversion group must be written to the LT3[3:0] bits in the ADCx_SQ1R register.

The ADC_SQR1, ADC_SQR2, and ADC_SQR3 registers cannot be modified when a regular conversion may be in progress. Therefore, ADSTP=1 must be written first to stop ADC regular conversion.

- When the ADC selects channel 1100
  - If it is necessary to measure the output signal of OPAMP0 and the ADC needs to collect the OPAMP signal at the same time, configure the OPAMP0_EN, OPAMP0_IN_CTRL, and OPAMP0_OUT_CTRL in the SYS_OPAMPCR 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 SYS_OPAMPCR registers of the OPAMP module, with OPAMP0_EN, OPAMP0_IN_CTRL all set to 1 and OPAMP0_OUT_CTRL set to 0, to ensure the normal operation of OPAMP0.
  - If only the signal at the PA0 pin is measured, configure the SYS_OPAMPCR register of the OPAMP module to set OPAMP0_EN to 0 and OPAMP0_OUT_CTRL 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 the ADC needs to collect the OPAMP signal at the same time, configure the OPAMP1_EN, OPAMP1_IN_CTRL, and OPAMP1_OUT_CTRL of the SYS_OPAMPCR 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 SYS_OPAMPCR registers of the OPAMP module, with OPAMP1_EN, OPAMP1_IN_CTRL all set to 1 and OPAMP1_OUT_CTRL to 0, to ensure the normal operation of OPAMP1.
  - If only the signal at the PA5 pin is measured, configure the SYS_OPAMPCR register of the OPAMP module with OPAMP1_EN set to 0 and OPAMP1_OUT_CTRL set to 1, so that

OPAMP1 is turned off and the PA5 signal is available for ADC detection.

18.4.7 Configurable sampling time per channel (ADC_SMPR1)

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 SMP0[2:0] bits in the ADC_SMPR1 register, allowing independent sampling times for each channel.

The formula for total conversion time is as follows:

$$T_{CONV} = \text{Sampling time} + 13 \text{ ADC clock cycles}$$

18.4.8 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 ADSTART bit in the ADC_CR register to 1 (for regular channels)
- External hardware trigger event (for regular or injected channels)

In a regular sequence, after each conversion is completed:

- The conversion data is stored in the ADC_DR register
- The EOC (end of regular conversion) flag is set to 1
- An interrupt is generated when EOCIE is set to 1

After the regular sequence is completed:

- The EOS (end of regular sequence) flag is set to 1
- An interrupt is generated when EOSIE is set to 1

Subsequently, the ADC stops working until a new external regular or injected trigger occurs, or ADSTART is set to 1 again.

Note: To convert a single channel, program the sequence length to 1.

18.4.9 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 ADSTART 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 EOC (end of conversion) flag is set to 1
- An interrupt is generated when EOCIE is set to 1

After the conversion sequence is completed:

- The EOS (end of sequence) flag is set to 1
- An interrupt is generated when EOSIE 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.

18.4.10 Start conversion (ADSTART)

Software starts ADC regular conversion by setting ADSTART to 1.

After ADSTART 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 ADSTP program called by software

Note: In continuous mode, since the sequence restarts automatically, the ADSTART bit will not be cleared by hardware when EOS is set to 1.

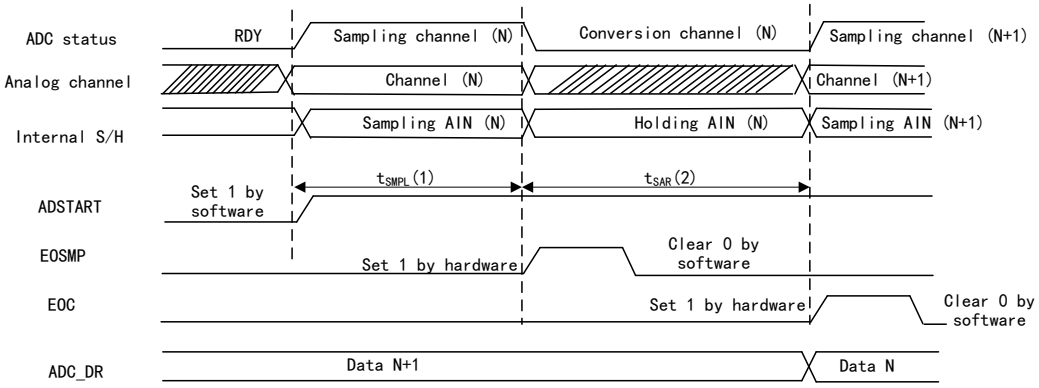
If hardware triggering is selected in single mode (CONT=0 and EXTEN is not equal to 00), the ADSTART bit will not be cleared by hardware when EOS is set to 1. This allows the software to avoid resetting ADSTART for the next hardware trigger event. This ensures that no subsequent hardware triggers are missed.

Note: When software triggering is selected, the ADSTART bit should not be set to 1 if the EOC flag is still high.

18.4.11 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 73 Analog-to-Digital Conversion Time



Note:

(1) t_{SMPL} depends on SMP[2:0]

(2) t_{SAR} is the conversion time: 13 ADC_CLK cycles

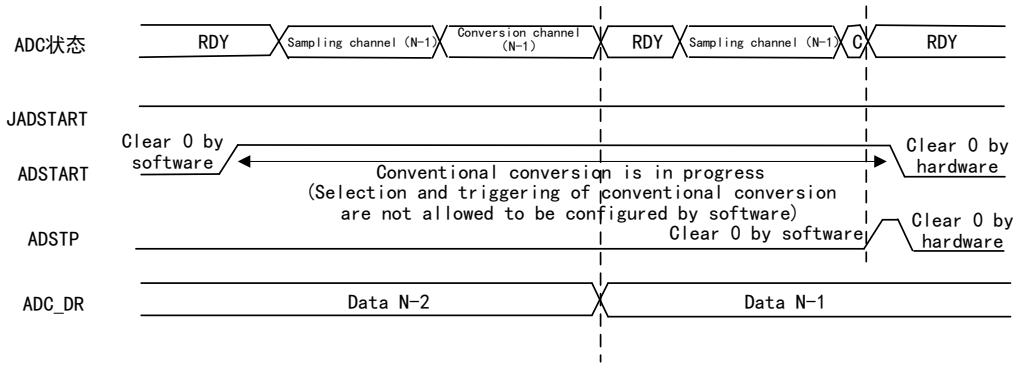
18.4.12 Stop ongoing conversion (ADSTP)

The software decides whether to stop the conversion. To stop an ongoing regular conversion, ADSTP 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 ADSTP 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 ADSTP/ADSTART bit will be cleared by hardware. The software must poll ADSTART until it is reset to determine that the ADC has completely stopped running.

Figure 74 Stop Ongoing Regular Conversion



18.4.13 External trigger conversion and trigger polarity (EXTSEL, 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 ADSTART bit to 1, the regular trigger selection becomes valid. Hardware triggers occurring during conversion will be ignored. If the ADSTART 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 55 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
11	Perform hardware trigger detection on falling edges and falling edges

The EXTSEL 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 and injected conversions.

Table 56 External Trigger of Regular Channel

Name	Source	Type	EXTSEL[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

18.4.14 Discontinuous mode (DISCEN, DISCNUM)

Regular group mode

This mode can be enabled by setting the DISCEN bit in the ADC_CFGR register to 1.

This mode is used to convert short sequences containing n (n $\leqslant$ 8) conversions, which are part of the conversion sequence selected in the ADC_SQRx register. The value of n can be specified by writing to the DISCNUM[2:0] bits in the ADC_CFGR register.

When an external trigger occurs, the next n conversions selected in the

ADC_SQRx register will start until all conversions in the sequence are completed. The total sequence length is defined by the LT3[3:0] bits in the ADC_SQR1 register.

Each conversion generates an EOC event, and the last conversion also generates an EOS 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.

18.4.15 End of conversion (EOC)

The ADC notifies the application each time a regular conversion end event occurs. The ADC immediately sets the EOC flag to 1 when new regular conversion data appears in the ADC_DR register. An interrupt can be generated if the EOCIE bit is set to 1. The EOC flag can be cleared by software writing 1 to it or reading the ADC_DR.

18.4.16 End of conversion sequence (EOS)

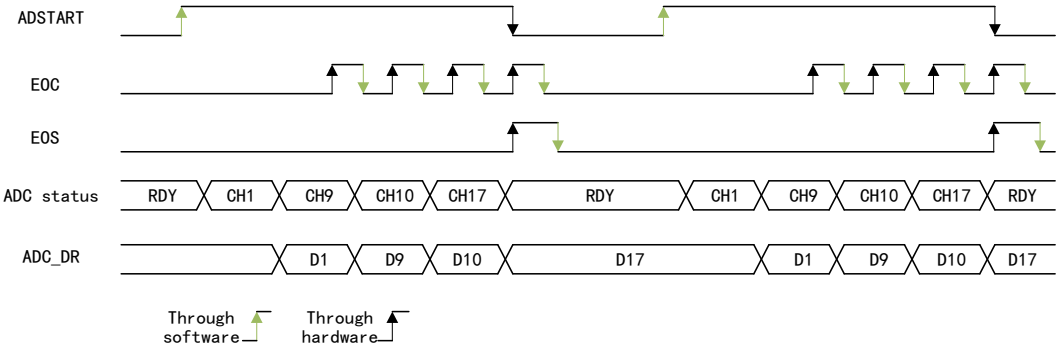
The ADC notifies the application each time a regular sequence end event occurs.

The ADC immediately sets the EOS 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 EOSIE bit is set to 1. The EOS flag can be cleared by software writing 1 to it.

18.4.17 Example timing diagrams (single mode/continuous mode, hardware/software trigger)

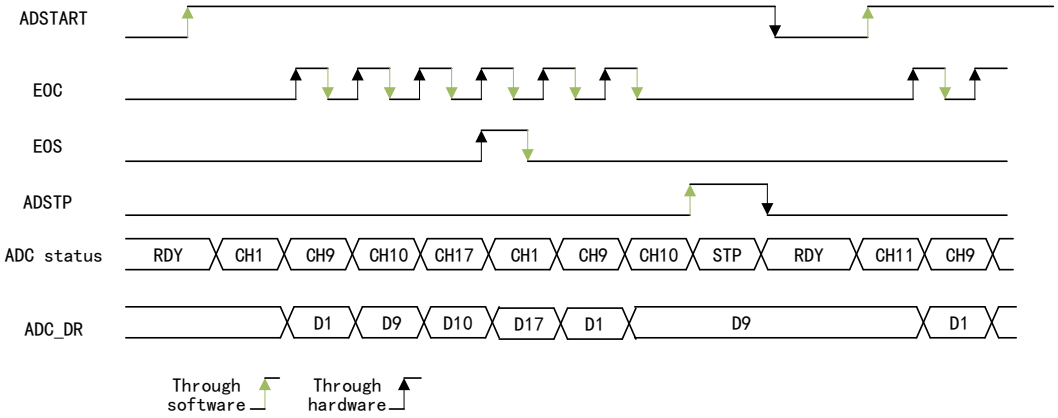
When EXTEN=00, CONT=0, and the selected channels=1, 9, 10, 17, the timing diagram is as follows:

Figure 75 Single sequence conversion, software trigger



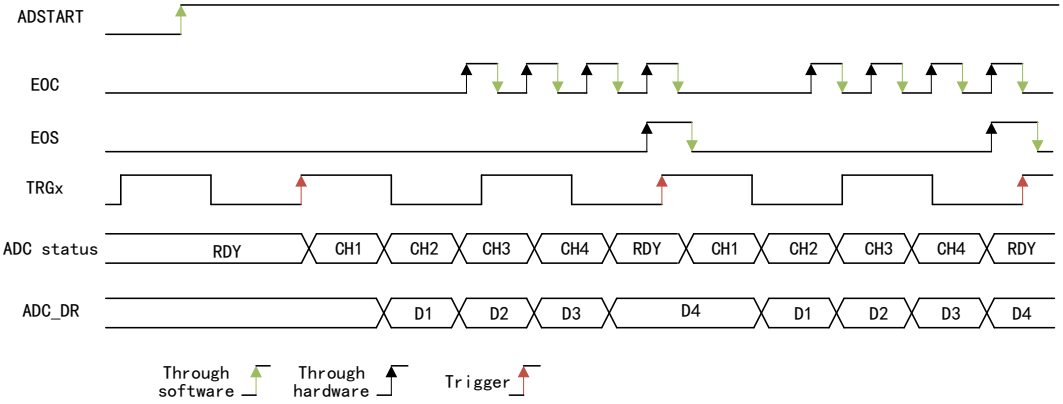
When EXTEN=00, CONT=1, and the selected channels=1, 9, 10, 17, the timing diagram is as follows:

Figure 76 Continuous sequence conversion, software trigger



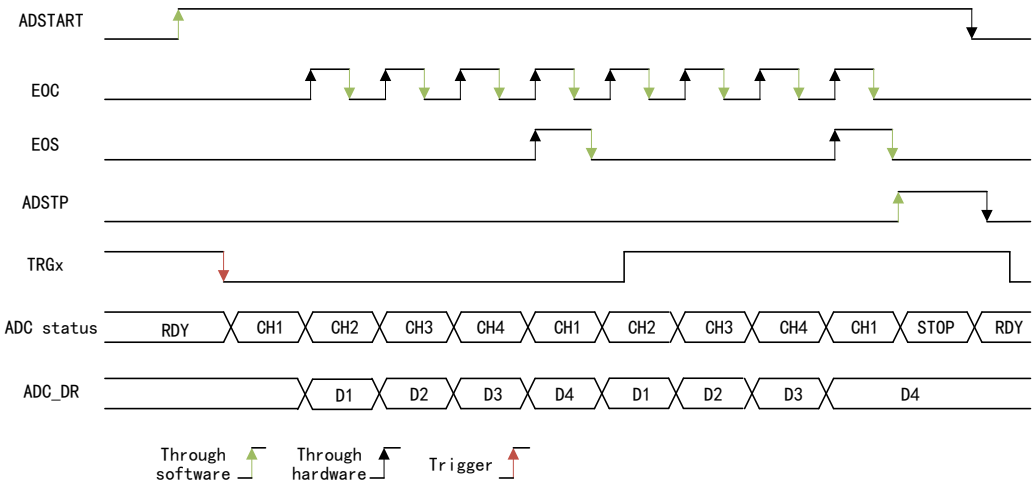
When TRGx (overfrequency) is selected as the trigger source, EXTEN=01, CONT=0, and the selected channels=1, 2, 3, 4, the timing diagram is as follows:

Figure 77 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 78 Continuous sequence conversion, hardware trigger



18.4.18 Data alignment

The ALIGN bit in the ADC_CFGR 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 79 Data right alignment

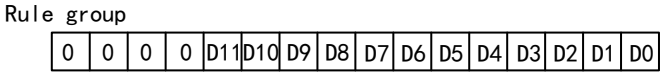
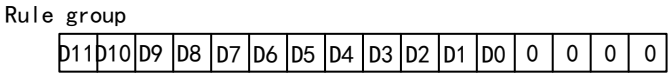


Figure 80 Data left alignment



The data of the sequential segmented sampling mode is stored in ADC_DATA(0-7) register and can be left or right-aligned.

18.4.19 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 ADSTP 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

18.4.20 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 EOC bit is 1 but another new conversion has been completed, an overrun event will occur, and OVR bit of register ADC_ISR will be set to 1; if OVRIE bit is set to 1, an overrun interrupt will be generated.

It is determined by OVRMOD bit of configuration register ADC_CFGR1 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_DATA register
- OVRMOD is set to 1: When an overrun event is detected, ADC_DATA register will overwrite the data by the last converted data

18.4.21 ADC (low-power mode)

Automatic delay conversion mode

The ADC executes the automatic delay conversion mode controlled by the AUTDLY 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 AUTDLY=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_CFGR1 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.

18.4.22 ADC interrupt

Interrupts are generated in the following cases:

- At the end of any conversion of the regular group (EOC flag)
- At the end of the conversion sequence of the regular group (EOS flag)
- Interrupt at the end of the first segment of sequence segmented sampling (SEQ_NUM1_FIN flag)
- Interrupt at the end of the second segment of sequence segmented sampling (SEQ_NUM2_FIN flag)
- Interrupt at the end of the third segment of sequence segmented sampling (SEQ_NUM3_FIN flag)
- Overrun interrupt (OVR flag)
- Interrupt at the end of regular conversion sampling (EOSMP flag)
- Enable ADC ready interrupt (ADY flag)

Separate interrupt enable bits can be used for flexibility.

Table 57 ADC Interrupt

Interrupt event	Event flag	Enable control bit
End of conversion of regular group	EOC	EOCIE
End of conversion sequence of regular group	EOS	EOSIE
End of the first segment of sequential segmented sampling	SEQ_NUM1_FIN	SEQ_NUM1_FIN IE
End of the second segment of sequential segmented sampling	SEQ_NUM2_FIN	SEQ_NUM2_FIN IE
End of the third segment of sequential segmented sampling	SEQ_NUM3_FIN	SEQ_NUM3_FIN IE
Data overrun of regular group	OVR	OVRIE
End of regular conversion sampling	EOSMP	EOSMPIE
Enable ADC ready	RDY	RDYIE

18.4.23 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) D_{sample}= Tensor voltage at the actual temperature

(3) Ts_{sample}= For the 25℃ voltage, please refer to the value at address 0x0010 0C14 for details

(4) slope= Average slope，slope=4.5 mV/ºC

18.4.24 Direction for use

The process of enabling ADC through software:

(1) Write "1" to the ADRDY bit in the ADCx_ISR register to clear it to zero.

(2) Set ADEN 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 ADCx_ISR register to zero it (optional).

The process of disabling ADC through software:

(1) Check whether both ADSTART and JADSTART are 0 to ensure that no conversion is currently being performed. If necessary, set ADSTP to 1 and JADSTP to 1, and then wait until ADSTP=0 and JADSTP=0 to stop any ongoing regular and injection conversions.

(2) Clear ADDEN 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 ADSTART position 1 in the ADCx_CR register (for regular channels)
- set JADSTART position 1 in the ADCx_CR register (for injection channel)
- External hardware trigger event (applicable to regular channels or injection channels)

In a regular sequence, after each conversion is completed:

- The conversion data is stored in the 16-bit ADCx_DR register

- Set the EOC (End of Regular Conversion) flag to 1
- An interruption will occur when EOCIE is at position 1

In the injection sequence, after each conversion is completed:

- The conversion data is stored in one of the four 16-bit ADCx_JDRy registers
- Set the JEOC (Injection Conversion Completed) flag to 1
- An interrupt will occur when JEOCIE is at position 1

After the regular sequence is completed:

- Set the EOS (End of Regular Sequence) flag to 1
- An interrupt will occur when EOSIE is at position 1

After the injection sequence is completed:

- Set the JEOS (End of Injection sequence) flag to 1
- An interrupt will occur when JEOSIE is at position 1

Subsequently, the ADC will cease operation until a new external routine or injection trigger occurs, or the ADSTART or JADSTART 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 ADSTART 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 EOC (Conversion Completed) flag to 1
- An interruption will occur when EOCIE is at position 1

After the conversion sequence is completed:

- Set the EOS (sequence End) flag to 1
- An interruption will occur when EOSIE is at position 1

Subsequently, the new sequence will be restarted immediately, and the ADC will continue to repeat the conversion sequence.

18.5 Register address mapping

Table 58 ADC Register Address Mapping Table

Register name	Description	Offset address
ADC_ISR	ADC status register	0x00
ADC_IER	ADC interrupt enable register	0x04
ADC_CR	ADC control register	0x08

Register name	Description	Offset address
ADC_CFGR1	ADC configuration register 1	0x0C
ADC_CFGR2	ADC configuration register 2	0x10
ADC_SMPR1	ADC sampling configuration register 1	0x14
ADC_CFGR3	ADC configuration register 3	0x1C
ADC_SQR1	ADC sequence configuration register 1	0x2C
ADC_SQR2	ADC sequence configuration register 2	0x30
ADC_SQR3	ADC sequence configuration register 3	0x34
ADC_DR	ADC regular conversion data register	0x38
ADC_DATAx	ADC data register x	0x3C+0x04*x (x=0~7)
ADC_SEQ_NUM	ADC sequential segmented sampling control register	0x5C
ADC_TMR	ADC test configuration register	0x84

18.6 Register functional description

18.6.1 ADC status register (ADC_ISR)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:11	Reserved		
10	SEQ_NUM3_FIN	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	SEQ_NUM2_FIN	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	SEQ_NUM1_FIN	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	OVR	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 EOC 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)

Field	Name	R/W	Description
			1: An overrun has occurred
3	EOS	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	EOC	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 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	EOSMP	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	RDY	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

18.6.2 ADC interrupt enable register (ADC_IER)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:11	Reserved		
10	SEQ_NUM3_FINIE	R/W	NUM.3 of Sequential Section Sampling Finish Interrupt Enable 0: Disable 1: Enable
9	SEQ_NUM2_FINIE	R/W	NUM.2 of Sequential Section Sampling Finish Interrupt Enable 0: Disable 1: Enable
8	SEQ_NUM1_FINIE	R/W	NUM.1 of Sequential Section Sampling Finish Interrupt Enable 0: Disable 1: Enable
7:5	Reserved		

Field	Name	R/W	Description
4	OVRIE	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 OVR bit is set to 1. Note: Software is only allowed to write to this bit when ADSTART=0 (this ensures that no regular conversion is currently in progress).
3	EOSIE	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. 0: EOS interrupt disabled 1: EOS interrupt enabled. An interrupt is generated when the EOS bit is set to 1. Note: Software is only allowed to write to this bit when ADSTART=0 (this ensures that no regular conversion is currently in progress).
2	EOCIE	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 EOC bit is set to 1. Note: Software is only allowed to write to this bit when ADSTART=0 (this ensures that no regular conversion is currently in progress).
1	EOSMPIE	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: EOSMP interrupt disabled 1: EOSMP interrupt enabled. An interrupt is generated when the EOSMP bit is set to 1. Note: Software is only allowed to write to this bit when ADSTART=0 (this ensures that no regular conversion is currently in progress).
0	RDYIE	R/W	ADC Ready Interrupt Enable 0: Disable 1: Enable

18.6.3 ADC control register (ADC_CR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31	VREF_SEL	R/W	VREF connect select 0: VREF connected to VDD5 1: VREF connected to IO input

Field	Name	R/W	Description
30:5	Reserved		
4	ADSTP	R/W	ADC stop of regular conversion command This bit is set to 1 by software to stop and discard the ongoing regular conversion (ADSTP 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 (ADSTART 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 ADSTP command is being executed. Note: Software is only allowed to set ADSTP to 1 when ADSTART=1 and ADDIS=0 (the ADC is enabled, will eventually perform regular conversion, and there are no pending requests to disable the ADC). In automatic injection mode (JAUTO=1), setting ADSTP to 1 will abort both regular and injected conversions (do not use JADSTP).
3	Reserved		
2	ADSTART	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 single conversion mode, if software trigger is selected (EXTSEL=0x0): cleared when the end of regular conversion sequence (EOS) flag appears. - In all cases: cleared by hardware when the ADSTP bit is cleared after the execution of the ADSTP 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 ADSTART to 1 when ADEN=1 and ADDIS=0 (the ADC is enabled, and there are no pending requests to disable the ADC). In automatic injection mode (JAUTO=1), setting ADSTART to 1 starts regular conversion and automatic injection conversion (JADSTART must remain cleared).
1	Reserved		
0	ADEN	R/W	ADC enable control This bit is set to 1 and cleared by software to enable the ADC. 0: ADC disabled (OFF state) 1: Write 1 to enable the ADC

18.6.4 ADC configuration register 1 (ADC_CFGR1)

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 001: 2 channels ... 110: 7 channels 111: 8 channels Note: Software is only allowed to write to these bits when ADSTART=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. Automatic injection mode and discontinuous mode cannot be used simultaneously: When JAUTO=1, the DISCEN and JDISCEN bits must be kept cleared by software. Software is only allowed to write to this bit when ADSTART=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 ADSTART=0 (to ensure no conversion is currently in progress).
14	AUTDLY	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 ADSTART=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 ADSTART=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.

Field	Name	R/W	Description
			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 ADSTART=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 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 ADSTART=0 (to ensure no regular conversion is currently in progress).
9	Reserved		
8:6	EXTSEL1	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 ADSTART=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 ADSTART=0 and JADSTART=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 ADSTART=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 ADSTART=0 (to ensure no regular conversion is currently in progress).

18.6.5 ADC configuration register 2 (ADC_CFGR2)

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	SEQE	R/W	Sequential Section Sampling Enable 0: Disable 1: Enable

18.6.6 ADC sampling configuration register (ADC_SMPR1)

Offset address: 0x14

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:6	Reserved		
5:3	SMP1_SLOW	R/W	Channel x sampling time selection (for slow channels ADC_IN8-ADC_IN11) 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: 128 ADC clock cycles 001: 160 ADC clock cycles 010: 192 ADC clock cycles 011: 224 ADC clock cycles 100: 256 ADC clock cycles Others: Reserved, handled as 128 ADC clock cycles Note: Software is only allowed to write to these bits when ADSTART=0 (to ensure no conversion is currently in progress).
2:0	SMP1_FAST	R/W	Channel x sampling time selection (for fast channels) 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: 3 ADC clock cycles 001: 8 ADC clock cycles 010: 16 ADC clock cycles 011: 32 ADC clock cycles 100: 64 ADC clock cycles

Field	Name	R/W	Description
			101: 96 ADC clock cycles 110: 128 ADC clock cycles 111: 256 ADC clock cycles Note: Software is only allowed to write to these bits when ADSTART=0 (to ensure no conversion is currently in progress).

18.6.7 ADC configuration register 3 (ADC_CFGR3)

Offset address: 0x1C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:13	Reserved		
12:10	EXTSEL3	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 ADSTART=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 ADSTART=0 (to ensure no regular conversion is currently in progress).
7:5	Reserved		
4:2	EXTSEL2	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 ADSTART=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)

Field	Name	R/W	Description
			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 ADSTART=0 (to ensure no regular conversion is currently in progress).

18.6.8 ADC sequence configuration register 1 (ADC_SQR1)

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 ADSTART=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 ADSTART=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 ADSTART=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 ADSTART=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 ADSTART=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 ADSTART=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: select vin0 - PA7 0001: choose vin1 - PA8 0010: choose vin2 - PB5 0011: choose vin3 - PB6 0100: choose vin4 - PB1

Field	Name	R/W	Description
			0101: Choose vin5 - PB2 0110: select vin6 - PB3 0111: choose vin7 - PB4 1000: choose vin8 - TS 1001: choose vin9 - VBG 1010: Choose vin10 VBB 1011-1/12: choose vin11-1/4 VDD5 1100: choose vin12 - OPAMP0 1101: choose vin13 – OPAMP1 Note: Software is only allowed to write to these bits when ADSTART=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 the ADC needs to collect the OPAMP signal at the same time, configure the OPAMP0_EN, OPAMP0_IN_CTRL, and OPAMP0_OUT_CTRL in the SYS_OPAMPCR 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 SYS_OPAMPCR registers of the OPAMP module, with OPAMP0_EN, OPAMP0_IN_CTRL all set to 1 and OPAMP0_OUT_CTRL set to 0, to ensure the normal operation of OPAMP0. - If only the signal at the PA0 pin is measured, configure the SYS_OPAMPCR register of the OPAMP module to set OPAMP0_EN to 0 and OPAMP0_OUT_CTRL 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 the ADC needs to collect the OPAMP signal at the same time, configure the OPAMP1_EN, OPAMP1_IN_CTRL, and OPAMP1_OUT_CTRL of the SYS_OPAMPCR 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 SYS_OPAMPCR registers of the OPAMP module, with OPAMP1_EN, OPAMP1_IN_CTRL all set to 1 and OPAMP1_OUT_CTRL to 0, to ensure the normal operation of OPAMP1. - If only the signal at the PA5 pin is measured, configure the SYS_OPAMPCR register of the OPAMP module with OPAMP1_EN set to 0 and OPAMP1_OUT_CTRL set 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 ADSTART=0 (to ensure no regular conversion is currently in progress).

18.6.9 ADC sequence configuration register 2 (ADC_SQR2)

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 ADSTART=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 ADSTART=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 ADSTART=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 ADSTART=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 ADSTART=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 ADSTART=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 ADSTART=0 (to ensure no regular conversion is currently in progress).

18.6.10 ADC sequence configuration register 3 (ADC_SQR3)

Offset address: 0x34

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:8	Reserved		

Field	Name	R/W	Description
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 ADSTART=0 (to ensure no regular conversion is currently in progress).
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 ADSTART=0 (to ensure no regular conversion is currently in progress).

18.6.11 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.

18.6.12 ADC data register 0 (ADC_DATA0)

Offset address: 0x3C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA0	R	Sequence segmented sampling result data 0

18.6.13 ADC data register 1 (ADC_DATA1)

Offset address: 0x40

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA1	R	Sequence segmented sampling result data 1

18.6.14 ADC data register 2 (ADC_DATA2)

Offset address: 0x44

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA2	R	Sequence segmented sampling result data 2

18.6.15 ADC data register 3 (ADC_DATA3)

Offset address: 0x48

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA3	R	Sequence segmented sampling result data 3

18.6.16 ADC data register 4 (ADC_DATA4)

Offset address: 0x4C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA4	R	Sequence segmented sampling result data 4

18.6.17 ADC data register 5 (ADC_DATA5)

Offset address: 0x50

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA5	R	Sequence segmented sampling result data 5

18.6.18 ADC data register 6 (ADC_DATA6)

Offset address: 0x54

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA6	R	Sequence segmented sampling result data 6

18.6.19 ADC data register 7 (ADC_DATA7)

Offset address: 0x58

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:16	Reserved		
15:0	RDATA7	R	Sequence segmented sampling result data 7

18.6.20 ADC sequential segmented sampling control register (ADC_SEQ_NUM)

Offset address: 0x5C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:18	Reserved		
17:16	SG_NUM	R/W	The Number of Sequencial Section 00: 1 section

Field	Name	R/W	Description
			01: 2 sections 10: 3 sections 11: Invalid
15:11	Reserved		
10:8	SEQ_NUM3	R/W	Section3 Transmission Time Setup 000: Transmit 1 time 001: Transmit 2 times ..... 101: Transmit 6 times Others: Invalid
7	Reserved		
6;4	SEQ_NUM2	R/W	Section2 Transmission Time Setup 000: Transmit 1 time 001: Transmit 2 times ..... 110: Transmit 7 times Others: Invalid
3	Reserved		
2:0	SEQ_NUM1	R/W	Section1 Transmission Time Setup 000: Transmit 1 time 001: Transmit 2 times ..... 111: Transmit 8 times

18.6.21 ADC test configuration register (ADC_TMR)

Offset address: 0x84

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	CALC_BYPASS	R/W	ADC calibration value bypass 0: Output calibration value 1: Output original conversion result

19 Comparator (COMP)

19.1 Full Name and Abbreviation Description of Terms

Table 59 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

19.2 Introduction

Two independent general-purpose comparators (COMP0 and COMP1) are embedded in SoC, and they can be used in combination with the timer.

19.3 Main characteristics

(1) Each comparator has configurable positive and negative inputs for flexible voltage selection;
- Positive input: 3 IOs, 1/4, 2/4, 3/4, 1 of BG voltage
- Negative input: COMP0 has 5 IOs (COMP1 has 3 IOs), 2 amplifier outputs

(2) Programmable hysteresis
- Positive hysteresis
- Hysteresis voltage: no hysteresis, 20mV, 40mV, 80mV

(3) Digital filtering: 8 levels , 10 levels

(4) Output phase-inverting

(5) Comparator interrupt

(6) Output as PWM brake trigger signal

19.4 Structure block diagram

Figure 81 COMP0 Structure Block Diagram

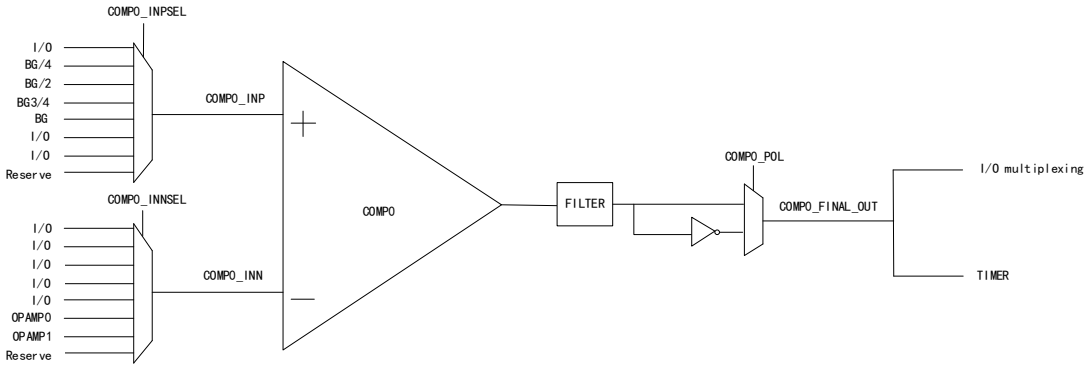
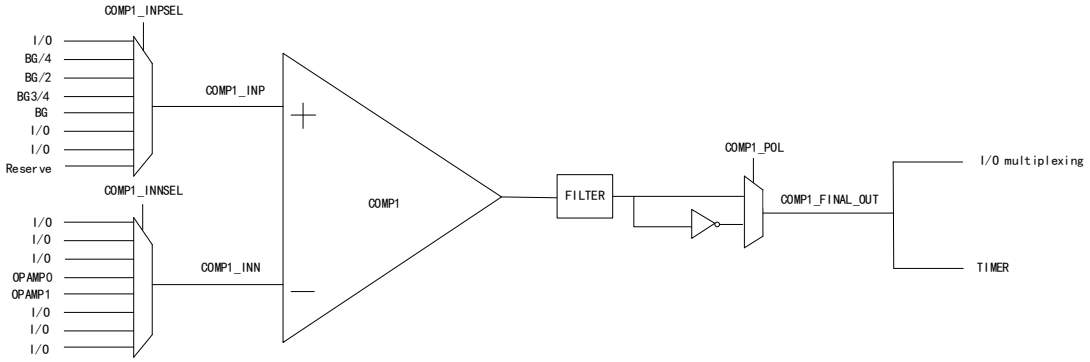


Figure 82 COMP1 Structure Block Diagram



19.5 Functional description

19.5.1 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.

19.5.2 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.

19.5.3 COMP interrupt

The comparator output internally generates an interrupt event.

The COMPx interrupt can be enabled by following these steps:

(1) Configure and enable the interrupt mode corresponding to the COMPx 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 COMPx

19.5.4 Direction for use

(1) Configure the COMP_CSTSx register and set the LOCK value, CFG value, EN_LV value, VP_EN[2:0], VN_EN[2:0], and HYS_EN[1:0];

(2) Configure the COMP_CSTSx register and set the COMP_TMR_SEL[2:0], POL value, R_EN value, F_EN value, RF_EN value, and SWINT value.

19.6 Register address mapping

Table 60 COMP Register Address Mapping

Register name	Description	Offset address
COMP_CSTS0	Control status register 0	0x00
COMP_CSTS1	Control status register 1	0x04
COMP_STS0	Status register 0	0x08
COMP_STS1	Status register 1	0x0C

19.7 Register functional description

19.7.1 Control status register x (COMP_CSTSx)

Offset address: 0x00+0x04*x (x=0~1)

Reset value: 0xC000 0000

Field	Name	R/W	Description
31	LOCK	R/W	COMP_CSTSx 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_CSTSx [31:0]. 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	VALUE	R	Comparator output flag This read-only flag reflects the level value of the comparator output before the polarity selector and masking.

Field	Name	R/W	Description
29	Reserved		
28:26	PSC	R/W	Digital filtering clock frequency division configure 000: 0 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128
25:22	CFG	R/W	Digital filtering counting cycle configuration 0000: 0 0001: 2 0010: 4 0011: 8 0100: 16 0101: 32 0110: 64 0111: 128 1000: 256 1001: 512 1010~1111: Reserved
21	SWINT	R/W	Software interrupt Set to 1 by software, write 1 and set to 0 on STS; When this bit is 0, writing 1 will set STS and generate an interrupt. 0: No effect 1: Software generates an interrupt
20	RF_EN	R/W	Comparator interrupt enables both the rising and falling edges simultaneously 0: Disable 1: Enable
19	F_EN	R/W	Comparator interrupts the falling edge enable 0: Disable 1: Enable
18	R_EN	R/W	Comparator interrupts the rising edge enable 0: Disable 1: Enable
17:16	Reserved		
15:14	HYS_EN	R/W	Comparator hysteresis enable Default value is 11. 00: Disable 01: 1st level hysteresis (20mV) 10: 2nd level hysteresis (40mV)

Field	Name	R/W	Description
			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.
12:10	COMPx_TMR_SEL	R/W	TMR channel Select 000: No connection to TMR 001: ATMR_BKIN 010: GTMR capture channel 0 011: GTMR capture channel 1 100: GTMR capture channel 2 101: GTMR capture channel 3 110: GTMR_ETR
9:7	VP_EN	R/W	Comparator positive input voltage select All switches are disabled by default. 1 IO, 1/4, 2/4, 3/4, 1 of BG voltage. COMP0: 000: PA5 001: BG/4 010: BG/2 011: BG3/4 100: BG 101: PA7 110: PB7 111: Reserved COMP1: 000: PA0 001: BG/4 010: BG/2 011: BG3/4 100: BG 101: PB3 110: PB8 111: Reserved
6:4	VN_EN	R/W	Comparator negative terminal input voltage select All switches are disabled by default. COMP0 has 5 IOs, COMP1 has 3 IOs, 2 amplifier outputs. COMP0: 000: PA9 001: PA8 010: PB0 011: PB1 100: PB2 101: OPAMP0_OUT

Field	Name	R/W	Description
			110: OPAMP1_OUT 111: Reserved COMP1: 000: PA1 001: PA2 010: PA5 011: OPAMP0_OUT 100: OPAMP1_OUT 101: PB4 110: PB5 111: PB6
3:1	Reserved		
0	EN_LV	R/W	Comparator enable 0: Disable 1: Enable

19.7.2 Status Register x (COMP_STSx)

Offset address: 0x08+0x04*x (x=0~1)

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:1	Reserved		
0	STS	RW1C	Interrupt status Set to 1 by hardware when an edge trigger request corresponding to R_EN/F_EN/RF_EN occurs, 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.

20 Operational Amplifier (OPAMP)

20.1 Introduction

The SoC is embedded with two independent operational amplifiers (OPAMP0 and OPAMP1), which can be used in combination with COMP and ADC.

20.2 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

20.3 Structure Block Diagram

Figure 83 OPAMP Structure Block Diagram

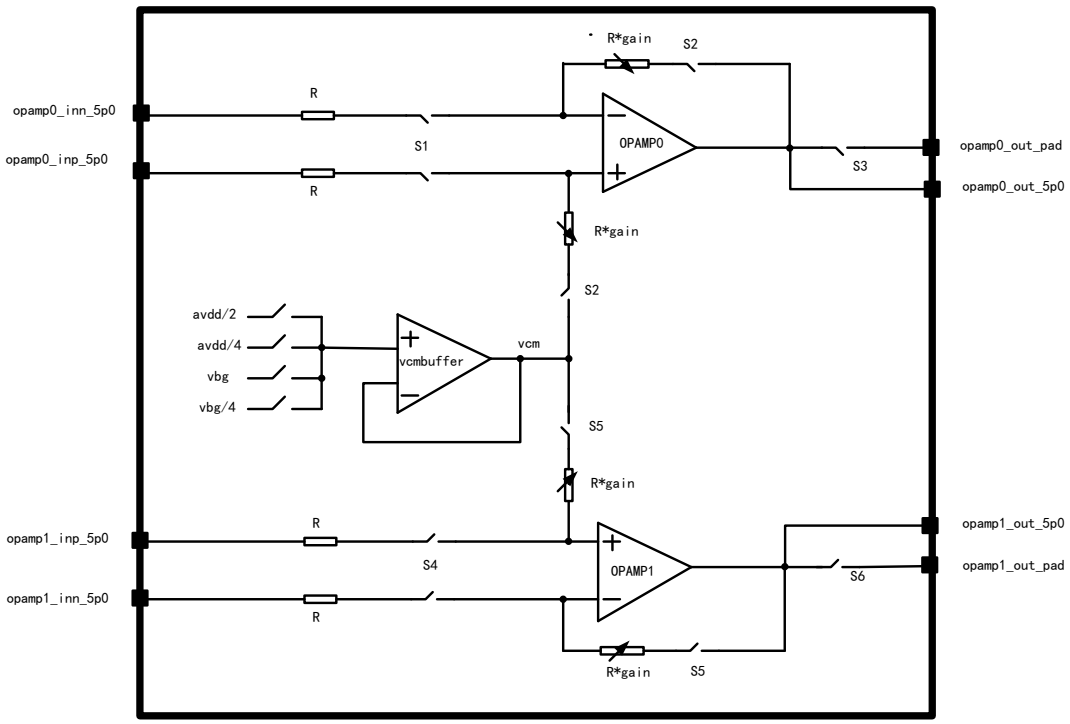


Figure 84 Internal gain application diagram

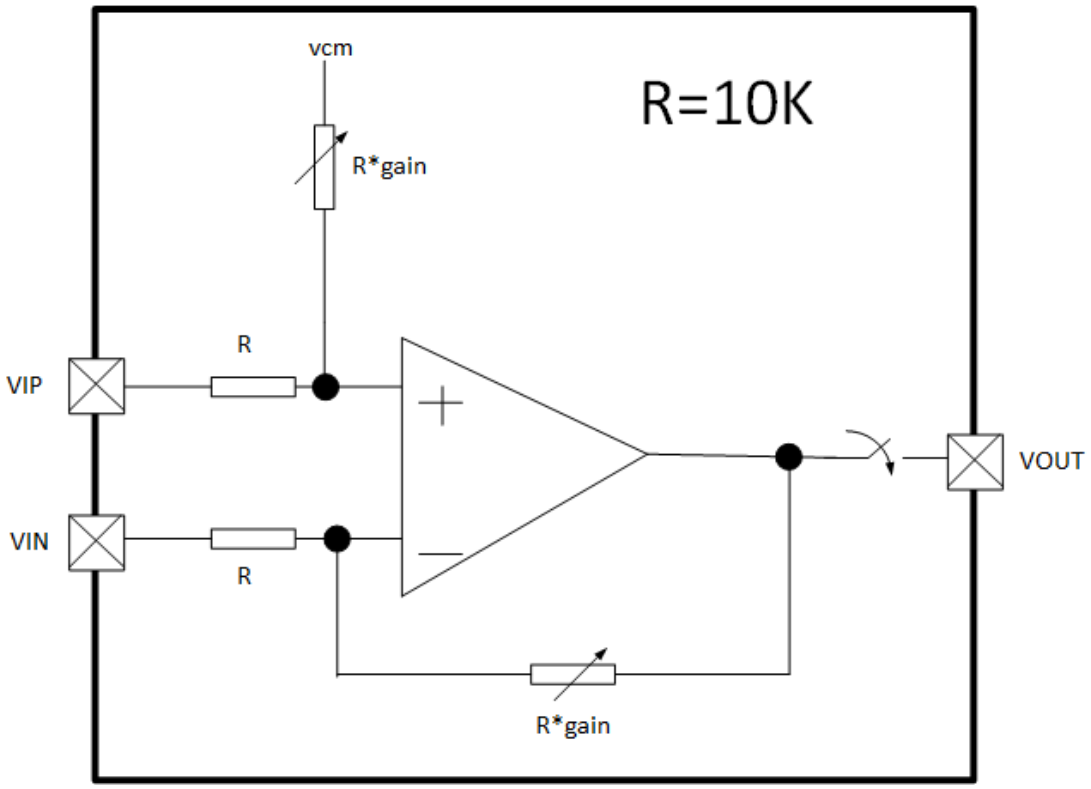
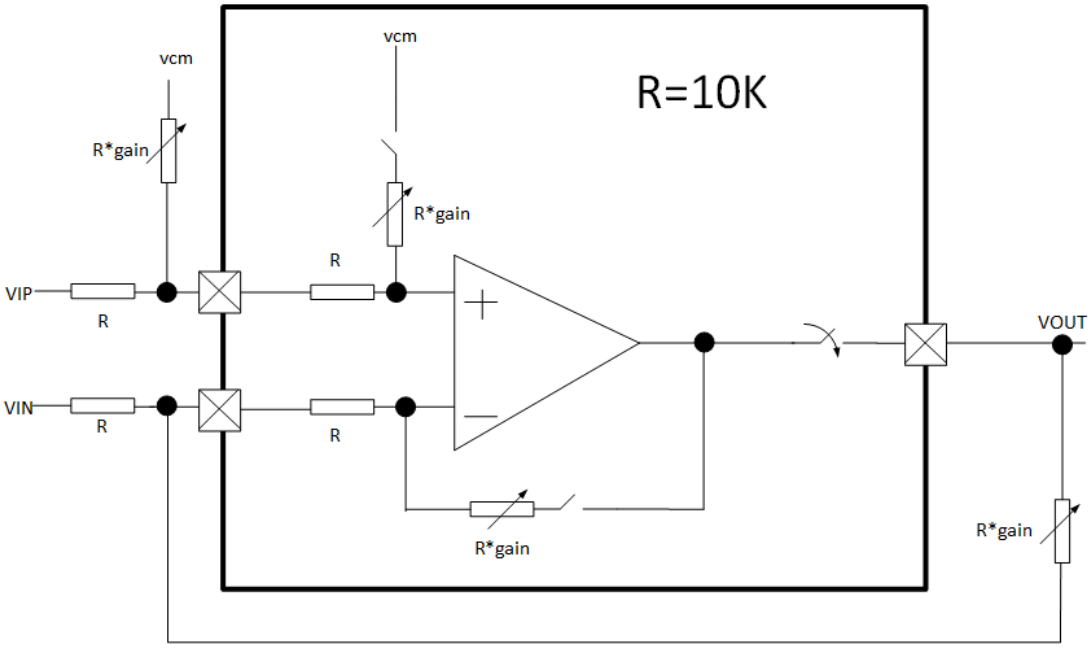


Figure 85 External gain application diagram



20.4 Register functional description

For the description of register functions, please refer to Section 4.2.12.

21 Cyclic redundancy check computing unit (CRC)

21.1 Introduction

The cyclic redundancy check (CRC) computing unit can get 8/16/32-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.

21.2 Main characteristics

(1) Process 8-bit, 16-bit and 32-bit data

(2) Supports CRC16-CCITT and CRC32

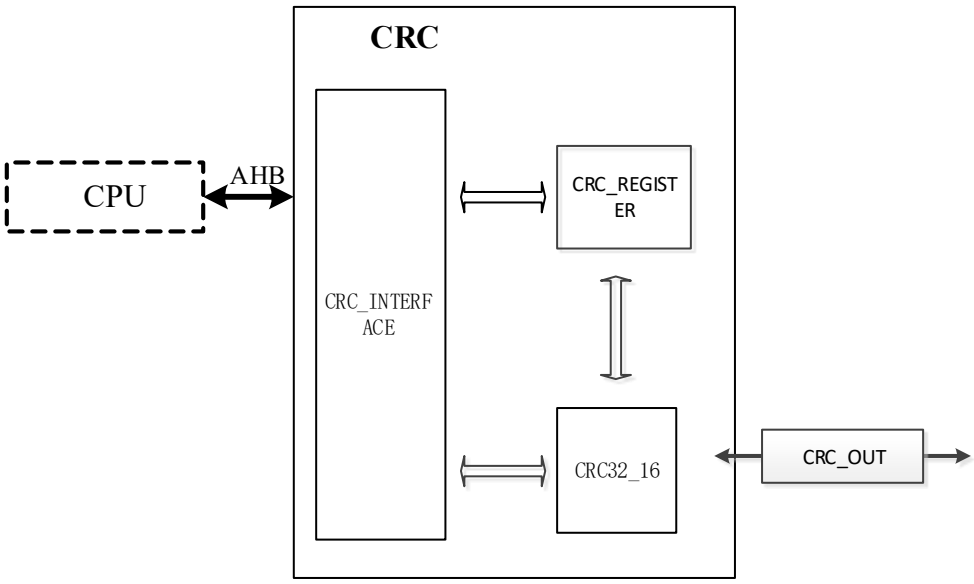
(3) Programmable CRC initial value

(4) 32-bit data register

(5) 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)

21.3 Structure block diagram

Figure 86 Structure Block Diagram



21.4 Functional description

21.4.1 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: ($X^{16}+X^{12}+X^{5}+1$)

21.4.2 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
- When processing 8-bit data, the calculation time is 1 AHB clock cycle

21.4.3 Functional characteristics

CRC unit contains a 32-bit read/write register CRC_DATA, 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 8-bit, 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.

21.4.4 Direction for use

(1) Configure the CRC_DATA register and set the data to be verified

(2) Configure the CRC_CTRL register, set CRC32 or CRC16, input data flip form and output data

(3) Configure the CRC_INITVAL register and set the initial value of CRC

21.5 Register address mapping

Table 61 CRC Register Address Mapping

Register name	Description	Offset address
CRC_DATA	Data register	0x00
CRC_CTRL	Control register	0x04
CRC_INITVAL	CRC initial value register	0x08

21.6 Register functional description

21.6.1 Data register (CRC_DATA)

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.

21.6.2 Control register (CRC_CTRL)

Offset address: 0x04

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:4	Reserved		
3	REVO	R/W	Output Data Reverse 0: Not reverse 1: Reverse
2:1	REVI	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

21.6.3 CRC initial value register (CRC_INITVAL)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	VALUE	R/W	Initial CRC Value The CRC initial value is programmable, and this bit is used to set the initial value of CRC.

22 Divider (DIV)

22.1 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.

22.2 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

(6) Signed or unsigned integer division operation

22.3 Register address mapping

Table 62 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_CR	Control register	0x14

22.4 Register functional description

22.4.1 Dividend Register (DIV_DVDR)

Offset address: 0x00

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	DIVIDEND	R/W	Dividend data

22.4.2 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.

22.4.3 Quotient Register (DIV_QUOTR)

Offset address: 0x08

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	QUOTIENT	R	Quotient data

22.4.4 Remainder Register (DIV_RMDR)

Offset address: 0x0C

Reset value: 0x0000 0000

Field	Name	R/W	Description
31:0	REMAINDER	R/W	Remainder data

22.4.5 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

23 Chip electronic signature

23.1 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 128-bit (or 96-bit can be composed of UID0 to UID2) product unique identity identifier UID is unique to any SoC under any circumstances. Under no circumstances can the user read this identity identifier and cannot modify it.

The storage space address of UID0 is: 0x00100E88

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: 0x00100E8C

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: 0x00100E90

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.

The storage space address of UID3 is: 0x00100E94

Field	Name	R/W	Description
31:0	UID3	R	The [127:96] bit position of the unique identity marker of the product is assigned at the time of factory shipment.

23.2 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: 0x00100D80

Field	Name	R/W	Description
31:16	Reserved		

Field	Name	R/W	Description
15:0	CHIPID	R	chip version:0x4401

24 Gate Driver

24.1 Introduction

Integrated three-phase 36V gate driver capable of driving P/NMOS power transistors, integrated 5V LDO to provide power for the internal SoC. It is equipped with multiple built-in protection functions, including under-voltage protection. The under-voltage interrupt function (LVD) can be set, and the under-voltage flag bit is triggered when the VBB is lower than 3.4V. Input direct protection: When the inputs are of the same high or low, HO and LO will turn off the external P/NMOS power transistors. It also has a dead time of 240ns to effectively ensure the normal operation of the system. The four adjustable output slopes of HO/LO can be selected through system configuration.

25 Revision history

Table 63 Document Revision History

Date	Version	Revision History
December,2025	1.0	• initial version
April, 2026	1.1	• Modified the"System Reset" reset circuit diagram and description • Delete the sampling accuracy of the main features of the ADC module • Modified the formula of the temperature sensor • Modified the bit25:22 naming of "COMP_CSTSx" • Delete the bit0 content of "COMP_CSTSx"
June, 2026	1.2	• Modified Pre-Drive to Gate driver • Modified the "SYS_PREDRVCR" register • Modified the content of bit7:4 in "ADC sequence Configuration Register 1"

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.

©2026 Zhuhai Geehy Semiconductor Co., Ltd. All Rights Reserved