LogicGreen Technologies Co., LTD

8-bit LGT8XM

RISC Microcontroller with
In-System Programmable
FLASH Memory

LGT8F88P

LGT8F168P

LGT8F328P

Data book
Version 1.0.5

Application field

Home appliances

Motor drive

Automatic control

Functional Overview

z **High performance and low power consumption 8 Bit LGT8XMKernel**

z **advanced RISC Framework**

131 Instructions,80%The above is a single cycle execution

32x8 General working register

32MHz Up to when working 32MIPS Execution efficiency

Internal single cycle multiplier (8x8)

z **Non-volatile program and data storage space**

32Kbytes On-chip online programming **FLASH** Program memory

2Kbytes Internal data **SRAM**

Programmable E2PROMAnalog interface, support byte access

Brand new program encryption algorithm to ensure user code security

z **Peripheral Controller**

Two with independent prescaler 8 Bit timer, support compare output mode

Two with independent prescaler 16 Bit timer, support input capture and compare output

internal 32KHz Can be calibrated **RC** Oscillator realizes real-time counter function

Can support up to 9 road PWMOutput, three groups of complementary programmable dead zone control

12 aisle 12 Bit high-speed analog-to-digital converter (**ADC)**

- Optional internal and external reference voltage
- Programmable gain (X1/8/16/32)Differential amplifier input channel
- automatic threshold voltage monitoring mode

Two analog comparators (**AC)**, Support comes from ADC Expansion of input channels

internal 1.024V/2.048V/4.096V ±1%Calibrable reference voltage source

One 8 Bit programmable **DAC**, Can be used to generate a reference voltage

source programmable watchdog timer (**WDT**)

Programmable synchronous/asynchronous serial interface (**USART/SPI**)

Synchronous Peripheral Interface (**SPI),** Programmable master/slave working

mode two-wire serial interface (**TWI),** compatible **I2C** Master-slave mode

16 Bit digital operation acceleration unit (**DSC)**, Supports direct 16 Bit data access

z **Special processor function**

SWD Two-wire on-chip debugging/mass production interface

External interrupt source and I/O Interrupt-on-change support

Built-in power-on reset circuit (**POR**) And programmable low-voltage detection circuit (**LVD**) Built-in 1%Can be calibrated 32MHz RC Oscillator, support frequency multiplication output built-in 1%

Can be calibrated 32KHz RC Oscillator external support 32.768KHz as well as 400K~32MHz

Crystal input

6x High current push-pull drive IO, Support high speed PWMapplication

z **I/O And package:**QFP48/32L, SSOP20L

z **Lowest power consumption:** 1uA@3.3V

z **working environment**

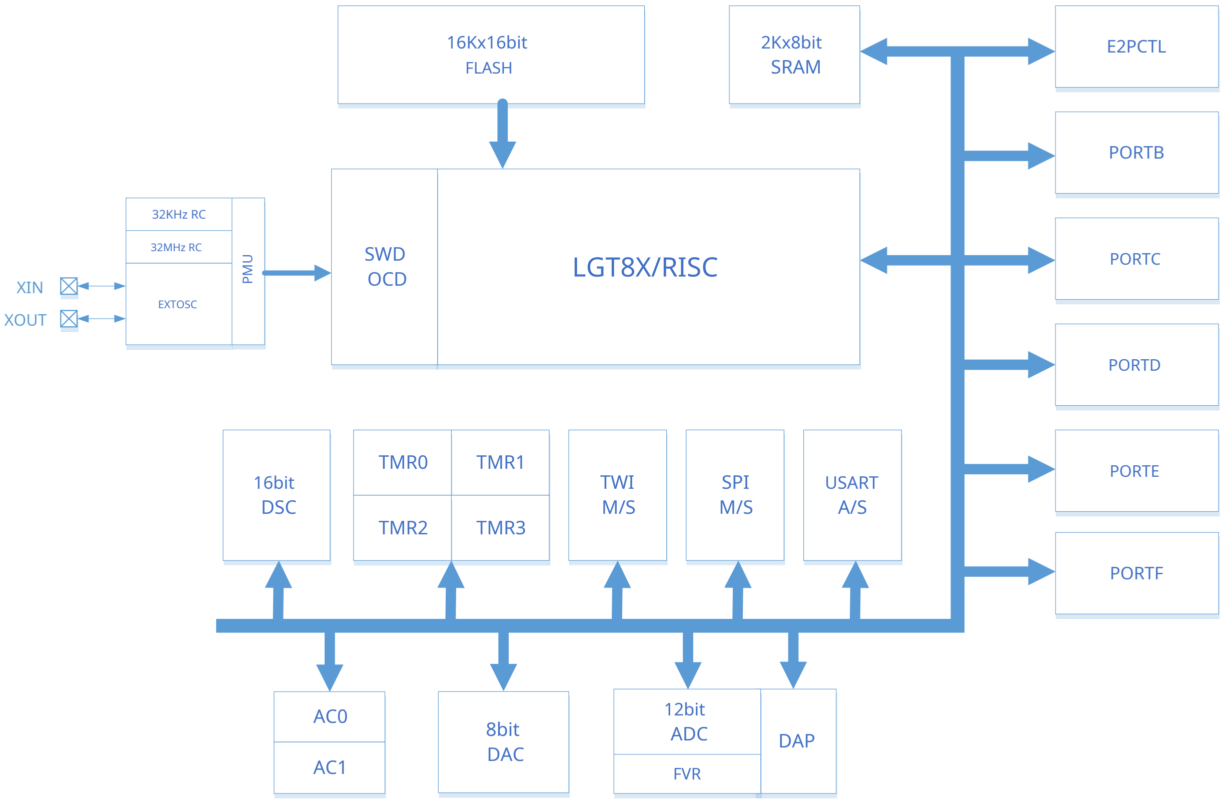
Operating Voltage: 1.8V ~ 5.5V

working frequency: 0 ~ 32MHz

Operating temperature: -40C

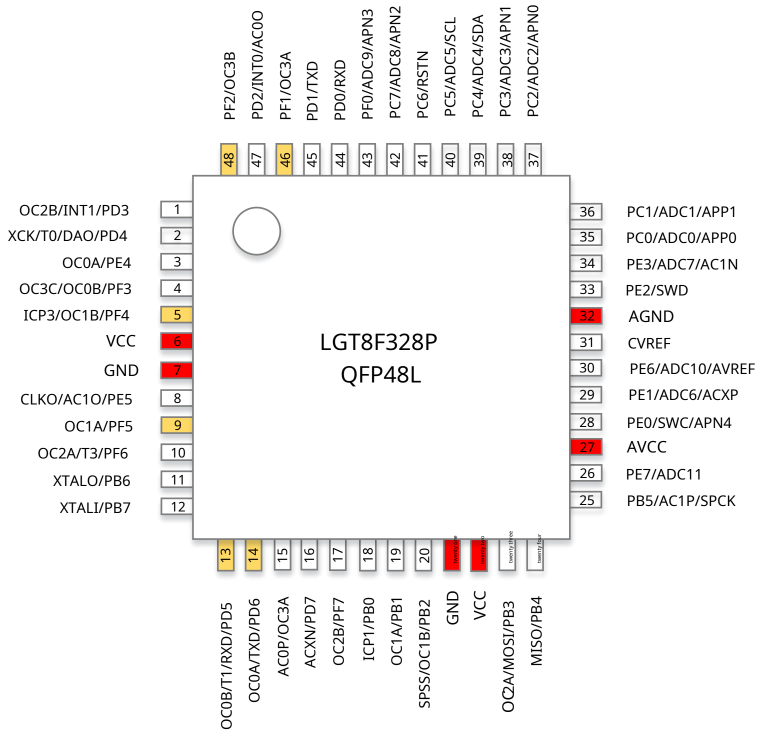
~ +85C HBM ESD:> 4KV

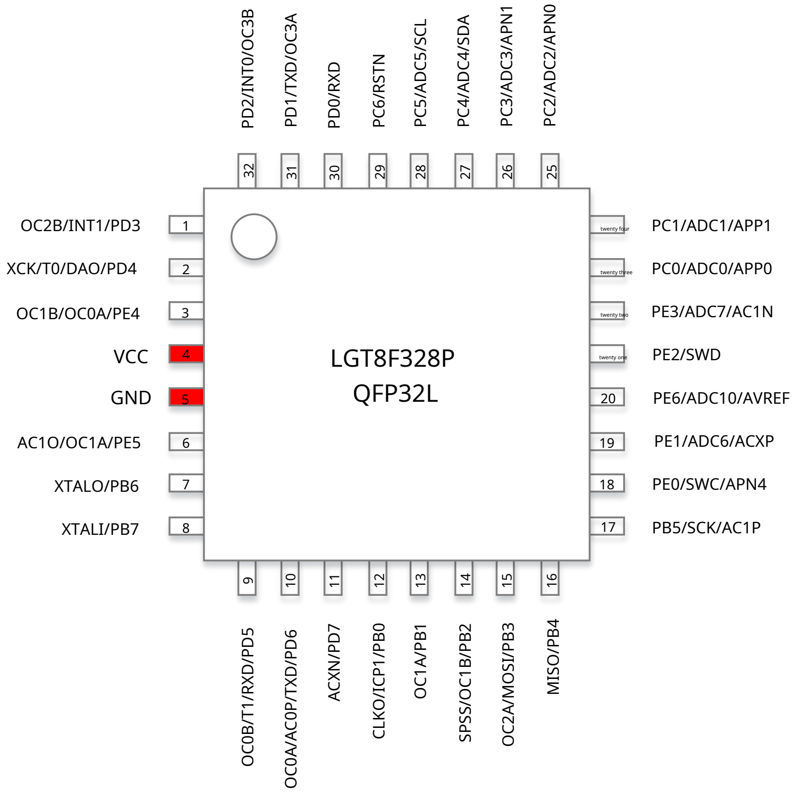
system framework

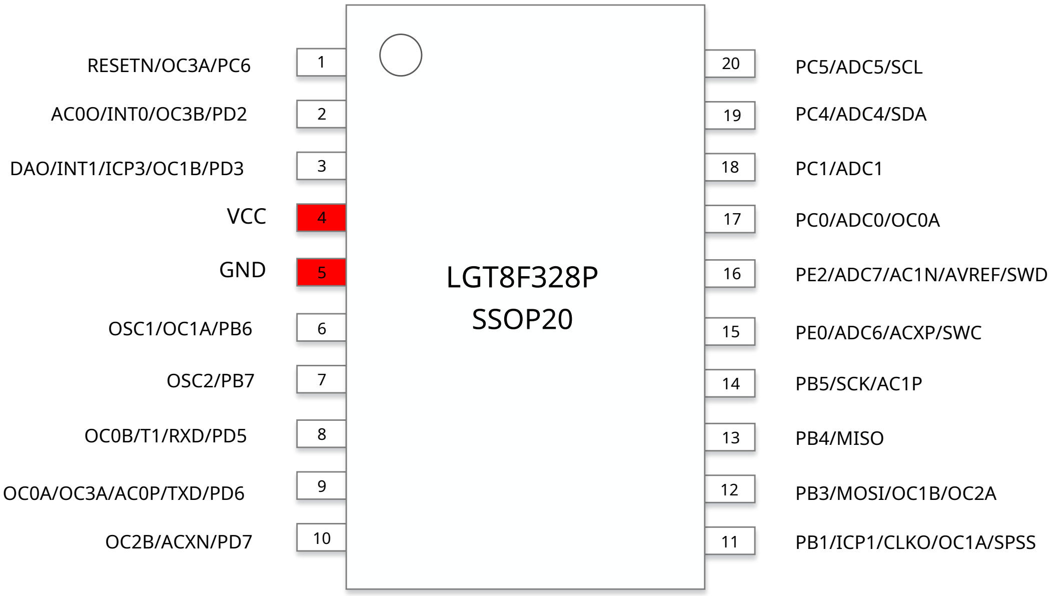


Module name	Module function
SWD	Debug module, at the same time realize online debugging and ISP Features
LGT8X	8bit high performance RISC Kernel data
E2PCTL	FLASH Access interface controller
PMU	Power consumption management module, responsible for managing the transition between the working states of the system.
PORTB/C/D/E/F	Universal programmable input and output ports
DSC	16 Bit digital operation acceleration unit
ADC	8 aisle 12 Bit analog to digital converter
DAP	Programmable gain differential amplifier
IVREF	1.024V/2.048V/4.096V Internal reference
AC0/1	analog comparator
TMR0/1/2/3	8/16 Bit timer/counter,PWMController
WDT	watchdog reset module
SPI M/S	Master-slave SPI Controller
TWI M/S	Master-slave two-wire interface controller, compatible I2C protocol
USART	Synchronous/asynchronous serial transceiver
DAC	8 Digital-to-analog converter

Package definition






LGT8F328P
SSOP20
RESETN/OC3A/PC6
1
AC0O/INT0/OC3B/PD2
2
DAO/INT1/ICP3/OC1B/PD3
3
VCC
4
GND
5
OSC1/OC1A/PB6
6
OSC2/PB7
7
OC0B/T1/RXD/PD5
8
OC0A/OC3A/AC0P/TXD/PD6
9
OC2B/ACXN/PD7
10
20
PC5/ADC5/SCL
19
PC4/ADC4/SDA
18
PC1/ADC1
17
PC0/ADC0/OC0A
16
PE2/ADC7/AC1N/AVREF/SWD
15
PE0/ADC6/ACXP/SWC
14
PB5/SCK/AC1P
13
PB4/MISO
12
PB3/MOSI/OC1B/OC2A
11
PB1/ICP1/CLKO/OC1A/SPSS

Pin description

LGT8FX8P In the series package,QFP48L The package leads to all pins. Other packages are inQFP48 Multiple internal I/O Generated by binding to a pin. Pay special attention when configuring the pin direction. The following table lists the binding of various package pins:

QFP48	QFP32	SSOP20	Function Description
01	01	03	PD3/INT1/OC2B* PD3: Programmable port D3 INT1: External interrupt input 1 OC2B: Timer 2 Compare match output B
02	02		PD4/DAO/T0/XCK PD4: Programmable port D4 DAO: internal DAC Output T0: Timer0 External clock input XCK: USART Synchronous transmission clock
03	03	-	PE4/0C0A* PE4: Programmable port E4 OC0A: Timer 0 Compare match output A
04	-	-	PF3/OC3C/OC0B* PF3: Programmable port F3 OC3C: Timer 3 Compare match output C OC0B: Timer 0 Compare match output B
05	03	03	PF4/OC1B*/ICP3 PF4: Programmable port F4 OC1B: Timer 1 Compare match output B ICP3: Timer 3 Capture input
06	04	04	VCC
07	05	05	GND
08	06	-	PE5/AC1O/CLKO* PE5: Programmable port E5 C1O: Analog comparator AC1 Output CLKO: System clock output
09		06	PF5/OC1A* PF5: Programmable port F5 OC1A: Timer 1 Compare match output
10	-	-	A PF6/T3/OC2A* PF6: Programmable port F6 T3: Timer 3 External clock input OC2A: Timer 2 Compare match output
11	07	06	A PB6/XTALO PB6: Programmable port B6 XTALO: Crystal oscillator IO Output port

12	08	07	PB7/XTALI PB7: Programmable port B7` XTALI: Crystal oscillator IO Input port
13	09	08	PD5/RXD*/T1/OC0B PD5: Programmable port D5 RXD: USART Data reception (optional) T1: Timer 1 External clock input OC0B: Timer 0 Compare match output
14	10	09	B PD6/TXD*/OC0A PD6: Programmable port D6 TXD: USART Data transmission (optional) OC0A: Timer 0 Compare match output
15			A AC0P/0C3A AC0P: Analog comparator 0 Positive input OC3A: Timer 3 Compare match output
16	11	10	A PD7/ACXN PD7: Programmable port D7 ACXN: Analog comparator 0/1 Common negative input
17	-		PF7/OC2B PF7: Programmable port F7 OC2B: Timer 2 Compare match output
18	12	11	B PB0/ICP1 PB0: Programmable port B0 ICP1: Timer 1 Capture input
19	13		PB1/OC1A PB1: Programmable port B1 OC1A: Timer 1 Compare match output
20	14	12	A PB2/OC1B/SPSS PB2: Programmable port B2 OC1B: Timer 1 Compare match output B SPSS: SPI Slave mode chip select
twenty one	-	-	GND
twenty two	-	-	VCC
twenty three	15	12	PB3/MOSI/OC2A PB3: Programmable port B3 MOSI: SPI Host output/Slave input OC2A: Timer 2 Compare match output
twenty four	16	13	A PB4/MISO PB4: Programmable port B4 MISO: SPI Host input/Slave output
25	17	14	PB5/SPCK/AC1P PB5: Programmable port B5 SPCK: SPI Clock signal AC1P: Analog comparator 1 Positive input

26	**-**	**-**	PE7/ADC11 PE7: Programmable port E7 ADC11: ADC Analog input channel 11 AVCC:
27	**-**	**-**	Internal analog circuit power supply
28	**18**	**15**	PE0/SWC/APN4 PE0: Programmable port E0 SWC: SWD Debug interface clock APN4: Differential amplifier reverse input channel 4
29	**19**		PE1/ADC6/ACXP PE1: Programmable port E1 ADC6: ADC Analog input channel 6 ACXP: Analog comparator 0/1 Common positive input
30	**20**	**16**	PE6/ADC10/AVREF PE6: Programmable port E6 ADC10: ADC Analog input channel 10 AVREF: ADC External reference input
31	**-**	**-**	CVREF: ADC Reference voltage output Only for external 0.1uF Filter capacitor
32	**-**	**-**	AGND: Internal analog circuit ground
33	**twenty one**	**16**	PE2/SWD PE2: Programmable port E2 SWD: SWD Debug interface data line
34	**twenty two**		PE3/ADC7/AC1N PE3: Programmable port E3 ADC7: ADC Analog input channel 7 AC1N: Analog comparator negative input
35	**twenty three**	**17**	PC0/ADC0/APP0 PC0: Programmable port C0 ADC0: ADC Analog input channel 0 APP0: Differential amplifier forward input channel 0
36	**twenty four**	**18**	PC1/ADC1/APP1 PC1: Programmable port C1 ADC1: ADC Analog input channel 1 APP1: Differential amplifier forward input channel 1
37	**25**	**-**	PC2/ADC2/APN0 PC2: Programmable port C2 ADC2: ADC Analog input channel 2 APN0: Differential amplifier reverse input channel 0
38	26	-	PC3/ADC3/APN1 PC3: Programmable port C3 ADC3: ADC Analog input channel 3 APN1: Differential amplifier reverse input channel 1

39	27	19	PC4/ADC4/SDA PC4: Programmable port C4 ADC4: ADC Analog input channel 4 SDA: I2C Controller data line
40	28	20	PC5/ADC5/SCL PC5: Programmable port C5 ADC5: ADC Analog input channel 5 SCL: I2C Controller clock line
41	29	1	PC6/RESETN PC6: Programmable port C6 RESETN: External reset input
42	-	-	PC7/ADC8/APN2 PC7: Programmable port C7 ADC8: ADC Analog input channel 8 APN2: Differential amplifier reverse input channel 2
43	-	-	PF0/ADC9/APN3 PF0: Programmable port F0 ADC9: ADC Analog input channel 9 APN3: Differential amplifier reverse input channel 3
44	30	-	PD0/RXD PD0: Programmable port D0 RXD: USART Data receiving input
45	31	-	PD1/TXD PD1: Programmable port D1 TXD: USART Data transmission output
46		1	PF1/OC3A PF1: Programmable port F1 OC3A: Timer 3 Compare match output
47	32	2	A PD2/INT0/AC0O PD2: Programmable port D2 INT0: External interrupt input 0 AC0O: Analog comparison 0 Output
48			PF2/OC3B PF2: Programmable port F2 OC3B: Timer 3 Compare match output B

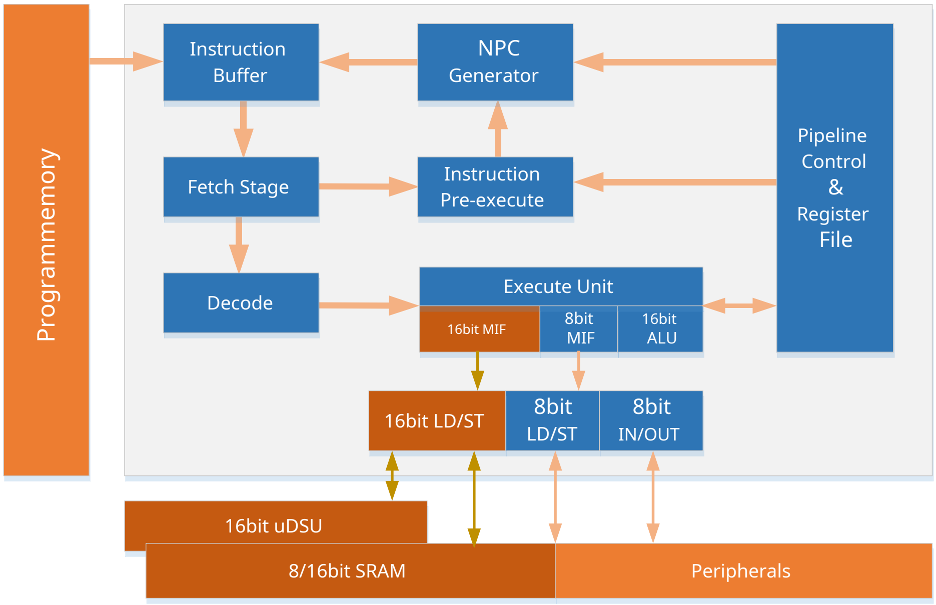
LGT8XMKernel

- Z Low power design
- Z high efficiency RISC Framework
- Z 16 Bit LD/ST Extension(uDSU dedicated)
- Z 130 Instructions, of which 80%The above is a single-cycle
- Z embedded online debugging (OCD)stand by

Overview

This chapter mainly describes LGT8XM Kernel architecture and functions. Kernel isMCU The brain is responsible for ensuring the correct execution of the program, so the kernel must be able to accurately perform calculations, control peripherals, and handle various interrupts.

LGT8XMThe structure of the kernel



In order to achieve greater efficiency and parallelism,LGT8XM The kernel adopts the Haval architecture - Independent data and program bus. Instructions are executed through an optimized two-stage pipeline. The two-stage pipeline can reduce the number of invalid instructions in the pipeline and reduce the number of FLASH Access to the program memory, so the power consumption of the kernel can be reduced. SimultaneouslyLGT8XMThe kernel adds an instruction cache in the pre-stage of fetching instructions (can be cached at the same time 2 Instructions), through the pre-execution module in the instruction fetch cycle, further reducing the FLASH Access frequency of program memory; after extensive testing,LGT8XM It can reduce about 50%Correct FLASH The access, greatly reducing the operating power consumption of the system.

LGT8XMThe kernel has 32 A 8 General-purpose working register for high-speed access (Register file), Which helps to realize single-cycle arithmetic and logic operations (ALU). In general,ALU Both operands of the operation come from the general working register,ALU The result of the operation is also written into the register file in one cycle.

32 Through the working register 6 One is used to combine two to form three 16 Bit register, can be used for indirect addressing address pointer, used to access external storage space and FLASH Program space.LGT8XM Support single cycle 16 Bit arithmetic operations greatly improve the efficiency of indirect addressing.LGT8XMThese three special 16 The bit register is named X, Y, Z Register, will be introduced in detail later.

ALU Supports arithmetic and logic operations between registers and between constants and registers, and a single register can also be operated in ALU Executed.ALU After the operation is completed, the effect of the operation result on the state of the core is updated to the status register (SREG). Program flow control is realized by conditional and unconditional jump/call, which can be addressed to all program areas. most LGT8XMThe instruction is 16 Bit. One for each program address space16 Bit or 32 Bit LGT8XMinstruction.

After the kernel responds to the interrupt or subroutine call, the return address (PC)Is stored in the stack. The stack is allocated in the general data of the systemSRAMMedium, so the size of the stack is only limited by the system SRAMThe size and usage. All applications that support interrupts or subroutine calls must first initialize the stack pointer register (SP),SP able to pass IO Space access. data SRAM able to pass 5 A different addressing mode access.LGT8XM The internal storage space is linearly mapped to a unified address space. For details, please refer to the introduction in the storage chapter.

LGT8XM The core contains a flexible interrupt controller, the interrupt function can be controlled by a global interrupt enable bit in the status register. All interrupts have an independent interrupt vector. The priority of the interrupt has a corresponding relationship with the interrupt vector address. The smaller the interrupt address, the higher the priority of the interrupt.

I/O Space contains 64 Can pass IN/OUT The register space directly addressed by the instruction. These registers show the control and status registers of the core,SPI And other I/O Control functions of peripherals. This part of the space can passIN/OUT Instructions are directly accessed, or they can be accessed through their addresses mapped to the data memory space (0x20 – 0x5F). In addition, LGT8FX8P Also includes extended I/O Space, they are mapped to data storage space 0x60 – 0xFF, Can only be used here ST/STS/STD as well as LD/LDS/LDD Command access.

For enhancement LGT8XM The computing power of the kernel has been added to the popular line of instructions 16 Bit LD/ST Extension. this16 Bit LD/ST Extended fit 16 Digital operation acceleration unit (uDSU)Work to achieve efficient 16 Bit data operations. At the same time, the kernel also addsRAM Spatial 16 Bit access capability. therefore16 Bit LD/ST The extension can be in uDSU, RAM, And transfer between working registers 16 Bit data. For specific details, please refer to"Digital computing accelerator"chapter.

Arithmetic and logical operation unit (ALU)

LGT8XMContains a 16 Bit arithmetic logic operation unit can be completed in one cycle 16 Arithmetic operations for data. EfficientALU versus 32 Two general-purpose working registers are connected. The arithmetic logic operation between two registers or between the register and the immediate data can be completed in one cycle.ALU There are three types of operations: arithmetic, logic and bit operations. SimultaneouslyALU The part also includes a single-cycle hardware multiplier, which can achieve two 8 Direct signed or unsigned operations on bit registers. Please refer to the detailed introduction in the instruction set section.

Status register (SREG)

The status register mainly saves the execution Last time ALU The result information generated by the operation. This information is used Control program execution flow. The status register is inALU Update after the operation is completely finished, so that the use of a separate comparison instruction can be omitted, and a more compact and efficient code implementation can be brought. The value of the status register is not automatically saved and restored when responding to the interrupt and exiting from the interrupt, which requires software to achieve.

SREG Register definition

SREG System status register								
address: 0x3F (0x5F)				Defaults: 0x00				
Bit	7	6	5	4	3	2	1	0
Name	I	T	H	S	V	N	Z	C
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit definition		
[0]	C	Carry flag, indicating that an arithmetic or logical operation caused a carry, please refer to the instruction description for details
[1]	Z	Zero flag, indicating that the result of arithmetic or logic operation is zero, please refer to the instruction description section
[2]	N	Negative sign, which means that arithmetic or logical operation produces a negative number, please refer to the instruction description section
[3]	V	Overflow flag, indicating that the result of the two's complement operation has overflowed, please refer to the instruction description section
[4]	S	Sign bit, equivalent to N versus V The result of the exclusive OR operation, please refer to the instruction description section for details
[5]	H	Half-carry flag, in BCD Useful in calculations, representing the half-carry generated by byte operations
[6]	T	Temporary bit, bit copy (BLD)And bit storage (BST)Used in instructions,T The bit will be used as a temporary storage bit to temporarily store the value of a bit in the general-purpose register. For details, please refer to the instruction description section
[7]	I	Global interrupt enable bit, this bit must be set to 1 In order to enable the core to respond to interrupt events. Different interrupt sources are controlled by independent control bits. The global interrupt enable bit is the back barrier that controls the interrupt signal to enter the core.I The bit is automatically cleared by the hardware after the core responds to the interrupt vector, and the interrupt return instruction (RETI)Set automatically afterwards. I Bits can also be used SEI with CLI Command changes, please refer to the command description section

General working register

General working register according to LGT8XM Instruction set architecture optimization. In order to achieve the required efficiency and flexibility of kernel execution, LGT8XMThe internal general working register supports the following access modes:

Z One 8 Read one bit at the same time 8 Bit write operation

Z two 8 Read one bit at the same time 8 Bit write operation

Z two 8 Read one bit at the same time 16 Bit write operation

Z 16 Read one bit at the same time 16 Bit write

LGT8XM General working register

through
use
work
Make
send
Save
Device

7 0	Addr.	
R0	0x00	
R1	0x01	
R2	0x02	
…		
R13	0x0D	
R14	0x0E	
R15	0x0F	
R16	0x10	
R17	0x11	
…		
R26	0x1A	XRegister low byte
R27	0x1B	XRegister high byte
R28	0x1C	YRegister low byte
R29	0x1D	YRegister high byte
R30	0x1E	ZRegister low byte
R31	0x1F	ZRegister high byte

Most instructions can directly access all general-purpose working registers, and most of them are single-cycle instructions. As shown in the figure above, each register corresponds to an address in the data storage space, and these general working registers are mapped to the data storage space. As soon as they don't really exist inSRAMHowever, this unified mapping storage organization brings great flexibility to access them.X/Y/Z The register can be used as a pointer to index to any general-purpose register.

X/Y/Z register

register R26...R31 Can be combined in two to form three 16 Bit register. These three16 Bit registers are mainly used for indirect Address pointer for addressing access,X/Y/Z The register structure is as follows:

	15 XH	XL 0
Xregister	7 0	7 0
	R27 (0x1B)	R26 (0x1A)

	15 YH	YL 0
Yregister	7 0	7 0
	R29 (0x1D)	R28 (0x1C)

	15 ZH	ZL 0
Zregister	7 0	7 0
	R31 (0x1F)	R30 (0x1E)

In different addressing modes, These registers are used as fixed offset, auto-increment and auto-decrement address pointers, For details, please refer to the instruction description section.

Stack pointer

The stack is used to store temporary data, local variables, and the return address of interrupts and subroutine calls. It is important to note that the stack is not designed to grow from a high address to a low address. Stack pointer register (SP)Always point to the top of the stack. The stack pointer points to the data SRAMThe physical space where it is located, where the stack space necessary for subroutines or interrupt calls is stored.PUSH The instruction will decrement the stack pointer.

Stack in SRAMThe position in must be correctly set by software before subroutine execution or interrupt enable. Under normal circumstances, the stack pointer is initialized to point toSRAMHigh address. The stack pointer must be set to highSRAMStart address.SRAMFor the address mapped in the system data storage, please refer to the system data storage section.

Stack pointer related instructions

instruction	Stack pointer	description
PUSH	increase 1	Data is pushed onto the stack
CALL ICALL RCALL	increase 2	The return address of the interrupt or subroutine call is pushed onto the stack
POP	cut back 1	Data is taken from the stack
RET RETI	cut back 2	The return address of the interrupt or subroutine call is taken from the stack

The stack pointer is allocated by I/O Two of space 8 Bit register structure. The actual length of the stack pointer is related to the system implementation. inLGT8XMIn some chip implementations of the architecture, the data space is so small that only SPL Can meet the addressing needs, in this case,SPH The register will not appear.

SPH/SPL Stack pointer register definition

SPH/SPL Stack pointer register		
SPH: 0x3E (0x5E)		Defaults: RAMEND
SPL: 0x3D (0x5D)		
SP	SP[15:0]	
R/W	R/W	
Bit definition		
[7:0]	SPL	Stack pointer low 8 Bit
[15:8]	SPH	Stack pointer high 8 Bit

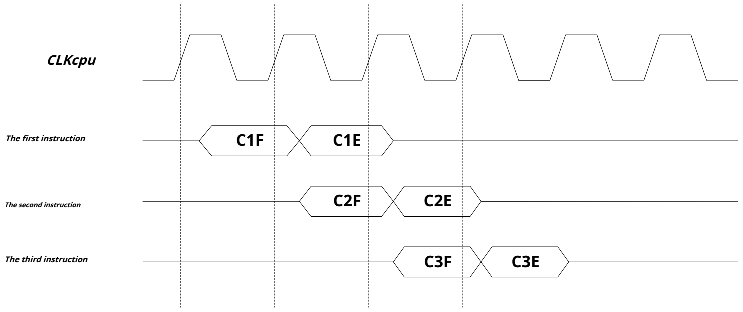
Instruction execution timing

This chapter describes the general timing concepts of instruction execution.LGT8XMThe core is clocked by the core (CLKcpu)Drive, this clock comes directly from the clock source selection circuit of the system.

The following figure shows the execution sequence of the instruction pipeline based on the Haval architecture and the concept of fast access to the register file. This is to make

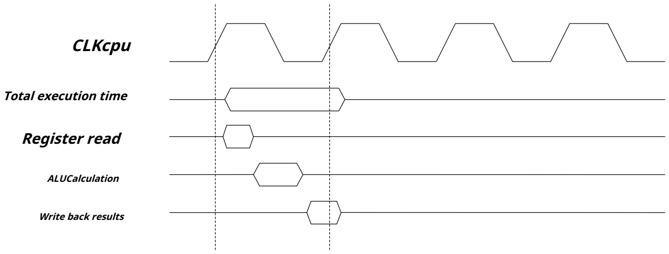
Get the kernel to get 1MIPS/MHz The physical guarantee of the efficiency of the implementation.

As can be seen from the figure above, the second instruction will be read during the execution of the first instruction. When the second instruction enters the execution



During the run, the third instruction will be read at the same time. In this way, during the entire execution period, there is no need to spend extra cycles for fetching instructions. From the perspective of the pipeline, the efficiency of executing one instruction every Monday is realized.

The following figure shows the access timing of the general working register. In one cycle,ALU The operation uses two registers as operands, and in this cycle ALU The execution result is written into the target register.



Reset and interrupt handling

LGT8XM Support multiple interrupt sources. These interrupt and reset vectors correspond to an independent program vector entry in the program space. Generally speaking, all interrupts are controlled by separate control bits. When the control bit is set and the global interrupt enable bit of the core is enabled, the core can respond to this interrupt.

The low program space is reserved as the reset and interrupt vector area by default.LGT8FX8P For a complete list of interrupts supported, please refer to the introduction in the interrupt chapter. This list also determines the priority of different interrupts. The lower the vector address of the interrupt, the higher the corresponding interrupt priority. Reset (RESET)Has a high priority, and then INT0 – External interrupt request 0.

The start address of the interrupt vector table (except the reset vector) can be redefined to any 256 At the beginning of byte alignment, you need to pass MCU Control register (MCUCR)middle IVSEL Bit and IVBASE Vector base address register implementation.

When the kernel responds to the interrupt, the global interrupt enable flag is I Will be automatically cleared by hardware. Users can passI Bit enable realizes interrupt nesting. In this way, any subsequent interrupts will interrupt the current interrupt service routine.I Bit in the execution of the interrupt return instruction (RETI)It is automatically set afterwards, so that it can respond normally to subsequent interrupts.

There is a basic type of interrupt. The first type is triggered by an event, and the interrupt flag bit is set after an interrupt event occurs. For this kind of interrupt, after the kernel responds to the interrupt request, the currentPC The value is directly replaced with the actual interrupt vector address, the corresponding interrupt service subroutine is executed, and the hardware automatically clears the interrupt flag bit. The interrupt flag bit can also be written to the position of the interrupt flag bit1 Clear. If the interrupt enable bit is cleared when an interrupt occurs, the interrupt flag bit will still be set to record the interrupt event. After the interrupt is enabled, the recorded interrupt event will be immediately responded. Similarly, if an interrupt occurs, the global interrupt enable bit (SERG.I)Is cleared, the corresponding interrupt flag bit will also be set to record the interrupt event, etc.

After the global interrupt enable bit is set, these recorded interrupts will be executed in order according to their priority.

The second type of interrupt is that when the interrupt condition always exists, the interrupt will always respond. This type of interrupt does not require an interrupt flag bit. If the interrupt condition disappears before the interrupt is enabled, the interrupt will not be responded.

when LGT8XMAfter the kernel exits from the interrupt service subroutine, the execution flow will return to the main program. After executing one or several instructions in the main program, it can respond to other waiting interrupt requests.

It should be noted that the system status register (SREG)It will not be automatically saved after entering the interrupt service, nor will it be automatically restored after returning from the interrupt service. It must be handled by the software.

When using CLI After the instruction disables the interrupt, the interrupt will be immediately disabled. inCLI All interrupts that occur after the instruction will not be responded. Even withCLI The interrupts that occur at the same time when the instruction is executed will not be responded. The following example shows how to useCLI Avoid interruption EEPROM Write timing:

Interrupt response time

LGT8XM The kernel is optimized for interrupt response, so that any interrupt 4 A response must be received within one system clock cycle.4 After a system clock cycle, the interrupt service subroutine enters the execution cycle. At this4 Within clocks, before the interrupt PC The value is pushed onto the stack, and the system execution flow jumps to the interrupt service routine corresponding to the interrupt vector. If the interrupt occurs during the execution of a multi-cycle instruction, the kernel will ensure that the current instruction ends correctly. If the interrupt occurs while the system is sleeping (SLEEP), The interrupt response needs to be added 4 Clock cycles. This increased clock period is used to synchronize the period of wake-up operation from the selected sleep mode. For a detailed description of the sleep mode, please refer to the relevant chapters on power management.

Need to return from the interrupt service subroutine 2 Clock cycles. At this2 Within clock cycles,PC Recover from the stack, stack pointer plus 2, And automatically enable the global interrupt control bit.

Storage unit

Overview

This chapter mainly describes LGT8FX8P Different storage units within the series.LGT8XMThe framework supports two main internal storage spaces, namely data storage space and program storage space.LGT8FX8P Also contains data FLASH, Can be achieved through the internal controller EEPROMData storage function of the interface. In addition,LGT8FX8P The system also contains a special storage unit for storing system configuration information and the chip's global device number (GUID).

LGT8FX8P Series chips include LGT8F88P/168P/328P Four different models; the peripherals and packages of the four models are fully compatible, the difference is FLASH Program storage space and internal data SRAM, The following table is more clear Described LGT8FX8P Different storage space configurations of series chips:

DEVICE	FLASH	SRAM	E2PROM	Interrupt vector
LGT8F88P	8KB	1KB	2KB	1 Instruction word
LGT8F168P	16KB	1KB	4KB	2 Instruction word
LGT8F328P	32KB	2KB	Can be configured as 0K/1K/2K/4K/8K (versus FLASH shared)	2 Instruction word

LGT8F328P There is no independent internal use for simulation E2PROMInterface FLASH Space; for simulation E2PROMSave Storage space and program FLASH Sharing, users can choose the appropriate configuration according to application requirements.

Due to simulation E2PROMUnique implementation of the interface, the system requires twice as many programs FLASH Space simulation E2PROM Storage space, for example for LGT8F328P, If the user configures 1KB of E2PROM Space, there will be 2KB byte The program space is reserved, leaving 30KB of FLASH Space is used to store programs.

LGT8F328P program FLASH versus E2PROMShared configuration table:

DEVICE	FLASH	E2PROM
LGT8F328P	32KB	0KB
	30KB	1KB
	28KB	2KB
	24KB	4KB
	16KB	8KB

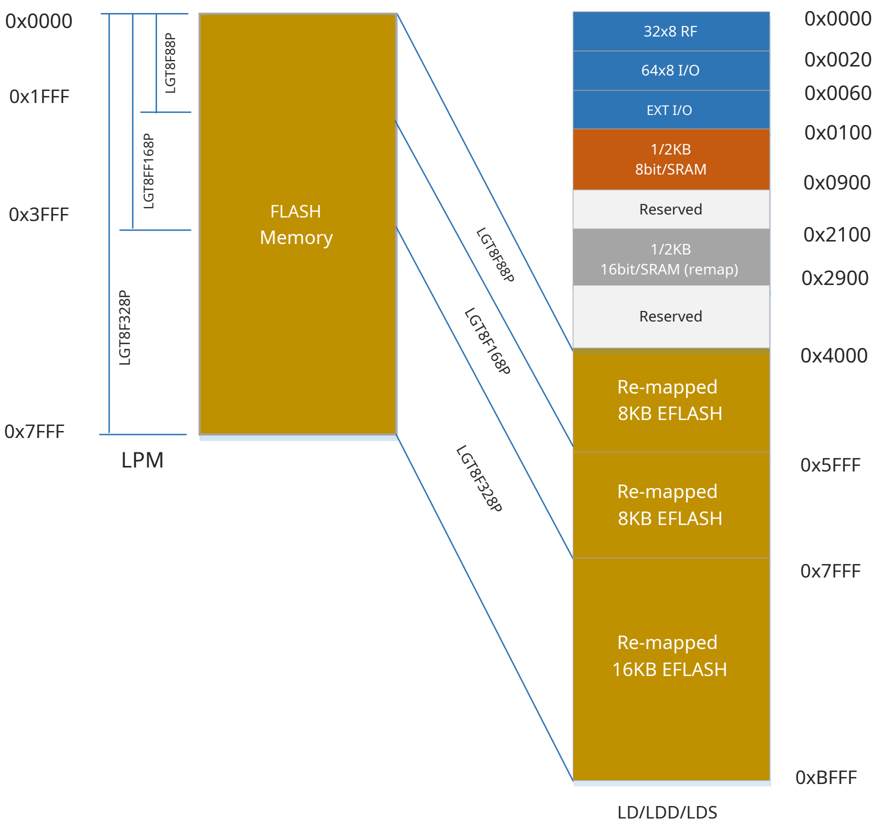
System programmable FLASH Program storage unit

LGT8FX8PThe series of microcontrollers include8K/16K/32KByte on-chip online programmableFLASHProgram storage unit.

program FLASH Can guarantee at least 100,000 More than two erasing and writing cycles.LGT8FX8P Internal integration FLASH Interface controller, capable of in-system programming (ISP)And the self-upgrading function of the program. For specific implementation details, please refer to this chapter on FLASH The description of the interface controller part.

Program space can also be passed LPMDirect access to instructions (read), this feature can realize application-related constant search table. SimultaneouslyFLASH The program space is also mapped to the system data storage space, and the user can also use it LD/LDD/LDS Realize right FLASH Space access. Program space is mapped to data storage space0x4000 Within the starting address range.

As shown below:



SRAMData storage unit

LGT8FX8P A series of microcontrollers is a relatively complex microcontroller, it supports a variety of different types of peripherals, the controllers of these peripherals are allocated in 64 A I/O In the register space. Can pass directlyIN/OUT Command access. The control registers of other peripherals are allocated in0x60 ~ 0xFF In the area, since this part of the space is mapped to the data storage space, it can only be passed through ST/STS/STD as well as LD/LDS/LDD Wait for instruction to visit.

LGT8FX8P System data storage space from 0 Starting from the address, the general working register files are mapped respectively,I/O Space, expansion I/O Space and internal data SRAM space. The beginning32 Byte address correspondence LGT8XM Kernel 32 A general-purpose working register. Next64 Addresses can be passed IN/OUT Direct access standard for instructions I/O space. Then160 Addresses are extensions I/O Space, the next step is more 2K Bytes of data SRAM. From0x4000 Start to 0xBFFF This part of the ending space is mapped FLASH Program storage unit.

In the system 1K/2K byte SRAM Are mapped to two spaces respectively. From0x0100 Start to 0x0900 This space ended by the kernel with 8 The width of the bit byte is read and written. From0x2100 Start to 0x2900 End this area as 16 Bit-width access space. systemRAMIs mapped to 0x2100 The first high address is mainly used to cooperate uDSU Module work to achieve efficient 16 Bit data storage. When programming, the ordinary8 Bit addressing variable address plus 0x2000 , You can switch to 16 Bit access mode.

System Support 5 Different addressing modes can cover the entire data space: direct access, indirect access with offset, indirect access, indirect access with decrement address before access, and indirect access with increment address after access. General working register R26 To R31 Address pointer used for indirect access. Indirect access can address the entire data storage space. Indirect access with offset address can be addressed toY/Z Register is near the base address 63 Address spaces.

When using the indirect register access mode that supports address auto-increment/decrement, the address register X/Y/Z It will be automatically decremented/incremented by the hardware before/after the access occurs. For details, please refer to the instruction set description section.

16 Bit register X/Y/Z And the related automatic addressing mode (increment, decrement), in 16 The bit extension mode also plays a very important role.16 Bit extension mode can be used LD/ST Increment/decrement mode for realizing automatic increment and decrement addressing with variables. This mode is very effective when performing operations on arrays. For specific implementation, please refer to"Digital Computing Accelerator (uDSU)"Related chapters.

Universal I/O register

LGT8FX8P of I/O There are three general I/O register GPIOR2/1/0, These three registers can be used IN/OUT Command access, used to store user-defined data.

Peripheral register space

I/O For detailed definition of space, please refer to LGT8FX8P Data sheet"Register overview"chapter.

LGT8FX8P All peripherals are assigned to I/O space. allI/O Space address can be LD/LDS/LDDD as well as ST/STS/STD Command access. The data accessed is all through32 A general-purpose working register is passed. in0x00 ~ 0x1F between I/O Registers can be addressed by bit addressing instructions SBI with CBI access. In these registers, the value of a certain bit can be usedSBIS with SBIC Instruction detection is used to control the execution flow of the program. For details, please refer to the instruction set description section.

When using IN/OUT Command access I/O Must be addressed when registering 0x00 ~ 0x3F Between addresses. When usingLD or ST Command access I/O Space, must pass I/O The space in the system data memory is uniformly mapped to the mapped address access of the space (plus 0x20 Offset). Some other allocations are in expansionI/O Peripheral registers of space (0x60 ~ 0xFF), Can only use ST/STS/STD with LD/LDS/LDD Command access.

In order to be compatible with future devices, reserved bits must be written during write operations 0. Can not be reservedI/O Perform a write operation on the space.

Some registers include status flags and need to be written 1 To be cleared. have to be aware of is,CBI with SBI The instructions only support specific bits, so CBI/SBI It can only work on the registers that contain these status flags. In addition, CBI/SBI The instruction can only work on 0x00 To 0x1F Registers in this address range.

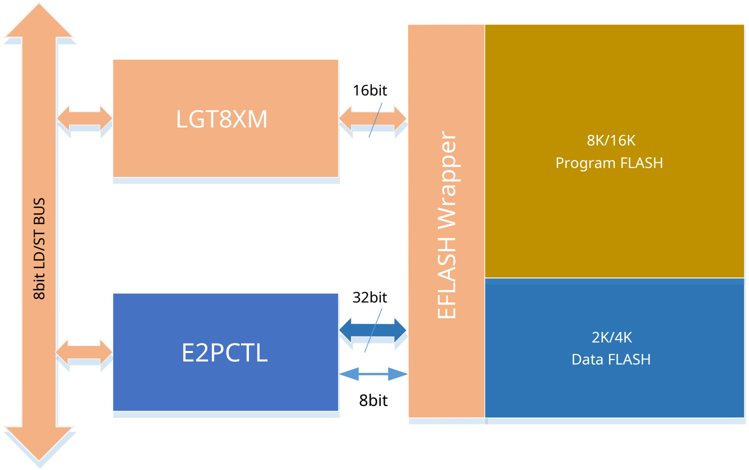
FLASH Controller (E2PCTL)

LGT8FX8P The internal implementation integrates a flexible and reliable EFLASH Read and write controller, can use the existing data in the systemFLASHStorage space, storage space for byte read and write access, similar toE2PROMStorage applications;E2PROM The interface simulation adopts the algorithm of erasing and writing equalization, which can transfer the data FLASH Increased life cycle 1 About times, can guarantee 100,000 More than two erasing and writing cycles.

E2PCTL The controller also implements FLASH The online erasing and writing operation of the program space can be realized online by the software.

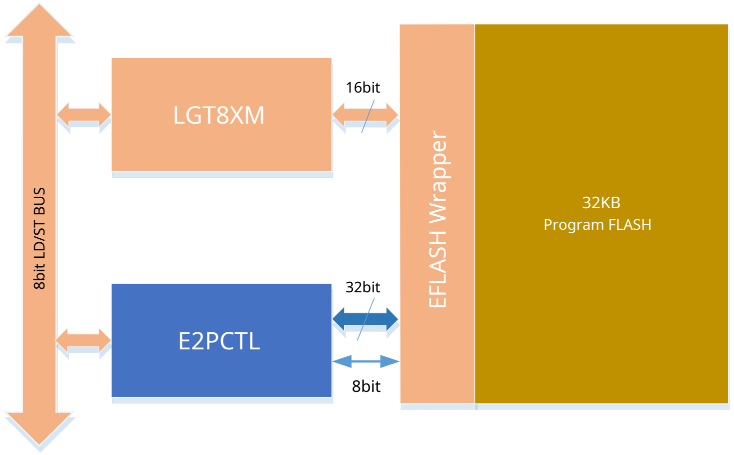
The function of upgrading the firmware automatically. byFLASH Controller access program FLASH Program space, only supports page erase (1024 Bytes) and 32 Bit width read and write access.

LGT8F88D/168D E2PCTL Controller structure diagram



E2PCTL simulation E2PROM Function access data FLASH Space, can support 8 Bit,32 Bit read and write width. visit Ask the program FLASH Space, support page erasing and 32 Bit data read and write. due toLGT8FX8P internal FLASH The small storage unit is 32 Bit, so it is recommended to use 32 Bit access mode, especially for write operations.32 Bit-accessed read and write operations are not only efficient, but also conducive to protection FLASH The erasable life of the storage unit.

LGT8F328P E2PCTL Controller structure diagram



LGT8F328P No extra data inside FLASH. therefore,LGT8XM Kernel and E2PCTL Shared internal 32K word Section FLASH storage. Users can change32K byte FLASH The space is divided into program space and data space. By configurationE2PCTL Controller, can be set to simulate E2PROM The size of the space.E2PCTL Use page swap mode to achieve simulation E2PROM Logic, algorithm in page (1K Byte) as the unit. So simulate1K Byte E2PROM Space, need to occupy 2K Byte FLASH Space, and so on, to achieve 4K Byte E2PROM, Need to occupy 8K Byte FLASH space. For specific implementation methods, please refer toE2PCTL The description of the algorithm implementation.

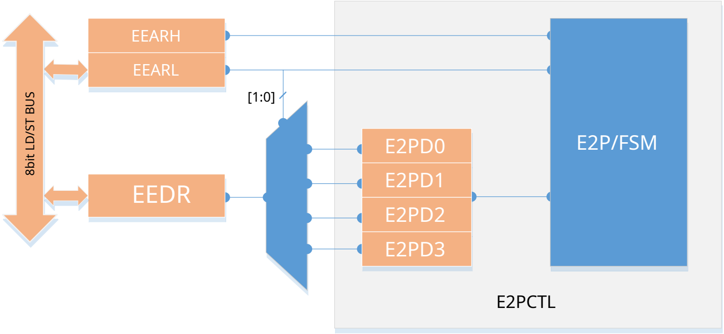
E2PCTL Data register

E2PCTL Inside the controller 4 Bytes of data buffer (E2PD0~3),this 4 Byte buffer composition Last visit FLASH Spatial 32 Bit data interface.

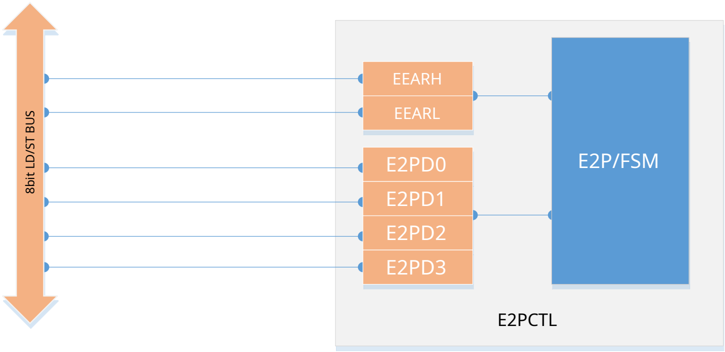
when E2PCTL When the controller works in byte read and write mode,EEDR As an interface for reading and writing byte data, E2PCTL more EEARL[1:0]Load the data into the correct data cache based on the current address information FLASH The data of the target address complements the other three bytes of data, and the combination will be complete 32 Bit data updated to FLASH in.

when E2PCTL work at 32 In bit read/write mode, it can still be used at this time EEDR Register as a common data interface, through EEARL[1:0]As an address to address the internal data buffer, a complete read and write 32 Bit data. In addition, you can directly use the data cache to map toIO Direct access to the registers of the space (E0~3).

E2PCTL work at 8 Schematic diagram of data access in bit byte read and write mode:



E2PCTL work at 32 Schematic diagram of data access in bit word read and write mode:



Byte mode is used for backward compatibility LGT8FX8D The byte read and write mode.LGT8FX8P Built-in FLASH for 32 Bit interface width, use 32 Bit read and write mode will give read and write efficiency and FLASH The erasing life brings great benefits, so it is recommended to use 32 Bit read and write mode.

E2PCTL simulation E2PROMInterface algorithm

we know,FLASH The memory must be erased before being written, and the erase operation is in units of pages.LGT8FX8P Built-in FLASH The size of one page of memory is 1K byte. Therefore, in order to update a byte of data in the page, it is also necessary to erase the data of the entire page first, then update the target address data, and restore the data of other bytes in the page at the same time. The entire operation is not only time-consuming, but also brings Risk of accidental data loss due to power supply.

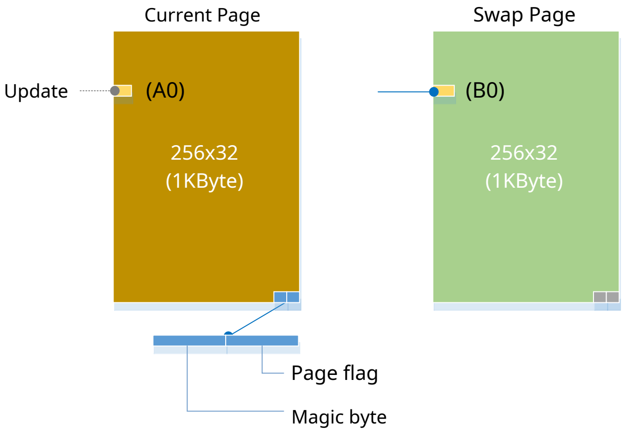
E2PCTL Internal use of page swap algorithm to achieve simulation E2PROM. The page swap algorithm mode can ensure that the original data will not be lost due to unexpected conditions such as power failure when the page erase operation is performed. At the same time exchange algorithm use2 Page space

The exchange method is used alternately, and simulation is also added E2PROMThe service life of the space.

In terms of efficiency, E2PCTL The controller implements a continuous data update mode, which reduces the repeated erasing and writing process caused by repeated data update.

In terms of implementation, E2PCTL Manage each page separately, and after occupying one page 2 Bytes are used as page status information. So users are using greater than1K of E2PROMWhen simulating the space, you need to pay attention to the address crossing 1K Spatial Special treatment. Because every1K Spatial Rear 2 Bytes reserved for E2PCTL Use, the user can't 2 Bytes The space performs normal reading and writing.

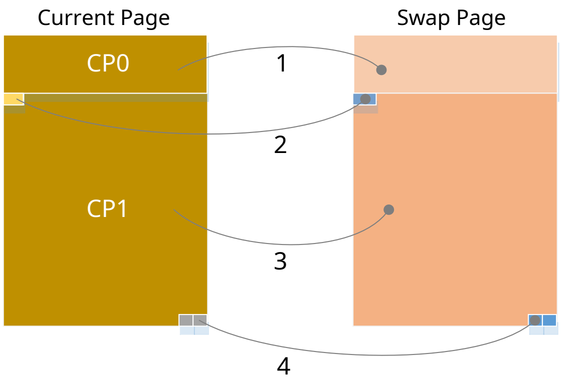
The picture below is E2PCTL Logic diagram based on page swap algorithm:



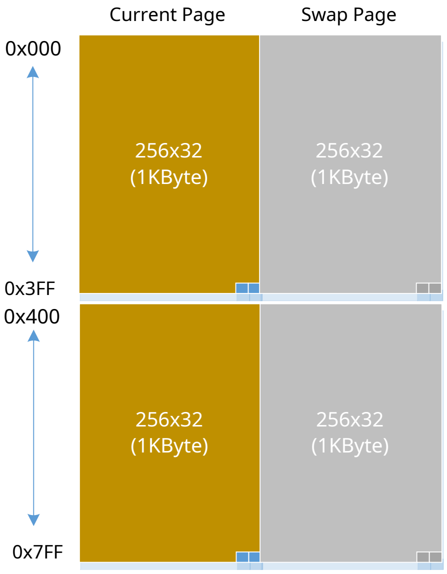
as the picture shows,E2PCTL internal use 2 Each page simulates one page size E2PROM space. These two pages One is marked as the current page, and the other is an exchange page.E2PCTL After using the page 2 Bytes store page information. When we need to update a certain byte in the page, such as the picture aboveA0 byte. First, we will not erase the current page, but the swap page. Then divide the current page into3 Part of the operation. First is inA0 In the previous data, we call this part of the space CP0,After that A0 After the data, this part of the space is CP1.E2PCTL According to user configuration, CP0 The corresponding data is copied to the corresponding address of the exchange page, and then the data that needs to be updated is written to the address corresponding to the exchange page (B0), Followed by copy CP1 The data to the exchange page.

After completing the above operations, the data has been exchanged, but the page status has not been updated. Therefore, if a power failure or other abnormality occurs before this, because this update operation has not been completed, the previous data will not be destroyed, ensuring the integrity of the data. If everything goes well,E2PCTL Will be at CP1 After the data is exchanged, the updated page status is written to the page information of the previous exchanged page to realize the replacement of the face-to-face page. After that, the exchange page becomes the current page.

E2PCTL The page exchange process is shown in the figure below (1->2->3->4):



When the system is configured E2PROMSimulation space is greater than1KTime,E2PCTLStill take the page as Small unit realization E2PROM Space simulation algorithm. For example, if the user configures2K of E2PROMArea, actually E2PCTL Will occupy 4 Pages (4K)Space. among them2 Pages as a group, used to simulate a page size E2PROMspace.



Need to pay attention to the user configuration 2K Byte E2PROM The space is not continuous, because the bytes of each Rear 2 A page will be used to store page status information.

E2PCTL Continuous programming mode

Due to passing E2PCTL Updates will cause page swaps. The swap page will be erased during the page swap process. Page erasing not only takes time, but also increases FLASH Loss of life. thereforeE2PCTL Added continuous write mode. In continuous write mode, users can update continuouslyE2PROM Area, the page swap operation will only be performed after consecutive addresses. For applications that need to continuously update a whole block of data, the continuous mode is more effective.

Continuous programming mode E2PCTL Control register ECCR of SWM Bit enable. After the continuous mode is enabled, subsequent write operations will directly write the data to the address corresponding to the swap page.SWMIn mode, the write operation will not be executed CP0/1 Area data copy operation. Before writing the last byte, the software passesSWMDisable continuous mode, then execute write, after that E2PCTL Will execute the complete CP0/1 Copy operation and update page status information.

E2PCTL Read and write FLASH Program space

by E2PCTL Controller, can realize the program FLASH Read and write access to the space. And simulationE2PROM The difference is that through E2PCTL On the program FLASH Access to the space requires software control. Proceed as follows:

1. To erase the target page, the target page needs to be erased before updating the data, and the page address is passed EEAR The register is given. CorrectFLASH For page erasing command control, please refer to EECR The definition of the register;
2. programming FLASH Space must be 32 Bits are small units. byE2PD0~3 Set data;
3. The destination address is determined by EEAR Register given, address EEAR[1:0]Will be ignored

by E2PCTL Read and write program FLASH Space, can realize online program update (IAP)The function is very useful in some applications that need to update application data on site and provide customized product updates.

E2PCTL Interface operation process

E2PCTLThe controller is working mainly through4Implementation of a register, respectivelyE2PCTLControl status registerEECR, ECCR; Data register EEDR(E2PD0~E2PD3)And address register EEAR(EEARL/EEARH).

ECCR Register is used to set E2PCTL Working status, most of the status needs to be in E2PCTL The setup is completed before work, and this process is generally implemented during system initialization.ECCR In the register SWM This bit is used to enable continuous write mode. This control bit needs to be set during continuous write operation.

EECR The register is used to control the selection operation type, and is used to select operation instructions, such as setting read and erase commands.

EEDR Register is used 8 Bit byte mode interface,E2PD0~3 Used for 32 Bit pattern read and write operations;

EEAR The register is used to set the target address of reading and writing, and also used to set the page address of the page erase operation. Page address is Aligned in page units, the size of a page is 1K Bytes, need attention EEAR The specified address is a byte address.

by E2PCTL Interface access FLASH Program space:

by E2PCTL The interface can be implemented FLASH Reading and writing and erasing of program space. CorrectFLASH Reading and writing of space only supports hold32 Bit access width. The erase operation is in page units, the size of each page1K byte(256x32). WritingFLASH Before the program space, first erase the page where the target address is located.E2PCTL write FLASH Program space Continuous mode is not supported, users need to complete write operations in sequence. The following is erasedFLASH The flow of the program space:

1. program FLASH Page erase operation

Z Set up EEAR[14:0]For the target page address that needs to be erased, the program FLASH One page size is 1K Bytes, therefore EEAR[14:10]Will be used as the page address,EEAR[9:0]Set as 0

Z Set up EEPM[3:0] = 1X01,among them EEPM[2]Can be set to 0 or 1

Z Set up EEMPE = 1, Simultaneously EEPE = 0

Z In four cycles, set EEPE = 1,starting program FLASH Wipe process

2. program FLASH Programming operation

Z write E2PD0~3,ready 32 Bit programming data setting EEAR Is the target

Z address, where the address is 4 Byte alignment setting EEPM[3:0] = 1X10,

Z among them EEPM[2]Can be set to 0 or 1

Z Set up EEMPE = 1,Simultaneously EEPE = 0

Z In four cycles, set EEPE = 1,start up FLASH Programming process *by E2PCTL Interface access E2PROMSimulation space:*

E2PCTL Controller through simulation E2PROM Interface logic to access data FLASH space. simulationE2PROM stand by 8 Bit,16 Bit and 32 Read and write access to bit data width.8 Bit byte pattern pair E2PROMThe interface has better compatibility. 32 Bit mode helps to improve storage efficiency and FLASH Life, therefore 32 The bit read and write mode is the recommended read and write mode.E2PROMThe analog interface supports continuous read and write mode. It has obvious advantages in data applications that need to update multiple continuous addresses at a time and is recommended.

for LGT8F88P/168P,data FLASH For independent storage space. No need to passECCR Register configuration and use canFLASH Data space.LGT8F328P There is no independent data FLASH Space, data FLASH And program FLASH Total enjoy 32K byte FLASH space. Need to passECCR Register enable data FLASH Partition function and pass ECCR Deposit Mechanical ECS[1:0]Bit configuration data FLASHthe size of. After the configuration takes effect, Other methods of use and LGT8F88P/168P the same.

FLASH The controller is implementing E2PROMWhen the interface is used, the internal data has been automatically erased when necessary FLASH Logic, so EPROMThe erase command is optional. This command is only used when the user needs to perform the erase alone. EECR Register control FLASH Erasing/writing timing, including program FLASH with E2PROM. The specific operation type needs to pass EECR Register EEPME with EEPM[3:0]set up. CorrectE2PROMThe read operation is relatively simple, after setting the target address and mode, write EERE Corresponding to the target address 32 Bit data read in FLASH Inside the controller, the user can pass EEDR The register reads the byte of interest.FLASH The controller does not implement the program FLASH Space read operation, users can use it conveniently LPM Or through the program FLASH Used at the address of the data unified mapping space LD/LDD/LDS The instruction is read.

1. 8 Bit pattern, programming E2PROM

Z Set the target address to EEARH/L register

Z Set new data to EEDR register

Z Set up EEPM[3:1] = 000,EEPM[0]Can be set to 0 or 1

Z Set up EEMPE = 1, Simultaneously EEPE = 0

Z In four cycles, set EEPE = 1

When the setting is complete,FLASH The controller will start the programming operation, during programming CPU It will remain at the current instruction address and will not continue to run until the operation is completed. During the programming process, if you need to erase dataFLASH, FLASH The controller will automatically start the erasing process.

2. 32 Bit pattern, programming E2PROM

Z by E2PD0~3,ready 32 Bit data sets the target address to EEARH/L register. Note that this is a byte-

Z aligned address,FLASH For controller EEAR[15:2]As a visit FLASH the address of. Set upEEPM[3:1] = 010,EEPM[0]Can be set to 0 or 1

Z

Z Set up EEMPE = 1, Simultaneously EEPE = 0

Z In four cycles, set EEPE = 1

3. 8 Bit pattern, read E2PROM

Z Set the target address to EEARH/L register

Z Set up EEPM[3:1] = 000

Z Set up EERE = 1 start up E2PROMRead operation waiting 2 Cycles

Z (execute two NOP Operation) The data corresponding to the target

Z address is updated to EEDR register

4. 32 Bit pattern, read E2PROM

Z Set up EEARH/L Is the target address, the address is 4 Byte

Z alignment setting EEPM[3:1] = 010, Turn on 32 Bit interface mode

Z setting EERE = 1,start up E2PROMRead operation waiting 2 System

Z clock cycles (execute two NOP instruction)

E2PCTL Access simulation E2PROMSpace, support continuous programming mode, continuous access mode is very efficient for applications that need to update one data block at a time, and also help to improve FLASH Service life. Continuous programming mode only supports32 Data programming operation of bit width.

Continuous access mode passed ECCR Register SWMBit enable.SWMAfter enabling, the next pass E2PCTL Write simulation E2PROMSpace operations are in continuous programming mode. In continuous programming mode,E2PCTL The controller will automatically process the page change according to the data situation in the target address. However, if a page change occurs during the continuous programming mode, the controller will not automatically change the CP0/1 The data exchange of the area will not update the page information.

When continuous programming to the next operation, clear through SWMBit to turn off the continuous programming mode, and then in the non- SWMA programming operation after starting in the mode, after the programming is over,E2PCTL Will automatically CP0/1 The data of the area is copied to the swap page, and the information of the swap page is updated to make it the current valid page, thereby completing the entire continuous programming operation.

5. Continuous programming mode operation process:

1. by ECCR Configuration Data FLASH Size and enable SWMBit
2. use 32 Bit mode programming simulation E2PROMarea
3. If it is not the next operation, go back to step 2 Continue to program the next data
4. If it reaches the next programming, first pass SWMDisable continuous programming mode, then use steps 2 Programming once after the completion of the operation flow

E2PCTL Efficient FLASH Data management

E2PCTL In addition to the continuous programming mode, the controller can also pass ECCR Register CP0/1 Bits carry out independent control of data exchange and copying during the page swapping process.ECCR Register CP0/1 Respectively used to control the page exchange process for the current page CP0/1 Exchange operation of area data. ClearedCP0/1 Bit, the data of the corresponding area in the current page will not be exchanged during the page exchange process. An efficient management method provided in this section will take advantage of this feature.

in FLASH During the data update process, a time-consuming operation occurs during the swap page erasing process. Therefore, we can address a data management method that greatly reduces the number of page erases, which can improve programming efficiency and reduce life loss.

Here we provide a reference algorithm for data management applications based on data blocks:

1. Assuming that user data is just a complete data block, the size of the data block 4 Integer multiples of bytes;
2. Each data update will update a complete data block
3. In addition to storing user data, data block information also needs to store a block management information

Under the above three conditions, we can make full use of E2PCTL Continuous programming mode and automatic page switching mechanism to achieve An efficient FLASH Data management methods.

Since the data updated each time is a data block of the same size, and the address information pointing to the next block of data is stored in the data structure of each block, we can program in the order of the address each time the data is updated FLASH, No need to do CP0/1 Data replication. At the same time, since the data is updated to an erased area each time, page erasure will not occur.

When the data is written, the next data area pointed to by its structure information returns to the starting address of the page. After that, data write operations occur again,E2PCTL A page erasing process will be initiated and the current active page will be updated.

FLASH Operational protection measures

in case VCC The voltage is low,FLASH Erase and write operations may cause errors due to low voltage.

FLASH/There may be two reasons for the error of data erasing and writing operation under low pressure. First of all, normalFLASH Erase and write operations require a small operating voltage. Below this voltage, the operation will fail and cause data errors. The second reason is that the core is running at a certain frequency and also requires a small voltage requirement. When the voltage is lower than this voltage, it will cause an error in the execution of instructions, which willFLASH An error occurred in the operation.

You can avoid similar problems in the following simple ways:

When the supply voltage is low, let the system enter the reset state. This can be achieved by configuring the internal low-voltage detection circuit (VDT) achieve. in caseVDT It is detected that the current working voltage is lower than the set threshold,VDT A reset signal will be output. in case VDT The threshold value of can not meet the needs of the application, you can consider adding a reset circuit externally.

Register description

FLASH Address register- EEARH/EEARL

EEARH/EEARL		
EEARH: 0x22 (0x42) EEARL: 0x21 (0x41)		Defaults: 0x0000
bits	EEAR[15:0]	
R/W	R/W	
Bit definition		
[7:0]	EEARL	EFLASH/E2PROMLow access address 8 Bit.
[14:8]	EEARH	EFLASH/E2PROMHigh access address 7 Bit
[15]	-	reserved

When using E2PCTL Controller access program FLASH Area,EEAR[14:2]For access to 4 Byte aligned integer A program space.EEAR[1:0]Only accessing data registers EEDR When used. For details, please refer to the following aboutEEDR The description of the data register.E2PCTL Controller support 8/16/32 Bit mode, no matter which mode, here EEAR All addressing is byte-aligned.

FLASH Data register- EEDR/E2PD0

EEDR/E2PD0 – FLASH/E2PROMData register 0		
EEDR/E2PD0: 0x20 (0x40)		Defaults: 0x00
bits	EEDR[7:0]	
R/W	R/W	
Bit definition		
[7:0]	EEDR E2PD0	E2PCTL Data register 16/32 In bit mode, it is used to access the low byte

FLASH Data register- E2PD1

E2PD1 – E2PCTL Data register 1	
E2PD1: 0x5A	Defaults: 0x00
bits	E2PD1[7:0]

R/W	R/W	
Bit definition		
[7:0]	E2PD1	16 Used for storage in bit mode 16 High of bit data 8 Bit 32 Used to store low in bit mode 16 High of bit data 8 Bit

FLASH Data register- E2PD2

E2PD2 – FLASH Data register 2		
E2PD2: 0x57	Defaults: 0x00	
Bits	E2PD2[7:0]	
R/W	R/W	
Bit definition		
[7:0]	E2PD2	32 Used to store high in bit mode 16 Bit data low 8 Bit

FLASH Data register- E2PD3

E2PD3 – FLASH Data register 3		
E2PD3: 0x5C	Defaults: 0x00	
Bits	E2PD3[7:0]	
R/W	R/W	
Bit definition		
[7:0]	E2PD3	32 Used to store high in bit mode 16 High of bit data 8 Bit

FLASH Mode control register- ECCR

ECCR – FLASH/E2PROMConfiguration register								
ECCR: 0x36 (0x56)					Defaults: 0x0C			
bits	WEN	EEN	ERN	SWM	CP1	CP0	ECS1	ECS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	1	1	0	0
Bit definition								
[7]	WEN	ECCRWrite enable control Modifying ECCR Before, you must first WEN write 1And then in 6 Update ECCR The contents of the register						
[6]	EEN	E2PROMEnable, only for LGT8F328P effective 1:Enable E2PROMSimulation, will start from 32KFLASH Reserve some space in 0: Disable E2PROMsimulation,32KFLASH All for program space writing 1 Will						
[5]	ERN	reset E2PCTL Controller continuous write mode, suitable for simulation						
[4]	SWM	E2PROMController operation page exchange CP1 Zone enable control						
[3]	CP1							
[2]	CP0	Page swap CP0 Zone enable control						
[1:0]	ECS[1:0]	E2PROMSpace configuration 00:1KB E2PROM, 30KB program FLASH 01 :2KB E2PROM, 28KB program FLASH						

		10:4KB E2PROM, 24KB program FLASH 11 :8KB E2PROM, 16KB program FLASH

FLASH Access control register- EECR

EECR – FLASH/E2PROMControl register								
EECR: 0x1F (0x3F)				Defaults: 0x00				
bits	EEPM3	EEPM2	EEPM1	EEPM0	EERIE	EEMPE	EEPE	EERE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial value	0	0	0	0	0	0	0	0
Bit definition								
[7:4]	EEPM[3:0]	EFLASH/EPROMAccess mode control bit						
		[3]	[2]	[1]	[0]	Mode description		
		0	0	0	x	8 Bit pattern read/write E2PROM(default)		
		0	0	1	x	16 Bit pattern read/write E2PROM		
		0	1	0	x	32 Bit pattern read/write E2PROM		
		1	x	0	0	E2PROMErase (optional operation)		
		1	x	0	1	program FLASH Erase (page erase)		
		1	x	1	0	program FLASH Programming		
		1	x	1	1	Reset FLASH/E2PROMController		
[3]	EERIE	FLASH/E2PROMReady interrupt enable control. write1 Enable, write 0 Prohibited. when EEPE After being automatically cleared by hardware,E2PROM The ready interrupt is active. inEPROM During operation, this interrupt will not be generated						
[2]	EEMPE	FLASH/E2PROMProgram operation enable control bit EEMPE For control EEPE Is it valid, when set EEMPE for 1,EEPE for 0 After, in the next four cycles, set EEPE for 1 The programming operation will start. Otherwise, the programming operation is invalid. After four cycles,EEMPE Automatically cleared						
[1]	EEPE	FLASH/E2PROMProgram operation enable bit						
[0]	EERE	E2PROMRead the enable bit, the data will be valid after two system cycles						

Universal I/O register- GPIOR2

GPIOR2 – Universal I/O register 2		
GPIOR2: 0x2B (0x4B)		Defaults: 0x00
Bits	GPIOR2[7:0]	
R/W	R/W	
Initial value	0x00	
Bit definition		
[7:0]	GPIOR2	Universal I/O register 2, Used to store user-defined data

Universal I/O register- GPIOR1

GPIOR1 – Universal I/O register 1		
GPIOR1: 0x2A (0x4A)		Defaults: 0x00
Bits	GPIOR1[7:0]	
R/W	R/W	
Initial value	0x00	
Bit definition		
[7:0]	GPIOR1	Universal I/O register 1, Used to store user-defined data

Universal I/O register- GPIOR0

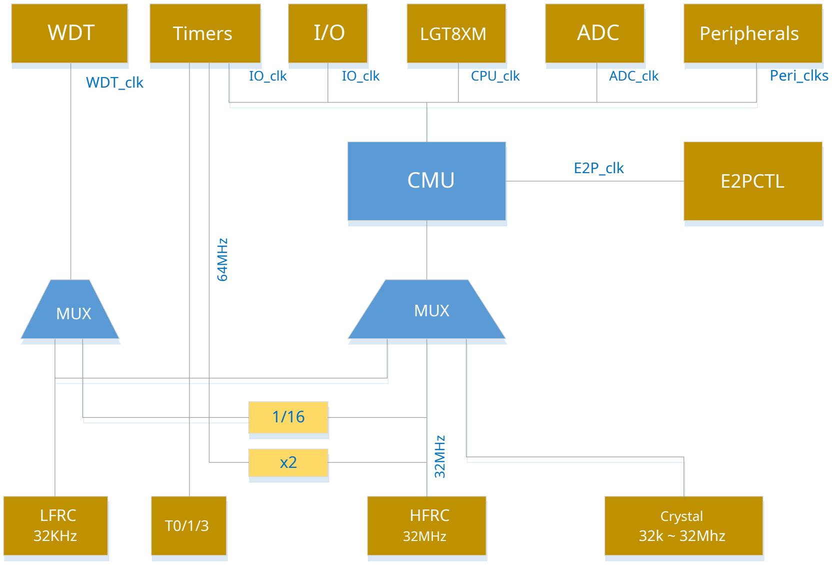
GPIOR0 – Universal I/O register 0		
GPIOR0: 0x1E (0x3E)		Defaults: 0x00
Bits	GPIOR0[7:0]	
R/W	R/W	
Initial value	0x00	
Bit definition		
[7:0]	GPIOR0	Universal I/O register 0, Used to store user-defined data

System clock and configuration

System clock distribution

LGT8FX8P Support multiple clock inputs. The system can work in three main clock sources, which are internal32KHz Can be calibrated RC Oscillator, internal 32MHz Can be calibrated RC Oscillator and external 400KHz ~ 32MHz Crystal input. The picture below isLGT8FX8P Clock system distribution,CMU It is the center of the entire clock management, responsible for the frequency division of the system clock, generating independent clocks for different modules and controlling the clocks, etc. In general applications, it is not necessary for all clocks to work at the same time. In order to reduce system power consumption, system power consumption management shuts off unused module clocks according to different sleep modes.

For specific operation details, please refer to the relevant chapters on power management.



CPU_clk

Used to drive LGT8XM Kernel and SRAM Running. Such as driving general working registers, status registers, etc. CPU After the clock stops, the core will not continue to execute instructions and perform calculations. System executionSLEEP After the instruction enters the sleep mode, the core clock will be turned off.

Peri_clk

Used to drive most peripheral modules, such as timers/counters,SPI,USART Wait.IO The clock is also used to drive the external interrupt module. When the peripheral clock is stopped due to sleep, some peripheral parts that can be used to wake up the system work in independent clock or asynchronous mode. such asTWI The address recognition function can wake up most sleep modes, and the address recognition part works in asynchronous mode at this time.

E2P_clk

E2P_clk The clock is used to generate FLASH Interface access timing.E2P_clk Generate visits E2PCTL access FLASH Timing of the interface.E2P_clk Fixed from inside 32MHz HFRC Oscillator 32 Frequency division (1MHz). If the user needs to useE2PCTL Module reads and writes internal program FLASH Or data FLASH Space, need to enable internal 32MHz Oscillator.

Asy_clk

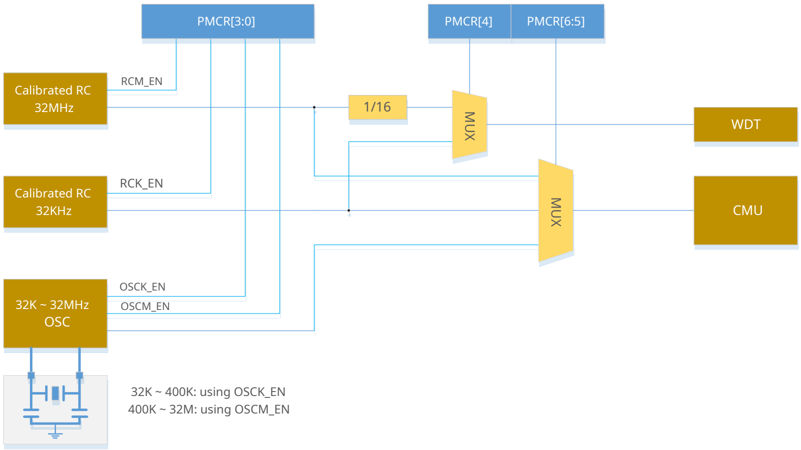
Asynchronous timer clock. The timer/counter can directly use an external clock or crystal oscillator (32.768K)drive. This independent clock mode can keep the timer running while the system is processing sleep mode.

WDT_clk

Internal watchdog timer clock source, can be configured to select internal 32KHz LFRC Oscillator, or from internal 32MHz HFRC of 16 Frequency division (2MHz). After the system is powered on, the default clock source of the watchdog is32KHz LFRC Oscillator.

Clock source selection

LGT8FX8P stand by 4 Clock source input, the user can pass PMCR Register realizes the enable control of the clock source And complete the switch of the main clock. Below isPMCR The control structure diagram:



LGT8FX8P internal OSC The oscillator can work in two modes, high frequency and low frequency. The actual size of the control internal OSC The oscillator is working in the correct mode. Same internalRC Oscillators are also divided into high frequency and low frequency.PMCR Register low 4 Bits are used to control these four clock sources. The control relationship is as follows:

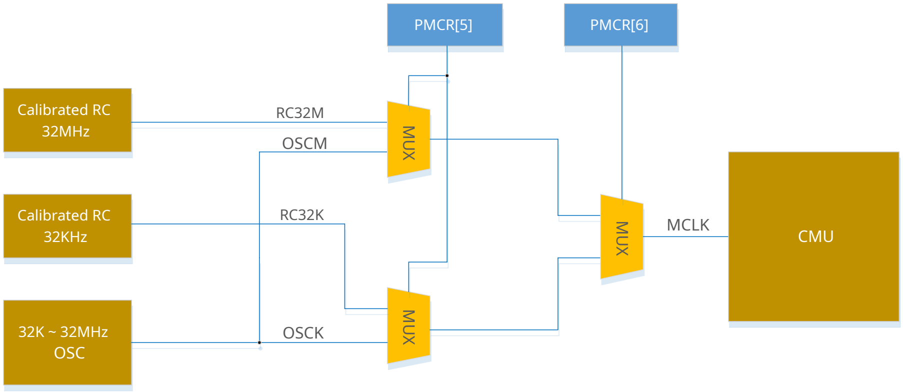
PMCR	Corresponding clock source
PMCR[0]	32MHz RC Enable control,1 Enable,0 shut down
PMCR[1]	32KHz RC Enable control,1 Enable,0 shut down
PMCR[2]	400K ~ 32MHz OSC Mode enable,1 Enable,0 shut down
PMCR[3]	32K ~ 400K OSC Mode enable, 1 Enable,0 shut down

LGT8FX8P After the system is powered on, it is used by default 32MHz RC As the system clock source, the core works on the clock source 8 Minute frequency(4MHz). Users can setPMCR Register and system prescaler register (CLKPR)Change the default configuration.

If the user needs to change the configuration of the main clock source, it is necessary to ensure that the switched clock source is in a stable working state before switching the clock. Therefore, before switching the main clock source, passPMCR[3:0]Enable the required clock source and wait for the clock to stabilize before switching.

When the user switches the main clock to an external crystal oscillator, although the user has enabled the external crystal oscillator, it cannot be ruled out that the crystal oscillator cannot start oscillating due to configuration errors or failure of the crystal oscillator. If you switch to an external crystal oscillator at this time, the system will stop working after the switch. Therefore, from the perspective of system reliability, it is recommended to turn on the watchdog timer to avoid such problems from the perspective of software design.

After the clock source is enabled and waits for stability, you can pass PMCR[6:5]Switch the main clock. among themPMCR[5]Used to select is inside unitRC Oscillator and external crystal oscillator,PMCR[6]Used to select high-speed clock source and low-speed clock source.



Main clock source selection:

PMCR[6]	PMCR[5]	Main clock source
0	0	internal 32MHz RC Oscillator (system default)
0	1	external 400K ~ 32MHz Inside the high-speed crystal
1	0	oscillator 32KHz RC Oscillator
1	1	external 32K ~ 400KHz Low-speed crystal oscillator

Clock source control timing

For protectionPMCRThe register was accidentally modified, rightPMCRThe modification of the register needs to strictly install the specified time sequence.

PMCR The high bits of the register (PMCR[7])Used to achieve timing control. User is editingPMCR Before the other bits, you must first PMCR[7]Set 1, In the home 1 After operation 6 Within cycles, change PMCR The value of other registers.6 Cycles later, right PMCR The direct modification of will be invalid.

Take switching to an external high-speed crystal oscillator as an example, and list the recommended operating steps:

(1) Enable clock source

Z Set up PMCR[7] = 1

Z In six cycles, set PMCR[2] = 1, Enable the external high-speed mode, the external crystal oscillator waits for the external

Z crystal oscillator to stabilize (the waiting time varies with different crystal oscillators, generally us Level wait)

(2) Switch the main clock source

Z Set up PMCR[7] = 1

Z In six cycles, set PMCR[6:5] = 01, The system will automatically switch the working clock to the external crystal oscillator to perform

Z several NOP Operation to improve stability (optional operation)

[Note]: In the above operation of switching the main clock, ensure that the current system clock is working normally. After switching to the external crystal oscillator, you can turn off the previous internal RC Oscillator.

System clock prescaler control

LGT8FX8P There is a system clock prescaler inside, which can be passed through the clock prescaler register (CLKPR)Take control. This function can be used to reduce system power consumption when the system does not require very high processing power. The prescaler setting is valid for all clock sources supported by the system. The clock prescaler can affect the core execution clock and all synchronized peripherals.

When switching between different clock prescaler settings, the system clock prescaler ensures that there will be no glitches during the switching process, but will also ensure that there will be no intermediate states of excessive high frequency. The frequency division switch is executed immediately. When the register change takes effect,

More in 2~3 After the current system clock cycle, the system clock is switched to the new frequency-divided clock.

In order to avoid misoperation of the clock divider register, CLKPR The modification must also follow a special timing flow Cheng:

z Set the clock prescaler change enable bit (CLKPCE)for 1,CLKPR The other positions are 0

z In four cycles, write the required value CLKPS,Simultaneously CLKPCE write 0

Before changing the clock prescaler register, the interrupt function needs to be disabled to ensure that the write sequence can be completed. About the main clock prescaler registerCLKPR For the specific definition, please refer to the register description part of this chapter.

internal RC Oscillator calibration

LGT8FX8P Internally contains two calibratable RC Oscillator, after calibration, can reach±1%Accuracy within. its in 32MHz RC The default is used for the system working clock.

LGT8FX8P Before delivery, internal 32MHz HFRC with 32KHz LFRC Have been calibrated, and the calibration value has been written into the system System configuration information area. In the process of system power saving, these calibration values will be read into the internal register, and the register will beRC Frequency recalibration.

The calibration register is located in IO Address space, user program can read and write. For applications with special frequency requirements, the frequency output of the internal oscillator can be adjusted by modifying the calibration register. Modifying the calibration register will not change the factory configuration information. When the system is re-powered or the configuration bit reload operation initiated by the user, the calibration register will be restored to the factory settings.

Register definition

32MHz HFRC Oscillator Calibration Register- RCMCAL

RCMCAL – 32MHz HFRC Calibration register		
RCMCAL: 0x66		Default value: factory configuration
Bits	RCCAL[7:0]	
R/W	R/W	
Bit definition		
[7:0]	RCCAL	After the system is powered on, the value of the register will be RC The calibration value is replaced.

32KHz RC Oscillator Calibration Register- RCKCAL

RCKCAL – 32MHz RC Calibration register	
RCKCAL: 0x67	Default value: Factory setting
Bits	RCKCAL[7:0]
R/W	R/W
Bit definition	
[7:0]	RCKCAL Write calibration value RCKCAL Register completion 32KHz RC Oscillator calibration

Clock source management register- PMCR

PMCR – Clock source management register								
PMCR: 0xF2				Defaults: 0x03				
Bits	PMCE	CLKFS/CLKSS	WCLKS	OSCKEN	OSCMEN	RCKEN	RCMEN	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Bit definition								
[0]	RCMEN	internal 32MHz RC Oscillator enable control,1 Enable,0 Prohibited inside						
[1]	RCKEN	32KHz RC Oscillator enable control,1 Enable,0 Disable external high-frequency						
[2]	OSCMEN	crystal oscillator enable control,1 Enable,0 Disable external low-frequency						
[3]	OSCKEN	crystal oscillator enable control,1 Enable,0 Prohibit						
[4]	WCLKS	WDT Clock source selection, 0 – Select internal 32MHz HFRC Oscillator 16 Frequency division 1 - internal 32KHz LFRC Oscillator						
[5]	CLKSS	Main clock source selection control, select the type of clock source, please refer to the clock source selection section						
[6]	CLKFS	Main clock source frequency control, select the clock frequency type, please refer to the clock source selection section						
[7]	PMCE	PMCR Register change enable control bit. Changing PMCR Before other positions, you must first set this bit, and then set the value of other bits within four cycles.						

Main clock prescaler register- CLKPR

CLKPR – Main clock prescaler register								
CLKPR: 0x61				Defaults: 0x03				
Bits	WCE	CKOEN1	CKOEN0	-	PS3	PS2	PS1	PS0
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
Bit definition								
[3:0]	CLKPS	Clock prescaler selection bit						
		PS3	PS2	PS1	PS0	Frequency division parameters		
		0	0	0	0	1		
		0	0	0	1	2		
		0	0	1	0	4		
		0	*0*	*1*	*1*	*8(default allocation)*		

		0	1	0	0	16
		0	1	0	1	32
		0	1	1	0	64
		0	1	1	1	128
		1	0	0	0	256
		Other value				Keep
[4]	-	Keep unused				
[5]	CKOEN0	Set whether the system clock is PB0 Output on pin				
[6]	CKOEN1	Set whether the system clock is PE5 Output on pin				
[7]	WCE	Clock prescaler change clock control Changing CLKPR Before other bits of the register, it must be set separately CKWEN for 1, And then set other bits in the next four system cycles. After the four cycles are over,CKWEN Automatically cleared.				

Power management

Overview

Sleep mode reduces system power consumption by turning off the system clock and clock module.LGT8FX8P Provides a very flexible and diverse sleep mode and module controller, users can realize the ideal low-power configuration according to the application.

LGT8FX8P When entering the sleep mode, the analog function module will not be automatically closed, such as ADC,DAC,Comparators (AC), Low voltage reset module (LVD)And so on, the software needs to turn off unnecessary simulation functions before going to sleep according to application requirements, and restore the correct state after the system wakes up.

LGT8FX8P Support multiple sleep modes, including ADC Dedicated noise cancellation mode, used to eliminate ADC The digital part of the conversion process ADC Power interference. In addition, the others are all power control modes, which are divided into five types:

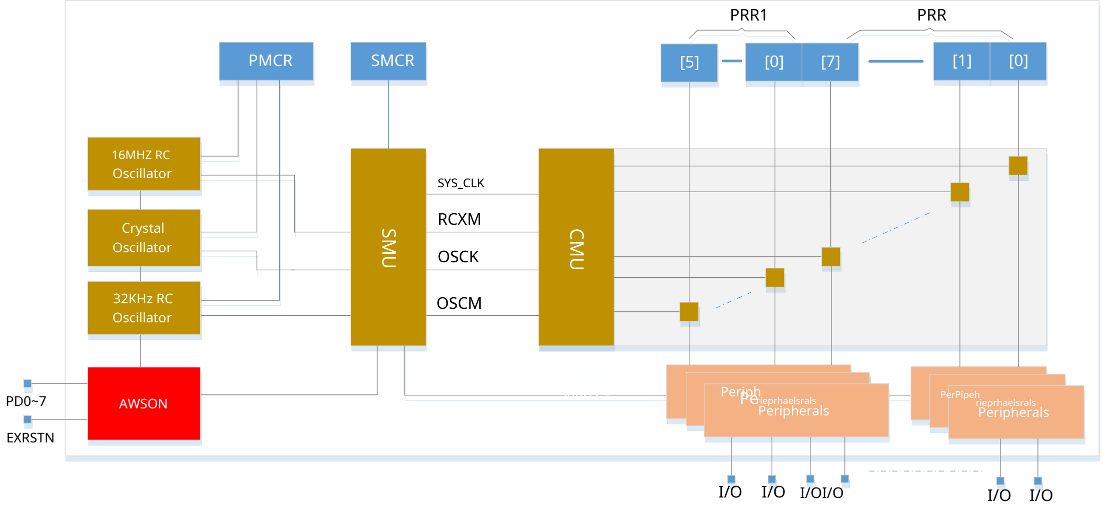
Sleep mode	Function Description
Idle mode (IDLE)	Just turn off the core clock, other peripheral modules work normally, all valid interrupt sources can wake up the core
Power saving mode (Save)	versus DPS0 The same pattern,Save The mode is and LGT8FX8D Stay compatible with Save The
Power down mode (DPS0)	mode is the same, and the supported wake-up sources include: z All pin level changes z Watchdog wakes up at regular intervals z Asynchronous mode TMR2 wake
Power down mode (DPS1)	Turn off all internal and external oscillators, support wake-up sources include: z External level change of all pins z External Interrupt 0/1 z Work on 32K LFRC The watchdog timer turns off the core power, low-
Power down mode (DPS2)	power mode, and the supported wake-up sources include: z External reset z PORTD Pin level change z LPRC Timed wake up (128ms/256ms/512ms/1s) Need to pay attention to, from DPS2 The wake-up process is the same as power-on reset

LGT8FX8P Support deep sleep DPS2, In this mode, the system LDO In the power-down state, the core registers, All peripheral controllers and SRAMEtc. are in the power-down state, the data in it will not be maintained.FLASH The storage unit will also be in a power-down state, so DPS2 Mode can achieve low power consumption of the system. Power-down mode can be passed through the portD(PORTD) Wake up on pin change, you can also choose5Level timing wake up. Used to wake upDPS2Because the timer does not support calibration, the accuracy is 15%It is only suitable for low-precision timing wake-up applications.

System from DPS2 Mode wake up, it will be turned on first LDO, This process is the same as the power-on process. The chip will perform a complete power-on reset startup process, load the configuration information, and then run the program from the address pointed to by the reset vector.

except DPS2 In other modes, the internal power supply will not be turned off. During the sleep process, all register information and RAM No data will be lost. After waking up, the core continues execution from the next instruction before sleep.

Schematic diagram of system power management:



As shown in FIG,LGT8FX8P Mainly through the sleep mode controller (SMU)And the clock management unit (CMU)Control The power consumption of each system. From the level of power saving, we can divide power consumption into4 Levels:

The first level is through PRR The register controls the working clock of the module, and saves the dynamic power consumption of system operation by turning off the clock of the unused module. Under normal circumstances, the power consumption that this level can save is not obvious.

The second stage is to switch the main clock source to the low-frequency clock, and turn off the unused clock source modules and other analog modules. This mode can basically get very considerable system running power consumption and sleep power consumption.

The third level is by letting the system enter the power-down mode (DPS1),DPS1 Mode LGT8FX8P Polar standby power consumption can be obtained. After waking up from the power-down mode, the software can passMCUSR The register reads the state before reset.

The fourth level is power-down mode (DPS2), This mode will turn off the core power supply, which can achieve low system power consumption. Because the kernel power is turned off, all data information will be lost in this mode. After waking up, a power-on reset process is executed immediately, and the system restarts from the reset vector.

AWSON Power management

versus LGT8FX8D Compared to power-down mode DPS2 It is a brand new power consumption mode.DPS2 The mode is used for applications that have higher requirements for sleep power consumption. enterDPS2 After the mode, the system only maintains a static module (AWSON)In the working state, other circuits are in a completely power-down state.

AWSON Module dedicated to responsible DPS2 Mode sleep and wake-up control,AWSON The module is mainly composed of IO Wake-up control logic and a low-power LPRC composition. Software can passIOCWK Register and DPS2R Register implementation pair AWSON control.

IOCWK Register is used to control PD0~7 Wake-up function of level change.DPS2R Register is used to control DPS2 Pattern and LPRC Function mode. For specific information, please refer to the register definition section at the end of this section.

use DPS2 Before mode, software settings IOCWK Enable required wake-up IO, Or through DPS2R Register enable LPRC And configure the timing wake-up period, and then pass DPS2R Register DPS2EN Bit enable DPS2 mode. After the setting is completed, the software needs to passSMCR Register settings DPS2 Sleep mode, then execute SLEEP Command to enter sleep.

Sleep mode and wake-up source

LGT8FX8P stand by 5 A sleep mode, the user can select the appropriate sleep mode according to application requirements.SMCR The register contains the control settings of the sleep mode, and executes SLEEP After the instruction, the core enters sleep mode. In order to obtain a more ideal sleep power consumption, it is recommended to turn off all unused clocks and analog modules before the core enters the sleep mode. But it should be noted that some wake-up sources require a working clock. If you need to use this kind of wake-up source, please keep the relevant clock source

Working status.

Sleep mode and wake-up mode:

Sleep mode	Effective clock				Wake-up source							
	Inside nuclear Time bell	outer Assume Time bell	ADC Time bell	different step Time bell	lead foot Electricity level change Transform	outer unit in Cut off 0/1	TWI Ground site match Match	TMR2 in Cut off	ADC turn change Knot bundle	Look Gate dog overflow Out	outer Assume in Cut off	PD Electricity level change Transform
Idle mode (IDLE)		X	X	X	X	X	X	X	X	X	X	X
ADC Noise suppression			X	X	X	X	X	X	X	X		X
Power saving mode (SAVE)				X	X	X	X	X		X		X
Power down mode (DPS0) (With RC32K)				X	X	X		X		X		X
Power down mode (DPS1) (Without RC32K)				X	X	X		X				X
Power down mode (DPS2) (Without LDO)												X

If you need to enter the above 5 A sleep mode,SMCR middle SE Position must be set 1To enable sleep mode control. then Execute one SLEEP Just order.SMCR middle SM0/1/2 Used to select different sleep modes. For specific information, please refer to the description below.

in MCU In sleep mode, if the wake-up source is valid,MCU Will be at 4 Was awakened after three cycles and continued to execute instructions. If the interrupt remains valid, the interrupt will respond immediately and enter the interrupt service subroutine. If inSLEEP A system reset occurred in the mode,MCU It will also be awakened and start execution from the reset vector.

when MCU In Power/Off In mode, the system can pass external interrupt INT0/1 Wake up, after wake up MCU Will be from sleep Continue execution at the previous position.

Idle mode (IDLE)

when SM2...0 Set as 000,carried out SLEEP After the instruction,MCU Into IDLE mode,IDLE The mode will turn off the core working clock, and other peripherals can work normally.

IDLE The mode can be awakened by external interrupts and internal interrupts. If you don't need to use a comparator andADC As a wake-up source, it is recommended to turn it off.

IDLE Because the mode only shuts down the clock that the core runs, it does not get a significant reduction in power consumption.IDLE In mode, the kernel will also stop executing and fetching instructions, so the internal program can be reduced FLASH The operating power consumption.

but IDLE The mode has a more flexible wake-up mode, and users can obtain a more ideal operating power consumption by reducing the system main clock and turning off unnecessary modules.

ADC Noise suppression mode

when SM2…0 Set as 001,carried out SLEEP After the instruction,MCU enter ADC Noise suppression mode. In this mode, the core and most peripherals will stop working,ADC,External Interrupt,TWI Addresses match,WDT And the timer/ counter working in asynchronous clock mode 2 All can work normally.

ADC The noise always mode is mainly used for ADC Conversion provides a good working environment. Reduce the high frequency interference of the digital module to the analog conversion. After entering this mode,ADC Sampling conversion will be started automatically, and the converted data will be saved to ADC After the data register,ADC End of conversion interrupt will be MCU From ADC Wake up in noise mode.

Power saving mode (Save)

when SM2…0 Set as 010,carried out SLEEP After the instruction,MCU Into Save mode. In this mode, the system will turn off the working clocks of all modules. Because this mode turns off the working clocks of all modules, it can only be awakened through asynchronous mode, external interrupts,TWI Address matching and working in independent clock source mode WDT Both can generate a wake-up signal in this mode.

This mode can turn off all modules except the main clock source. In order to achieve a more ideal operating power consumption, it is recommended to switch the system main clock to internal before entering this mode32K RC Or external 32KHz The low-frequency crystal oscillator is then turned off so the unused clock sources and analog modules are turned off.

Power down mode DPS0

when SM[2:0]Set as 110,carried out SLEEP After the instruction,MCU Will enter DPS0 mode. enterDPS0 After, except for the internal 32KHz RC In addition, other clock sources are turned off. This mode can be interrupted by externalINT0/1 Wake up; if enabled WDT The interrupt function can also be passedWDT Realize timing wake-up.

Power down mode DPS1

when SM[2:0]Set as 011,carried out SLEEP After the instruction,MCU Will enter DPS1 mode. enterDPS1 After that, all clock sources of the system are shut down. This mode can be usedIO The level changes, the watchdog wakes up.

Power down mode DPS2

Set up SM[2:0]for 111And pass DPSR2 Register DPS2EN Enable AWSON Module, execute SLEEP After the instruction will enter DPS2 mode. enterDPS2 After the mode, the system turns off the kernel power. So the register andRAMThe data will be lost. FromDSP2 The wake-up process is the same as the power-on reset process.

DPS2 In the mode, because the core voltage is turned off and the register information is lost, the control state of the port will also be restored to the input state. IO The output driver and pull-up control will also be turned off.

FLASH Power control and fast wake-up

When the system is in SLEEP After the mode, the kernel will not continue to execute instructions, at this time you can choose to close FLASH Power supply to obtain lower standby power consumption. This function can be passedMCUCR Register FPDEN Bit control realization;

In power-down mode, the system can use external interrupts orWDT Wake-up, in order to filter out possible interference from external signals, the internal wake-up circuit includes a configurable filter circuit, and the user can select an appropriate filter width according to their needs. The configuration of the filter circuit can be passedMCUCR Register FWKPEN achieve.

MCUCR[FWKPEN]Filter width control:

FWKPEN	Filter width
0	260us (default)
1	32us

Register description

Sleep mode control register- SMCR

SMCR – Sleep mode control register					
SMCR: 0x33(0x53)		Defaults: 0x00			
Bits		SM2	SM1	SM0	SE
R/W	-	R/W	R/W	R/W	R/W
Bit definition					
[0]	SE	Sleep mode enable control bit, set to 1 After executing SLEEP Instruction, the core will enter sleep mode. SE The bit can protect the system from accidentally entering sleep mode. After waking up, it is recommended to clear it immediatelySE Bit.			
[3:1]	SM	Sleep mode selection			
		SM2	SM1	SM0	Mode description
		0	0	0	IDLE mode
		0	0	1	ADC Noise suppression mode
		0	1	0	Save mode
		0	1	1	DPS1 mode
		1	1	0	DPS0 mode
		1	1	1	DPS2 mode
		Others			Keep unused
[7:4]	-	Keep unused			

Power saving control register- PRR

PRR – Power saving control register							
PRR: 0x64				Defaults: 0x00			
PRR	PRTWI	PRTIM2	PRTIM0	-	PRTIM1 PRSPI	PRUART0	PRADC
R/W	R/W	R/W	R/W	-	R/WR/W	R/W	R/W
Bit definition							
[0]	PRADC	Set as 1,shut down ADC The controller clock is set to 1					
[1]	PRUART0	,shut down USART0 The clock of the module is set to 1					
[2]	PRSPI	,shut down SPI The clock of the module is set to 1,					
[3]	PRTIM1	Turn off the timer/counter 1 The clock is kept unused					
-	-						
[5]	PRTIM0	Set as 1, Turn off the timer/counter 0 The clock is					
[6]	PRTIM2	set to 1, Turn off the timer/counter 2 The clock is					
[7]	PRTWI	set to 1,shut down TWI Module clock					

Power saving control register- PRR1

PRR1 – Power saving control register 1								
PRR1: 0x65				Defaults: 0x00				
PRR1			PRWDT	-	PRTIM3	PREFL	PRPCI	-
R/W			R/W	-	R/W	R/W	R/W	-

Bit definition		
[0]	-	Keep unused
[1]	PRPCI	Set as 1, Turn off external pin changes and external interrupt module clock
[2]	PREFL	Set as 1,shut down FLASH The controller interface
[3]	PRTIM3	clock is set to 1,shut down TMR3 Controller's clock
[4]	-	Keep unused
[5]	PRWDT	Set as 1,shut downWDT The counter clock is
[7:6]	-	reserved

MCU Control register- MCUCR

MCUCR – MCU Control register								
MCUCR: 0x35(0x55)				Defaults: 0x00				
MCUCR	FWKEN	FPDEN	EXRFD	PUD	IRLD	IFAIL	IVSEL	WCE
R/W	R/W	R/W	R/W	R/W	W/O	R/O	R/W	R/W

Bit definition		
[0]	WCE	MCUCR Update the enable bit, in the update MCUCR Before, you need to set this bit first, and then 6 Complete the pairing in cycles MCUCR The update interrupt vector selection bit of the register, this position
[1]	IVSEL	1 After that, the interrupt vector address will be based on IVBASE The value of the register is mapped to the new address
[2]	IFAIL	The system configuration bit loading failure flag bit, 0 = Configuration information passed 1 = Failed to load configuration information
[3]	IRLD	write 1 System configuration information will be reloaded
[4]	PUD	Global pull-up prohibition bit 0 = Enable global pull-up control 1 = Close all IO Pull-up resistor
[5]	EXRFD	external reset filter disable bit 0 = Enable external reset (190us)Digital filter 1 = Digital filter circuit with external reset disabled
[6]	FPDEN	Flash Power/down Enable control 0:system SLEEP Rear FLASH Keep powered on 1:system SLEEP Rear FLASH Power-off fast wake-up mode enable
[7]	FWKEN	control, only for Power/Off Mode is valid 0:260us Filter delay 1:32us Filter delay

PD Group wake-up on change control register- IOCWK

IOCWK – PD Group wake-up on change control register								
IOCWK: 0xAE					Defaults: 0x00			
Bits	IOCD7	IOCD6	IOCD5	IOCD4	IOCD3	IOCD2	IOCD1	IOCD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	IOCWK	Set 1 Corresponding bit, enable PD group IO Wake-up on pin change function						

DPS2 Mode control register- DPS2R

DPS2R – DPS2 Mode control register								
DPS2R: 0xAF					Defaults: 0x00			
Bits	-	-	-	-	DPS2E	LPRCE	TOS1	TOS0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Bit definition								
[1:0]	TOS	LPRC Timed wake-up settings: 00 = 128ms 01 = 256ms 10 = 512ms 11 = 1s						
[2]	LPRCE	LPRC Enable control 0 = Disable LPRC Timer 1 = Enable LPRC Timer						
[3]	DPS2E	DPS2 Mode enable control bit 0 = Disable DPS2 mode 1 = Enable DPS2 mode						
[7:4]	-	Keep						

System control and reset

Overview

After the system reset, all I/O The registers are all set to their initial values, and the program is executed from the reset vector.LGT8FX8P Must use a RJMP – The relative jump instruction jumps to the reset handler. If the program does not use interrupts and the interrupt source is not enabled, the interrupt vector will not be used. The interrupt vector area can be used to store the user's program code.

After the reset is effective, all I/O The ports immediately enter their initial state. mostI/O The initial state is input and the internal pull-up resistor is turned off. With analog input functionI/O, Also initialized to a number I/O Features.

When the reset becomes invalid,LGT8FX8P The internal timer counter starts to be used for stretching reset. Stretching the width of the reset signal is used to ensure that the power and clock modules in the system enter a stable state.

Reset source

LGT8FX8P A total of six reset sources are supported:

Z Power-on reset: When the operating voltage of the system is low-voltage internal POR At the reset threshold of the module, the power-on reset

Z is valid. External reset: A low-level pulse with a certain width on the external reset pin of the chip, the external reset is effective. Watchdog reset:

Z After the watchdog module is enabled, if the watchdog timer expires, the system will reset. Low voltage reset:LGT8FX8P There is a low voltage

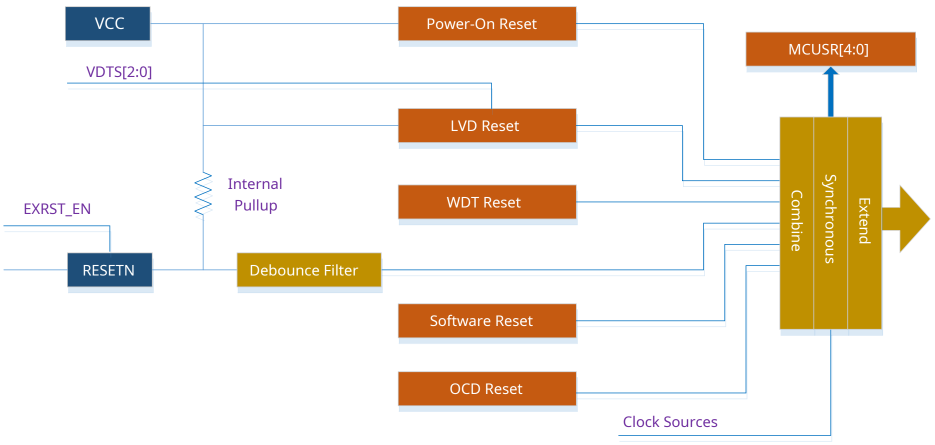
Z detection module inside (LVD), When the system working power is lower than LVD

When the reset threshold is set,MCU It will also be reset.

Z Software reset: LGT8FX8P There is a dedicated software-triggered reset register inside, the user can reset at any time through this register MCU.

Z OCD Reset:OCD The reset is issued by the debugger module for direct resetMCU Kernel.

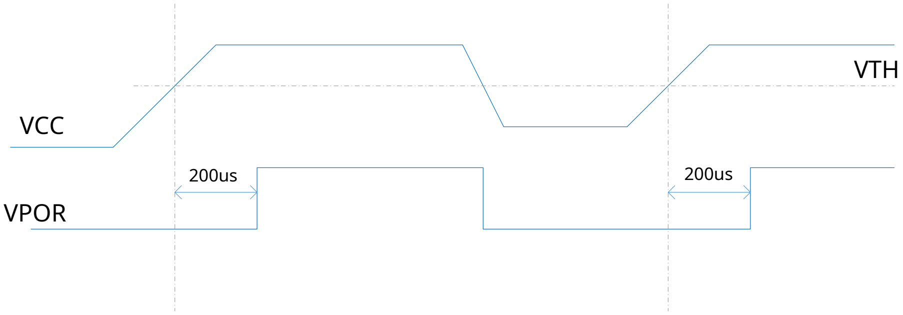
Reset system structure diagram:



Power-on reset

The power-on reset signal is generated by the internal voltage detection circuit. When the system power (VCC)When it is lower than the detection threshold, the power-on reset signal is valid. For the detection threshold of power-on reset, please refer to the electrical parameters section.

The power-on reset circuit can ensure that the chip is in a reset state during the power-on process, and the chip can start to run from a known stable state after power-on. The power-on reset signal will also be expanded by the counter inside the chip to ensure that the internal Kind of analog module, such as RC The oscillator, etc. can enter a stable working state.

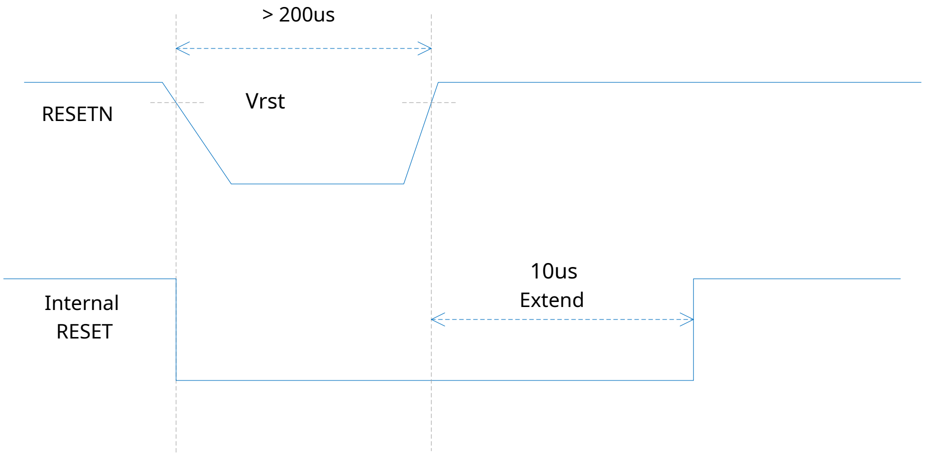


External reset

External reset pin (RSTN)When a low level is applied, the external reset is effective immediately. The width of the low level is greater than the requirement of a small reset

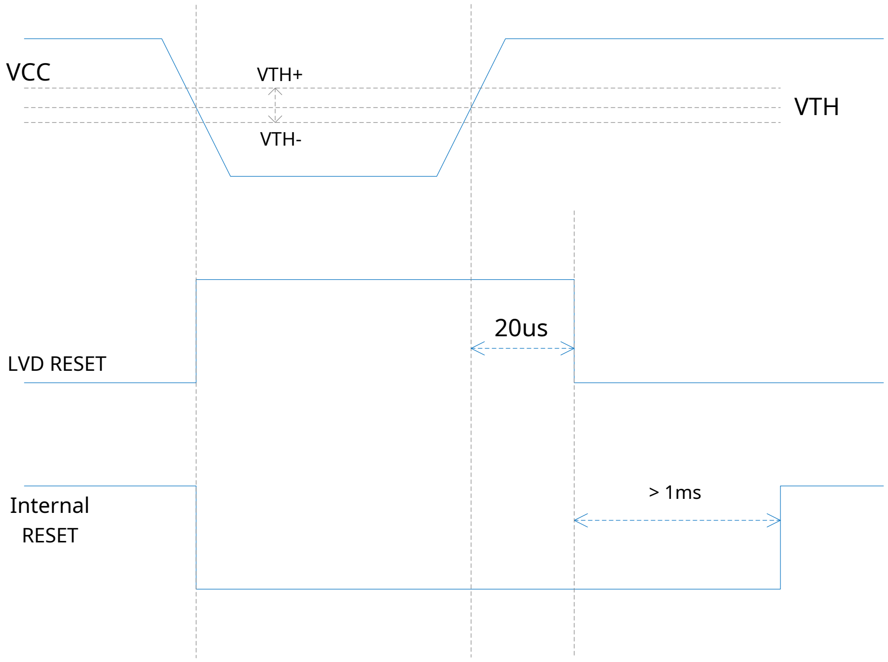
A pulse width. The external reset is an asynchronous reset. Even if the chip does not have a clock, the external reset can still be used

Enough to reset the chip.LGT8FX8P The external reset pin can also be used as a general purpose I/O use. After the chip is powered on, it defaults to an external reset function. The user can turn off the external reset function of this pin through register configuration, so that it can be used as an ordinaryI/O use. For specific use, please refer toIOCR The description part of the register.



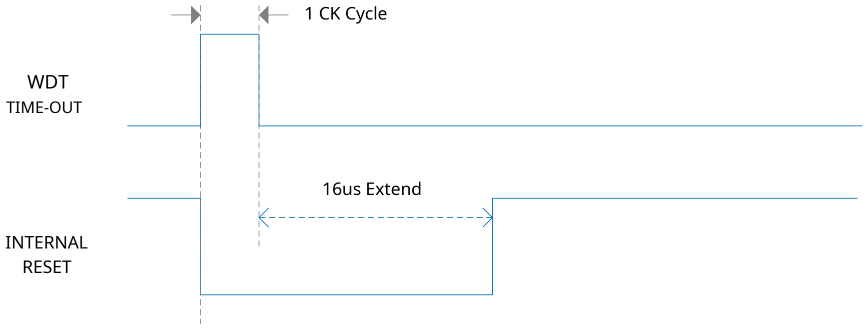
Low voltage detection (LVD)Reset

LGT8FX8P Contains a programmable low-voltage detection (LVD)Circuit.LVD The same is testing VCC Voltage changes, But unlike the power-on reset,LVD You can select the threshold of the detection voltage. The user can directly pass the operationVDTCR The register selects between different voltage thresholds.LVD The voltage detection circuit has ±10mV~±50mV Hysteresis characteristics, used to filter VCC Voltage jitter. whenLVD After enabling, if VCC The voltage drops to the set reset threshold,LVD The reset will be effective immediately. whenVCC After increasing to above the reset threshold, the internal reset expansion circuit is activated, and the reset will continue to expand at least 1 millisecond.



Watchdog reset

When the watchdog timer overflows, if the watchdog system reset function is enabled, a periodic system will be generated immediately Reset signal. The common watchdog reset signal is also stretched by the internal delay counter. For detailed operation of the watchdog controller, please refer to the detailed introduction section below.



Software reset,OCD Reset

Software reset is the user's operation VDTCR The sixth bit of the register is triggered, and the timing of the software reset is the same as the watchdog reset Completely similar. The reset signal is stretched internally16us.

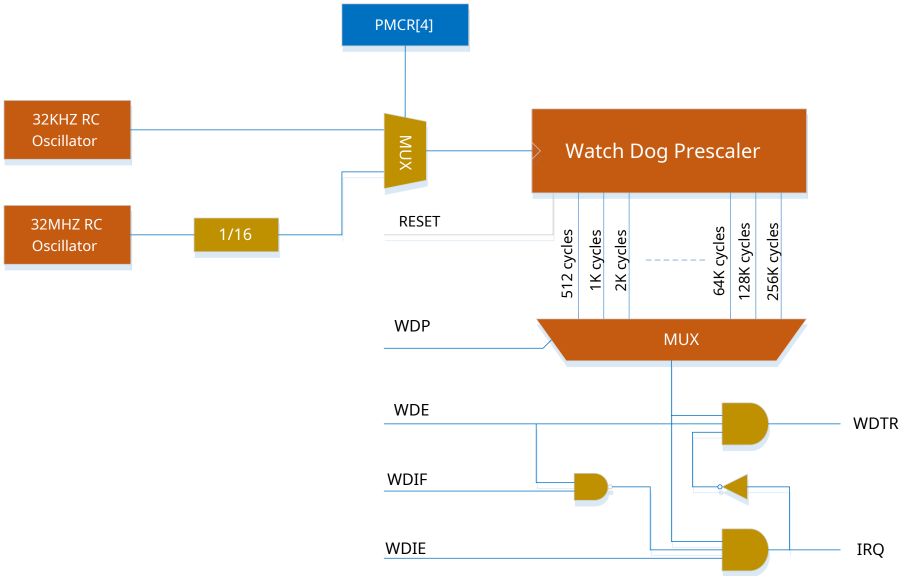
OCD The reset is generated by the debugger unit inside the chip,OCD The reset is generally controlled by a debugger and cannot be triggered by user software OCD Reset.

Watchdog timer

- Clock optional internal 32KHz RC Or internal 32MHz RC of 16 Frequency division (2MHz)
- Support interrupt mode, reset mode and reset interrupt mode
- The timer expires 8 second

LGT8FX8P It contains an enhanced watchdog timer (WDT)Module.WDT The working clock of the timer can be internal 32KHz RC Oscillator, can also be internal 32MHz RC Oscillator 16 Frequency division.WDT After the counter overflows, an interrupt or a system reset signal can be output. In normal use, the software needs to execute a WDR –The watchdog timer reset instruction restarts the counter before overflowing. If the system does not even executeWDR instruction,WDT An interrupt or system reset will be generated.

The structure diagram of the watchdog timer is shown in the figure below:



In interrupt mode,WDT An interrupt request signal will be generated after overflow. You can use this interrupt as a wake-up signal in sleep mode, or as a general system timer. For example, you can use this interrupt to limit the execution time of an operation and terminate a current task during overflow. In the system reset mode,WDT A system reset signal is generated immediately after the counter overflows. The typical use is to prevent the system from crashing or running away. The third mode is the reset interrupt mode, which combines interrupt and reset functions. First the system will respondWDT Interrupt function, exit WDT After interrupting the reset procedure, immediately switch to the reset mode. This function can support saving some key parameter information before resetting.

to preventWDT Was accidentally banned, closedWDT The operation must be carried out in accordance with a strictly defined sequence. The following code describes how to turn off the watchdog timer. The following example assumes that interrupts have been disabled, so that the entire operation flow will not be interrupted.

Sample code of watchdog enable and close operation:

Assembly code

```
WDT_OFF:
    ; Turn off global interrupt
    CLI
    ; Reset watchdog timer
    WDR
    ; Clear WDRF in MCUSR
    IN r16, MCUSR
    ANDI r16, ~(1 << WDRF)
    OUT MCUSR, r16
    ; Write logical one to WDCE and WDE
    ; Keep old Prescaler setting to prevent unintentional time-out
    LDS r16, WDTCSR
    ORI r16, (1 << WDCE) | (1 << WDE)
    STS WDTCSR, r16
    ; Turn off WDT
    LDI r16, (0 << WDE)
    STS WDTCSR, r16
    ; Turn on global interrupt
    SEI
    RET
```

C Language code

```
void WDT_OFF(void)
{
    __disable_interrupt();
    __watchdog_reset();
    /* Clear WDRF in MCUSR */
    MCUSR &= ~(1 << WDRF);
    /* Write logical one to WDCE and WDE */
    /* Keep old Prescaler setting to prevent unintentional time-out */
    WDTCSR |= (1 << WDCE) | (1 << WDE);
    /* Turn off WDT */
    WDTCSR = 0x00;
    __enable_interrupt();
}
```

[Use suggestions]

in caseWDT If it is accidentally enabled, such as a program running away, the chip will be reset, butWDT It is still enabled. If there is no processing in the user codeWDT, Which will cause the cycle to reset. To avoid this situation, it is recommended that the user software clear the watchdog reset flag bit (WDRF)withWDE Control bit.

The following code describes how to change the timeout value of the watchdog timer.

Assembly code

```
WDT_TOV_Change:
    ; Turn off global interrupt
    CLI
    ; Reset watchdog timer
    WDR
    ; Start timed sequence
    LDS r16, WDTCSR
    ORI r16, (1 << WDCE) | (1 << WDE)
    STS WDTCSR, r16
    ; - Got for cycles to set the new value from here -;
    Set new time-out value = 64k cycles
    LDI r16, (1 << WDE) | (1 << WDP2) | (1 << WDP0)
    STS WDTCSR, r16
    ; - Finished setting new value, used 2 cycles;
    Turn on global interrupt
    SEI
    RET
```

C Language code

```
void WDT_TOV_Change(void) {

    __disable_interrupt();
    __watchdog_reset();
    /* Start timed sequence */
    WDTCSR |= (1 << WDCE) | (1 << WDE);
    /* Set new time-out value = 64K cycles */
    WDTCSR |= (1 << WDE) | (1 << WDP2) | (1 << WDP0);
    __enable_interrupt();
}
```

【Instructions for use】

ChangingWDP Before configuring the bit, it is recommended to reset the watchdog timer. Because of changesWDP Bit to a relatively small time-out period is likely to cause a watchdog time-out reset.

Register definition

Low pressure detection (LVD)Control register- VDTCR

VDTCR – LVD Control register								
VDTCR: 0x62				Defaults: 0x00				
Bits	WCE	SWR	-	VDTS2	VDTS1	VDTS0	VDREN	VDTEN
R/W	R/W	W/R	-	R/W	R/W	R/W	R/W	R/W
Bit definition								

[0]	VDTEN	Low-voltage detection module enable control,1 Enable,0 Disable low-
[1]	VDREN	voltage reset function enable control,1 Enable,0 Prohibit low pressure
[4:2]	VDTS	detection threshold configuration bit 000 = 1.8V 001 = 2.2V 010 = 2.5V 011 = 2.9V 100 = 3.2V 101 = 3.6V 110 = 4.0V 111 = 4.4V
[5]	-	Keep unused
[6]	SWR	Soft reset enable bit, clearing this bit will generate a software reset
[7]	WCE	VDTCR Value change enable bit Users are changing VDTCR Before the value of the register, this bit must be written first 1, After 6 Within clock cycles, change VDTCR The value of the other bits. After four cyclesWCE Automatically clear, right VDTCR The update operation of the register is invalid.

IO Function multiplexing register- PMX2

PMX2 – IO Function multiplexing register								
PMX2: 0xF0				Defaults: 0x00				
Bits	WCE	STSC1	STSC0	-	-	XIEN	E6EN	C6EN
R/W	R/W	R/W	R/W	-	-	R/W	R/W	R/W
Bit definition								

0	C6EN	PC6 The pin defaults to reset function, set this bit to 1 External reset function will be disabled Yes, after the reset function is disabled,PC6 Can be used as an ordinary I/O use
1	E6EN	PE6 The pin defaults to analog input function, set this bit to 1, Will close the analog input function, this pin can be used as GPIO use
2	XIEN	External clock input enable control
4:3	-	Keep unused
5	STSC0	Low-speed crystal oscillator start control
6	STSC1	High-speed crystal oscillator startup control
7	WCE	IOCR Value change enable bit Users are changing IOCR Before the value of the register, this bit must be written first 1,in

		After 6 Within clock cycles, change IOCR The value of the other bits. Four cycles Rear IOCE Automatically clear, right IOCR The update operation of the register is invalid.

MCU Status register- MCUSR

MCUSR-IO Special Function Control Register								
MCUSR: 0x34(0x54)				Defaults: 0x00				
Bits	SWDD	-	PDRF	OCDRF	WDRF	BORF	EXTRF	PORF
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit definition		
[0]	PORF	Power-on reset flag, write 0 Cleared
[1]	EXTRF	External reset flag, reset automatically after power-on, or write 0 Clear low
[2]	BORF	voltage detection reset, power-on reset automatically clear, or write 0 Clear the
[3]	WDRF	watchdog reset flag, reset automatically after power-on reset, or write 0 Cleared
[4]	OCDRF	OCD Debugger reset flag, reset automatically after power-on reset, or write 0 Cleared from Power/off Mode
[5]	PDRF	wake-up flag, please refer to the power management chapter for specific description. Keep unused
[6]	-	
[7]	SWDD	SWD Interface disable bit. write1 Will close SWD interface. SWD After the interface is closed, debugging and ISP operating. If closed in the user programSWD Interface, can be pulled down during power-on RESET Way to prohibit the operation of internal programs, and then debug and ISP operating.SWD After the interface is closed,SWD Two occupied I/O The interface can be used as a universal I/O use. To avoidSWDD Misoperation, the user needs to update for the first time SWDD Write again within four cycles after the bit SWDD To take effect.

[Use suggestions]:

In order to use the reset flag information more accurately and effectively, it is recommended that the user try to read the reset flag in the early stage of the program initialization and then

Clear it.

Watchdog control status register- WDTCSR

WDTCSR – WDT Control and status registers								
address: 0x60					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	WDIF	WDIE	WDP3	WDTOE	WDE	WDP2	WDP1	WDP0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
[7]	WDIF	WDT Interrupt flag bit. whenWDT Set when working in interrupt mode and overflow occursWDIF Bit. whenWDT Interrupt enable bit WDIE for"1"And when the global interrupt is set,WDT An interrupt is generated. carried out WDT Cleared when interruptedWDIF Bit, rightWDIF Bit write"1"This bit can also be cleared.
[6]	WDIE	WDT Interrupt enable control bit. When setWDIE Bit is"1", And when the global interrupt is set,WDT The interrupt is enabled.

<table>
<tr><td></td><td></td><td>When setWDIE Bit is"0"Time,WDT Interrupts are disabled.
WDIE Bit sumWDE The bits determine the working mode of the watchdog together, as shown in the following table.
<table>
<tr><th>WDE</th><th>WDIE</th><th>mode</th><th>Action after overflow</th></tr>
<tr><td>0</td><td>0</td><td>Stop</td><td>no</td></tr>
<tr><td>0</td><td>1</td><td>Interrupt mode</td><td>Interrupt</td></tr>
<tr><td>1</td><td>0</td><td>Reset mode</td><td>Reset</td></tr>
<tr><td>1</td><td>1</td><td>Interrupt reset mode</td><td>Reset after interrupt</td></tr>
</table></td></tr>
<tr><td>[5]</td><td>WDP3</td><td>WDT Prescaler factor selection control 3 Bit.
WDP[3]withWDP[2:0]compositionWDT Prescaler factor selection bitWDP[3:0]To setWDT The overflow period.</td></tr>
<tr><td>[4]</td><td>WDTOE</td><td>WDT Turn off the enable control bit.
Should beWDE When the bit is cleared,WDTOE Bit must be set, otherwiseWDT Will not be closed.
whenWDTOE After the bit is set, the hardware will 4 Cleared after clock cyclesWDTOE Bit.</td></tr>
<tr><td>[3]</td><td>WDE</td><td>WDT Enable control bit.
When set WDE Bit is"1"Time,WDT Is enabled. When setWDE Bit is"0"Time, WDT banned.
only at WDTOE Position WDE To be cleared. To turn off the enabled WDT, Must operate in accordance with the following sequence:
1. Set at the same timeWDTOE withWDE Bit evenWDE Has been set and must be checked before the shutdown operation starts.WDE Bit write"1";
2. In the following 4 Within clock cycles, WDE Bit write"0". This will close WDT.
when WDE Bit is"1"And WDT Set after overflow resets the system WDT Reset system flagWDRF(lie in MCUSR register). whenWDRF Set when bit is setWDE Bit. So to clearWDE Bit, must be cleared firstWDRF Bit.</td></tr>
<tr><td>[2:0]</td><td>WDP</td><td>WDT Prescaler factor selection control.
Used to set WDT The overflow period. Suggest atWDT Change when not counting WDP The value of, changes during the counting processWDP The value of will produce unexpectedWDT overflow.</td></tr>
</table>

Watchdog prescaler selection list:

WDP3	WDP2	WDP1	WDP0	Watchdog timer Number of overflow cycles	32KHz clock	2MHz clock
0	0	0	0	2K cycles	64ms	1ms
0	0	0	1	4K cycles	128ms	2ms
0	0	1	0	8K cycles	256ms	4ms
0	0	1	1	16K cycles	512ms	8ms
0	1	0	0	32K cycles	1s	16ms
0	1	0	1	64K cycles	2s	32ms
0	1	1	0	128K cycles	4s	64ms
0	1	1	1	256K cycles	8s	128ms
1	0	0	0	512K cycles	16s	256ms
1	0	0	1	1024K cycles	32s	512ms

1	0	1	0	Keep unused
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

Interrupts and interrupt vectors

z 28 Interrupt sources

z Programmable vector start address

LGT8F88P/168P/328P The interrupt resources are basically the same, the main differences are:LGT8F88P The interrupt vector is 1 Instruction words (16 Bit), while LGT8F168P/328P The interrupt vector is 2 Instruction words.

LGT8F88P Interrupt vector list

LGT8F88P List of interrupt vectors:

Numbering	Vector address	Interrupt source signal	Interrupt source description
1	0x0000	RESET	External reset, power-on reset, watchdog reset, SWD Debug reset, low voltage reset
2	0x0001	INT0	External interrupt request 0
3	0x0002	INT1	External interrupt request 1
4	0x0003	PCI0	Pin level interrupt 0
5	0x0004	PCI1	Pin level interrupt 1
6	0x0005	PCI2	Pin level interrupt 2
7	0x0006	WDT	Watchdog overflow interrupt
8	0x0007	TC2 COMPA	Timer 2 Comparison match A Interrupt
9	0x0008	TC2 COMPB	Timer 2 Comparison match B Interrupt
10	0x0009	TC2 OVF	Timer 2 Overflow interrupt
11	0x000A	TC1 CAPT	Timer 1 Input capture interrupt
12	0x000B	TC1 COMPA	Timer 1 Comparison match A Interrupt
13	0x000C	TC1 COMPB	Timer 1 Comparison match B Interrupt
14	0x000D	TC1 OVF	Timer 1 Overflow interrupt
15	0x000E	TC0 COMPA	Timer 0 Comparison match A Interrupt
16	0x000F	TC0 COMPB	Timer 0 Comparison match B Interrupt
17	0x0010	TC0 OVF	Timer 0 Overflow interrupt
18	0x0011	SPI STC	SPI End of serial transmission interrupt
19	0x0012	USART RXC	USART Receive end interrupt
20	0x0013	USART UDRE	USART Data register empty interrupt
twenty one	0x0014	USART TXC	USART End of transmission interrupt
twenty two	0x0015	ADC	ADC End of conversion interrupt
twenty three	0x0016	EE_RDY	EEPROMReady interrupt
twenty four	0x0017	ANA_COMP	Analog comparator 0 Interrupt
25	0x0018	TWI	Two-wire serial interface interrupt
26	0x0019	ANA_COMP1	Analog comparator 1 Interrupt
27	0x001A	-	Keep
28	0x001B	PCI3	Pin level interrupt 3
29	0x001C	PCI4	Pin level interrupt 4
30	0x001D	TC3_INT	Timer 3 Interrupt

LGT8F168P/328P Interrupt vector list

LGT8F168P/328P List of interrupt vectors:

Numbering	Vector address	Interrupt source signal	Interrupt source description
1	0x0000	RESET	External reset, power-on reset, watchdog reset, SWD Debug reset, low voltage reset
2	0x0002	INT0	External interrupt request 0
3	0x0004	INT1	External interrupt request 1
4	0x0006	PCI0	Pin level interrupt 0
5	0x0008	PCI1	Pin level interrupt 1
6	0x000A	PCI2	Pin level interrupt 2
7	0x000C	WDT	Watchdog overflow interrupt
8	0x000E	TC2 COMPA	Timer 2 Comparison match A Interrupt
9	0x0010	TC2 COMPB	Timer 2 Comparison match B Interrupt
10	0x0012	TC2 OVF	Timer 2 Overflow interrupt
11	0x0014	TC1 CAPT	Timer 1 Input capture interrupt
12	0x0016	TC1 COMPA	Timer 1 Comparison match A Interrupt
13	0x0018	TC1 COMPB	Timer 1 Comparison match B Interrupt
14	0x001A	TC1 OVF	Timer 1 Overflow interrupt
15	0x001C	TC0 COMPA	Timer 0 Comparison match A Interrupt
16	0x001E	TC0 COMPB	Timer 0 Comparison match B Interrupt
17	0x0020	TC0 OVF	Timer 0 Overflow interrupt
18	0x0022	SPI STC	SPI End of serial transmission interrupt
19	0x0024	USART RXC	USART Receive end interrupt
20	0x0026	USART UDRE	USART Data register empty interrupt
twenty one	0x0028	USART TXC	USART End of transmission interrupt
twenty two	0x002A	ADC	ADC End of conversion interrupt
twenty three	0x002C	EE_RDY	EEPROMReady interrupt
twenty four	0x002E	ANA_COMP	Analog comparator interrupt
25	0x0030	TWI	Two-wire serial interface interrupt
26	0x0032	ANA_COMP1	Analog comparator 1 Interrupt
27	0x0034	-	Keep
28	0x0036	PCI3	Pin level interrupt 3
29	0x0038	PCI4	Pin level interrupt 4
30	0x003A	TC3_INT	Timer 3 Interrupt

LGT8FX8P Reset vector slave address 0x0000 Begin execution. Except for the reset vector, other vector addresses can be communicated PassMCUCR In the register IVSEL as well as IVBASE The register is redirected to 512 The byte-aligned start address.

Interrupt vector processing

The following code only LGT8F88P As an example, it is used to illustrate reset and interrupt vector programming, for reference only:

Assembly code example – LGT8F88P

address	Code	Description
0x000	RJMP RESET	Reset vector
0x001	RJMP EXT_INT0	External Interrupt 0
0x002	RJMP EXT_INT1	External Interrupt 1
0x003	RJMP PCINT0	Pin change interrupt 0
0x004	RJMP PCINT1	Pin change interrupt 1
0x005	RJMP PCINT2	Pin change interrupt 2
0x006	RJMP WDT	Watchdog timer interrupt
0x007	RJMP TIM2_COMPA	Timer 2 Comparison match A Group
0x008	RJMP TIM2_COMPB	interrupt timer 2 Comparison match B
0x009	RJMP TIM2_OVF	Group interrupt timer 2 Overflow interrupt
0x00A	RJMP TIM1_CAPT	Timer 1 Capture interrupt
0x00B	RJMP TIM1_COMPA	Timer 1 Comparison match A Group
0x00C	RJMP TIM1_COMPB	interrupt timer 1 Comparison match B
0x00D	RJMP TIM1_OVFR	Group interrupt timer 1 Overflow interrupt
0x00E	RJMP TIM0_COMPA	Timer 0 Comparison match A Group
0x00F	RJMP TIM0_COMPB	interrupt timer 0 Comparison match B
0x010	RJMP TIM0_OVF	Group interrupt timer 0 Overflow interrupt
0x011	RJMP SPI_STC	SPI Transfer complete interrupt
0x012	RJMP USART_RXC	USART Receive complete interrupt
0x013	RJMP USART_UDRE	USART Data register empty interrupt
0x014	RJMP USART_TXC	USART Send complete interrupt
0x015	RJMP ADC	ADC Conversion complete interrupt
0x016	RJMP EE_RDY	EEPROMThe controller is ready to interrupt
0x017	RJMP ANA_COMP	Comparator interrupt
0x018	RJMP TWI	TWI Controller interrupt
0x019	NOP	Reserved address
0x01A	NOP	Reserved address
0x01B	RJMP PCI3	Pin change interrupt 3
;		
0x01C (**RESET :)**	LDI r16, high(RAMEND)	The main program starts
0x01D	OUT SPH, r16	Set the stack pointer to RAMTop address
0x01E	LDI r16, low(RAMEND)	
0x01F	OUT SPL, r16	
0x020	SEI	Enable global interrupt
0x021	...	

Register definition

MCU Control register- MCUCR

MCUCR – MCU Control register								
MCUCR: 0x35(0x55)				Defaults: 0x00				
MCUCR	FWKEN	FPDEN	EXRFD	PUD	IRLD	IFAIL	IVSEL	WCE
R/W	R/W	R/W	R/W	R/W	W/O	R/O	R/W	R/W

Bit definition		
[0]	WCE	MCUCR Update the enable bit, in the update MCUCR Before, you need to set this bit first, and then 6 Complete the pairing in cycles MCUCR The update interrupt vector selection bit of the register, this position
[1]	IVSEL	1 After that, the interrupt vector address will be based on IVBASE The value of the register is mapped to the new address
[2]	IFAIL	The system configuration bit loading failure flag bit, 0 = Configuration information passed 1 = Failed to load configuration information
[3]	IRLD	write 1 System configuration information will be reloaded
[4]	PUD	Global pull-up prohibition bit 0 = Enable global pull-up control 1 = Close all IO Pull-up resistor
[5]	EXRFD	external reset filter disable bit 0 = Enable external reset (190us)Digital filter 1 = Digital filter circuit with external reset disabled
[6]	FPDEN	Flash Power/down Enable control 0:system SLEEP Rear FLASH Keep powered on 1:system SLEEP Rear FLASH Power-off fast wake-up mode enable
[7]	FWKEN	control, only for Power/Off Mode is valid 0:260us Filter delay 1:32us Filter delay

Interrupt vector base address register- IVBASE

IVBASE – Interrupt vector base address register	
IVBASE: 0x75	Defaults: 0x00
IVBASE	IVBASE[7:0]
R/W	R/W

Bit definition		
[7:0]	IVBASE	in case IVSEL for 1, The interrupt vector (except the reset vector) will be IVBASE As the base address in 512 Remap on the byte page. The mapped base address of the interrupt vector is: (IVBASE << 8) + table 1 Corresponding vector address in

External Interrupt

Z 2 External interrupt source

Z Configurable level or edge trigger interrupt

Z Can be used as a wake-up source in sleep mode

Overview

External interrupts are caused by INT0 with INT1 Pin trigger. As long as the external interrupt is enabled, even this2 The configuration of a pin as an output can also trigger an interrupt. This can be used to generate software interrupts. External interrupts can be triggered by rising edge, falling edge or low level, controlled by the external interrupt control registerEICRA To configure. When the external interrupt is enabled and configured as level trigger (onlyINT0 with INT1 Pin), as long as the pin level is low, the interrupt will always be generated.INT0 with INT1 Needed for interrupt trigger on rising or falling edge of the pin IOThe clock works normally, and INT0with INT1The low-level trigger interrupts of the pins are all detected asynchronously. In addition to idle mode, other sleep modesIO The clock is stopped. So this2 Each external interrupt can be used as a wake-up source in other sleep modes except idle mode.

If level-triggered interrupt is used as a wake-up source in power saving mode, the changed level must be maintained for a certain period of time to wake upMCUTo reduce MCU Sensitivity to noise. The required level must be maintained long enough forMCU End the wake-up process, Then trigger the level interrupt.

Register definition

Register list

register	address	Defaults	description
EICRA	0x69	0x00	External interrupt control register A
EIMSK	0x3D	0x00	External interrupt mask register
EIFR	0x3C	0x00	External interrupt flag register

External interrupt control register A- EICRA

EICRA - External interrupt control register A								
address: 0x69				Defaults: 0x00				
Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	ISC11	ISC10	ISC01	ISC00
R/W	-	-	-	-	R/W	R/W	R/W	R/W

Bit	Name	description
7:4	-	Reserved.
3	ISC11	INT1 The pin interrupt trigger mode control bit is high.
2	ISC10	INT1 The low bit of pin interrupt trigger mode control bit. When the global interrupt is set and GICR When the corresponding interrupt mask control bit of the register is set, the external interrupt 1 by INT1 The pin is excited. The interrupt trigger mode is described in the table. Before edge detectionMCU Sample first INT1 The level on the pin. If edge trigger mode or level change trigger mode is selected, then the duration is greater than1 A pulse of one system clock cycle will trigger an interrupt, and a pulse that is too short cannot guarantee an interrupt. If you choose low power

		Level trigger mode, then the low level must be maintained until the current instruction execution is completed before triggering the interrupt.
1	ISC01	INT0 The pin interrupt trigger mode control bit is high.
0	ISC00	INT0 The low bit of pin interrupt trigger mode control bit. When the global interrupt is set and GICR When the corresponding interrupt mask control bit of the register is set, the external interrupt 0 by INT0 The pin is excited. The interrupt trigger mode is described in the table. Before edge detectionMCU Sample first INT0 The level on the pin. If edge trigger mode or level change trigger mode is selected, then the duration is greater than1 A pulse of one system clock cycle will trigger an interrupt, and a pulse that is too short cannot guarantee an interrupt. If the low level trigger mode is selected, then the low level must be maintained until the current instruction execution is completed before triggering the interrupt.

External Interrupt 1 The trigger mode is shown in the table below.

External Interrupt 1 Trigger mode control

ISC1[1:0]	description
0	External pin INT1 Low level trigger
1	External pin INT1 Rising or falling edge trigger
2	External pin INT1 Falling edge trigger
3	External pin INT1 Rising edge trigger

External Interrupt 0 The trigger mode is shown in the table below.

External Interrupt 0 Trigger mode control

ISC0[1:0]	description
0	External pin INT0 Low level trigger
1	External pin INT0 Rising or falling edge trigger
2	External pin INT0 Falling edge trigger
3	External pin INT0 Rising edge trigger

External interrupt mask register- EIMSK

EIMSK **- External interrupt mask register**								
address: 0x3D					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name	-	-	-	-	-	-	INT1	INT0
R/W	-	-	-	-	-	-	R/W	R/W

Bit	Name	description
7:2	-	Keep
1	INT1	External pin 1 Interrupt enable control bit. When set INT1 Bit is"1"When the global interrupt is set, the external pin 1 The interrupt is enabled and the wake-up function is enabled. even ifINT1 The pin is configured as an output, as long as the pin level changes accordingly, an interrupt will be generated. When set INT1 Bit is"0"When the external pin 1 The interrupt is disabled, and the wake-up function is also disabled.

0	INT0	External pin 0 Interrupt enable control bit. When set INT0 Bit is"1"When the global interrupt is set, the external pin 0 The interrupt is enabled and the wake-up function is enabled. even ifINT0 The pin is configured as an output, as long as the pin level changes accordingly, an interrupt will be generated. When set INT0 Bit is"0"When the external pin 0 The interrupt is disabled, and the wake-up function is also disabled.

External interrupt flag register- EIFR

EIFR - External interrupt flag register								
address: 0x3C					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name	-	-	-	-	-	-	INTF1	INTF0
R/W	-	-	-	-	-	-	R/W	R/W
Bit	Name	description						
7:2	-	Reserved.						
1	INTF1	External pin 1 Interrupt flag bit. When an edge triggers an external pin 1 When interrupted,INTF1 Is set. When the low level triggers the external pin1 Not set when interrupted INTF1 Bit. If the external pin1 Interrupt enable INT1EN Bit is"1" And the global interrupt flag is set, an external pin will be generated 1 Interrupted. When this interrupt service routine is executedINTF1 Will be automatically cleared, or INTF1 Bit write "1"This bit can also be cleared.						
0	INTF0	External pin 0 Interrupt flag bit. When an edge triggers an external pin 0 When interrupted,INTF0 Is set. When the low level triggers the external pin0 Not set when interrupted INTF0 Bit. If the external pin0 Interrupt enable INT0EN Bit is"1" And the global interrupt flag is set, an external pin will be generated 0 Interrupted. When this interrupt service routine is executedINTF0 Will be automatically cleared, or INTF0 Bit write "1"This bit can also be cleared.						

Computing accelerator (uDSC)

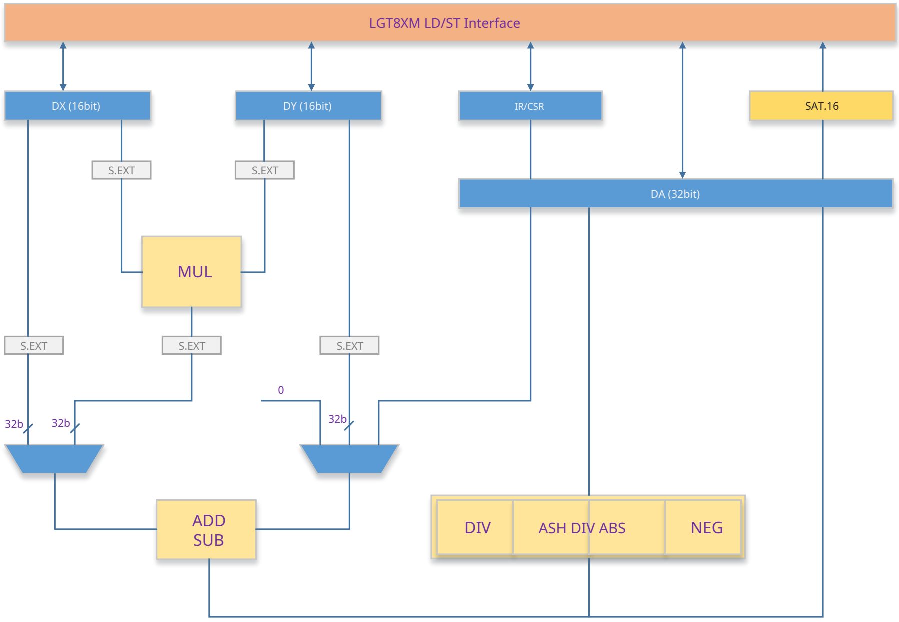
- Z 16 Bit storage mode (LD/ST)
- Z 32 Bit accumulator (DX)
- Z Single cycle 16 Bit multiplier (MUL) 32 Bit
- Z arithmetic logic operation unit (ALU)
- Z 16 Bit saturation operation (SD)
- Z 8 cycle 32/16 Divider
- Z Single cycle multiply-add/multiply-subtract operations (MAC/MSC)

Overview

Digital Computing Accelerator (uDSC) As LGT8XM A computing co-processing module of the kernel, with LGT8XM Kernel 16 Bit LD/ST Pattern to achieve a 16 Bit digital signal processing unit. Can meet most of the control digital signals deal with.

uDSC Function internal and function:

1. 16 Bit operand register DX/DY
2. 32 Bit accumulation register DA
3. Single cycle 17 Bit multiplier (can be implemented 16 Bit with/unsigned multiplication operation)
4. 32 Bit ALU (can be realised 16/32 Bit addition, subtraction and shift operations)
5. 16 Bit saturation operation (used to store the result of the operation in RAMspace)
6. 32/16 Divider,8 Completion of calculations in cycles



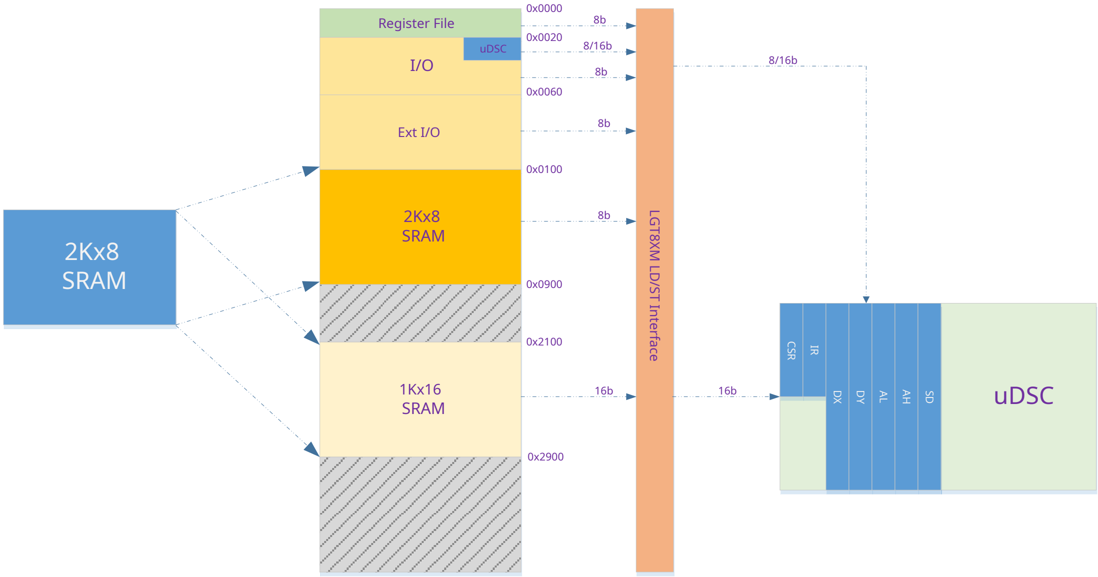
uDSC Structure chart

16 Bit LD/ST Operating mode

To improve uDSC The efficiency of processing large amounts of data,LGT8XM The kernel implements a dedicated 16 Bit LD/ST Storage channel, you can use LDD/STD Instructions efficiently in uDSC versus SRAMAnd general register files 16 Bit data exchange.

In order not to disrupt normalLD/STCommand system,LGT8XMKernelSRAMSpace remapped to0x2100~0x28FF. useLD/ST Instruction from 0x2100~0x28FF Space access SRAMWhen the kernel is automatically turned on 16 Bit LD/ST Function, hit openSRAMversus uDSC Direct access channel between. The picture below is

LGT8XMThe kernel's data space address distribution:



As shown in FIG,LGT8XMThe kernel can be used by LD/ST Instructions in uDSC of DX/DY/DA Register and SRAMDirectly between 16 Bit data storage access. SimultaneouslyuDSC The internal registers are also mapped to I/O Space, visit uDSC The register is divided into 8/16 Two modes.

uDSC In addition to the internal calculation DX/DY/DA In addition to the register, it also contains another 2 A 8 Bit register: uDSC Control status register CSR And arithmetic instruction register IR.CSR/IR Can only pass I/O Space is accessed in bytes; access DX/DY/AL/AH When 16 Bit pattern. can useIN/OUT as well as LD/ST/LDD/STD/LDS/STD Wait for instruction to visit.

uDSC The related control status and data registers are mapped to IO Space, direct use IN/OU Instruction addressing can be completed in one instruction cycle 8/16 Bit data access.

CSRFor controluDSC Work mode and record the currentuDSCThe status flag bit of the execution operation.IRcontroluDSC The specific operation implemented.uDSC Most of the operations supported will be completed in one cycle, and the division operation requires 7 Wait cycles, you can also pass CSR The flag bit in the register judges whether the current division operation is completed.

standard LD/ST Instruction use LGT8XM The internal general working register is used as LD/ST Data using X/Y/Z As the target address. When the target address falls on16 Bit SRAM When mapping space, at this time LD/ST The meaning of the instruction operand has changed, where X/Y/Z Still as the target address, the meaning of general working register addressing is based on uDSC The mapping mode will be handled in two ways.uDSC The mapping mode only applies to 0x2100~0x28FF Address access access. Mapping mode passedCSR Register 6 Bit (MM)Set up.

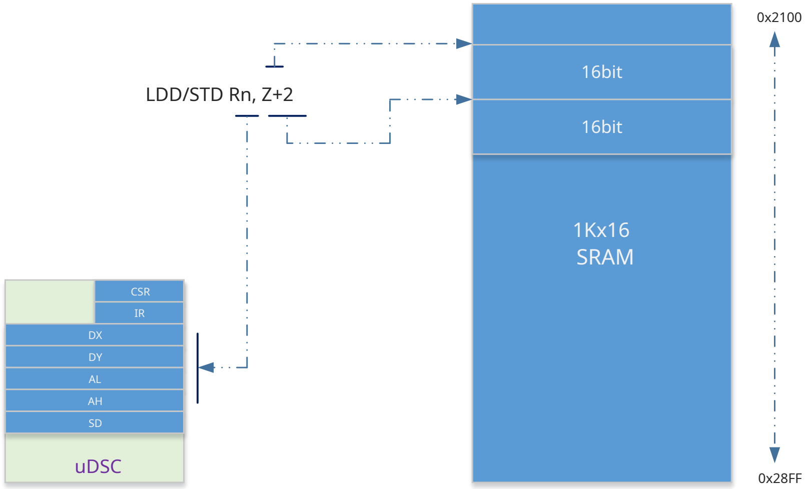
16 Bit LD/ST In mode, the command”LDD Rn, Z+q” Means to put [Z]Address 16 Bit data loaded into uDSC In the data register, and then Z Add an offset to the value of"Q". HereRn The meaning and mapping mode CSR[MM]Off

Department is as follows:

LDD Rn, Z/Y+q			
CSR[MM]	[Z+q]	Opcode	Operations
0	0x2100~0x28FF	LDD R0, Z+q	DX = [Z]; Z = Z + q; R0 kept unchanged
		LDD R1, Z+q	DY = [Z]; Z = Z + q; R1 kept unchanged
		LDD R2, Z+q	AL = [Z]; Z = Z + q; R2 kept unchanged
		LDD R3, Z+q	AH = [Z] ; Z = Z + q; R3 kept unchanged
1	0x2100~0x28FF	LDD Rn, Z+q	{Rn} address for DX/DY/AL/AH in I/O region [DX/DY/AL/AY] = [Z]; Z = Z + q Rn keep unchanged
STD Rn, Z/Y+q			
0	0x2100~0x28FF	STD Z+q, R0	[Z] = DX; Z = Z + q; R0 kept unchanged
		STD Z+q, R1	[Z] = DY; Z = Z + q; R1 kept unchanged
		STD Z+q, R2	[Z] = AL; Z = Z + q; R2 kept unchanged
		STD Z+q, R3	[Z] = AH; Z = Z + q; R3 kept unchanged
		STD Z+q, R4	[Z] = SD; Z = Z + q; R4 kept unchanged
1	0x2100~0x28FF	STD Z+q, Rn	{Rn} address for DX/DY/AL/AH/SD in I/O region [Z] = [DX/DY/AL/AH/SD] addressed by {Rn} Rn keep unchanged

LGT8XM Instruction set LD/ST, LDS/STS All accessible 0x2100~0x28FF Area, but LDD/STD Y/Z+q The addressing mode of is more effective.LDD/STD Way of addressing is based on a base address, we can put Y/Z Set as RAMThe base address of the data, by using LDD/STD Instructional Y/Z+q Addressing mode can execute instructions and access data in one cycle, and automatically move the address pointer to the next target address.

LGT8XM Kernel standard LDD/STD Instructional Y/Z+q Offset addressing mode, used during instruction execution [Y/Z+q]As 8 The address of the bit data, after the execution is complete Y/Z The value of does not increase. When usingLDD/STD Addressing 0x2100~0x28FF When the address of the interval,LDD/STD The behavior of the instruction has changed: When the instruction is executed, use [Y/Z]As 16 Bit data addressing address, after execution,Y/Z Increase in value"Q"The specified offset. This feature can improve the efficiency of our continuous addressing, by changing"Q=2"Can achieve continuous 16 Addressing of bit data.



Variable address and 16 Mapping between bit pattern addresses

LGT8XMfor 8 Bit processor, data access is in bytes.LGT8F328P Built-in 2K Bytes of data space. This part of the space is mapped to 0x0100~0x08FF the address of.C/C++The compilation automatically assigns the variable to 0x0100~0x08FF between. If we areC/C++One defined in 16 Array of bits needs to use uDSC To perform operations, you need to first map the address of the variable to 16 Bit LD/ST Address area visited (0x2100~0x28FF). The method is very simple, just increase the address of the variable 0x2000 The offset can be.

uDSC Operation instruction definition

Software passed uDSC of IR The register specifies the operation that needs to be implemented.uDSC All operations are in DX/DY/DA Between. Users can use16 Bit LD/ST Channel in DX/DY/DA as well as SRAMExchange data directly and quickly.

classification	IR[7:0]								Function description
ADD/SUB	0	0	S_1	0	0	1	0	1	DA = DX + DY
	0	0	S_1	0	0	0	0	1	DA = DX – DY
	0	0	0	1	1	1	0	1	DA = DY
	0	0	S_1	1	1	0	0	1	DA = -DY
	0	0	S_1	1	0	1	1	1	DA = DA + DY
	0	0	S_1	1	0	0	1	1	DA = DA – DY
MAC/MSC	0	1	$S1_2$	$S0_2$	0	1	0	0	DA = DX * DY
	0	1	$S1_2$	$S0_2$	0	0	0	0	DA = -DX * DY
	0	1	$S1_2$	$S0_2$	1	1	0	0	DA = (DX * DY) >> 1
	0	1	$S1_2$	$S0_2$	1	0	0	0	DA = (-DX * DY) >> 1
	0	1	$S1_2$	$S0_2$	0	1	1	S	DA = DA + DX * DY
	0	1	$S1_2$	$S0_2$	1	1	1	S	DA = (DA + DX * DY) >> 1
	0	1	$S1_2$	$S0_2$	0	0	1	S	DA = DA – DX * DY
	0	1	$S1_2$	$S0_2$	1	0	1	S	DA = (DA – DX * DY) >> 1
MISC	1	0	0	0	0	0	0	0	DA = 0

	1	0	0	0	0	1	0	S	DA = NEG(DA)
	1	0	0	0	1	0	0	S	DA = DX^2`
	1	0	0	0	1	0	1	S	DA = DY^2
	1	0	1	0	0	0	0	S	DA = ABS(DA)
	1	0	1	1	0	0	0	0	DA = DA/DY
	1	0	1	1	0	0	0	1	DA = DA/DY, DY = DA%DY
SHIFT	1	1	0	0	N3	N2	N1	N0	DA = DA << N
	1	1	S	1	N3	N2	N1	N0	DA = DA >> N

Description:

1. S Indicates whether the operation is signed or unsigned
2. S1 Means DX Is it a signed number, S2 Means DY Is it a signed number
3. N3...0 Is four bit shifts, which can achieve more 15 Bit shift operation
4. - Indicates that the value of this bit is not meaningless and can be set to 0 or 1, It is recommended to set to 0

Register definition

name	IO address	Function description
DCSR	0x20(0x00)	uDSC Control status register
DSIR	0x21(0x01)	Operation instruction register
DSSD	0x22(0x02)	accumulator DSA of 16 Bit saturation operation result
DSDX	0x10(0x30)	operand DSDX, 16 Bit read and write access operands
DSDY	0x11(0x31)	DSDY, 16 Bit read and write access
DSAL	0x38(0x58)	32 Bit accumulator DSA[15:0], 16 Bit read and write access
DSAH	0x39(0x59)	32 Bit accumulator DSA[31:16], 16 Bit read and write access

DSCR- Control status register

DSCR **– UDSC Control status register**								
address: 0x20 (0x00)						Defaults: 0010_xxxx		
Bit	7	6	5	4	3	twenty one		0
Name	DSUEN	MM	D1	D0	-	NZ		C
R/W	R/W	R/W	R/W	R/W	-	R/WR/W		R/W

Bit	Name	description
7	DSUEN	uDSC Module enable control; 1 = Enable,0 = Disable
6	MM	uDSC Register mapping mode; please refer to the detailed definition 16 Introduction of bit working mode. 0 = Quick access mode, 1 = IO The division operation completion
5	D1	flag of the mapping mode, 1 = Operation complete division
4	D0	operation division 0 Flag bit
3	-	Unimplemented
2	N	The result of the operation is a negative flag
1	Z	The result of the operation is a zero flag
0	C	32 Adder carry/borrow flag

DSIR – Operation instruction register

DSIR **– UDSC Operation instruction register**							
address: 0x21 (0x01)						Defaults: 0000_0000	
Bit	7	6	5	4	3	twenty one	0
Name	DSIR[7:0]						
R/W	R/W						
Bit	Name	description					
7:0	IR	uDSC Operation instructions. Please refer to"Operation instruction definition"Chapter description					

DSDX- Operand register DSDX

DSDX **– UDSC Fuck Count register DX**												
address: 0x30 (0x10)								Defaults: 0000_0000				
Bit	15 14 13 12 11 10			9	8	7	6	5 4 3 2			1	0
Name	DSDX[15:0]											
R/W	R/W											
Bit	Name	description										
15:0	DSDX	16 Bit operand register DSDX										

DSDY- Operand register DSDY

DSDY **– UDSC Fuck Count register DY**												
address: 0x31 (0x11)								Defaults: 0000_0000				
Bit	15 14 13 12 11 10			9	8	7	6	5 4 3 2			1	0
Name	DSDY[15:0]											
R/W	R/W											
Bit	Name	description										
15:0	DSDY	16 Bit operand register DSDY										

DSAL-32 Bit accumulator DA The low 16 Bit

DSAL **– UDSC Operand register DSA The low 16 Bit**												
address: 0x58 (0x38)								Defaults: 0000_0000				
Bit	15 14 13 12 11 10			9	8	7	6	5 4 3 2			1	0
Name	DSA[15:0]											
R/W	R/W											
Bit	Name	description										
15:0	DSAL	32 Bit accumulator DSA The low 16 Bit										

DSAH-32 Bit accumulator DA height of 16 Bit

***DSAH* – UDSC Operand register DSA height of 16 Bit**												
address: 0x59 (0x39)								Defaults: 0000_0000				
Bit	15 14 13 12 11 10			9	8	7	6	5 4 3 2			1	0
Name	DSA[31:16]											
R/W	R/W											
Bit	Name	description										
15:0	DSAH	32 Bit accumulator DSA height of 16 Bit										

DSSD-DA Saturation Operation Register

***DSSD*- 16 Bit DA Saturation calculation result**												
address: 0x22 (0x02)								Defaults: 0000_0000				
Bit	15 14 13 12 11 10			9	8	7	6	5 4 3 2			1	0
Name	DSSD[15:0]											
R/W	R/W											
Bit	Name	description										
15:0	DSSD	32 Bit accumulator DSA of 16 Bit saturation operation result										

uDSC Applications

Instance 1. Basic configuration and operation

The following is a simple subroutine (AVRGCC)To achieve a 16 Multiplication of bits, return 32 Bit result.:

unsigned long **dsu_xmuluu** (unsigned short dy, unsigned short dx);

The following is the C The assembly code of the function:

```
# include       "Udsc_def.inc"              ; opcode definitions
                . global     dsu_xmuluu     ; declare for called from C/C++ code


dsu_xmuluu:
                out          DSDX, r24      ; load DX
                out          DSDY, r22      ; load DY
                ldi          r20, XMULUU    ; load opcode
                out          DSIR, r20      ; do multiply
                in           r22, DSAL      ; {r23, r22} = AL;
                in           r24, DSAH      {r25, r24} = AH
                ret
```

Universal Programmable Port (GPIO)

Overview

All based on LGT8XM Realized by the kernel series MCU All have I/O Port read-modify-write function. This means that the status of a certain port can be usedSBI with CBI The instructions are changed individually without affecting any other I/O. The same is true for changing the direction of a port or controlling its pull-up resistor.

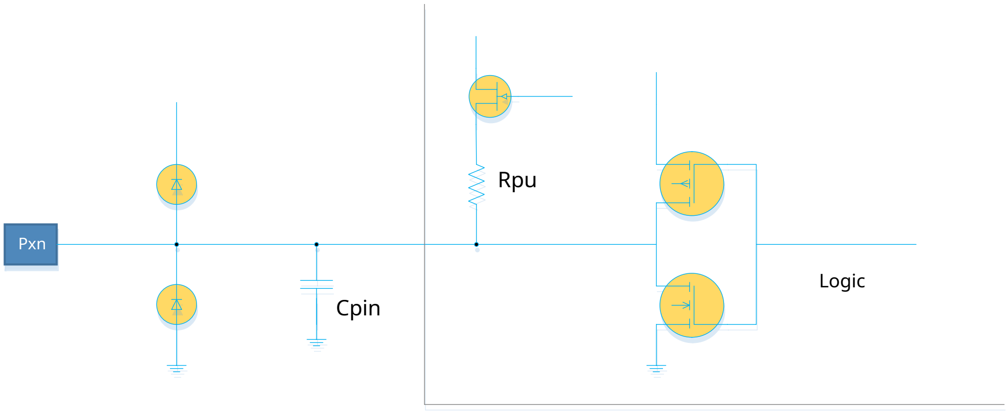
LGT8FX8P the most part of I/O With symmetrical driving characteristics, it can drive and absorb larger currents.I/O With two-level drive capability, the user can control each group I/O The driving ability.I/O The driving ability can directly drive some led.

LGT8FX8P the most part of I/O Can drive up to 30mA The current can be directly used to drive the segment code led. allI/O of

VCC with GND Directly have independent ESD Protection diode, designed to withstand at least up to

5000V of ESD pulse.

I/O Equivalent circuit diagram:



All registers below in this chapter are described in a unified manner, in lower case"X"Represents the alphabetical serial number name of the port, lowercase"N" Indicates the bit number in the port. But when using the port register in the program, you must use the exact register name. such as PORTB3, It means PORTB The third place, here is used uniformly PORTxn Said.I/O For detailed definitions of related registers, please refer to the register description section.

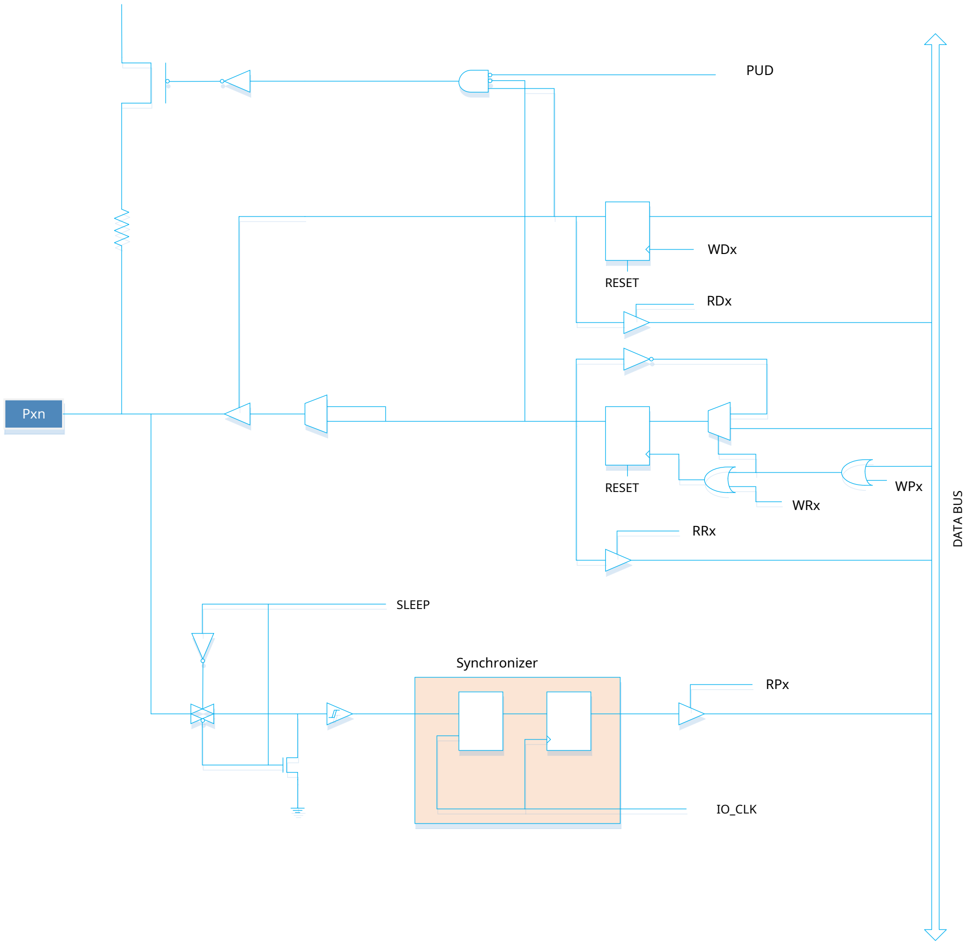
Three for each port I/O Register space, they are: port data output register (PORTx), Port direction register (DDRx), Port data input register (PINx). The port data input register is a read-only register. The data output register and port direction register can be read or rewritten.MCUCR In the register PUD Bits, used to control all I/O Pull-up resistor, when PUD Bit is 1 When it will ban so I/O The pull-up resistor.

most I/O In addition to having general input/output functions, it will also be multiplexed into other peripheral functions. For specific multiplexing functions, please refer to the chapter on port function multiplexing.

It should be noted that enabling the multiplexing function of some ports will not affect these ports as digital I/O use. And some reuse functions may also need to passI/O The register controls the input/output direction of the port. The specific settings will be introduced in the documentation of each multiplexing module.

Universal input/output port

As universal I/O When the port is bidirectionally driven I/O Port, internal programmable pull-up. The picture below is generalI/O The equivalent circuit diagram of the port:



PUD: PULLUP DISABLE

SLEEP: SLEEP CONTROL

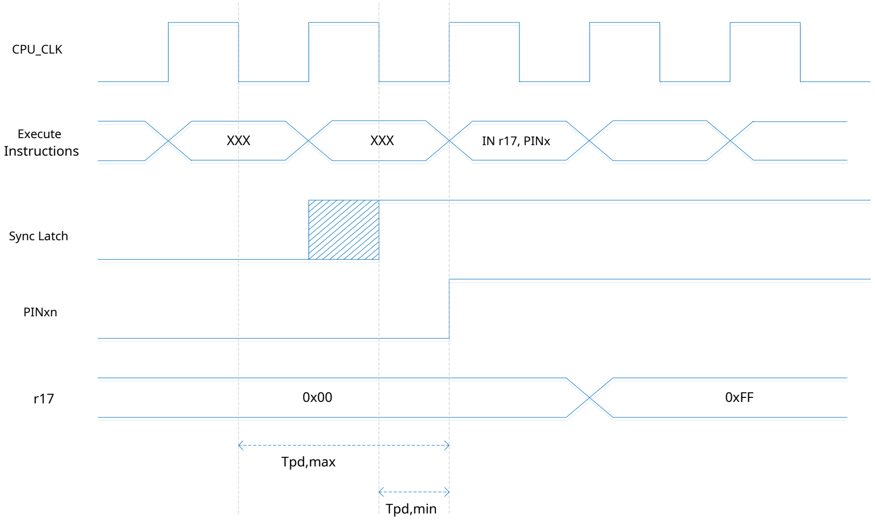
IO_CLK: I/O CLOCK

WDx:	WRITE DDRx
RDx:	READ DDRx
WRx:	WRITE PORTx
RRx:	READ PORTx REGISTER
RPx:	READ PORTx PIN
WPx:	WRITE PINx REGISTER

Port usage configuration

Each port is controlled by three register bits:DDxn,PORTxn with PINxn. among themDDxn Used to pass DDRx Register access,PORTxn able to pass PORTx Register access,PINxn able to pass PINx Register access.

DDRxn The register bits are used to set the input/output direction of the port. in caseDDxn Set as 1,Pxn The port is configured as an output port. in caseDDxn Set as 0,Pxn It is configured as an input port.



in case PORTxn Bit is written 1At the same time, this port is configured as an input port, and the pull-up resistor of this port is valid. If you want to disable the pull-up resistor of the port,PORTxn Must be written as 0 Or configure this port as an output port.

The reset initialization state of the port is input state, and the pull-up resistor is invalid.

PORTxn Set as 1At the same time, this port is configured as an output port, and the external port will be driven high. in casePORTxn Set as 0, The port will be driven low.

Input/output switch

when I/O The state is in three states ([DDxn, PORTxn]) = 0b00)And output high level ([DDxn, PORTxn] = 0b11)When switching between, there will be an intermediate state where the port is pulled up or the output is low. Generally, pull-up resistors are acceptable because in a high-impedance environment, the difference between driving high and pull-up is not important. If this is not the case, you can passMCUCR In the register PUD The pull-up function of the port is turned off.

Similarly, the same problem occurs when switching between input and output low levels enabled by the pull-up. Users must use tri-state ([DDxn, PORTxn] = 0b00)Or output high ([DDxn, PORTxn] = 0b11)As an intermediate state.

Port driver configuration table:

DDxn	PORTxn	PUD	Port status	pull up	Function Description
0	0	X	enter	Prohibit	Three states (High-Z)
0	1	0	enter	Enable	Input + internal pull-up mode
0	1	1	enter	Prohibit	Three states (High-Z)
1	0	X	Output	Prohibit	Low output (fan-in)
1	1	X	Output	Prohibit	High output (fan out)

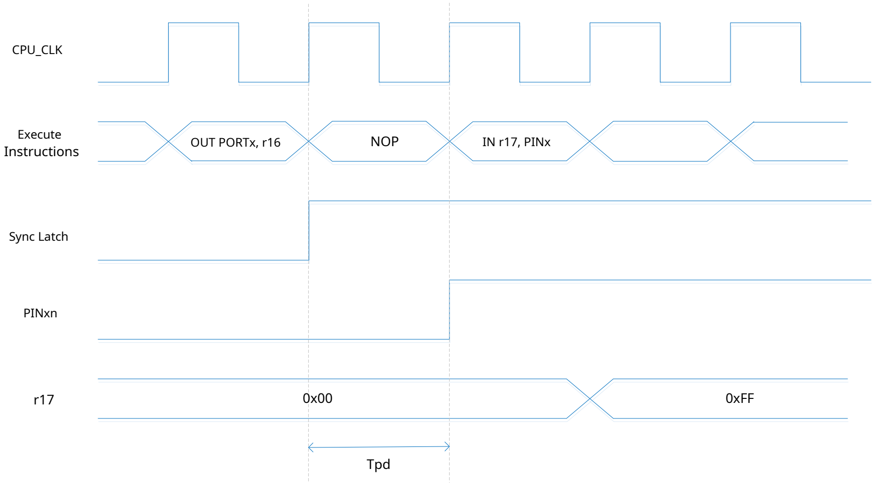
Read port value

Regardless of the port direction bit DDxn How to set it up PINxn The register bit reads the current status of the port. To avoid directly reading the metastable state generated by the port,PINxn The register bit is the result of the port passing through a synchronizer. The synchronizer is composed of a latch and a register, soPINxn There is a small delay between the value of and the current port. This delay is due to the existence of the synchronizer, and the delay time is mostly1 Half system cycle.

We assume that the system cycle starts from the first falling edge of the system clock, the latch latches data when the clock is low, and the data passes through the latch when the clock is high, as shown in the shaded part in the above figure. When the clock is low, the port data is locked

Save, and will be registered to the rising edge of the next clock PINxn register. In the picture aboveTpd,max as well as Tpd,min Is the large and small delay of the port data, divided into 1.5 Period sum 0.5 cycle.

If you want to read the port value set by the software, you need to I/O The write and read bytes support the insertion of a no-operation instruction (NOP). The sequence is shown in the figure below:



The following code shows how to set the port B Pin 0/1 Is high,2/3 Is low, define the pin 4~7 For input and Pin enabled 6,7 The pull-up resistor. Then the value of the pin is read back into the general working register. According to the previous description, a pin is directly inserted into the output and input of the pin.NOP instruction.

Assembly code

```
; Define Pull-ups and set outputs high;
Define directions for port pins
LDI r16, (1<<PB7)|(1<<PB6)|(1<<PB1)|1<<PB0)
LDI r17, (1<<DDB3)|(1<<DDB2)|(1<<DDB1)|(1<<DDB0)
OUT PORTB, r16
OUT DDRB, r17
; Insert nop for synchronization
NOP
; Read port pins
IN r16, PINB
```

C Language code

```
unsigned char I;
/* Define pull-ups and set outputs high */ /*
Define directions for port pins */
PORTB = (1<<PB7)|(1<<PB6)|(1<<PB1)|(1<<PB0); DDRB =
(1<<DDB3)|(1<<DDB2)|(1<< DDB1)|(1<<DDB0);
/* Insert nop for synchronization */
__no_operation();
/* Read port pins */
I = PINB;
```

Input enable and sleep control

From I/O As we can see in the equivalent circuit diagram, the digital input can be SLEEP The signal is clamped to the ground level under the control of the signal.SLEEP Signal by MCU Sleep controller and various sleep mode control. This can ensure that after entering sleep, the system will not cause leakage due to port input floating.

Port of SLEEP The control function will be replaced by the external interrupt function. If the external interrupt request is invalid,SLEEP The control can still work.SLEEP The control function will also be replaced by some other secondary functions. For details, please refer to the introduction of the port secondary function below.

Quickly flip port status

Port status is set to output IO,able to pass PORTn The register changes the port state. If you need to flip the current The output status of the port, usually you need to read the current port status first PINx, And then inverted and written back to PORTn The register is flipped.LGT8FX8P Provide another more efficient way to flip the port status, by directly PINx Register write 1 Then the specified port status can be reversed. For example, we writePINB[3]for 1, You can achieve PB3 The port status is reversed. For applications that need to generate an output clock, this method is very practical.

Digital/analog multiplex port

LGT8FX8P Some ports are mixed and multiplexed ports with digital and analog functions. Except internalDAC Output PD4 Outside, other mixed end Both ports are used as analog input. When the port is used as an analog function, the software needs to set the port to input mode and turn off the internal pull-up as needed to avoid affecting the analog income.DIDR0~2 The register is used to close the digital input channel of the mixed function port to avoid the redundant power loss caused by the analog input to the digital circuit.DIDRx The digital output function of the port will not be turned off.

High current push-pull drive port

LGT8FX8P Support to many 6 High-current push-pull drive port, supporting large 80mA The push-pull drive. Considering the core sheet VCC Large over-current capability limit, it is not recommended to open at the same time 6 High-current drive. Especially for those with only one set of power portsQFP32 Package, it is recommended not to open and drive at the same time 4 High current load above the circuit.

The driver of the ordinary port is 12mA, The software needs to pass HDR The register enables the high current drive function of the port. The ports with high current drive capability are as follows:

HDR port	QFP48	QFP32	HDR	Function Description
PD5	PD5	PD5	HDR[0]	N/A
PD6	PD6	PD6	HDR[1]	N/A
PF1	PF1	PD1 PF1	HDR[2]	QFP32 Encapsulated PD1 Internal equivalent QFP48 of PD1 versus PF1 in parallel
PF2	PF2	PD2 PF2	HDR[3]	QFP32 Encapsulated PD2 Internal equivalent QFP48 of PD2 versus PF2 in parallel
PF4	PF4	PE4 PF4	HDR[4]	QFP32 Encapsulated PE4 Internal equivalent QFP48 of PF4 versus PE4 in parallel
PF5	PF5	PE5 PF5	HDR[5]	QFP32 Encapsulated PE5 Internal equivalent QFP48 of PF5 versus PE5 in parallel

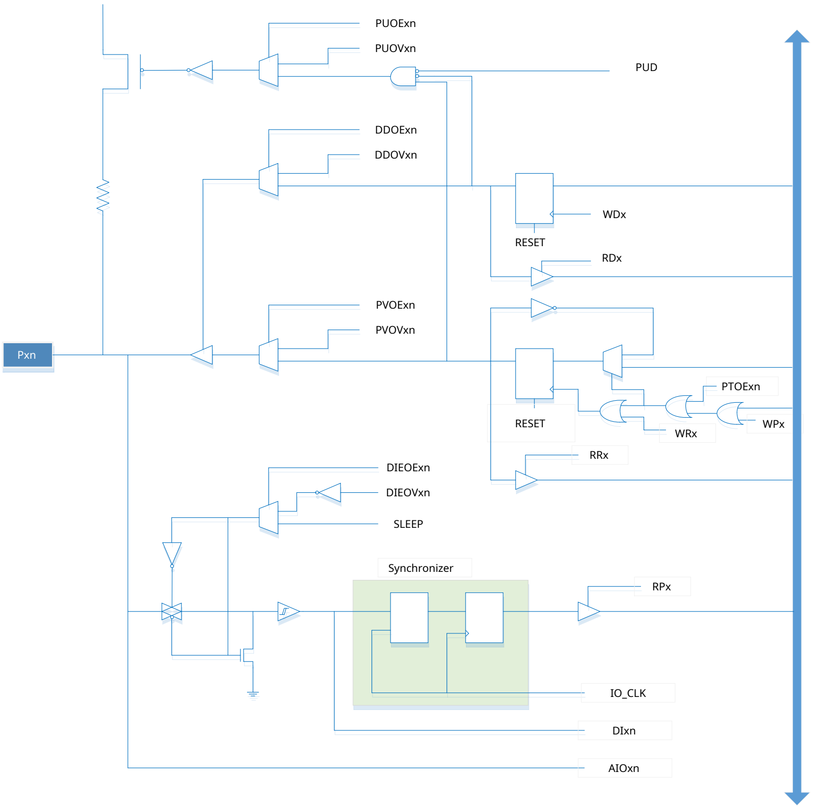
Free port handling

If some ports are not used, it is recommended to drive them to a fixed level. In any case, floating pins will bring more power consumption, and will cause the system to become unstable under strong interference.

The simple way to give a fixed level to the port is to turn on the pull-up resistor of the port. It should be noted that the pull-up resistor is prohibited during the power-on reset process. The method of pull-up resistor will also bring excess leakage. Therefore, it is recommended to use an external pull-up or pull-down resistor connection. It is not recommended to directly connect the port to the power supply or ground, because if these pins are configured as outputs, it may cause a very large current to pass through the port and cause a destructive effect on the chip.

Port multiplexing function

Most ports have multiplexing functions. The following equivalent circuit illustrates the control of the port multiplexing function on the ports. These multiplexing functions do not necessarily exist with all port pins.



PUOExn:	Pxn PULL-UP OVERRIDE ENABLE	PUD:	PULLUP DISABLE
PUOVxn:	Pxn PULL-UP OVERRIDE VALUE	WDx:	WRITE DDRx
DDOExn:	Pxn DATA DIRECTION OVERRRIDE ENABLE	RDx:	READ DDRx
DDOVxn:	Pxn DATA DIRECTION OVERRIDE VALUE Pxn	RRx:	READ PORTx REGISTER
PVOExn:	PORT VALUE OVERRIDE ENABLE Pxn PORT	WRx:	WRITE PORTx
PVOVxn:	VALUE OVERRIDE VALUE Pxn INPUT-ENABLE	RPx:	READ PORTx PIN
DIEOExn:	OVERRIDE ENABLE Pxn INPUT-ENABLE	WPx:	WRITE PINx
DIEOVxn:	OVERRIDE VALUE SLEEP CONTROL	IO_CLK:	I/O CLOCK
SLEEP:		DIxn:	INPUT PIN n ON PORTx
PTOExn:	Pxn PORT TOGGLE OVERRIDE ENABLE	AIOxn:	ANALOG I/O PIN n ON PORTx

General description of multiplex function control signal:

signal	Full name	Function description
PUOE	Pull-up multiplexing enable	This bit is 1, The pull-up is enabled by PVOV Control; if this bit is 0, The pull-up enable is DDxn, PORTxn as well as PUD Joint control
PUOV	Pull up multiplexed value	in case PUOE for 1, This bit is 1 The pin pull-up resistor will be enabled, otherwise the pin pull-up resistor will be disabled
DDOE	Port direction multiplexing enable	The second place is 1, The pin output is enabled by DDOE Control, otherwise by DDxn control
DDOV	Port direction multiplexing value	in case DDOE for 1, The second place is 1, The output function of the pin will be enabled, otherwise the output of the pin will be disabled
PVOE	Port data multiplexing enable	If the second place is 1, And the pin output is enabled, the output value of the pin will be determined by PVOV Control, otherwise by PORTxn Control reference PVOE
PVOV	Port data multiplexing value	Function description
PTOE	Port flip multiplexing enable	The second place is 1,PORTxn Bit will flip if the second bit is 1, The port digital
DIEOE	Digital input enable multiplex enable	input enable will be controlled by DIEOV Control; otherwise there will be MCU Operating status control
DIEOV	Digital input enable multiplexed value	in case DIEOE for 1, The digital input function of the port will be controlled by the next bit, and MCU Independent of operating status
DI	Digital input	This is the digital input signal input to the replacement function module. From I/O As you can see in the circuit diagram below, this value is after the Schmitt trigger, but after I/O Before entering the synchronizer. This signal is connected to the peripheral module, and the peripheral module will To be synchronized
AIO	Analog input	Analog input/output signal, this signal is directly related toI/OofPAD Connected, can be used as an analog two-way signal. This signal is straight Internal ADC, Comparator and other analog modules connected to the port

The following subsection will briefly describe the multiplexed function of each pin and related control signals.

port B Reuse function

Pin multiplexing function description	
PB7	XTALI/TOSC2 (External main crystal oscillator pin XI) PCINT7 (Pin change interrupt 7) XTALO/TOSC1
PB6	(External main crystal oscillator pin XO) PCINT6 (Pin change interrupt 6) SCK (SPI Bus master clock
PB5	input) PCINT5 (Pin change interrupt 5) MISO
PB4	(SPI Bus master input/slave output) PCINT4 (Pin change interrupt 4)

PB3	MOSI (SPI Bus master output/slave input) OC2A (Timer/counter 2 Compare match output A) PCINT3 (Pin change interrupt 3) SSN (SPI
PB2	(Bus slave select input) OC1B (Timer/counter 1 Compare match output B) PCINT2 (Pin change interrupt 2) OC1A (
PB1	Timer/counter 1 Compare match output A) PCINT1 (Pin change interrupt 1) ICP1 (Timer/
PB0	counter 1 Capture input) CLKO (System clock output) PCINT0 (Pin change interrupt 0)

XTALI/TOSC2/PCINT7 – port B Pin 7

XTALI: External crystal pin XI. When used as the clock signal of the crystal oscillator, this pin cannot be used asI/O use.

TOSC2: Timer external crystal oscillator pin 2. When insideRC It is configured as the main working clock of the chip, and the asynchronous timer function is enabled (ASSR Register configuration), this pin will be used as the external crystal oscillator pin of the timer. whenASSR Register AS2 set as 1,EXCLK Is set to 0, The timer/counter is enabled 2 Use the asynchronous clock function of the external crystal oscillator, PB7 Will be with internal I/O The port is disconnected and becomes the inverted output pin of the internal oscillator amplifier. In this mode, the external crystal oscillator is connected to the pin.

PCINT7: Pin change interrupt 7.PB7 It is an external interrupt source. in casePB7 Is used for crystal pins,DDB7,PORTB7 with PINB7 The value of will have no meaning.

XTALO/TOSC1/PCINT6- port B Pin 6

XTALO: External crystal pin XO.

TOSC1: Timer external crystal oscillator pin 1. When insideRC It is configured as the main working clock of the chip, and the asynchronous timer function is enabled (ASSR Register configuration), this pin will be used as the external crystal oscillator pin of the timer. whenASSR Register AS2 set as 1,EXCLK Is set to 0, The timer/counter is enabled 2 Use the asynchronous clock function of the external crystal oscillator, PB6 Will be with internal I/O The port port becomes the input pin of the internal oscillator amplifier. In this mode, the external crystal oscillator is connected to the pin.

PCINT6: Pin change interrupt 6.PB6 It is an external interrupt source. in casePB6 Is used for crystal pins,DDB6,PORTB6 with PINB6 The value of will have no meaning.

SCK/PCINT5- port B Pin 5

SCK: SPI Controller master clock output, slave clock input. whenSPI The controller is configured as a slave device, this pin will be configured as an input pin, Not affected by DDB5 control. whenSPI The controller is configured as the master device, The direction of this pin is determined by DDB5 control. When this pin isSPI After being forced to input, it can still pass PORTB5 Bit controls the pull-up resistor.

PCINT5: Interrupt on pin change.PB5 It is an external interrupt source.

MISO/PCINT4- port B Pin 4

MISO: SPI Control the data input of the master device and the data output of the slave device. whenSPI Is configured as a master device, this pin will be forced to be an input, and is not affected by DDB4 control. whenSPI As a slave device, the data side of this pin

Xiang Yu DDB4 control. When this pin isSPI After the controller is forced as an input, its pull-up resistor can still pass PROTB4 control.

PCINT4: Interrupt on pin change.PB4 It is an external interrupt source.

MOSI/OC2A/PCINT3- port B Pin 3

MOSI: SPI Controller main device data output, slave device data input. whenSPI Is configured as a slave device, this pin will be forced as an input, and is not affected by DDB3 control. whenSPI The controller is configured as the master device, and the method of this pin is determined by DDB3 control. When this pin isSPI The control is forced to be input, and it can still be passed PORTB3 Control its pull-up resistance.

OC2A: Timer/counter 2 of A Group comparison match output.PB3 Can be used as a timer/counter 2 External output of compare match. Must pass at this timeDDB3 Set the pin as an output. Simultaneously,OC2A Also timer 2 of PWM Mode output pin.

PCINT3: Interrupt on pin change.PB3 It is an external interrupt source.

SSN/OC1B/PCINT2- port B Pin 2

SSN: SPI Chip select input from the device. whenSPI When the controller is configured as a slave device, this pin will be forced as an input, and is not affected by DDB2 control. As a slave device,SPI The controller is in SSN It is effective to be driven low. whenSPI The controller is configured as the master device, and the direction of this pin is determined by DDB2 control. When this pin isSPI After the controller is forced to input, it can still pass PORTB2 Control the pull-up resistor.

OC1B: Timer/counter 1 of B Group comparison match output.PB2 Can be used as a timer/counter 1 External output of compare match. Must pass at this timeDDB2 Set the pin as an output. Simultaneously,OC1B Also timer 1 of PWM Mode output pin.

PCINT2: Interrupt on pin change.PB2 It is an external interrupt source.

OC1A/PCINT1- port B Pin 1

OC1A: Timer/counter 1 of A Group comparison match output.PB1 Can be used as a timer/counter 1 External output of compare match. Must pass at this timeDDB1 Set the pin as an output. Simultaneously,OC1A Also timer 1 of PWM Mode output pin.

PCINT1: Interrupt on pin change.PB1 It is an external interrupt source.

ICP1/CLKO/PCINT0- port B Pin 0

ICP1: Timer/counter 1 Capture input pin

CLKO: System working clock output, when CLKPR In the register CLKOE Bit is 1, This pin will be forced to be an output, not DDB0 control. The output frequency is the working clock frequency of the current system.

PCINT0: Interrupt on pin change.PB0 It is an external interrupt source.

port C Reuse function

Pin	Reuse function description

PC7	ADC8(ADC Input channel 8)
	APN2(DAP Reverse input 2)
	PCINT15(Pin level change input 15)

PC6	RESETN (External reset input) PCINT14 (Pin level change input 14)
PC5	ADC5 (ADC Input channel 5) SCL (TWI Clock line) PCINT13 (Pin level change input 13)
PC4	ADC4 (ADC Input channel 4) SDA (TWI Data cable) PCINT12 (Pin level change input 12)
PC3	ADC3 (ADC Input channel 3) PCINT11 (Pin level change input 11) ADC2 (ADC
PC2	Input channel 2) PCINT10 (Pin level change input 10) ADC1 (ADC Input
PC1	channel 1) PCINT9 (Pin level change input 9) ADC0 (ADC Input channel 0)
PC0	PCINT8 (Pin level change input 8)

ADC8/APN2/PCINT15- port C Pin 6

ADC8: ADC External input channel 8

APN2: Inverting input port of differential amplifier 2

PCINT15: Interrupt on pin change. After turning off the external reset input function of this pin,PC7 Can be used as an external interrupt source.

RESETN/PCINT14- port C Pin 6

RESETN: External reset input pin. After power-on reset, this pin defaults to an external reset function. able to passIOCR The register disables the external reset function. After turning off the external reset function, this pin can be used as a general purposeI/O use. However, it should be noted that during power-on and other reset processes, this pin defaults to reset input, so if the user needs to use the generalI/O Function, the external circuit cannot affect the power-on and reset process of the chip, it is recommended to configure this pin as the output function I/O, And add an appropriate pull-up resistor externally.

PCINT14: Interrupt on pin change. After turning off the external reset input function of this pin,PC6 Can be used as an external interrupt source.

SCL/ADC5/PCINT13- port C Pin 5

SCL: TWI Interface clock signal.TWCR In the register TWEN position 1 After, enable TWI interface,PC5 will be TWI Control, become TWI The clock signal of the interface.

ADC5: ADC Input channel 5.DIDR Register is used to turn off digital-analog multiplexing I/O Digital function to avoid the digital part

The impact of points on analog circuits. Please refer toADC Related chapters.

PCINT13: Pin change interrupt 13

SDA/ADC4/PCINT12- port C Pin 4

SDA: TWI Interface data signal.TWCR In the register TWEN position 1 After, enable TWI interface,PC4 will be TWI Control, become TWI The data signal of the interface.

ADC4: ADC Input channel 4.DIDR Register is used to turn off digital-analog multiplexing I/O The digital function to avoid the influence of the digital part on the analog circuit. Please refer toADC Related chapters.

PCINT12: Pin change interrupt 12

ADC3/APN1/PCINT11- port C Pin 3

ADC3: ADC Input channel 3.DIDR Register is used to turn off digital-analog multiplexing I/O The digital function to avoid the influence of the digital part on the analog circuit. Please refer toADC Related chapters.

APN1: Differential amplifier reverse input 1

PCINT11: Pin change interrupt 11

ADC2/APN0/PCINT10- port C Pin 2

ADC2: ADC Input channel 2.DIDR Register is used to turn off digital-analog multiplexing I/O The digital function to avoid the influence of the digital part on the analog circuit. Please refer toADC Related chapters.

APN0: Differential amplifier reverse input 0

PCINT10: Pin change interrupt 10

ADC1/APP1/PCINT9- port C Pin 1

ADC1: ADC Input channel 1.DIDR Register is used to turn off digital-analog multiplexing I/O The digital function to avoid the influence of the digital part on the analog circuit. Please refer toADC Related chapters.

APP1: Differential amplifier forward input 1

PCINT9: Pin change interrupt 9

ADC0/APP0/PCINT8- port C Pin 0

ADC0: ADC Input channel 0.DIDR Register is used to turn off digital-analog multiplexing I/O The digital function to avoid the influence of the digital part on the analog circuit. Please refer toADC Related chapters.

APP0： Differential amplifier positive input 0

PCINT8: Pin change interrupt 8

port D Reuse function

Pin	Reuse function description
PD7	ACXN (Analog comparator 0/1 Common negative input) PCINT23 (Pin change interrupt twenty three)
PD6	AC0P (QFP32: Analog comparator 0 Positive input) OC0A (Timer/counter 0 Compare match output A) OC3A (QFP32: Timer/counter 3 Compare match output A) PCINT22 (Pin change interrupt 22) T1 (Timer/
PD5	counter 1 (External count clock input) OC0B (Timer/counter 0 Compare match output B) PCINT21 (Pin change interrupt 21) XCK
PD4	(USART External clock input/output) DAO(internal 8bit DAC Analog output) T0 (Timer/counter 0 (External count clock input) PCINT20 (Pin change interrupt 20) INT1 (External
PD3	interrupt input 1) OC2B (Timer/counter 2 Compare match output B) PCINT19 (Pin change interrupt 19) INT0 (External interrupt input 0) AC0O (
PD2	Comparators 0 Output) OC3B (QFP32: Timer/counter 3 Compare match output B) PCINT18 (Pin change interrupt 18)
PD1	TXD (USART Data output) OC3A (QFP32: Timer/counter 3 Compare match output A) PCINT17 (Pin change interrupt 17)
PD0	RXD (USART data input) PCINT16 (Pin change interrupt 16)

ACXN/OC2B/PCINT23- port D Pin 7

ACXN: Analog comparator 0/1 Common negative input

OC2B: Timer/counter 2 of B Group comparison match output.PD7 Can be used as a timer/counter 2 External output of compare match. Must pass at this timeDDD7 Set the pin as an output. Simultaneously,OC2B Also timer 2 of PWM Mode output pin;

PCINT23: Pin change interrupt twenty three

AC0P/OC0A/PCINT22- port D Pin 6

AC0P: Analog comparator 0 Positive input.

OC0A: Timer/counter 0 of A Group comparison match output.PD6 Can be used as a timer/counter 0 External output of compare match. Must pass at this timeDDD6 Set the pin as an output. Simultaneously,OC0A Also timer 0 of PWM Mode output pin

PCINT22: Pin change interrupt twenty two

T1/OC0B/PCINT21- port D Pin 5

T1: Timer/counter 1 External count clock input

OC0B: Timer/counter 0 of B Group comparison match output.PD5 Can be used as a timer/counter 0 External output of compare match. Must pass at this timeDDD5 Set the pin as an output. Simultaneously,OC0B Also timer 0 of PWM Mode output pin

PCINT21: Pin change interrupt twenty one

XCK/T0/DAO/PCINT20- port D Pin 4

XCK: Sync mode USART External clock signal

T0: Timer/counter 0 External count clock input

DAO: internal 8 Bit DAC Analog output

PCINT20: Pin change interrupt 20

INT1/OC2B/PCINT19- port D Pin 3

INT1: External interrupt input 1

OC2B: Timer/counter 2 of B Group comparison match output.PD3 Can be used as a timer/counter 2 External output of compare match. Must pass at this timeDDD3 Set the pin as an output. Simultaneously,OC2B Also timer 2 of PWM Mode output pin

PCINT19: Pin change interrupt 19

INT0/OC3B/AC0O/PCINT18- port D Pin 2

INT0: External interrupt input 0

OC3B: Timer counter 3 Compare match output B. only atQFP32 When encapsulating,PD2 versus QFP48/PF2 Merge into one IO ,therefore PF2 Up OC3B Function will also change from PD2 Upper output

AC0O: Analog comparator 0 The comparison result is output directly. byAC0FR Register control

PCINT18: Pin change interrupt 18

TXD/OC3A/PCINT17- port D Pin 1

TXD: transfer data(USART Data output).USART After the transmitter is enabled,PD1 Will be forced to output, not subject to DDD1 control

OC3A: Timer counter 3 Compare match output A. only atQFP32 When encapsulating,PD1 versus QFP48/PF1 Merge into one IO ,therefore PF1 Up OC3A Function will also change from PD1 Upper output

PCINT17: Pin change interrupt 17

RXD/PCINT16- port D Pin 0

RXD: transfer data(USART data input).USART After the receiver is enabled,PD0 Will be forced as input, not subject to DDD0 control. When the pin isUSART After being forced as an input, the pull-up resistor can still pass PORTD0 Bit control

PCINT16: Pin change interrupt 16

port E Reuse function

Pin	Reuse function description
PE7	ADC11 (ADC Input channel 11) PCINT31 (Pin change interrupt 31)
PE6	AVREF (QFP32: ADC External reference voltage) ADC10 (ADC Input channel 10) PCINT30 (Pin change interrupt 30) CLKO (System
PE5	clock output) AC1O (Analog comparator 1 Output) PCINT29 (Pin change interrupt 29)
PE4	OC0A (Timer/counter 0 Compare configuration output A) PCINT28 (Pin change interrupt 28) ADC7 (ADC Input
PE3	channel 7) AC1N (Analog comparator 1 Negative input) PCINT27 (Pin change interrupt 27) SWD
PE2	(SWD Debugger data cable) PCINT26 (Pin change interrupt 26) ADC6 (ADC
PE1	Input channel 6) ACXP (Analog than machine 0/1 Common positive input) PCINT25 (Pin change interrupt 25)
PE0	SWC (SWD Debugger clock input) APN4 (Differential amplifier reverse input 4) PCINT24 (Pin change interrupt twenty four)

ADC11/PCINT31- port E Pin 7

ADC11: ADC External input channel 11

PCINT31: Pin change interrupt 30

AVREF/ADC10/PCINT30- port E Pin 6

AVREF: ADC External reference power input, when used as an analog function, the corresponding digital I/O Set as input and turn off the pull-up resistor to avoid interference from digital circuits to analog circuits

ADC10: ADC Analog input channel 10

PCINT30: Pin change interrupt 30

CLKO/AC1O/PCINT29- port E Pin 5

CLKO: This function is the same as PB0 of CLKO The function is the same. can be used asPB0/CLKO Spare pin

AC1O: Analog comparator 1 Output

PCINT29: Pin change interrupt 29

OC0A/PCINT28- port E Pin 4

OC0A: Timer/counter 0 of A Group comparison match output.PE4 Can be used as a timer/counter 0 External output of compare match. Must pass at this timeDDE4 Set the pin as an output. Simultaneously,OC0A Also timer 0 of PWMMode output pin.

PCINT28: Pin change interrupt 28

ADC7/ AC1N/PCINT27- port E Pin 3

ADC7: ADC Input channel 7.DIDR Register is used to turn off digital-analog multiplexing I/O The digital function to avoid the influence of the digital part on the analog circuit. Please refer toADC Related chapters

AC1N: Analog comparator 1 Negative input

PCINT27: Pin change interrupt 27

SWD/PCINT26- port E Pin 2

SWD: SWD Debugger data cable.PE2 The default is SWD Features. Users can passMCUSR register SWDD position 1 shut down SWD Debugger function.SWD After being closed, the debugging function will not be available.

PCINT26: Pin change interrupt 26

ADC6/ACXP/PCINT25- port E Pin 1

ADC6: ADC Input channel 6.DIDR Register is used to turn off digital-analog multiplexing I/O The digital function to avoid the influence of the digital part on the analog circuit. Please refer toADC Related chapters

ACXP: Analog comparator 0/1 Common positive input

PCINT25: Pin change interrupt 25

SWC/APN4/PCINT24- port E Pin 0

SWC: SWD Debugger clock line.PE0 The default is SWC Features. Users can passMCUSR register SWDD position 1 shut down SWD Debugger function.SWD After being closed, the debugging function will not be available

APN4: Differential amplifier reverse input 4

PCINT24: Pin change interrupt twenty four

port F Reuse function

Pin	Reuse function description
PF7	OC2B (Timer/counter 2 Compare match output B) PCINT39 (Pin change interrupt 39) T3 (Timer/
PF6	counter 3 External clock input) OC2A (Timer/counter 2 Compare match output A) PCINT38 (Pin change interrupt 38) OC1A (Timer/
PF5	counter 1 Compare match output A) PCINT37 (Pin change interrupt 37) OC1B(Timer/counter 1
PF4	Compare configuration output B) ICP3 (Timer/ counter 3 External capture input) PCINT36 (Pin change interrupt 36) OC0B (Timer/
PF3	counter 0 Compare configuration output B) PCINT35 (Pin change interrupt 35) OC3B (Timer/
PF2	counter 3 Compare match output B) PCINT34 (Pin change interrupt 34) OC3A (Timer/counter 3
PF1	Compare match output A) PCINT33 (Pin change interrupt 33) ADC9 (ADC External input channel 9)
PF0	APN3 (Differential amplifier reverse input 3) PCINT32 (Pin change interrupt 32)

OC2B/PCINT39- port F Pin 7

OC2B: Timer/counter 2 Compare match output B. Output selection is subject toPMX1 Register control

PCINT39: Pin change interrupt 39

OC2A/T3/PCINT38- port F Pin 6

OC2A: Timer/counter 2 Compare match output A. Output selection is subject toPMX1 Register control

T3: Timer/counter 3 External clock input

PCINT38: Pin change interrupt 38

0C1A/PCINT37- port F Pin 5

OC1A: Timer/counter 1 Compare match output A. Output selection is subject toPMX0 Register control

PCINT37: Pin change interrupt 37

ICP3/OC1B/PCINT36- port F Pin 4

OC1B: Timer/counter 1 of B Group comparison match output. Output selection is subject toPMX0 Register control

ICP3: Timer/counter 3 External capture input

PCINT36: Pin change interrupt 36

OC3C/OC0B/PCINT35- port F Pin 3

OC0B: Timer/counter 0 of B Group comparison match output. Output selection is subject toPMX0 Register control

OC3C: Timer/counter 3 of C Group comparison match output

PCINT35: Pin change interrupt 35

OC3B/PCINT34- port F Pin 2

OC3B: Timer/counter 3 of B Group comparison match output

PCINT34: Pin change interrupt 34

OC3A/PCINT33- port F Pin 1

OC3A: Timer/counter 3 of B Group comparison match output. Output selection is subject toPMX1 Register control

PCINT33: Pin change interrupt 33

ADC9/APN3/PCINT32- port F Pin 0

ADC9: ADC External mode input channel 9

APN3: Differential amplifier reverse input 3

PCINT32: Pin change interrupt 32

Register definition

port B Output data register- PORTB

PORTB – port B Output data register								
PORTB: 0x05(0x25)				Defaults: 0x00				
Bits	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PORTB	B Group port output register						

port B Direction register- DDRB

DDRB – port B Direction register								
DDRB: 0x04(0x24)				Defaults: 0x00				
DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	DDB	port B Group direction control bit; 1 = Output, 0 = enter						

port B Input data register- PINB

PINB – port B Input data register								
PINB: 0x03(0x23)				Defaults: 0x00				
PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PINB	B Group port status register. readPINB Get the current status of the port directly; write PINBn Bit 1 Will flip PORTBn Output status						

port C Output data register- PORTC

PORTC – port C Output data register								
PORTC: 0x08(0x28)				Defaults: 0x00				
PORTC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PORTC	C Group port output register						

port C Direction register- DDRC

DDRC – port C Direction register								
DDRC: 0x07(0x27)				Defaults: 0x00				
DDRC	DDC7	DDC6	DDC5	DDC4	DDC3	DDC2	DDC1	DDC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	DDC	C Group port direction control bit; 1 = Output,0 = enter						

port C Input data register- PINC

PINC – port C Input data register								
PINB: 0x06(0x26)				Defaults: 0x00				
PINC	PINC7	PINC6	PINC5	PINC4	PINC3	PINC2	PINC1	PINC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PINC	C Group port status register; read PINC Get the current port status write PINC Will flip the current port output						

port D Output data register- PORTD

PORTD – port D Output data register								
PORTD: 0x0B(0x2B)				Defaults: 0x00				
Bits	PD7	PD6	PD5	PD4 PD3		PD2	PD1	PD0
R/W	R/W	R/W	R/W	R/WR/W		R/W	R/W	R/W
Bit definition								

[7:0]	PORTD	D Group port output register

port D Direction register- DDRD

DDRD – port D Direction register								
DDRD: 0x0A(0x2A)				Defaults: 0x00				
DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	DDD	D Group port output direction control register						

port D Input data register- PIND

PIND – port D Input data register								
PIND: 0x09(0x29)				Defaults: 0x00				
PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PIND	D Group port status register read PIND Obtain the current port level status write PINDn for 1, Flip PORTDn The state of the corresponding bit						

port E Output data register- PORTE

PORTE – port E Output data register								
PORTE: 0x0E(0x2E)				Defaults: 0x00				
Bits	PE7	PE6	PE5	PE4	PE3	PE2	PE1	PE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PORTE	E Group port output register						

port E Direction register- DDRE

DDRE – port E Direction register								
DDRE: 0x0D(0x2D)				Defaults: 0x00				
DDRE	DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	DDE	E Group Port Direction Control Register						

port E Input data register- PINE

PINE – port E Input data register								
PINE: 0x0C(0x2C)				Defaults: 0x00				
PINE	PINE7	PINE6	PINE5	PINE4	PINE3	PINE2	PINE1	PINE0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PINE	E Group port status register read PINE Obtain the current port level status write PINEn for 1, Flip PORTEn Bit state						

port F Output register- PORTF

PINF – port F Input data register								
PORTF: 0x14(0x34)				Defaults: 0x00				
Bits	PF7	PF6	PF5	PF4	PF3	PF2	PF1	PF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PORTF	F Group port status register Input mode port, corresponding bit is written 1 Will open the internal pull-up The port of output mode, the corresponding bit is written 1 Will drive output high						

port F Direction control register- DDRF

DDRF – port F Direction control register								
DDRF: 0x13(0x33)				Defaults: 0x00				
Bits	DDF7	DDF6	DDF5	DDF4	DDF3	DDF2	DDF1	DDF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	DDRF	F Group Port Direction Control Register						

port F Status register- PINF

PINF – port F Status register								
PINF: 0x12(0x32)				Defaults: 0x00				
Bits	PINF7	PINF6	PINF5	PINF4	PINF3	PINF2	PINF1	PINF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								
[7:0]	PINF	F Group port status register read PINF Get port F Current level status PINFn write 1, Flip PORTFn The state of the corresponding bit						

Port driver control register- HDR

HDR0 – Port driver control register								
HDR: 0xE0				Defaults: 0x00				
Bit	-	-	HDR5	HDR4	HDR3	HDR2	HDR1	HDR0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Bit definition								

[7:6]	-	Keep unused
5	HDR5	PF5 Output drive control; 1 = 80mA drive, 0 = 12mA drive
4	HDR4	PF4 Output drive control; 1 = 80mA drive, 0 = 12mA drive
3	HDR3	PF2 Output drive control; 1 = 80mA drive, 0 = 12mA drive
2	HDR2	PF1 Output drive control; 1 = 80mA drive, 0 = 12mA drive
1	HDR1	PD6 Output drive control; 1 = 80mA drive, 0 = 12mA drive
0	HDR0	PD5 Output drive control; 1 = 80mA drive, 0 = 12mA drive

Port multiplexing control register 0- PMX0

PMX0 – Port multiplexing control register 0								
PMX0: 0xEE				Defaults : 0x00				
Bit	WCE	C1BF4	C1AF5	C0BF3	C0AC0	SSB1	TXD6	RXD5
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit definition		
7	WCE	PMX0/1 Update enable control; in update PMX0/1 Before registering, you need to write WCE Bit is 1, After 6 Complete the pairing within two system cycles PMX0/1 Update.
6	C1BF4	OC1B Auxiliary output control 1 = OC1B Output to PF4 0 = OC1B Output to PB2 OC1A
5	C1AF5	Auxiliary output control 1 = OC1A Output to PF5 0 = OC1A Output to PB1 OC0B
4	C0BF3	Auxiliary output control 1 = OC0B Output to PF3 0 = OC0B Output to PD5 OC0A
3	C0AC0	Auxiliary output control 0C0A Output by C0AC0 Bit and TCCR0B Register C0AS Bit joint control: {C0AC0, C0AS} = 00 = OC0A Output to PD6 01 = 0C0A Output to PE4 10 = 0C0A Output to PC0 11 = OC0A Simultaneously output to PE4 with PC0 SPSS Auxiliary output control
2	SSB1	1 = SPSS Output to PB1 0 = SPSS Output to PB2
1	TXD6	Serial port TXD Auxiliary output control 1 = TXD Output to PD6, 0 = TXD Output to PD1
0	RXD5	Serial port RXD Auxiliary input control 1 = RXD Input from PD5, 0 = RXD Input from PD0

Port multiplexing control register 1- PMX1

PMX1 – Port multiplexing control register 1								
PMX1: 0xED				Defaults: 0x00				
Bit	-	-	-	-	-	C3AC	C2BF7	C2AF6
R/W	-	-	-	-	-	R/W	R/W	R/W
Bit definition								
[7:3]	-	Keep unused						
2	C3AC	OC3A Auxiliary output control 1 = OC3A Output to QFP48/AC0P 0 = OC3A Output to PF1 OC2B Auxiliary						
1	C2BF7	output control 1 = OC2B Output to PF7 0 = OC2B Output to PD3 OC2A						
0	C2AF6	Auxiliary output control 1 = OC2A Output to PF6 0 = OC2A Output to PB3						
Instructions for use								
PMX0/1 Shared register update protection control bit PMX0[7], Update PMX1 When, please refer to PMX0 register Correct PMX0[7]Control instructions.								

Port multiplexing control register 2 – PMX2

PMX2 – Port multiplexing control register 2								
PMX2: 0xF0				Defaults: 0x00				
Bit	WCE	STSC1	STSC0	-	-	XIEN	E6EN	C6EN
R/W	R/W	R/W	R/W	-	-	R/W	R/W	R/W
Bit definition								
[7]	WCE	PMX2 Update enable control; in update PMX2 Before register, you need to write WCE Bit is 1, After 6 Complete the pairing within two system cycles PMX2 Update. High-speed						
[6]	STSC1	crystal oscillatorIO Start circuit control by PMCR After enabling the high-speed crystal oscillator,STSC1 Automatically enabled. When switching systems After the clock reaches the external high-speed crystal oscillator, STSC1 Automatically clear. The software can also manually clear after the crystal oscillator is stableSTSC1, The crystal oscillator startup circuit has been turned off to save power consumption. Low-						
[5]	STSC0	speed crystal oscillatorIO Start circuit control by PMCR After enabling the low-speed crystal oscillator,STSC0 Automatically enabled. After switching the system clock to an external low-speed crystal oscillator,STSC0 Automatically clear. The software can also manually clear after the crystal oscillator is stableSTSC0, The crystal oscillator startup circuit has been turned off to save power consumption.						
[4:3]	-	Keep unused						
[2]	XIEN	Enable external clock input, need to enable external crystal at the same time						
[1]	E6EN	Enable PE6 Universal IO Function; default PE6 for AVREF Function						
[0]	C6EN	enable PC6 Universal IO Function; default PC6 External reset input						

Pin change interrupt

Z 40 Interrupt source on pin change

Z 5 Interrupt entry

Summary

The interrupt on pin change is controlled by PBn,PCn,PDn, PEn with PFn Pin trigger. As long as the interrupt on pin change is enabled, the interrupt can be triggered even if these pins are configured as outputs. This can be used to generate software interrupts.

Any one enabled PBn Pin flipping will trigger pin level interrupt PCI0, Enabled PCn Pin flip will trigger PCI1, Enabled PDn Pin flip will trigger PCI2, Enabled PEn Pin flip will trigger PCI3. The enable of each pin change interrupt is determined byPCMSK0~4 Register to control. All pin-on-change interrupts are detected asynchronously.

Used as a wake-up source in some sleep modes.

Register definition

Pin Change Interrupt Register list

register	address	Defaults	description
PCICR	0x68	0x00	Pin change interrupt control register
PCIFR	0x3B	0x00	Pin change interrupt flag register
PCMSK0	0x6B	0x00	Pin change interrupt mask register 0
PCMSK1	0x6C	0x00	Pin change interrupt mask register 1
PCMSK2	0x6D	0x00	Pin change interrupt mask register 2
PCMSK3	0x73	0x00	Pin change interrupt mask register 3
PCMSK4	0x74	0x00	Pin change interrupt mask register 4

PCICR – Pin change interrupt control register

PCICR - Pin change interrupt control register								
address: 0x68					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	-	-	-	PCIE4	PCIE3	PCIE2	PCIE1	PCIE0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W

Bit	Name description	
7:5	-	Reserved.
4	PCIE4	Pin change interrupt enable control bit 4. When set PCIE4 Bit is"1"And when the global interrupt is enabled, the pin change interrupt 4 Is enabled. Any one enabledPFn Pin level changes will produce PCI4 Interrupted.PFn The enable of pin interrupts can be controlled by PCMSK4 Register to control. When setPCIE3 Bit is"0"When the pin changes interrupt 3 banned.
3	PCIE3	Pin change interrupt enable control bit 3. When set PCIE3 Bit is"1"And when the global interrupt is enabled, the pin change interrupt 3 Is enabled.

		Any one enabled PEn Pin level changes will produce PCI3 Interrupted.PEn The enable of pin interrupts can be controlled by PCMSK3 Register to control. When setPCIE3 Bit is"0"When the pin changes interrupt 3 banned.
2	PCIE2	Pin change interrupt enable control bit 2. When set PCIE2 Bit is"1"And when the global interrupt is enabled, the pin change interrupt 2 Is enabled. Any one enabledPDn Pin level changes will produce PCI2 Interrupted.PDn The enable of pin interrupts can be controlled by PCMSK2 Register to control. When setPCIE2 Bit is"0"When the pin changes interrupt 2 banned.
1	PCIE1	Pin change interrupt enable control bit 1. When set PCIE1 Bit is"1"And when the global interrupt is enabled, the pin change interrupt 1 Is enabled. Any one enabledPCn Pin level changes will produce PCI1 Interrupted.PCn The enable of pin interrupts can be controlled by PCMSK1 Register to control. When setPCIE1 Bit is"0"When the pin changes interrupt 1 banned.
0	PCIE0	Pin change interrupt enable control bit 0. When set PCIE0 Bit is"1"And when the global interrupt is enabled, the pin change interrupt 0 Is enabled. Any one enabledPBn Pin level changes will produce PCI0 Interrupted.PBn The enable of pin interrupts can be controlled by PCMSK0 Register to control. When setPCIE0 Bit is"0"When the pin changes interrupt 0 banned.

PCIFR – Pin change interrupt flag register

PCIFR - Pin change interrupt flag register								
address: 0x3B						Defaults: 0x00		
Bit	7	6	5	4	3 2		1	0
Name	-	-	-	PCIF4	PCIF3	PCIF2	PCIF1	PCIF0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:5	-	Reserved.
4	PCIF4	Pin change interrupt flag 4. Any one enabled PFn Pin level changes will be set PCIF4. whenPCIE4 And global interrupts are both set,MCU Will jump to PCI4 Interrupt entry address.PFn The enable of pin interrupts can be controlled by PCMSK4 Register to control. Execute interrupt service routine or go to PCIF4 Bit write"1"Will be cleared PCIF4 Bit.
3	PCIF3	Pin change interrupt flag 3. Any one enabled PEn Pin level changes will be set PCIF3. whenPCIE3 And global interrupts are both set,MCU Will jump to PCI3 Interrupt entry address.PEn The enable of pin interrupts can be controlled by PCMSK3 Register to control. Execute interrupt service routine or go to PCIF3 Bit write"1"Will be cleared PCIF3 Bit.
2	PCIF2	Pin change interrupt flag 2. Any one enabled PDn Pin level changes will be set PCIF2. whenPCIE2 And global interrupts are both set,MCU Will jump to PCI2 Interrupt entry address.PDn The enable of pin interrupts can be controlled by PCMSK2 Register to control. Execute interrupt service routine or go toPCIF2 Bit write"1"Will be cleared PCIF2 Bit.
1	PCIF1	Pin change interrupt flag 1.

		Any one enabled PCn Pin level changes will be set PCIF1. whenPCIE1 And global interrupts are both set,MCU Will jump to PCI1 Interrupt entry address.PCn The enable of pin interrupts can be controlled by PCMSK1 Register to control. Execute interrupt service routine or go to PCIF1 Bit write"1"Will be cleared PCIF1 Bit.
0	PCIF0	Pin change interrupt flag 0. Any one enabled PBn Pin level changes will be set PCIF0. whenPCIE0 And global interrupts are both set,MCU Will jump to PCI0 Interrupt entry address.PBn In the pin The enable of the off can be respectively determined by PCMSK0 Register to control. Execute interrupt service routine or go toPCIF0 Bit write"1"Will be cleared PCIF0 Bit.

PCMSK0 – Pin change interrupt mask register 0

PCMSK0 - Pin change mask register 0								
address: 0x6B					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	PCINT7	Pin change enable mask bit 7. When set PCINT7 Bit is"1"Time,PB7 The pin level change interrupt is enabled.PB7 The level change on the pin will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT7 Bit is"0"Time,PB7 The pin level change interrupt is disabled.
6	PCINT6	Pin change enable mask bit6. When set PCINT6 Bit is"1"Time,PB6 The pin level change interrupt is enabled.PB6 The level change on the pin will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT6 Bit is"0"Time,PB6 The pin level change interrupt is disabled.
5	PCINT5	Pin change enable mask bit5. When set PCINT5 Bit is"1"Time,PB5 The pin level change interrupt is enabled.PB5 The level change on the pin will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT5 Bit is"0"Time,PB5 The pin level change interrupt is disabled.
4	PCINT4	Pin change enable mask bit4. When set PCINT4 Bit is"1"Time,PB4 The pin level change interrupt is enabled.PB4 The level change on the pin will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT4 Bit is"0"Time,PB4 The pin level change interrupt is disabled.
3	PCINT3	Pin change enable mask bit3. When set PCINT3 Bit is"1"Time,PB3 The pin level change interrupt is enabled.PB3 The level change on the pin will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT3 Bit is"0"Time,PB3 The pin level change interrupt is disabled.
2	PCINT2	Pin change enable mask bit 2. When set PCINT2 Bit is"1"Time,PB2 The pin level change interrupt is enabled.PB2 The level change on the pin will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT2 Bit is"0"Time,PB2 The pin level change interrupt is disabled.
1	PCINT1	Pin change enable mask bit1. When set PCINT1 Bit is"1"Time,PB1 The pin level change interrupt is enabled.PB1 Pin

		The level change on will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT1 Bit is"0"Time,PB1 The pin level change interrupt is disabled.
0	PCINT0	Pin change enable mask bit0. When set PCINT0 Bit is"1"Time,PB0 The pin level change interrupt is enabled.PB0 The level change on the pin will be set PCIF0If PCIE0 Bit and global interrupt set, will generate PCI0 Interrupted. When setPCINT0 Bit is"0"Time,PB0 The pin level change interrupt is disabled.

PCMSK1 – Pin change interrupt mask register 1

PCMSK1 - Pin change mask register 1								
address: 0x6C					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	PCINT15	Pin change enable mask bit 15. When set PCINT15 Bit is"1"Time,PC7 The pin level change interrupt is enabled.PC7 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT15 Bit is"0"Time,PC7 The pin level change interrupt is disabled.
6	PCINT14	Pin change enable mask bit 14. When set PCINT14 Bit is"1"Time,PC6 The pin level change interrupt is enabled.PC6 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT14 Bit is"0"Time,PC6 The pin level change interrupt is disabled.
5	PCINT13	Pin change enable mask bit 13. When set PCINT13 Bit is"1"Time,PC5 The pin level change interrupt is enabled.PC5 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT13 Bit is"0"Time,PC5 The pin level change interrupt is disabled.
4	PCINT12	Pin change enable mask bit 12. When set PCINT12 Bit is"1"Time,PC4 The pin level change interrupt is enabled.PC4 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT12 Bit is"0"Time,PC4 The pin level change interrupt is disabled.
3	PCINT11	Pin change enable mask bit 11. When set PCINT11 Bit is"1"Time,PC3 The pin level change interrupt is enabled.PC3 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT11 Bit is"0"Time,PC3 The pin level change interrupt is disabled.
2	PCINT10	Pin change enable mask bit 2. When set PCINT10 Bit is"1"Time,PC2 The pin level change interrupt is enabled.PC2 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT10 Bit is"0"Time,PC2 Pin level change interrupt is disabled

		only.
1	PCINT9	Pin change enable mask bit 1. When set PCINT9 Bit is"1"Time,PC1 The pin level change interrupt is enabled.PC1 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT9 Bit is"0"Time,PC1 The pin level change interrupt is
0	PCINT8	disabled. Pin change enable mask bit0. When set PCINT8 Bit is"1"Time,PC0 The pin level change interrupt is enabled.PC0 The level change on the pin will be set PCIF1If PCIE1 Bit and global interrupt set, will generate PCI1 Interrupted. When setPCINT8 Bit is"0"Time,PC0 The pin level change interrupt is disabled.

PCMSK2 – Pin change interrupt mask register 2

PCMSK2 - Pin change mask register 2								
address: 0x6D					Defaults: 0x00			
Bits	7	6	5	4	3	2	1	0
	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	PCINT23	Pin change enable mask bit twenty three. When set PCINT23 Bit is"1"Time,PD7 The pin level change interrupt is enabled.PD7 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT23 Bit is"0"Time,PD7 The pin level change interrupt is disabled. Pin
6	PCINT22	change enable mask bit6. When set PCINT22 Bit is"1"Time,PD6 The pin level change interrupt is enabled.PD6 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT22 Bit is"0"Time,PD6 The pin level change interrupt is disabled. Pin
5	PCINT21	change enable mask bittwenty one. When set PCINT21 Bit is"1"Time,PD5 The pin level change interrupt is enabled.PD5 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT21 Bit is"0"Time,PD5 The pin level change interrupt is disabled. Pin
4	PCINT20	change enable mask bit20. When set PCINT20 Bit is"1"Time,PD4 The pin level change interrupt is enabled.PD4 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT20 Bit is"0"Time,PD4 The pin level change interrupt is disabled. Pin
3	PCINT19	change enable mask bit19. When set PCINT19 Bit is"1"Time,PD3 The pin level change interrupt is enabled.PD3 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT19 Bit is"0"Time,PD3 The pin level change interrupt is disabled. Pin
2	PCINT18	change enable mask bit18.

		When set PCINT18 Bit is"1"Time,PD2 The pin level change interrupt is enabled.PD2 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT18 Bit is"0"Time,PD2 The pin level change interrupt is disabled.
1	PCINT17	Pin change enable mask bit 17. When set PCINT17 Bit is"1"Time,PD1 The pin level change interrupt is enabled.PD1 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT17 Bit is"0"Time,PD1 The pin level change interrupt is disabled.
0	PCINT16	Pin change enable mask bit 16. When set PCINT16 Bit is"1"Time,PD0 The pin level change interrupt is enabled.PD0 The level change on the pin will be set PCIF2If PCIE2 Bit and global interrupt set, will generate PCI2 Interrupted. When set PCINT16 Bit is"0"Time,PD0 The pin level change interrupt is disabled.

PCMSK3 – Pin change interrupt mask register 3

PCMSK3 - Pin change mask register 3								
address: 0x73					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	PCINT31	PCINT30	PCINT29	PCINT28	PCINT27	PCINT26	PCINT25	PCINT24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	PCINT31	Pin change enable mask bit 31. When set PCINT31 Bit is"1"Time,PE7 The pin level change interrupt is enabled.PE7 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT31 Bit is"0"Time,PE7 The pin level change interrupt is disabled. Pin
6	PCINT30	change enable mask bit30. When set PCINT30 Bit is"1"Time,PE6 The pin level change interrupt is enabled.PE6 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT30 Bit is"0"Time,PE6 The pin level change interrupt is disabled. Pin
5	PCINT29	change enable mask bit39. When set PCINT29 Bit is"1"Time,PE5 The pin level change interrupt is enabled.PE5 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT29 Bit is"0"Time,PE5 The pin level change interrupt is disabled. Pin
4	PCINT28	change enable mask bit28. When set PCINT28 Bit is"1"Time,PE4 The pin level change interrupt is enabled.PE4 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT28 Bit is"0"Time,PE4 The pin level change interrupt is disabled. Pin
3	PCINT27	change enable mask bit27.

		When set PCINT27 Bit is"1"Time,PE3 The pin level change interrupt is enabled.PE3 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT27 Bit is"0"Time,PE3 The pin level change interrupt is disabled.
2	PCINT26	Pin change enable mask bit 26. When set PCINT26 Bit is"1"Time,PE2 The pin level change interrupt is enabled.PE2 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT26 Bit is"0"Time,PE2 The pin level change interrupt is disabled.
1	PCINT25	Pin change enable mask bit 25. When set PCINT25 Bit is"1"Time,PE1 The pin level change interrupt is enabled.PE1 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT25 Bit is"0"Time,PE1 The pin level change interrupt is disabled.
0	PCINT24	Pin change enable mask bit twenty four. When set PCINT24 Bit is"1"Time,PE0 The pin level change interrupt is enabled.PE0 The level change on the pin will be set PCIF3If PCIE3 Bit and global interrupt set, will generate PCI3 Interrupted. When set PCINT24 Bit is"0"Time,PE0 The pin level change interrupt is disabled.

PCMSK4 – Pin change interrupt mask register 4

PCMSK4 - Pin change mask register 4								
address: 0x74					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	PCINT39	PCINT38	PCINT37	PCINT36	PCINT35	PCINT34	PCINT33	PCINT32
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7	PCINT39	Pin change enable mask bit 39. When set PCINT39 Bit is"1"Time,PF7 The pin level change interrupt is enabled.PF7 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT39 Bit is"0"Time,PF7 The pin level change interrupt is disabled. Pin						
6	PCINT38	change enable mask bit38. When set PCINT38 Bit is"1"Time,PF6 The pin level change interrupt is enabled.PF6 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT38 Bit is"0"Time,PF6 The pin level change interrupt is disabled. Pin						
5	PCINT37	change enable mask bit37. When set PCINT37 Bit is"1"Time,PF5 The pin level change interrupt is enabled.PF5 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT37 Bit is"0"Time,PF5 The pin level change interrupt is disabled. Pin						
4	PCINT36	change enable mask bit36.						

		When set PCINT36 Bit is"1"Time,PF4 The pin level change interrupt is enabled.PF4 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT36 Bit is"0"Time,PF4 The pin level change interrupt is disabled.
3	PCINT35	Pin change enable mask bit 35. When set PCINT35 Bit is"1"Time,PF3 The pin level change interrupt is enabled.PF3 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT35 Bit is"0"Time,PF3 The pin level change interrupt is disabled.
2	PCINT34	Pin change enable mask bit 34. When set PCINT34 Bit is"1"Time,PF2 The pin level change interrupt is enabled.PF2 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT34 Bit is"0"Time,PF2 The pin level change interrupt is disabled.
1	PCINT33	Pin change enable mask bit 33. When set PCINT33 Bit is"1"Time,PF1 The pin level change interrupt is enabled.PF1 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT33 Bit is"0"Time,PF1 The pin level change interrupt is disabled.
0	PCINT32	Pin change enable mask bit 32. When set PCINT31 Bit is"1"Time,PF0 The pin level change interrupt is enabled.PF0 The level change on the pin will be set PCIF4If PCIE4 Bit and global interrupt set, will generate PCI4 Interrupted. When set PCINT32 Bit is"0"Time,PF0 The pin level change interrupt is disabled.

Timer/counter 0 (TMR0)

Z 8 Bit counter

Z Two independent comparison units

Z When a comparison match occurs, the counter is automatically cleared and automatically loaded

Z Phase-corrected without interference pulses PWMOutput

Z Frequency generator

Z External event counter

z 10 Bit clock prescaler

Z Overflow and compare match interrupt

Z With dead time control

Z 6 Selectable trigger sources are automatically turned off PWMOutput

Z Generate high-speed and high-resolution in high-speed clock mode (500KHz@7Bit)PWM

Overview

TC0 Is a universal 8 Bit timer counter module, support PWMOutput, can accurately generate waveform.TC0 contain 1 Count clock generation unit,1 A 8 Bit counter, waveform generation mode control unit and 2 Output comparison unit. Simultaneously,TC0 Can be combined with TC1 Share 10 Bit prescaler, can also be used independently 10 Bit prescaler. Prescaler to system clockclkio Or high-speed clock rcm2x(internal 32M RC Oscillator output clock rc32mof 2 Frequency multiplication) Divide the frequency to generate the count clock Clkt0. The waveform generation mode control unit controls the working mode of the counter and the generation of the comparison output waveform. According to different working modes, the counter counts each clockClkt0 Realize the operation of clearing, adding one or subtracting one.Clkt0 It can be generated by an internal clock source or an external clock source. When the count value of the counterTCNT0 Reach the maximum value (equal to the maximum value 0xFF Or output compare register OCR0A,defined as TOP, The maximum value is defined as MAX To show the difference), the counter will be cleared or decremented by one. When the count value of the counterTCNT0 Reach a small value (equal to 0x00,defined as BOTTOM), the counter will increment by one. When the count value of the counterTCNT0 Arrivals OCR0A/OCR0B When a comparison match occurs, it will be cleared or set to output a comparison signal OC0A/OC0BTo produce PWM Waveform. When the insertion of dead time is enabled, the set dead time (DTR0 The count clock corresponding to the register) will be inserted into the generated PWMIn the waveform. Software can be cleared byCOM0A/COM0B Bit is zero to turn off OC0A/OC0B Waveform output, or set the corresponding trigger source, the hardware will automatically clear when a trigger event occurs COM0A/COM0B Bit to close OC0A/OC0B Waveform output.

The count clock can be generated by an internal or external clock source. The selection of clock source and frequency division are determined by TCCR0B Register CS0 Bit to control, see detailed description TC0 with TC1 Prescaler chapter.

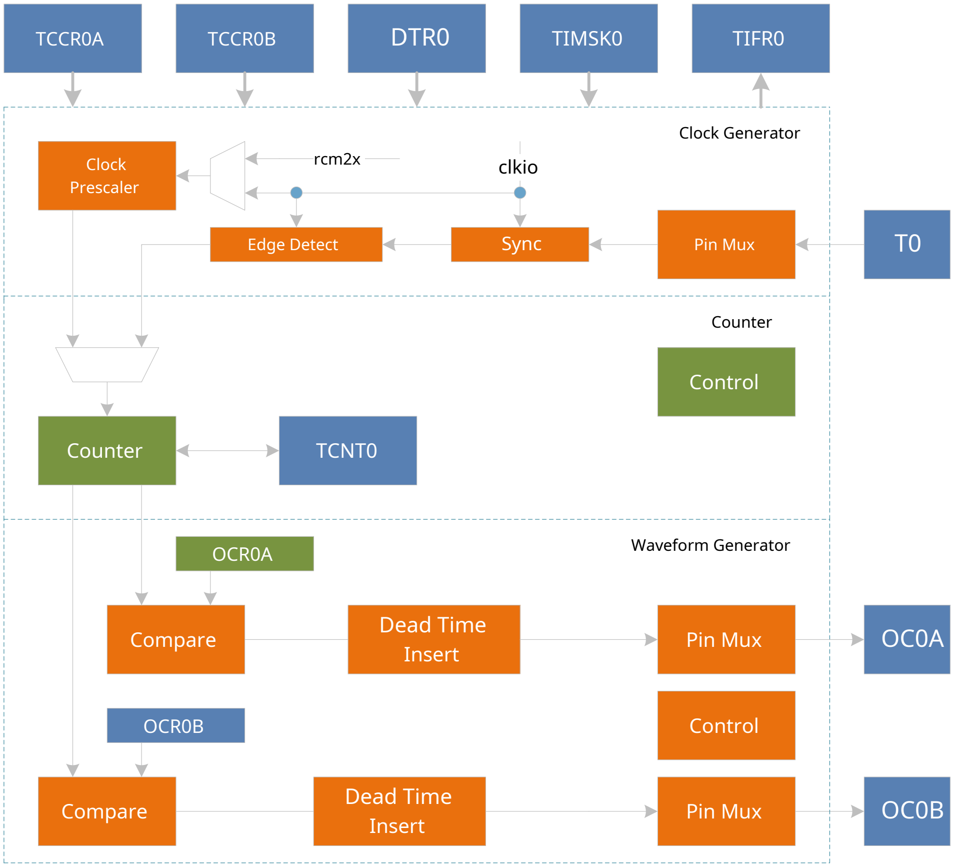
The length of the counter is 8 Bit, supports two-way counting. The waveform generation mode, that is, the working mode of the counter, is determined byTCCR0A with TCCR0B RegisterWGM0 Bit to control. According to different working modes, the counter counts each clockClkt0 Realize the operation of clearing, adding one or subtracting one. When the count overflows, it is located atTIFR0 Register overflow flag TOV0 The bit will be set. Can be generated when the interrupt is enabledTC0 Count overflow interrupt.

Output comparison unit pair count value TCNT0 And output compare register OCR0A with OCR0B To compare the value of when TCNT0 equal OCR0A or OCR0B Is called a comparison match, and is located at TIFR0 Register output compare flag OCF0A or OCF0B The bit will be set. Can be generated when the interrupt is enabledTC0 Output compare match interrupt. It should be noted that inPWMIn working mode,OCR0A with OCR0B The register is a double-buffered register. In normal mode and

CTC In mode, the double buffering function is invalid. When the count reaches a large value or a small value, the value in the buffer register is updated to the compare register synchronouslyOCR0A with OCR0B Go in. See the description of the working mode chapter for details.

The waveform generator uses comparison matching and counting overflow to generate output comparison waveform signals according to the waveform generation mode control and the comparison output mode control. OC0A with OC0B. For specific generation methods, see the description in the chapters on working modes and registers. To compare the output waveform signalOC0A with OC0B When outputting to the corresponding pin, the data direction register of the pin must also be set as output.

The picture below is TC0 Internal structure diagram.TC0 contain 1 Count clock generation unit,1 A 8 Bit counter,2 Lose Compare unit and 2 A waveform generation control unit.



TC0 Structure chart

Operating mode

Timer counter 0 There are four different working modes, including normal mode (Normal), cleared on comparison match (CTC) Mode, fast pulse width modulation (FPWM) Mode and phase correction pulse width modulation (PCPWM) Mode, which is controlled by the waveform generation modeWGM0[2:0]To choose. The four modes are described in detail below. Since there are two independent output comparison units, use"A"with"B"To indicate, in lowercase"X"To represent these two output comparison unit channels.

Normal mode

The normal mode is the simple working mode of the timer counter. At this time, the waveform generation mode control bitWGM0[2:0]=0, The maximum value of the count TOP for MAX(0xFF). In this mode, the counting method increases by one for each counting clock. When the counter reachesTOP Back after overflow BOTTOM Start accumulation again. Count valueTCNT0 The timer counter overflow flag is set in the same counting clock that becomes zero TOV0. In this modeTOV0 The sign is like the first 9 The counting bit is only set but not cleared. Overflow interrupt service routine will be automatically clearedTOV0 Mark, the software can use it to improve the resolution of the timer counter. There is no special situation to consider in the normal mode, and a new count value can be written at any time. Set upOC0xThe output comparison signal can only be obtained when the data direction register of the pin is outputOC0x的waveform. whenCOM0x=1

When a comparison match occurs, it will flip OC0x Signal, the frequency of the waveform in this case can be calculated with the following formula:

$$f_{oc0xnormal} = f_{sys/(}2*N*256)$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

The output compare unit can be used to generate interrupts, but it is not recommended to use interrupts in normal mode, which will take up too much CPU time.

CTC mode

Set up WGM0[2:0]=2 Time counter 0 enter CTC Mode, counted Big value TOP for OCR0A. in In this mode, the counting method is incremented by one for each counting clock. When the counter value TCNT0 equal TOP The hour counter is cleared.OCR0A Defines the maximum value of the count, that is, the resolution of the counter. This mode allows users to easily control the frequency of the compare match output, and also simplifies the operation of counting external events.

When the counter reaches the maximum value of the count, the compare match flag is output OCF0 Is set, an interrupt will be generated when the corresponding interrupt enable bit is set. Can be updated in the interrupt service routineOCR0A The register is the large value of the count. In this mode OCR0A Double buffering is not used. Be careful when updating the large value to close to the small value when the counter is working with no prescaler or a very low prescaler. If writeOCR0A Is less than the value at the time TCNT0 Value, the counter will lose a comparison match. Before the next comparison match occurs, the counter has to count toTOP, And then from BOTTOM Start counting toOCR0Avalue. Same as normal mode, Count value back toBOTTOMSet in the count clock TOV0Sign.

Set upOC0xThe output comparison signal can only be obtained when the data direction register of the pin is outputOC0x的waveform. whenCOM0x=1

When a comparison match occurs, it will flip OC0x Signal, the frequency of the waveform in this case can be calculated with the following formula:

$$f_{oc0xctc} = f_{sys/(}2*N*(1+OCR0x))$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

It can be seen from the formula that when setting OCR0A for 0x0 And when there is no prescaler, the waveform can The maximum frequency is $f_{sys/}2$ Output be obtained.

fast PWMmode

Set up WGM0[2:0]=3 or 7 Time counter 0 Enter fast PWM Mode, which can be used to generate high frequency PWMWaveform, count the big value TOP Respectively MAX(0xFF)or OCR0x. fastPWMMode and others PWMThe mode is different in that it is a one-way operation. Counter value0x00 Accumulate to TOP Back again BOTTOM Recount. When the count valueTCNT0 Arrivals OCR0x or BOTTOMWhen, output the comparison signal OC0x Will be set or cleared, depending on the comparison output mode COM0x See the register description for details. Due to the one-way operation, fastPWMThe operating frequency of the mode is phase correction with bidirectional operation PWM Twice the pattern. High frequency characteristics make fastPWM Mode is suitable for power regulation, rectification and DAC application. High-frequency signals can reduce the size of external components (inductance, capacitance, etc.), thereby reducing system costs.

When the count value reaches a large value, the timer counter overflow flag TOV0 Will be set and update the value of the compare buffer

To the comparison value. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routineOCR0x register. Set upOC0x The output comparison signal can only be obtained when the data direction register of the pin is output OC0x 的waveform. The frequency of the waveform can be calculated with the following formula:

$$f_{oc0xfpwm} = f_{sys/(}N*(1+TOP))$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

when TCNT0 with OCR0x When a comparison match occurs, the waveform generator is set (cleared)OC0x Signal when TCNT0 When cleared, the waveform generator is cleared (set)OC0x Signal to generate PWMwave. thusOCR0x Extreme values will produce special PWM Waveform. whenOCR0x Set as 0x00 When the output PWM For every (1+TOP)There is a narrow spike in each count clock. when OCR0x When set to a large value, the output waveform is a continuous high or low level.

Phase correction PWMmode

When set WGM0[2:0]=1 or 5 Time counter 0 Enter phase correction PWM Mode, the maximum value of the count TOP Respectively MAX(0xFF)or OCR0A. The counter adopts two-way operation, byBOTTOM Increment to TOP, And then decrease to BOTTOM, Repeat this operation again. Count arrivalTOP with BOTTOMChange the counting direction every time, the count value is in TOP or BOTTOM Only one counting clock stays on each of them. In the process of increment or decrement, the count valueTCNT0 versus OCR0x When matching, output comparison signal OC0x Will be cleared or set, depending on the comparison output mode COM0x setting. Compared with the one-way operation, the large frequency available for the two-way operation is smaller, but its excellent symmetry is more suitable for motor control. Phase correctionPWM Mode, when the count reaches BOTTOM Set when TOV0 Sign when the count arrives TOP When updating the value of the comparison buffer to the comparison value. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routineOCR0x register.

Set up OC0x The output comparison signal can only be obtained when the data direction register of the pin is output OC0x 的waveform. The frequency of the waveform can be calculated with the following formula:

$$f_{oc0xpcpwm} = f_{sys/(}N*TOP*2)$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

During the count-up process, when TCNT0 versus OCR0x When there is a match, the waveform generator is cleared (set)OC0x signal. In the process of counting down, whenTCNT0 versus OCR0x When there is a match, the waveform generator is set (cleared)OC0x signal. thus OCR0x The extreme value of will produce a special PWMwave. whenOCR0x Set Large value or At small values,OC0x Signal output to keep low or high all the time.

In order to ensure the output PWMThe symmetry of the wave on both sides of the small value, when there is no comparison match, it will be flipped in two cases OC0x signal. The first situation is whenOCR0x The value of the large value 0xFF When changing to other data. whenOCR0x Is a large value, when the count value reaches a large value,OC0x The output of is the same as the result of the comparison match when counting in descending order, that is, keep OC0x constant. At this time, the comparison value will be updated to the new oneOCR0x Value (not 0xFF),OC0x The value of will be maintained until a comparison match occurs during ascending counting and it is reversed. at this timeOC0x The signal is not symmetrical about the small value, so it needs to be TCNT0 Flip when reaching a large value OC0x Signal, that is, there is no rollover when a comparison match occurs OC0x The first case of the signal. The second situation is whenTCNT0 From OCR0x When the high value starts to count, a comparison match will be lost, which will cause asymmetry. Also need to flipOC0x The signal to achieve symmetry on both sides of the small value.

PWMAutomatic shutdown and restart of output

When set TCCR0A Register DOC0x When the bit is high,PWMThe automatic shutdown function of the output will be enabled. When the trigger condition is met, the hardware will clear the corresponding COM0x Bit, will PWM output signal OC0x Disconnect from its output pin and switch to general purpose IO Output, achieve PWM The output is automatically turned off. At this time, the state of the output pin can beIO To control the output of the port.

PWMAfter the automatic shutdown of the output is enabled, the trigger conditions need to be set. TCCR0C Register DSX0n Bit to select the trigger source. Trigger sources include analog comparator interrupt, external interrupt, pin level change interrupt and timer overflow interrupt. For details, please refer toTCCR0C Register description. When one or some trigger sources are selected as the trigger condition, the hardware will clear these interrupt flag bits at the same time as they are set.COM0x Bit to close PWMOutput.

Close when a trigger event occurs PWMAfter the output, the timer module does not have the corresponding interrupt flag bit, and the software needs to know the trigger condition and trigger event by reading the interrupt flag bit of the trigger source.

when PWM When the output is automatically turned off and the output needs to be restarted again, the software only needs to be reset COM0x Bit to switch OC0x The signal is output to the corresponding pin. It should be noted that after the automatic shutdown occurs, the timer does not stop working.
OC0x The status of the signal is also constantly being updated. Software can set after timer overflow or compare match COM0x Bit to output OC0x Signal so that you can get a clear PWMOutput status.

Dead time control

Set up DTEN0 Bit is"1"When the function of inserting the dead time is enabled,OC0A with OC0B The output waveform will be in B The set dead time is inserted on the basis of the waveform generated by the channel comparison output. The length of the time is DTR0 The time value corresponding to the count clock number of the register. As shown below,OC0A with OC0B The dead time insertion is based on the channel B The comparison output waveform is the reference. whenCOM0A with COM0B Same as"2"or"3"Time,OC0A The polarity of the waveform and OC0B The polarity of the waveform is the same, when COM0A with COM0B Respectively"2"or"3"Time,OC0A The waveform and OC0B The polarity of the waveform is opposite.

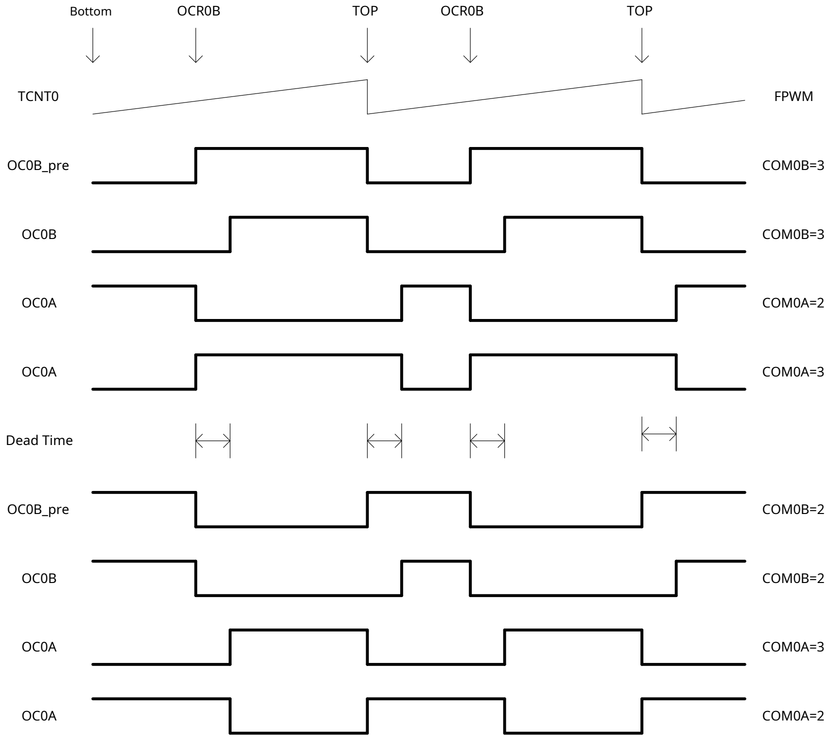


Figure 1 FPWMMode TC0 Dead time control

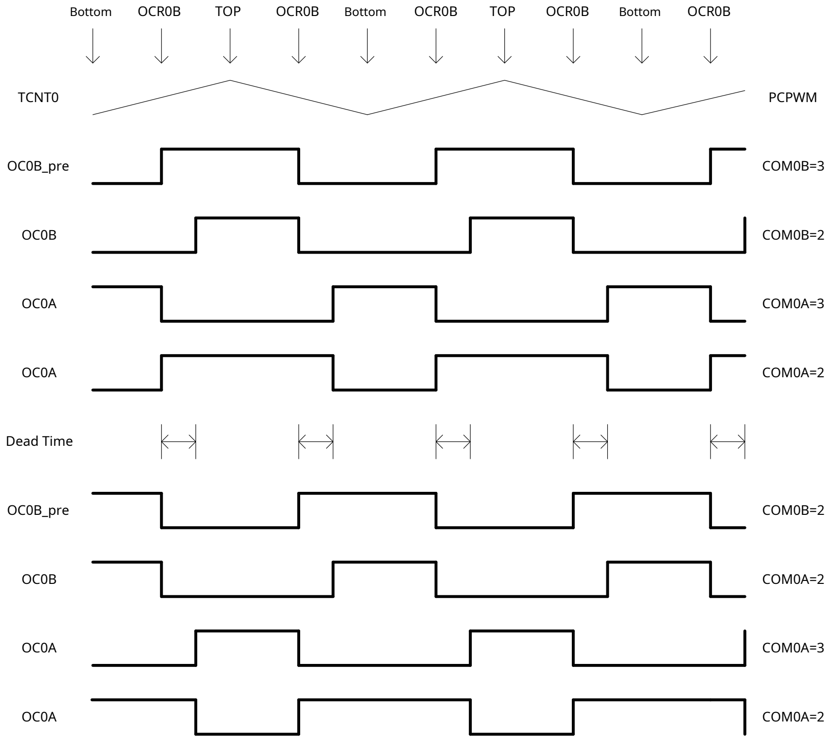


Figure 2 PCPWMMode TC0 Dead time control

Set up DTEN0 Bit is"0"Time, the function of inserting the dead time is disabled,OC0A with OC0B The output waveform of is the waveform generated by the respective comparison output.

High-speed clock mode

In high-speed clock mode, a higher frequency clock is used as the clock source for counting to generate higher speed and higher resolution PWMWaveform. This high-frequency clock is passed to the internal32M RC Oscillator's output clock rc32mget on 2 Multiplied to produce. Therefore, before entering the high frequency mode, you need to enable the internal32MRC The frequency multiplication function of the oscillator, Set TCKCSR Register F2XEN Bit and wait for a certain period of time until the output of the multiplier clock signal is stable. Then, you can setTCKCSR of TC2XS0 Bit to make the timer counter enter the high-speed clock mode.

In this mode, the system clock and high-speed clock are asynchronous, and some registers (see TC0 The register list) works in the high-speed clock domain. Therefore, the configuration and reading of this type of register are also asynchronous, and attention should be paid to the operation.

There are no special requirements for non-continuous read and write operations on registers in the high-speed clock domain. When continuous read and write operations are performed, you need to wait for a system clock. Follow the steps below:

1) Write register A；

2) Waiting for a system clock (NOP Or register under the operating system clock);

3) Read or write register A or B.

4) Waiting for a system clock (NOP Or registers under the operating system clock).

When reading the registers in the high-speed clock domain, except TCNT0 All external registers can be read directly. When the counter is still counting,TCNT0 The value of will change with the high-speed clock, and the counter can be paused (set CS0 Zero) and then read TCNT0 Value.

Register definition

TC0 Register list

register	address	Defaults	description
TCCR0A*	0x44	0x00	TC0 Control register A
TCCR0B*	0x45	0x00	TC0 Control register B
TCNT0*	0x46	0x00	TC0 Count value register
OCR0A*	0x47	0x00	TC0 Output compare register A
OCR0B*	0x48	0x00	TC0 Output compare register B
DSX0*	0x49	0x00	TC0 Trigger source control register
DTR0*	0x4F	0x00	TC0 Dead time register
TIMSK0	0x6E	0x00	Timer counter 0 Interrupt mask register
TIFR0	0x35	0x00	Timer counter 0 Interrupt flag register
TCKCSR	0xEC	0x00	TC Clock control and status register

【note】

band"*"The registers work in the system clock and high-speed clock domains, without"*"The registers only work in the system clock domain under.

TC0 Control register A- TCCR0A

TCCR0A -TC0 Control register A								
address: 0x44					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	COM0A1	COM0A0	COM0B1	COM0B0	DOC0B	DOC0A	WGM01	WGM00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	COM0A1	TC0 Comparison match A The output mode controls the high bit. COM0A1 with COM0A0 Together to form a comparative output mode control COM0A[1:0]To control OC0A The output waveform. in caseCOM0A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC0A Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM0A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
6	COM0A0	TC0 Comparison match A Output mode control low bit. COM0A0 with COM0A1 Together to form a comparative output mode control COM0A[1:0] To control OC0A The output waveform. in caseCOM0A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC0A Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM0A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
5	COM0B1	TC0 Comparison match B The output mode controls the high bit. COM0B1 with COM0B0 Together to form a comparative output mode control COM0B[1:0]To control OC0B The output waveform. in caseCOM0B of 1 Bit or 2 Bits are set,

		The output comparison waveform occupies OC0B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM0B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
4	COM0B0	TC0 Comparison match B Output mode control low bit. COM0B0 with COM0B1 Together to form a comparative output mode control COM0B[1:0] To control OC0B The output waveform. in caseCOM0B of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC0B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM0B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
3	DOC0B	TC0 Turn off the output compare enable control high bit. when DOC0B Bit is"1"When the trigger source turns off and outputs the comparison signal OC0B Is enabled. When a trigger event occurs, the hardware is automatically clearedCOM0B Bit, off OC0B Waveform output. Software through settingsCOMB Can be reopened PWMOutput. whenDOC0B Bit is"0" When the trigger source turns off and outputs the comparison signal OC0B banned.
2	DOC0A	TC0 Turn off the output compare enable control low bit. When set DOC0A Bit is"1"When the trigger source turns off and outputs the comparison signal OC0A Is enabled. When a trigger event occurs, the hardware automatically shuts downOC0A Waveform output. When setDOC0A Bit is"0"When the trigger source turns off and outputs the comparison signal OC0A banned. Will not close when a trigger event occursOC0A Waveform output.
1	WGM01	TC0 The waveform generation mode controls the neutral position. WGM01 with WGM00,WGM02 Form the waveform generation mode control together WGM0[2:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
0	WGM00	TC0 Waveform generation mode control low bit. WGM00 with WGM01,WGM02 Form the waveform generation mode control together WGM0[2:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.

TC0 Control register B- TCCR0B

TCCR0B -TC0 Control register B								
address: 0x45					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	FOC0A	FOC0B	OC0AS	DTEN0	WGM02	CS02	CS01	CS00
R/W	W	W	W/R	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	FOC0A	TC0 Force output comparison A Control bit. Work in nonPWM Mode, you can compulsorily output the comparison bit FOC0A write "1"Way to generate comparison matches. Force compare match will not be setOCF0A Flag, it will not reload or clear the timer, but the output pin OC0A Will be followed COM0A The settings are updated accordingly, just as if a comparison match actually occurred. ReadFOC0A The return value of is always zero.
6	FOC0B	TC0 Force output comparison B Control bit.

<table>
<tr><td></td><td></td><td colspan="2">Work in non PWM Mode, you can compulsorily output the comparison bit FOC0B write "1"Way to generate comparison matches. Force compare match will not be setOCF0B Flag, it will not reload or clear the timer, but the output pin OC0B Will be followed COM0B The settings are updated accordingly, just as if a comparison match actually occurred. ReadFOC0B The return value of is always zero.</td></tr>
<tr><td>5</td><td>OC0AS</td><td colspan="2">OC0A Output port selection control bit. When setOC0AS Bit is"0"Time,OC0A The waveform from the pin PD6 Output; when set OC0AS Bit is"1"Time,OC0A The waveform from the pin PE4 Output (QFP32 Valid under encapsulation).</td></tr>
<tr><td>4</td><td>DTEN0</td><td colspan="2">TC0 Dead time enable control bit.
When set DTEN0 Bit is"1"When the time, the dead time insertion is enabled.OC0A with OC0B All in B The dead time is inserted on the basis of the waveform generated by the channel comparison output, and the inserted dead time interval is determined by DTR0 The counting time corresponding to the register is determined.OC0A The polarity of the output waveform is determined by COM0 with COM0B The corresponding relationship is determined, see OC0A After inserting the dead time, the waveform polarity is shown in the table.
When set DTEN0 Bit is"0"Time, the dead time insertion is prohibited,OC0A with OC0B The waveform of is the waveform generated by the respective comparison output.</td></tr>
<tr><td>3</td><td>WGM02</td><td colspan="2">TC0 The waveform generation mode controls the high bit.
WGM02 with WGM00,WGM01 Form the waveform generation mode control together WGM0[2:0], Control the counting method and waveform generation method of the counter, see the wave for details Shape generation mode table description.</td></tr>
<tr><td>2</td><td>CS02</td><td colspan="2">TC0 The clock selection controls the high bit.
Used to select the timer counter 0 Clock source.</td></tr>
<tr><td>1</td><td>CS01</td><td colspan="2">TC0 Clock selection control center position.
Used to select the timer counter 0 Clock source.</td></tr>
<tr><td>0</td><td>CS00</td><td colspan="2">TC0 Clock selection control low bit.
Used to select the timer counter 0 Clock source.</td></tr>
<tr><td></td><td></td><td>CS0[2:0]</td><td>description</td></tr>
<tr><td></td><td></td><td>0</td><td>No clock source, stop counting</td></tr>
<tr><td></td><td></td><td>1</td><td>clk_{sys}</td></tr>
<tr><td></td><td></td><td>2</td><td>clk_{sys}/8, From the prescaler</td></tr>
<tr><td></td><td></td><td>3</td><td>clk_{sys}/64, From the prescaler</td></tr>
<tr><td></td><td></td><td>4</td><td>clk_{sys}/256, From the prescaler</td></tr>
<tr><td></td><td></td><td>5</td><td>clk_{sys}/1024, From the prescaler</td></tr>
<tr><td></td><td></td><td>6</td><td>External clock T0 Pin, falling edge trigger</td></tr>
<tr><td></td><td></td><td>7</td><td>External clock T0 Pin, rising edge trigger</td></tr>
</table>

The following table is not PWM Mode (i.e. normal mode and CTC Mode), the comparison output mode controls the output comparison waveform system.

COM0x[1:0]	description
0	OC0x Disconnect, general IO Flip when the port
1	operation compares match OC0x signal
2	Clear on comparison match OC0x signal
3	Set on compare match OC0x signal

The following table is fast PWMThe comparison output mode in the mode controls the output comparison waveform.

COM0x[1:0]	description
0	OC0x Disconnect, general IO Mouth operation
1	Keep
2	Clear on comparison match OC0x Signal, set when the maximum value matches OC0x Set when the
3	signal compare matches OC0x Signal, cleared when the maximum value matches OC0x signal

The following table shows the control of the output comparison waveform in the comparison output mode in the phase correction mode.

COM0x[1:0]	description
0	OC0x Disconnect, general IO Mouth operation
1	Keep
2	Cleared when compare matches in ascending order OC0x Signal, compare match in descending count Set when OC0x signal
3	Set when compare match in ascending count OC0x Signal, compare match in descending count Time clear OC0x signal

The following table shows the waveform generation mode control.

WGM0[2:0]	Operating mode	TOP value	Update OCR0X time	Position TOV0 time
0	Normal	0xFF	immediately	MAX
1	PCPWM	0xFF	TOP	BOTTOM
2	CTC	OCR0A	immediately	MAX
3	FPWM	0xFF	TOP	MAX
4	Keep	-	-	-
5	PCPWM	OCR0A	TOP	BOTTOM
6	Keep	-	-	-
7	FPWM	OCR0A	TOP	TOP

The following table shows when the dead time is enabled OC0A The polarity control of the signal output waveform.

In dead time enable mode OC0A Polarity control of signal output waveform

DTEN0	COM0A[1:0]	COM0B[1:0]	description
0	-	-	OC0A The signal polarity is determined by OC0A Comparison output mode control
1	0	-	OC0A Disconnect, general IO Mouth operation
1	1	-	Keep
1	2	2	OC0A Signal and OC0B Same signal polarity
		3	OC0A Signal and OC0B Signal polarity is opposite
1	3	2	OC0A Signal and OC0B Signal polarity is opposite
		3	OC0A Signal and OC0B Same signal polarity

【note】:

OC0B The polarity of the signal output waveform is determined by OC0B The comparison output mode control is the same as the dead time mode without enabling.

TC0 Control register C – TCCR0C

TCCR0C- ***TC0 Control register C***								
address: 0x49					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	DSX07	DSX06	DSX05	DSX04	-	-	DSX01	DSX00
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W

Bit	Name	description
7	DSX07	TC0 Trigger source selection control enable 7 Bit. When set DSX07 Bit is"1"Time,TC1 Overflow is used to turn off the output comparison signal waveform OC0A/OC0B The trigger source of is enabled. whenDOC0A/DOC0B Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC0A/OC0B Waveform output. When set DSX07 Bit is"0"Time,TC1 Overflow is used to turn off the output comparison signal waveform OC0A/OC0B The trigger source of is disabled.
6	DSX06	TC0 Trigger source selection control enable 6 Bit. When setDSX06 Bit is"1"Time,TC2 Overflow is used to turn off the output comparison signal waveform OC0A/OC0B The trigger source of is enabled. when DOC0A/DOC0B Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC0A/OC0B Waveform output. When set DSX06 Bit is"0"Time,TC2 Overflow is used to turn off the output comparison signal waveform OC0A/OC0B The trigger source of is disabled.
5	DSX05	TC0 Trigger source selection control enable 5 Bit. When setDSX05 Bit is"1"When the pin level changes 0 As a comparison signal waveform for turning off the output OC0A/OC0B The trigger source of is enabled. whenDOC0A/DOC0B Bit is"1" When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC0A/OC0B Waveform output. When set DSX05 Bit is"0"When the pin level changes 0 As a comparison signal waveform for turning off the output OC0A/OC0B The trigger source of is disabled.
4	DSX04	TC0 Trigger source selection control enable 4 Bit. When setDSX04 Bit is"1"External interrupt 0 As a comparison signal waveform for turning off the output OC0A/OC0B The trigger source of is enabled. whenDOC0A/DOC0B Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will automatically turn off OC0A/OC0B Waveform output. When set DSX04 Bit is"0"External interrupt 0 As a comparison signal waveform for turning off the output OC0A/OC0B The trigger source of is disabled.
3:2	-	Keep unused
1	DSX01	TC0 Trigger source selection control enable 1 Bit. When setDSX01 Bit is"1"When, analog comparator 1 As a comparison signal waveform for turning off the output OC0A/OC0B The trigger source of is enabled. whenDOC0A/DOC0B Bit is"1" When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC0A/OC0B Waveform output.

		When set DSX01 Bit is"0"When, analog comparator 1 As a comparison letter for turning off the output Waveform OC0A/OC0B The trigger source of is disabled.
0		TC0 Trigger source selection control enable 0 Bit. When set DSX00 Bit is"1"When, analog comparator 0 As a comparison letter for turning off the output Waveform OC0A/OC0B The trigger source of is enabled. whenDOC0A/DOC0B Bit is"1" DSX00 When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC0A/OC0B Waveform output. When set DSX00 Bit is"0"When, analog comparator 0 As a comparison signal waveform for turning off the output OC0A/OC0B The trigger source of is disabled.

The following table shows the selection and control of the trigger source of the waveform output.

shut down OC0A/OC0B Trigger source selection control of waveform output

DOC0x	DSX0n=1	Trigger source	description
0	-	-	DOC0x Bit is"0", The trigger source is turned off and the waveform output function is disabled
1	0	Analog comparator 0	ACIF0 The rising edge will turn off OC0x Wave output
1	1	Analog comparator 1	ACIF1 The rising edge will turn off OC0x Wave output
1	4	External Interrupt 0	INTF0 The rising edge will turn off OC0x Wave output
1	5	Pin level change 0	PCIF0 The rising edge will turn off OC0x Wave output
1	6	TC2 overflow	TOV2 The rising edge will turn off OC0x Wave output
1	7	TC1 overflow	TOV1 The rising edge will turn off OC0x Wave output

note:

1)DSX0n=1 Means DSX0 Register n Bit is 1 When, each register bit can be set at the same time.

TC0 Count value register- TCNT0

TCNT0 –*TC0 Count value register*								
address: 0x46					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	TCNT07	TCNT06	TCNT05	TCNT04	TCNT03	TCNT02	TCNT01	TCNT00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	TCNT0	TC0 Count value register. by TCNT0 The register can directly 8 Read and write access for the count value.CPU Correct TCNT0 The write operation of the register will prevent the comparison match from occurring in the next timer clock cycle, even if the timer has stopped. This allows initialization TCNT0 The value of the register is the same as OCR0 The values of are consistent without causing interruption. If writeTCNT0 Equal to or bypassed OCR0 Value, the comparison match will be lost, resulting in incorrect waveforms. The timer stops counting when the clock source is not selected, but CPU Still accessible TCNT0.CPU The write counter has a higher priority than clearing or adding and subtracting operations.

TC0 Output compare register A- OCR0A

OCR0A – *TC0 Output compare register A*								
address: 0x47					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR0A7	OCR0A6	OCR0A5	OCR0A4	OCR0A3	OCR0A2	OCR0A1	OCR0A0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	OCR0A	TC0 Output compare register. OCR0A Contains a 8 Bit data, uninterruptedly with the counter value TCNT0 Compare. Compare match can be used to generate output compare interrupt, or used toOC0A A waveform is generated on the pin. When using PWMMode,OCR0A The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can update OCR0A The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR0A Buffer register, double buffer function is prohibited Can time CPU Visiting is OCR0A itself.

TC0 Output compare register B- OCR0B

***OCR0B* – TC0 Output compare register B**								
address: 0x48					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OCR0B7	OCR0B6	OCR0B5	OCR0B4	OCR0B3	OCR0B2	OCR0B1	OCR0B0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial	0	0	0	0	0	0	0	0

Bit	Name	description
7:0	OCR0B	TC0 Output comparison B register. OCR0B Contains a 8 Bit data, uninterruptedly with the counter value TCNT0 Compare. Compare match can be used to generate output compare interrupt, or used toOC0B A waveform is generated on the pin. When using PWMMode,OCR0B The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR0B The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWM Pulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR0B Buffer register, when double buffering function is disabled CPU Visiting is OCR0B itself.

TC0 Interrupt mask register- TIMSK0

***TIMSK0* – TC0 Interrupt mask register**								
address: 0x6E					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0
R/W	-	-	-	-	-	R/WR/W		R/W

Bit	Name	description
7:3		Reserved.
2	OCIE0B	TC0 Output comparison B Match interrupt enable bit. when OCIE0B Bit is"1", And the global interrupt is set,TC0 Output comparison B Match interrupt Enable. When a comparison match occurs, that isTIFR0 in OCF0B When the bit is set, interrupt produce. when OCIE0B Bit is"0"Time,TC0 Output comparison B The match interrupt is disabled.
1	OCIE0A	TC0 Output comparison A Match interrupt enable bit. whenOCIE0A Bit is"1", And the global interrupt is set,TC0 Output comparison A Match interrupt Enable. When a comparison match occurs, that isTIFR0 in OCF0A When the bit is set, interrupt produce. when OCIE0A Bit is"0"Time,TC0 Output comparison A The match interrupt is disabled.
0	TOIE0	TC0 Overflow interrupt enable bit. when TOIE0 Bit is"1", And the global interrupt is set,TC0 The overflow interrupt is enabled. whenTC0 Overflow occurs, i.e. TIFR middle TOV0 When the bit is set, an interrupt is generated. when TOIE0 Bit is"0"Time,TC0 Overflow interrupts are disabled.

TC0 Interrupt flag register- TIFR0

***TIFR0* – TC0 Interrupt flag register**								
address: 0x35					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OC0A	OC0B	-	-	-	OCF0B	OCF0A	TOV0
R/W	R/O	R/O	-	-	-	R/W	R/W	R/W

Bit	Name	description
7	OC0A	Output comparison waveform signal OC0A. Output comparison waveform signal OC0A, The software is readable but not writable. Software can be disabled OC0A Signal output to its corresponding IO Before the pin is on, you can read it first OC0A Bit value to obtain the polarity of the comparison waveform signal to be output, and can be configured COM0A Position and set FOC0A Bit to change its polarity to avoid OC0A Signal output to its corresponding IO The extra interference pulse is generated after the pin. Output comparison waveform signal
6	OC0B	OC0B. Output comparison waveform signal OC0B, The software is readable but not writable. Software can be disabled OC0B Signal output to its corresponding IO Before the pin is on, you can read it first OC0B Bit

		Value to obtain the polarity of the comparison waveform signal to be output, and can be configured COM0B Position and set FOC0B Bit to change its polarity to avoid OC0B Signal output to its corresponding IO The extra interference pulse is generated after the pin. Keep
5:3		
2	OCF0B	TC0 Output comparison B Match flag bit. whenTCNT0 equal OCR0B When, the comparison unit gives a matching signal and sets the bit ratio More sign OCF0B. If the output is compared at this timeB Interrupt enable OCIE0B for"1"And all If the local interrupt flag is set, an output comparison will be generated B Interrupted. When this interrupt service routine is executedOCF0B Will be automatically cleared, or OCF0B Bit write"1"This bit can also be cleared.
1	OCF0A	TC0 Output comparison A Match flag bit. whenTCNT0 equal OCR0A When, the comparison unit gives a matching signal and sets the bit ratio More sign OCF0A. If the output is compared at this timeA Interrupt enable OCIE0A for"1"And all If the local interrupt flag is set, an output comparison will be generated A Interrupted. When this interrupt service routine is executedOCF0A Will be automatically cleared, or OCF0A Bit write"1"This bit can also be cleared.
0	TOV0	TC0 Overflow flag. When the counter overflows, the overflow flag is set TOV0. If overflow interrupt is enabled at this timeTOIE0 for"1"And the global interrupt flag is set, an overflow interrupt will be generated. When this interrupt service routine is executedTOV0 Will be automatically cleared, or TOV0 Bit write"1"also may Clear this bit.

DTR0-TC0 Dead time control register

DTR0 **– TC0 Dead time control register**								
address: 0x4F					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	DTR07	DTR06	DTR05	DTR04	DTR03	DTR02	DTR01	DTR00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
[7:4]	DTR0H	TC0 The high bit of the dead time register. when TCCR0B Register DTEN0 Bit is"1"Time,OC0A with OC0B Complementary composition Output, insert dead time control is enabled,OC0B When the dead zone is inserted on the channel Indirect DTR0H Decide, the length of time is DTR0H The time corresponding to a counting clock.
[3:0]	DTR0L	TC0 The low bit of the dead time register. when TCCR0B Register DTEN0 Bit is"1"Time,OC0A with OC0B Complementary composition Output, insert dead time control is enabled,OC0A When the dead zone is inserted on the channel Indirect DTR0L Decide, the length of time is DTR0H The time corresponding to a counting clock.

TCKCSR-TC Clock control and status register

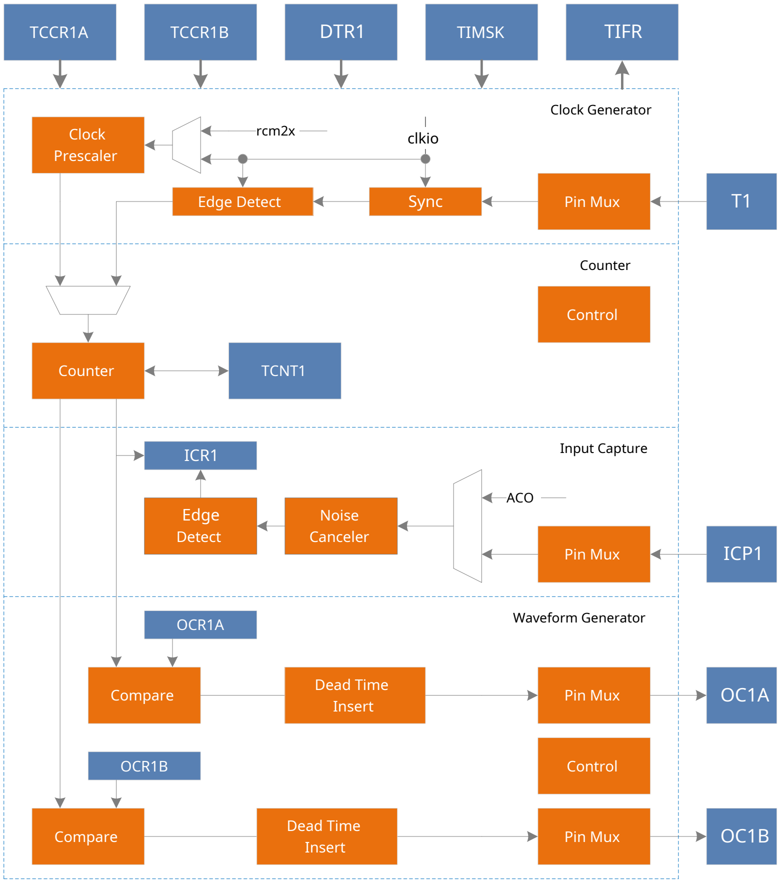
TCKSCR – TC Clock control and status register								
address: 0xEC				Defaults: 0x00				
Bit	7	6	5	4	3	2	1	0
Name	-	F2XEN	TC2XF1	TC2XF0	-	AFCKS	TC2XS1	TC2XS0
R/W	-	R/WRR			-	R/WR/WR/W		

Bit	Name	description
7	-	Keep
6	F2XEN	RC 32MMultiplier output enable control bit. When set F2XEN Bit is"1"Time,32M RC The frequency multiplier output of the oscillator is enabled, Output 64MHigh-speed clock. When set F2XEN Bit is"1"Time,32M RC The frequency multiplier output of the oscillator is prohibited and cannot be output 64MHigh-speed clock.
5	TC2XF1	TC High-speed clock mode flag 1. See timer counter 1 Register description.
4	TC2XF0	TC High-speed clock mode flag 0. When read TC2XF0 Bit is"1"Time, indicating that the timer counter 0 Working on high-speed clock Pattern for"0"Time, indicating that the timer counter 0 Work in system clock mode.
3:2	-	Reserved.
1	TC2XS1	TC High-speed clock mode selection control bit 1. See timer counter 1 Register description.
0	TC2XS0	TC High-speed clock mode selection control bit 0. When set TC2XS0 Bit is"1"When, select the timer counter 0 Working on high-speed clock mode. When set TC2XS0 Bit is"0"When, select the timer counter 0 Work in system clock mode.

Timer/counter 1 (TMR1)

- truly 16 Bit design, allowing 16 Bit PWM 2
- Independent output comparison unit
- Double-buffered output compare register
- 1 Input capture unit
- Input capture noise suppressor
- Automatically clear the counter and load automatically when compare matches
- Phase-corrected without interference pulses PWM
- Variable PWMcycle
- Frequency generator
- External event counter
- 4 Independent interrupt source
- Support dead time control PWM
- 6 Selectable trigger sources are automatically turned off PWMOutput
- High-speed and high-resolution (500KHZ@7BIT) PWM

Overview



TC1 Structure chart

TC1 Is a universal 16 Bit timer counter module, support PWMOutput, can accurately generate waveform.TC1 contain 1 A 16 Bit counter, waveform generation mode control unit,2 Independent output comparison unit and 1 Input capture unit. Simultaneously,TC1 Can be combined with TC0 Share 10 Bit prescaler, can also be used independently 10 Bit prescaler. Prescaler to system clockclkio Or high-speed clock rcm2x(internal 32M RC Oscillator output clock rc32m of 2 Frequency multiplication) divide the frequency to generate the count clock Clkt1. The waveform generation mode control unit controls the working mode of the counter and the generation of the comparison output waveform. According to different working modes, the counter counts each clockClkt1 Realize the operation of clearing, adding one or subtracting one.Clkt1 It can be generated by an internal clock source or an external clock source. When the count value of the counterTCNT1 Reach the maximum value (equal to the maximum value 0xFFFF Or fixed value or output compare register OCR1A Or input capture register ICR1,defined as TOP, The maximum value is defined as MAX To show the difference), the counter will be cleared or decremented by one. When the count value of the counterTCNT1 Reached the minimum value (equal to 0x0000,defined as BOTTOM), the counter will increment by one. When the count value of the counterTCNT1 Arrivals OCR1A or OCR1B When a comparison match occurs, it will be cleared or set to output a comparison signal OC1A or OC1BTo produce PWMWaveform. When the insertion of dead time is enabled, the set dead time (DTR1 The count clock corresponding to the register) will be inserted into the generated PWMIn the waveform. When the input capture function is turned on, the counter is triggered to start or stop counting,ICR1 The register will record the count value in the trigger period of the capture signal. Software can be cleared byCOM1A/ COM1B Bit is zero to turn off OC1A/OC1B Waveform output, or set the corresponding trigger source, the hardware will automatically clear when a trigger event occurs COM1A/COM1B Bit to close OC1A/OC1B Waveform output.

The count clock can be generated by an internal or external clock source. The selection of clock source and frequency division are determined by TCCR1B Register CS1 Bit to control, see detailed description TC0 with TC1 Prescaler chapter.

The length of the counter is16Bit, Support two-way counting. The waveform generation mode, that is, the working mode of the counter, is determined byTCCR1A with TCCR1B RegisterWGM1 Bit to control. According to different working modes, the counter counts each clockClkt1 Realize the operation of clearing, adding one or subtracting one. When the count overflows, it is located atTIFR1 Register overflow flag TOV1 The bit will be set. Can be generated when the interrupt is enabledTC1 Count overflow interrupt.

Output comparison unit pair count value TCNT1 And output compare register OCR1A with OCR1B To compare the value of when TCNT1 equal OCR1A or OCR1B Is called a comparison match, and is located at TIFR1 Register output compare flag OCF1A or OCF1B The bit will be set. Can be generated when the interrupt is enabledTC1 Output compare match interrupt. It should be noted that inPWMIn working mode,OCR1A with OCR1B The register is a double-buffered register. In normal mode and CTC In mode, the double buffering function is invalid. When the count reaches the maximum or minimum value, the value in the buffer register is synchronously updated to the compare registerOCR1A with OCR1B Go in. See the description of the working mode chapter for details.

The waveform generator uses comparison matching and counting overflow to generate output comparison waveform signals according to the waveform generation mode control and the comparison output mode control. OC1A with OC1B. For specific generation methods, see the description in the chapters on working modes and registers. To compare the output waveform signalOC1A with OC1B When outputting to the corresponding pin, the data direction register of the pin must also be set as output.

Operating mode

Timer counter 1 There are six different working modes, including normal mode (Normal), cleared on comparison match (CTC) Mode, fast pulse width modulation (FPWM) Mode, phase correction pulse width modulation (PCPWM) Mode, phase frequency correction pulse width modulation (PFCPWM) Mode, and input capture (ICP)mode. Control bit by waveform generation mode WGM1[3:0]To choose. The six modes are described in detail below. Since there are two independent output comparison units, use "A"with"B"To indicate, in lowercase"X"To represent these two output comparison unit channels.

Normal mode

Normal mode is the simplest working mode of the timer counter. At this time, the waveform generation mode control bitWGM1[3:0]=0, The maximum value of the count TOP for MAX(0xFFFF). In this mode, the counting method is incremented by each counting clock. When the counter reachesTOP Back after overflow BOTTOM Start accumulation again. Count valueTCNT1 The timer counter overflow flag is set in the same counting clock that becomes zero TOV1. In this modeTOV1 The sign is like the first 17 The counting bit is only set but not cleared. Overflow interrupt service routine will be automatically clearedTOV1 Mark, the software can use it to improve the resolution of the timer counter. There is no special situation to consider in the normal mode, and a new count value can be written at any time.

Set upOC1xThe output comparison signal can only be obtained when the data direction register of the pin is outputOC1x的waveform. whenCOM1x=1

When a comparison match occurs, it will be flipped OC1x Signal, the frequency of the waveform in this case can be calculated with the following formula:

$f_{oc1xnormal} = f_{sys/(}2*N*65536)$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

The output compare unit can be used to generate interrupts, but it is not recommended to use interrupts in normal mode, which will take up too much CPU time.

CTC mode

Set up WGM1[3:0]=4 or 12 Time counter 1 enter CTC mode. whenWGM1[3]=0 , The maximum value of the count TOP for OCR1A, whenWGM1[3]=1 , The maximum value of the count TOP for ICR1. The following isWGM1[3:0]=4 As an example to describeCTCIn this mode, the counting method is incremented by one for each counting clock. When the counter valueTCNT1 equal TOP The hour counter is cleared. This mode allows users to easily control the frequency of the compare match output, and also simplifies the operation of counting external events.

When the counter arrives TOP When, output compare match flag OCF1 Is set, an interrupt will be generated when the corresponding interrupt enable bit is set. Can be updated in the interrupt service routineOCR1A register. In this modeOCR1A Double buffering is not used. Be careful when updating the maximum value to close to the minimum value when the counter is working with no prescaler or a very low prescaler. If writeOCR1A Is less than the value at the time TCNT1 Value, the counter will lose a comparison match. Before the next comparison match occurs, the counter has to count toMAX, And then from BOTTOMStart counting to OCR1A. Like normal mode, the count value returns to0x0 Set in the count clock TOV1 Sign.

Set up OC1x The output comparison signal can only be obtained when the data direction register of the pin is output OC1x 的waveform. The frequency of the waveform can be calculated with the following formula:

$f_{oc1xctc} = f_{sys/(}2*N*(1+OCR1A))$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

It can be seen from the formula that when setting OCR1A for 0x0 And without prescaler, the maximum frequency that can be obtained is $f_{sys/}2$ The output waveform.

whenWGM1[3:0]=12 Time andWGM1[3:0]=4 Similar, just put OCR1A Related to ICR1 That's it.

fast PWMmode

Set upWGM1[3:0]=5,6,7,14 or 15 Time counter 1 Enter fast PWMMode, maximum count TOP Respectively 0xFF,0x1FF,0x3FF,ICR1 or OCR1A, Can be used to generate high frequency PWMWaveform. fast PWM Mode and others PWM The mode is different in that it is a one-way operation. Counter fromBOTTOM Accumulate to TOP Back again BOTTOM Recount. When the count valueTCNT1 Arrivals TOP or BOTTOMWhen, output the comparison signal OC1x Will be set or cleared, depending on the comparison output mode COM1 See the register description for details. Due to the one-way operation, fastPWM The operating frequency of the mode is phase correction with bidirectional operation PWM Twice the pattern. High frequency characteristics make fast

PWM Mode is suitable for power regulation, rectification and DAC application. High-frequency signals can reduce the size of external components (inductance, capacitance, etc.), thereby reducing system costs.

When the count value reaches TOP Time, timer counter overflow flag TOV1 Will be set, and the value of the compare buffer will be updated to the compare value. If the interrupt is enabled, it can be updated in the interrupt service routineOCR1A register.

Set up OC1x The output comparison signal can only be obtained when the data direction register of the pin is output OC1x 的waveform. The frequency of the waveform can be calculated with the following formula:

$f_{oc1xfpwm} = f_{sys/(}N*(1+TOP))$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

when TCNT1 with OCR1x When a comparison match occurs, the waveform generator is set (cleared)OC1x Signal when TCNT1 When cleared, the waveform generator is cleared (set)OC1x Signal to generate PWMwave. thusOCR1x Extreme values will produce special PWM Waveform. whenOCR1x Set as 0x00 When the output PWM For every (1+TOP)There is a narrow spike in each count clock. whenOCR1x Set as TOP When, the output waveform is a continuous high level or low level. If you useOCR1A As TOP And set COM1A=1, Output comparison signal OC1A Will produce a duty cycle of 50%of PWM wave.

Phase correction PWMmode

When setWGM0[3:0]=1,2,3,10 or 11 Time counter 1 Enter phase correction PWMMode, the maximum value of the count TOP Respectively 0xFF,0x1FF,0x3FF,ICR1 or OCR1A. The counter adopts two-way operation, byBOTTOM Increment to TOP, And then decrease to BOTTOM, Repeat this operation again. Count arrivalTOP with BOTTOM Change the counting direction every time, the count value is in TOP or BOTTOMThere is only one counting clock on each. In the process of increment or decrement, the count value

TCNT1 versus OCR1x When matching, output comparison signal OC1x Will be cleared or set, depending on the comparison output mode COM1 setting. Compared with unidirectional operation, the maximum frequency that can be obtained by bidirectional operation is smaller, but its excellent symmetry is more suitable for motor control.

Phase correction PWM Mode, when the count reaches BOTTOM Set when TOV1 Sign when the count arrives TOP Update the value of the comparison buffer to the comparison value at the time. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routineOCR1x Memory.

Set up OC1x The output comparison signal can be obtained only when the data direction register of the pin is output OC1x Waveform. The frequency of the waveform can be calculated with the following formula:

$f_{oc1xcpcpwm} = f_{sys/(}N*TOP*2)$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

During the count-up process, when TCNT1 versus OCR1x When there is a match, the waveform generator is cleared (set)OC1x signal. in

During the countdown process, when TCNT1 versus OCR1x When a match, the waveform generator is set (cleared)OC1x signal. thus OCR1x The extreme value of will produce a special PWMwave. whenOCR1x Set as TOP or BOTTOMTime,OC1x The signal output will always remain low or high. If you useOCR1A As TOP And set COM1A=1, Output comparison signal OC1A Will produce a duty cycle of 50%of PWMwave.

In order to ensure the output PWMWave in BOTTOM Symmetry on both sides, when there is no comparison match, there are two cases will be flipped OC1x signal. The first situation is whenOCR1x The value is determined by TOP When changing to other data. whenOCR1x for TOP, The count reaches TOP Time,OC1x The output of is the same as the result of the comparison match when counting in descending order, that is, keep OC1x constant. At this time, the comparison value will be updated to the new oneOCR1x Value (not TOP),OC1x The value of will be maintained until a comparison match occurs during ascending counting and it is reversed. at this timeOC1x The signal is not symmetrical about the minimum value, so it needs to be TCNT1 Flip when reaching maximum OC1x Signal, that is, there is no rollover when a comparison match occurs OC1x The first case of the signal. The second situation is whenTCNT1 From OCR1x When a high value starts to count, a comparison match will be lost, which will cause asymmetry. Also need to flipOC1x The signal to achieve symmetry on both sides of the minimum.

Phase frequency correction PWMmode

When setWGM0[3:0]=8 or 9 Time counter 1 Enter phase frequency correction PWMMode, the maximum value of the count TOP Respectively ICR1 or OCR1A. The counter adopts two-way operation, byBOTTOM Increment to TOP, And then decrease to BOTTOM, Repeat this operation again. Count arrivalTOP with BOTTOMChange the counting direction every time, the count value is in TOP or BOTTOM There is only one counting clock on each. In the process of increment or decrement, the count valueTCNT1 versus OCR1x When matching, output comparison signal OC1x Will be cleared or set, depending on the comparison output mode COM1 setting. Compared with unidirectional operation, the maximum frequency that can be obtained by bidirectional operation is smaller, but its excellent symmetry is more suitable for motor control.

Phase frequency correction PWM Mode, when the count reaches BOTTOM Set when TOV1 Mark and update the value of the comparison buffer to the comparison value. The time to update the comparison value is the phase frequency correction PWM Mode and phase correction PWM The biggest difference in the model. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routineOCR1x Memory. whenCPU change TOP Value is ORC1A or ICR1 Value, you must ensure that the new TOP The value is not less than the one already in use TOP Value, otherwise the comparison match will not happen again.

Set up OC1x The output comparison signal can be obtained only when the data direction register of the pin is output OC1x Waveform. The frequency of the waveform can be calculated with the following formula:

$$f_{oc1xcpfcpwm} = f_{sys/(}N*TOP*2)$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

During the count-up process, when TCNT1 versus OCR1x When there is a match, the waveform generator is cleared (set)OC1x signal. In the process of counting down, whenTCNT1 versus OCR1x When a match, the waveform generator is set (cleared)OC1x signal. thus OCR1x The extreme value of will produce a special PWMwave. whenOCR1x Set as TOP or BOTTOMTime,OC1x The signal output will always remain low or high. If you useOCR1A As TOP And set COM1A=1, Output comparison signal OC1A Will produce a duty cycle of 50%of PWMwave.

because OCR1x The register is in BOTTOM Updated all the time, so TOP The count length of the ascending order and the descending order is the same on both sides of the value, and a symmetrical waveform with the correct frequency and phase is generated.

When using fixed TOP Value, it’s best to use ICR1 Register as TOP Value, i.e. settingWGM1[3:0]=8,at this time OCR1A The register only needs to be used to generate PWM Output. If you want to produce frequency changesPWM Wave, must be changed TOP value, OCR1A The double-buffering characteristics of the "will be more suitable for this application.

Input capture mode

Input capture is used to capture external events and give them a time stamp to indicate the moment when this event occurs. It can be performed in the previous counting mode, but it is necessary to remove the use ICR1 Value as count TOP Value waveform generation mode.

The trigger signal of an external event is determined by the pin ICP1 The input can also be realized by an analog comparator unit. When the pinICP1 The logic level on the change, or the output of the analog comparator ACO The level changes, and this level change is captured by the input capture unit, the input capture is triggered, at this time 16 Bit count TCNT1 Data is copied to the input capture register ICR1, And enter the capture flag at the same time ICF1 Set if ICIE1 Bit is"1", The input capture flag will generate an input capture interrupt.

By setting analog comparison control and status registerACSRAnalog compare input capture control bitACICTo select the input capture trigger source ICP1 or ACO. It should be noted that changing the trigger source may cause an input capture, so after changing the trigger source, you mustICF1 Perform a clear operation to avoid false results.

The input capture signal is sent to the edge detector after an optional noise suppressor, and the control bit is selected according to the input capture ICES1 To see if the detected edge meets the trigger condition. The noise suppressor is a simple digital filter that performs4 Sub-sampling, only when 4 When the sub-sampled values are equal, its output will be sent to the edge detector. Noise suppressor by TCCR1B Register ICNC1 The bit controls its enabling or disabling.

When using the input capture function, when ICF1 After being set, it should be read as early as possible ICR1 The value of the register, because after the next capture event occurs ICR1 The value of will be updated. It is recommended to enable the input capture interrupt. In any input capture mode, it is not recommended to change the count during operationTOP value.

The time stamp captured by the input can be used to calculate the frequency, duty cycle, and other characteristics of the signal, as well as to create a log for trigger events. When measuring the duty cycle of an external signal, it is required to change the trigger edge after each capture, so readICR1 The signal edge of the trigger must be changed as soon as possible after the value.

PWMAutomatic shutdown and restart of output

When set TCCR1C Register DOC1x When the bit is high,PWMThe automatic shutdown function of the output will be enabled. When the trigger condition is met, the hardware will clear the corresponding COM1x Bit, will PWM output signal OC1x Disconnect from its output pin and switch to general purpose IO Output, achieve PWM The output is automatically turned off. At this time, the state of the output pin can beIO To control the output of the port.

PWMAfter the automatic shutdown of the output is enabled, the trigger conditions need to be set. TCCR1D Register DSX1n Bit to select the trigger source. Trigger sources include analog comparator interrupt, external interrupt, pin level change interrupt and timer overflow interrupt. For details, please refer toTCCR1D Register description. When one or some trigger sources are selected as the trigger condition, the hardware will clear these interrupt flag bits at the same time as they are set.COM1x Bit to close PWMOutput.

Close when a trigger event occurs PWMAfter the output, the timer module does not have the corresponding interrupt flag bit, and the software needs to know the trigger condition and trigger event by reading the interrupt flag bit of the trigger source.

when PWMWhen the output is automatically turned off and the output needs to be restarted again, the software only needs to be reset COM1x Bit to switch

OC1x The signal is output to the corresponding pin. It should be noted that after the automatic shutdown occurs, the timer does not stop working.

OC1x The status of the signal is also constantly being updated. Software can set after timer overflow or compare match COM1x Bit to output OC1x Signal so that you can get a clear PWMOutput status.

Dead time control

Set up DTEN1 Bit is"1"When the function of inserting the dead time is enabled,OC1A with OC1B The output waveform will be in B The set dead time is inserted on the basis of the waveform generated by the channel comparison output. The length of the time is DTR1 The time value corresponding to the count clock number of the register. As shown below,OC1A with OC1B The dead time insertion is based on the channel B The comparison output waveform is the reference. whenCOM1A with COM1B Same as"2"or"3"Time,OC1A The polarity of the waveform and OC1B The polarity of the waveform is the same, when COM1A with COM1B Respectively"2"or"3"Time,OC1A The waveform and OC1B The polarity of the waveform is opposite.

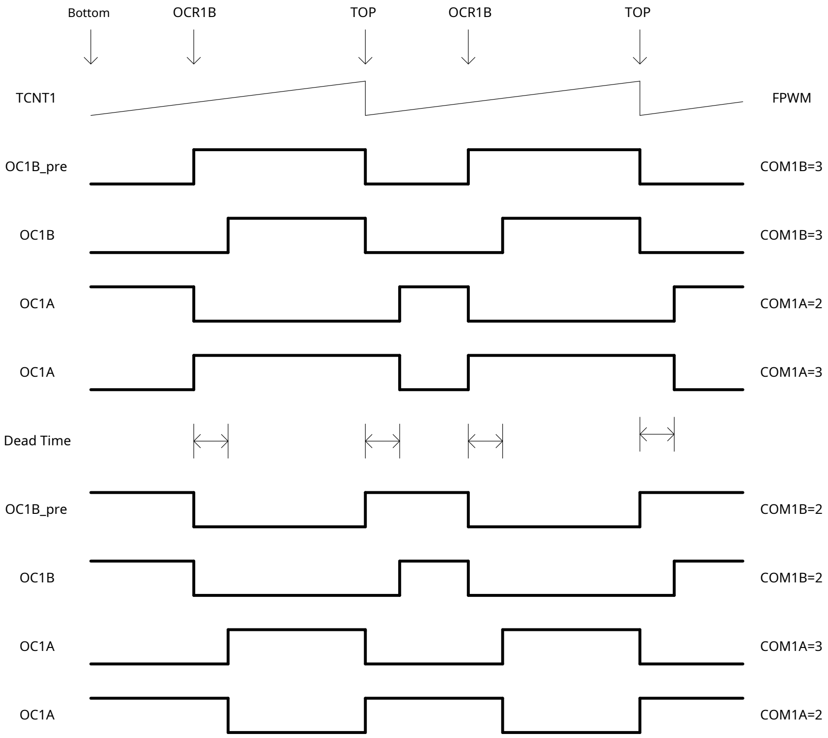


Figure 3 FPWMMode TC1 Dead time control

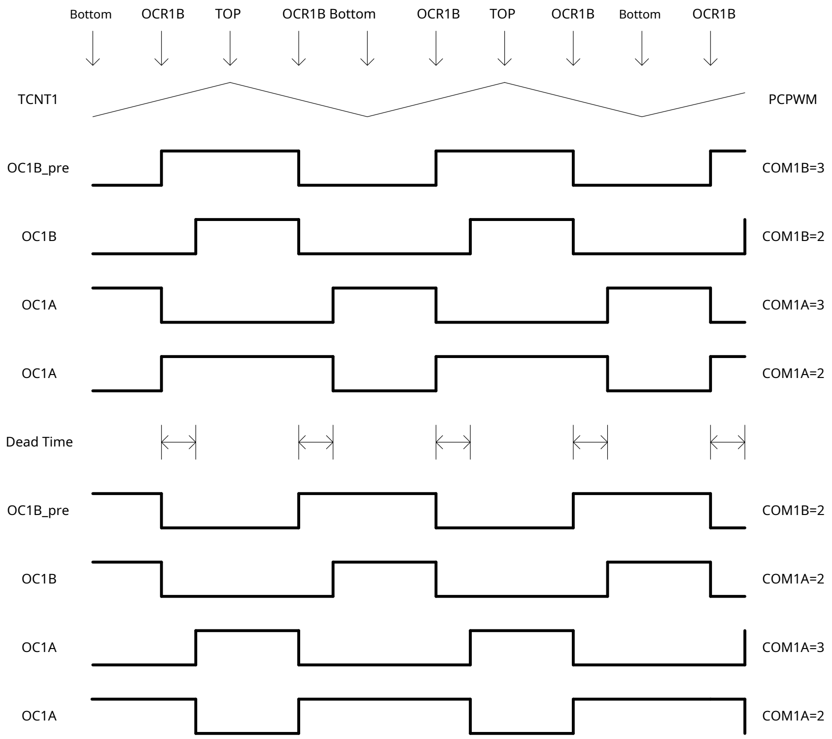


Figure 4 PCPWMMode TC1 Dead time control

Set up DTEN1 Bit is"0"Time, the function of inserting the dead time is disabled,OC1A with OC1B The output waveform of is the waveform generated by the respective comparison output.

High-speed counting mode

In high-speed clock mode, a higher frequency clock is used as the clock source for counting to generate higher speed and higher resolution PWM Waveform. This high-frequency clock is passed to the internal32M RC Oscillator's output clock rc32mget on 2 Multiplied to produce. Therefore, before entering the high frequency mode, you need to enable the internal32M RC Oscillator's frequency multiplication function, that is, set TCKCSRRegisterF2XENBit and wait for a certain period of time until the output of the multiplier clock signal is stable. Then, you can setTCKCSR of TC2XS1 Bit to make the timer counter enter the high-speed clock mode.

In this mode, the system clock and high-speed clock are asynchronous, and some registers (see TC1 The register list) works in the high-speed clock domain. Therefore, the configuration and reading of this type of register are also asynchronous, and attention should be paid to the operation.

There are no special requirements for non-continuous read and write operations on registers in the high-speed clock domain. When continuous read and write operations are performed, you need to wait for a system clock. Follow the steps below:

5) Write register A；

6) Waiting for a system clock (NOP Or register under the operating system clock);

7) Read or write register A or B.

8) Waiting for a system clock (NOP Or registers under the operating system clock).

When the register under the high-speed clock domain is read, the width is 8 Bit registers can be read directly, and read 16 The value of the bit register (OCR1A,OCR1B, ICR1, TCNT1), first read the value of the lower register and wait for a system clock

Then, read the value of the high register, and read TCNT1 When the counter is still counting,TCNT1 The value of will change with the high-speed clock, and the counter can be paused (set CS1 Zero) and then read TCNT1 Value.

Read OCR1A,OCR1B with ICR1 When, follow the steps below:

1) Read OCR1AL/OCR1BL/ICR1L；

2) Waiting for a system clock (NOP);

3) Read OCR1AH/OCR1BH/ICR1H.

Read TCNT1 When, follow the steps below:

1) Set CS1 Zero

2) Waiting for a system clock (NOP);

3) Read TCNT1L The value of

4) Waiting for a system clock (NOP);

read TCNT1H Value.

Register definition

TC1 Register list

register	address	Defaults	description
TCCR1A*	0x80	0x00	TC1 Control register A
TCCR1B*	0x81	0x00	TC1 Control register B
TCCR1C*	0x82	0x00	TC1 Control register C
DSX1	0x83	0x00	TC1 Trigger source control register
TCNT1L*	0x84	0x00	TC1 Count value register low byte
TCNT1H*	0x85	0x00	TC1 Count value register high byte
ICR1L*	0x86	0x00	TC1 Input capture register low byte
ICR1H*	0x87	0x00	TC1 Input capture register high byte
OCR1AL*	0x88	0x00	TC1 Output compare register A Low byte
OCR1AH*	0x89	0x00	TC1 Output compare register A High byte
OCR1BL*	0x8A	0x00	TC1 Output compare register B Low byte
OCR1BH*	0x8B	0x00	TC1 Output compare register B High byte
DTR1*	0x8C	0x00	TC1 Dead time control register
TIMSK1	0x6F	0x00	Timer counter interrupt mask register
TIFR1	0x36	0x00	Timer counter interrupt flag register
TCKCSR1	0xEC	0x00	TC1 Clock control status register

【note】

band"*"The registers work in the system clock and high-speed clock domains, without"*"The registers only work in the system clock domain.

TCCR1A –TC1 Control register A

TCCR1A -TC1 Control register A								
address: 0x80					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10
R/W	R/W	R/W	R/W	R/W	-	-	R/WR/W	

Bit	Name	description
7	COM1A1	Compare match output A Mode control high. COM1A1 with COM1A0 composition COM1A[1:0]To control the output comparison waveform OC1A. in caseCOM1A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC1A Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM1A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
6	COM1A0	Compare match output A Mode control low. COM1A1 with COM1A0 composition COM1A[1:0]To control the output comparison waveform OC1A. in caseCOM1A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC1A Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM1A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
5	COM1B1	Compare match output B Mode control high. COM1B1 with COM1B0 composition COM1B[1:0]To control the output comparison waveform OC1B. in caseCOM1B of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC1B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM1B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
4	COM1B0	Compare match output B Mode control low. COM1B1 with COM1B0 composition COM1B[1:0]To control the output comparison waveform OC1B. in caseCOM1B of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC1B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM1B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
3:2	-	Keep unchanged
1	WGM11	The waveform generation mode controls the next low bit. WGM11 with WGM13, WGM12, WGM10 Form the waveform generation mode control togetherWGM1[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
0	WGM10	The waveform generation mode controls the lowest bit. WGM10 with WGM13, WGM12, WGM11 Form the waveform generation mode control togetherWGM1[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.

The following table is not PWMMode (i.e. normal mode and CTC Mode), the comparison output mode controls the output comparison waveform.

COM1x[1:0]	description
0	OC1x Disconnect, general IO Flip when the port
1	operation compares match OC1x signal
2	Clear on comparison match OC1x signal
3	Set on compare match OC1x signal

The following table is fast PWMThe comparison output mode in the mode controls the output comparison waveform.

COM1x[1:0]	description
0	OC1x Disconnect, general IO Mouth operation
1	WGM1 for 15 Time: Flip when compare matches OC1A signal, OC1B disconnect WGM1 For other values:OC1x Disconnect, general IO Cleared when port operation compare match
2	OC1x Signal, set when the maximum value matches OC1x Set when the signal compare matches OC1x
3	Signal, cleared when the maximum value matches OC1x signal

The following table shows the control of the output comparison waveform in the comparison output mode in the phase correction mode.

COM1x[1:0]	description
0	OC1x Disconnect, general IO Mouth operation
1	WGM1 for 9 or 11 Time: Flip when compare matches OC1A signal, OC1B disconnect WGM1 For other values:OC1x Disconnect, general IO Compare match cleared under ascending
2	count of port operation OC1x Signal, compare match set in descending count OC1x signal
3	Compare match set in ascending count OC1x Signal, compare match cleared under descending count OC1x signal

TCCR1B –TC1 Control register B

TCCR1B -TC1 Control register B								
address: 0x81					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	ICNC1	ICES1	-	WGM13	WGM12	CS12	CS11	CS10
R/W	R/W	R/W	-	R/WR/W		R/W	R/W	R/W

Bit	Name	description
7	ICNC1	Input capture noise suppressor enable control bit. When set ICNC1 Bit is"1"When the input is enabled to capture the noise suppressor, the external pin ICP1 Input is filtered, continuous 4 The input signal is valid only when the sampling values are equal. This function delays the input capture 4 Clock cycles. When set ICNC1 Bit is"0"When the input capture noise suppressor is prohibited, the external pin ICP1 The input is directly valid.
6	ICES1	Input capture trigger edge selection control bit. When set ICES1 Bit is"1"Select the rising edge of the level to trigger the input capture; when set ICES1 Bit is"0"When, select the falling edge of the level to trigger the input capture. When an event is captured, the value of the counter is copied toICR1 Register, and set the input capture flag at the same time ICF1. If the interrupt is enabled, an input capture interrupt is generated. Reserved.
5	-	

4	WGM13	The waveform generation mode controls the high bit. WGM13 with WGM12, WGM11, WGM10 Form the waveform generation mode control together WGM1[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
3	WGM12	The waveform generation mode controls the second highest bit. WGM12 with WGM13, WGM11, WGM10 Form the waveform generation mode control together WGM1[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
2	CS12	The clock selection controls the high bit. Used to select the timer counter1 Clock source.
1	CS11	Clock selection control center position. Used to select the timer counter1 Clock source.
0	CS10	Clock selection control low bit. Used to select the timer counter 1 Clock source.

CS1[2:0]	description
0	No clock source, stop counting
1	clk$_{sys}$
2	clk$_{sys}$/8, From the prescaler
3	clk$_{sys}$/64, From the prescaler
4	clk$_{sys}$/256, From the prescaler
5	clk$_{sys}$/1024, From the external clock of the
6	prescaler T1 Pin, falling edge trigger
7	External clock T1 Pin, rising edge trigger

The following table shows the waveform generation mode control.

WGM1[3:0]	Operating mode	TOP value	Update OCR0	Always set TOV0 time
0	Normal	0xFFFF	immediately	MAX
1	8 Bit PCPWM	0x00FF	TOP BOTTOM	
2	9 Bit PCPWM	0x01FF	TOP BOTTOM	
3	10 Bit PCPWM	0x03FF	TOP BOTTOM	
4	CTC	OCR1A	immediately	MAX
5	8 Bit FPWM	0x00FF	BOTTOM TOP	
6	9 Bit FPWM	0x01FF	BOTTOM TOP	
7	10 Bit FPWM	0x03FF	BOTTOM TOP	
8	PFCPWM	ICR1	BOTTOM BOTTOM	
9	PFCPWM	OCR1A	BOTTOM BOTTOM	
10	PCPWM	ICR1	TOP BOTTOM	
11	PCPWM	OCR1A	TOP BOTTOM	
12	CTC	ICR1	immediately	MAX
13	Keep	-	-	-
14	FPWM	ICR1	TOP TOP	
15	FPWM	OCR1A	TOP TOP	

TCCR1C –TC1 Control register C

TCCR1C -TC1 Control register C								
address: 0x82					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	FOC1A	FOC1B	DOC1B	DOC1A	DTEN1	-	-	-
R/W	W	W	R/W	R/W	R/W	-	-	-

Bit	Name	description
7	FOC1A	Force output comparison A. Work in non PWMMode, you can compulsorily output the comparison bit FOC1A write"1" Way to generate comparison matches. Force compare match will not be setOCF1A Flag, it will not reload or clear the timer, but the output pin OC1A Will be followed COM1A The settings are updated accordingly, just as if a comparison match actually occurred. Work on PWMMode, write TCCR1A It must be cleared when registering. Read FOC1A The return value of is always zero.
6	FOC1B	Force output comparison B. Work in non PWMMode, you can compulsorily output the comparison bit FOC1B write"1" Way to generate comparison matches. Force compare match will not be setOCF1B Flag, it will not reload or clear the timer, but the output pin OC1B Will be followed COM1B The settings are updated accordingly, just as if a comparison match actually occurred. Work onPWMMode, write TCCR1A It must be cleared when registering. ReadFOC1B The return value of is always zero.
5	DOC1B	TC1 Turn off the output compare enable control high bit. When set DOC1B Bit is"1"When the trigger source turns off and outputs the comparison signal OC1B Is enabled. When a trigger event occurs, the hardware is automatically clearedCOM1B Bit, off OC1B Waveform output. The software can be set byCOM1B Reopen PWMOutput. When setDOC1B Bit is"0" When the trigger source turns off and outputs the comparison signal OC1B banned.
4	DOC1A	TC1 Turn off the output compare enable control low bit. When set DOC1A Bit is"1"When the trigger source turns off and outputs the comparison signal OC1A Is enabled. When a trigger event occurs, the hardware is automatically clearedCOM1A Bit, off OC1A Waveform output. The software can be set byCOM1A Reopen PWMOutput. When setDOC1A Bit is"0" When the trigger source turns off and outputs the comparison signal OC1A banned.
3	DTEN1	TC1 Dead time enable control bit. When set DTEN1 Bit is"1"When the time, the dead time insertion is enabled.OC1A with OC1B All in B The dead time is inserted on the basis of the waveform generated by the channel comparison output, and the inserted dead time interval is determined by DTR1 The counting time corresponding to the register is determined.OC1A The polarity of the output waveform is determined by COM1A with COM1B The corresponding relationship is determined, see OC1A After inserting the dead time, the waveform polarity is shown in the table. When set DTEN1 Bit is"0"Time, the dead time insertion is prohibited,OC1A with OC1B The waveform of is the waveform generated by the respective comparison output.
2:0	-	Keep

The following table shows when the dead time is enabled OC1A The polarity control of the signal output waveform.

In dead time enable mode OC1A Polarity control of signal output waveform

DTEN1	COM1A[1:0]	COM1B[1:0]	description
0	-	-	OC1A The signal polarity is determined by OC1A Comparison output mode control
1	0	-	OC1A Disconnect, general IO Port operation
1	1	-	reservation
1	2	2	OC1A Signal and OC1B Same signal polarity
		3	OC1A Signal and OC1B Signal polarity is opposite
1	3	2	OC1A Signal and OC1B Signal polarity is opposite
		3	OC1A Signal and OC1B Same signal polarity

【note】:

OC1B The polarity of the signal output waveform is determined by OC1B The comparison output mode control is the same as the dead time mode without enabling.

TCCR1D –TC1 Control register D

TCCR1D -TC Control register D								
address: 0x83					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	DSX17	DSX16	DSX15	DSX14	-	-	DSX11	DSX10
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W

Bit	Name	description
7	DSX17	TC1 Trigger source selection control enable 7 Bit. When setDSX17 Bit is"1"Time,TC0 Overflow is used to turn off the output comparison signal waveform OC1A/OC1B The trigger source of is enabled. whenDOC1A/DOC1B Bit is"1" When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC1A/OC1B Waveform output. When set DSX17 Bit is"0"Time,TC0 Overflow is used to turn off the output comparison signal waveform OC1A/OC1B The trigger source of is disabled.
6	DSX16	TC1 Trigger source selection control enable 6 Bit. When setDSX16 Bit is"1"Time,TC2 Overflow is used to turn off the output comparison signal waveform OC1A/OC1B The trigger source of is enabled. whenDOC1A/DOC1B Bit is"1" When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC1A/OC1B Waveform output. When set DSX16 Bit is"0"Time,TC2 Overflow is used to turn off the output comparison signal waveform OC1A/OC1B The trigger source of is disabled.
5	DSX15	TC1 Trigger source selection control enable 5 Bit. When setDSX15 Bit is"1"When the pin level changes 1 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is enabled. whenDOC1A/DOC1B Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC1A/OC1B Waveform output. When set DSX15 Bit is"0"When the pin level changes 1 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is disabled.
4	DSX14	TC1 Trigger source selection control enable 4 Bit. When setDSX14 Bit is"1"External interrupt 1 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is enabled. whenDOC1A/DOC1B Bit is

		"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC1A/OC1B Waveform output. When set DSX14 Bit is"0"External interrupt 1 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is disabled.
3:2	-	Keep
1	DSX11	TC1 Trigger source selection control enable 1 Bit. When setDSX11 Bit is"1"When, analog comparator 1 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is enabled. whenDOC1A/DOC1B Bit is "1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC1A/OC1B Waveform output. When set DSX11 Bit is"0"When, analog comparator 1 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is disabled.
0	DSX10	TC1 Trigger source selection control enable 0 Bit. When setDSX10 Bit is"1"When, analog comparator 0 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is enabled. whenDOC1A/DOC1B Bit is "1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC1A/OC1B Waveform output. When set DSX10 Bit is"0"When, analog comparator 0 As a comparison signal waveform for turning off the output OC1A/OC1B The trigger source of is disabled.

The following table shows the selection and control of the trigger source of the waveform output.

shut down OC1A/OC1B Trigger source selection control of waveform output

DOC1x	DSX1n=1	Trigger source	description
0	-	-	DOC1x Bit is"0", The trigger source is turned off and the waveform output function is disabled
1	0	Analog comparator 0	ACIF0 The rising edge will turn off OC1x Wave output
1	1	Analog comparator 1	ACIF1 The rising edge will turn off OC1x Wave output
1	4	External Interrupt 1	INTF1 The rising edge will turn off OC1x Wave output
1	5	Pin level change 1	PCIF1 The rising edge will turn off OC1x Wave output
1	6	TC2 overflow	TOV2 The rising edge will turn off OC1x Wave output
1	7	TC0 overflow	TOV0 The rising edge will turn off OC1x Wave output

【note】:

DSX1n=1 Means DSX1 Register n Bit is 1 When, each register bit can be set at the same time.

TCNT1L –TC1 Count value register low byte

TCNT1L -TC1 Count value register low byte								
address: 0x84					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	TCNT1L7	TCNT1L6	TCNT1L5	TCNT1L4	TCNT1L3	TCNT1L2	TCNT1L1	TCNT1L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name description							
7:0	TCNT1	TC1 The low byte of the count value.						

		TCNT1H with TCNT1L Combined together to form TCNT1,by TCNT1 The register can directly 16 The bit count value is accessed for read and write. Read and write16 The bit register requires two operations. write16 Bit TCNT1 Should be written first TCNT1H. read 16 Bit TCNT1 Should be read first TCNT1L. CPU Correct TCNT1 The write operation of the register will prevent the comparison match from occurring in the next timer clock cycle, even if the timer has stopped. This allows initialization TCNT1 The value of the register is the same as OCR1x The values of are consistent without causing interruption. If writeTCNT1 Equal to or bypassed OCR1x When the value is set, the comparison match will be lost, resulting in incorrect waveforms. The timer stops counting when the clock source is not selected, but CPU Still accessible TCNT1. CPU The write counter has a higher priority than clearing or adding and subtracting operations.

TCNT1H –TC1 Count value register high byte

TCNT1H -TC1 Count value register high byte								
address: 0x85					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	TCNT1H7	TCNT1H6	TCNT1H5	TCNT1H4	TCNT1H3	TCNT1H2	TCNT1H1	TCNT1H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	TCNT1H	TC1 The high byte of the count value. TCNT1H with TCNT1L Combined together to form TCNT1,by TCNT1 The register can directly 16 The bit count value is accessed for read and write. Read and write16 The bit register requires two operations. write16 Bit TCNT1 Should be written first TCNT1H. read 16 Bit TCNT1 Should be read first TCNT1L. CPU Correct TCNT1 The write operation of the register will prevent the comparison match from occurring in the next timer clock cycle, even if the timer has stopped. This allows initialization TCNT1 The value of the register is the same as OCR1x The values of are consistent without causing interruption. If writeTCNT1 Equal to or bypassed OCR1x When the value is set, the comparison match will be lost, resulting in incorrect waveforms. The timer stops counting when the clock source is not selected, but CPU Still accessible TCNT1. CPU The write counter has a higher priority than clearing or adding and subtracting operations.						

ICR1L –TC1 Input capture register low byte

ICR1L -TC1 Input capture register low byte								
address: 0x86					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	ICR1L7	ICR1L6	ICR1L5	ICR1L4	ICR1L3	ICR1L2	ICR1L1	ICR1L0
R/W	R/W	R/WR/W		R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	ICR1L	TC1 Enter the low byte of the captured value. ICR1H with ICR1L Combined together to form 16 Bit ICR1. Read and write16 The bit register requires two operations. write16 Bit ICR1 Should be written first ICR1H. read16 Bit						

		ICR1 Should be read first ICR1L. When the input capture is triggered, the count valueTCNT1 Will update and copy to ICR1 In the register.ICR1 Registers can also be used to define the count TOP value.

ICR1H –TC1 Input capture register high byte

ICR1H -TC1 Input capture register high byte								
address: 0x87					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	ICR1H7	ICR1H6	ICR1H5	ICR1H4	ICR1H3	ICR1H2	ICR1H1	ICR1H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name description							
7:0	ICR1H	TC1 Enter the high byte of the captured value. ICR1H with ICR1L Combined together to form 16 Bit ICR1. Read and write16 The bit register requires two operations. write16 Bit ICR1 Should be written first ICR1H. read16 Bit ICR1 Should be read first ICR1L. When the input capture is triggered, the count valueTCNT1 Will update and copy to ICR1 In the register.ICR1 Registers can also be used to define the count of TOP value.						

OCR1AL –TC1 Output compare register A Low byte

OCR1AL -TC1 Output compare register A Low byte								
address: 0x88					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR1AL7	OCR1AL6	OCR1AL5	OCR1AL4	OCR1AL3	OCR1AL2	OCR1AL1	OCR1AL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR1AL	Output compare register A The low byte. OCR1AL with OCR1AH Combined together to form 16 Bit OCR1A. Read and write16 The bit register requires two operationswrite16BitOCR1AShould be written firstOCR1AH. read 16 Bit OCR1A Should be read first OCR1AL. OCR1A Uninterruptedly with the counter value TCNT1 Compare. Compare match can be used to generate output compare interrupt, or used toOC1A A waveform is generated on the pin. When using PWMMode,OCR1A The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR1A The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR1A Buffer register, when double buffering function is disabled CPU Visiting is OCR1A itself.						

OCR1AH –TC1 Output compare register A High byte

OCR1AH -TC1 Output compare register A High byte								
address: 0x89					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR1AH7	OCR1AH6	OCR1AH5	OCR1AH4	OCR1AH3	OCR1AH2	OCR1AH1	OCR1AH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR1AH	Output compare register A The high byte. OCR1AL with OCR1AH Combined together to form 16 Bit OCR1A. Read and write16 The bit register requires two operationswrite16BitOCR1AShould be written firstOCR1AH. read 16 Bit OCR1A Should be read first OCR1AL. OCR1A Uninterruptedly with the counter value TCNT1 Compare. Compare match can be used to generate output compare interrupt, or used toOC1A A waveform is generated on the pin. When using PWMMode,OCR1A The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR1A The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR1A Buffer register, prohibited With double buffering function CPU Visiting is OCR1A itself.						

OCR1BL –TC1 Output compare register B Low byte

OCR1BL -TC1 Output compare register B Low byte								
address: 0x8A					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR1BL7	OCR1BL6	OCR1BL5	OCR1BL4	OCR1BL3	OCR1BL2	OCR1BL1	OCR1BL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR1BL	Output compare register B The low byte. OCR1BL with OCR1BH Combined together to form 16 Bit OCR1B. Read and write16 The bit register requires two operationswrite16BitOCR1BShould be written firstOCR1BH. read 16 Bit OCR1B Should be read first OCR1BL. OCR1B Uninterruptedly with the counter value TCNT1 Compare. Compare match can be used to generate output compare interrupt, or used toOC1B A waveform is generated on the pin. When usingPWMMode,OCR1B The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can update OCR1B The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR1B Buffer register, when double buffering function is disabled CPU Visiting is OCR1B itself.						

OCR1BH –TC1 Output compare register B High byte

OCR1BH -TC1 Output compare register B High byte								
address: 0x8B					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR1BH7	OCR1BH6	OCR1BH5	OCR1BH4	OCR1BH3	OCR1BH2	OCR1BH1	OCR1BH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR1BH	Output compare register B The high byte. OCR1BL with OCR1BH Combined together to form 16 Bit OCR1B. Read and write16 The bit register requires two operationswrite16BitOCR1BShould be written firstOCR1BH. read 16 Bit OCR1B Should be read first OCR1BL. OCR1B Uninterruptedly with the counter value TCNT1 Compare. Compare match can be used to generate output compare interrupt, or used toOC1B A waveform is generated on the pin. When using PWMMode,OCR1B The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR1B The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR1B Buffer register, when double buffering function is disabled CPU Visiting is OCR1B itself.						

TIMSK1 – TC1 Interrupt mask register

TIMSK1 **– TC1 Interrupt mask register**								
address: 0x6F					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	-	-	TICIE1	-	-	OCIE1A	OCIE1B	TOIE1
R/W	-	-	R/W	-	-	R/W	R/W	R/W
Bit	Name	description						
7:6	-	Reserved.						
5	TICIE1	TC1 Input capture interrupt enable control bit. when ICIE1 Bit is"1"Time, and the global interrupt is set,TC1 Input capture interrupt is enabled can. When the input capture is triggered, that isTIFR1 of ICF1 The flag is set and the interrupt is issued Raw. when ICIE1 Bit is"0"Time,TC1 The input capture interrupt is disabled.						
4:3	-	Reserved.						
2	OCIE1B	TC1 Output comparison B Match interrupt enable bit. whenOCIE1B Bit is"1", And the global interrupt is set,TC1 Output comparison B Matching Disable the enable. When a comparison match occurs, that isTIFR in OCF1B When the bit is set, An interrupt is generated. when OCIE1B Bit is"0"Time,TC1 Output comparison B The match interrupt is disabled.						
1	OCIE1A	TC1 Output comparison A Match interrupt enable bit. whenOCIE1A Bit is"1", And the global interrupt is set,TC1 Output comparison A Matching						

		Disable the enable. When a comparison match occurs, that isTIFR in OCF1A When the bit is set, an interrupt is generated. when OCIE1A Bit is"0"Time,TC1 Output comparison A The match interrupt is disabled.
0	TOIE1	TC1 Overflow interrupt enable bit. when TOIE1 Bit is"1", And the global interrupt is set,TC1 The overflow interrupt is enabled. whenTC1 Overflow occurs, i.e. TIFR middle TOV1 When the bit is set, an interrupt is generated. whenTOIE1 Bit is"0"Time,TC1 Overflow interrupts are disabled.

TIFR1 – TC1 Interrupt flag register

TIFR1 – TC1 Interrupt flag register								
address: 0x36					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	-	-	ICF1	-	-	OCF1B	OCF1A	TOV1
R/W	-	-	R/W	-	-	R/W	R/W	R/W

Bit	Name	description
7:6	-	Reserved.
5	ICF1	Input capture flag. When an input capture event occurs,ICF1 The flag is set. whenICR1 Used for counting TOP Value, and the count value reaches TOP Value,ICF1 The flag is set. IfICIE1 for "1"And the global interrupt flag is set, an input capture interrupt will be generated. When this interrupt service routine is executedICF1 Will be automatically cleared, or ICF1 Bit write"1"This bit can also be cleared. Reserved.
4:3	-	
2	OCF1B Sign	Output comparison B Match flag bit. when TCNT1 equal OCR1B When, the comparison unit gives a matching signal and sets the comparison OCF1B. If the output compare interrupt is enabled at this timeOCIE1B for"1"And global interrupt If the flag is set, an output compare interrupt will be generated. When this interrupt service routine is executedOCF1B Will be automatically cleared, or OCF1B Bit write"1"This bit can also be cleared.
1	OCF1A Sign	Output comparisonA Match flag bit. when TCNT1 equal OCR1A When, the comparison unit gives a matching signal and sets the comparison OCF1A. If the output compare interrupt is enabled at this timeOCIE1A for"1"And global interrupt If the flag is set, an output compare interrupt will be generated. When this interrupt service routine is executedOCF1A Will be automatically cleared, or OCF1A Bit write"1"This bit can also be cleared.
0	TOV1	Overflow flag. When the counter overflows, the overflow flag is set TOV1. If overflow interrupt is enabled at this time TOIE1 for"1"And the global interrupt flag is set, an overflow interrupt will be generated. When this interrupt service routine is executedTOV1 Will be automatically cleared, or TOV1 Bit write"1"You can also clear the Bit.

DTR1L –TC1 Dead time register low byte

DTR1-TC1 Dead time register								
address: 0x8C					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0

	DTR1L							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	DTR1L	Dead time register high byte. when DTEN1 When the bit is high,OC1A with OC1B Is complementary output,OC1A The dead time inserted on the output is determined by DTR1L It is determined by a count clock.						

DTR1H –TC1 Dead time register high byte

DTR1H-TC1 Dead time register high byte								
address: 0x8D					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	DTR1H							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	DTR1H	Dead time register high byte. when DTEN1 When the bit is high,OC1A with OC1B Is complementary output,OC1B The dead time inserted on the output is determined by DTR1H It is determined by a count clock.						

TCKCSR –TC Clock control status register

TCKCSR-TC Clock control status register								
address: 0xEC					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	-	F2XEN	TC2XF1	TC2XF0	-	AFCKS	TC2XS1	TC2XS0
R/W	-	R/W	R/O	R/O	-	R/W	R/W	R/W
Bit	Name	description						
7	-	Keep						
6	F2XEN	RC 32MMultiplier output enable control bit When set F2XEN Bit is"1"Time,32M RC The frequency multiplier output of the oscillator is enabled, and the output 64MHigh-speed clock When set F2XEN Bit is"1"Time,32M RC The frequency multiplier output of the oscillator is prohibited and cannot be output 64MHigh-speed clock						
5	TC2XF1	TC High-speed clock mode flag 1 When read TC2XF1 Bit is"1"Time, indicating that the timer counter 1 Working in high-speed clock mode, for"0"Time, indicating that the timer counter 1 Working in system clock mode						
4	TC2XF0	TC High-speed clock mode flag 0, Refer to the timer counter 0 Register description reserved						
3:2	-							
1	TC2XS1	TC High-speed clock mode selection control bit 1 When set TC2XS1 Bit is"1"When, select the timer counter 1 Working in high-speed clock mode When set TC2XS1 Bit is"0"When, select the timer counter 1 Working in system clock mode						
0	TC2XS0	TC High-speed clock mode selection control bit 0, Refer to the timer counter 0 Register description						

TMR0/1/3 Prescaler

z 3 A 10 Bit prescaler

z In multiplexing mode TC0,TC1 with TC3 Multiplex prescaler CPS310

z In standalone mode TC0 Dedicated prescaler CPS310,TC1 Dedicated prescaler CPS1,TC3 Dedicated prescaler CPS3

z Support software reset

Overview

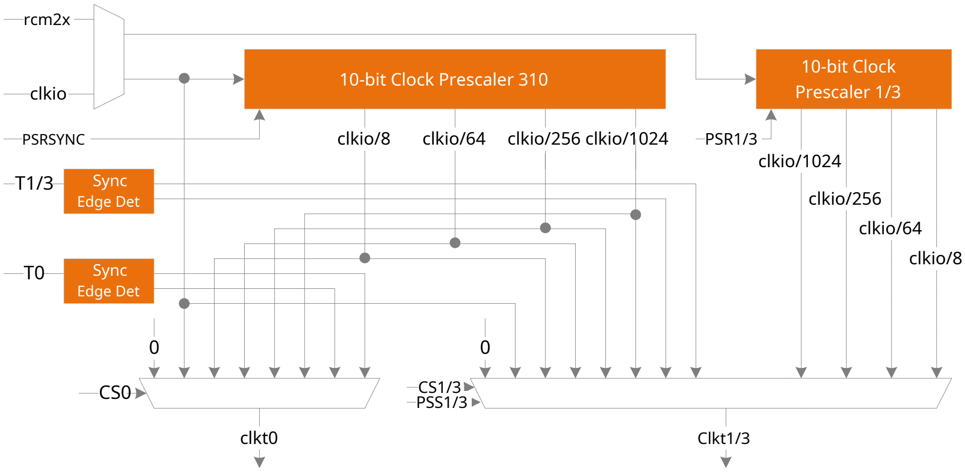
In multiplexing mode (PSS1=0 And PSS3=0),TC0,TC1 with TC3 Share one 10 Prescaler CPS310, But they have different crossover settings.

In single-use mode (PSS1=1 And PSS3=0),TC1 Use one independently 10 Prescaler CPS1,TC0 with TC3 Share one 10 Prescaler CPS310, But they have different crossover settings.

In single-use mode (PSS1=0 And PSS3=1),TC3 Use one independently 10 Prescaler CPS3,TC0 with TC1 Share one 10 Prescaler CPS310, But they have different crossover settings.

In standalone mode (PSS1=1 And PSS3=1),TC0 Use one independently 10 Prescaler CPS310,TC1 alone Use one immediately 10 Prescaler CPS1,TC3 Independent use of prescaler CPS3.

The following description is used TC0,TC1 with TC3,among them n representative 0,1 or 3.



TC0/TC1 /TC3 Prescaler Structure chart

Internal clock source

When set CSn[2:0]=1 Time, timer 3 Can only be used by the system clock clkio Drive, timer counter 0 or 1 Can be directly from the system clock clkio Or high-speed clock rcm2x(internal 32MRC Oscillator output clock 2 Frequency multiplier) drive. Prescaler can output4 Different clock frequencies, which are clkio/8,clkio/64,clkio/256 with clkio/1024.

Divider reset

Multiplexing mode

When set PSS1 Bit is"0"And PSS3 Bit is"0"Time,TC0,TC1 with TC3 Share a prescaler CPS310.

The prescaler operates independently, and its operation is independent of TC Clock selection logic, and it is determined by TC0,TC1 with TC3 shared. Since it is not affected by the clock selection control, the state of the prescaler will affect the application of the divided clock. When the timer is enabled and the output of the prescaler is selected as the count clock source (6>CSn[2:0]>1), the impact will occur. It may take from the timer enable to the first count1ToN+1System clocks, whereNIs the prescaler factor (8, 64,256 or 1024).

It is possible to synchronize the timer and program operation by resetting the prescaler. But it must be noted that if another timer is using this prescaler, resetting the prescaler will affect all timers connected to it.

Single use mode

When set PSS1 Bit is"1"Time,TC1 Independent use of prescaler CPS1, The reset of the prescaler is determined by PSR1 Bit to control. Each reset works independently and will not affect other prescalers.

When set PSS3 Bit is"1"Time,TC3 Independent use of prescaler CPS3, The reset of the prescaler is determined by PSR3 Bit to control. Each reset works independently and will not affect other prescalers.

When set PSS1 Bit is"1"And PSS3 Bit is"1"Time,TC0 Independent use of prescaler CPS310, The reset of the prescaler is determined by PSRSYNC Bit to control,TC1 Independent use of prescaler CPS1,TC3 Independent use of prescaler CPS3, The respective reset functions independently and will not affect other prescalers.

External clock source

by T0/T1/T3 The external clock source provided by the pin can be used as the counting clock source.T0/T1/T3 The signal of the pin is used as the clock source of the counter after passing through the synchronization logic and edge detector. Every rising edge (CSn[2:0]=7) Or falling edge (CSn[2:0]=6) Will generate a count pulse. The external clock source will not be fed into the prescaler.

Due to the existence of the synchronization and edge detection circuit on the pin,T0/T1/T3 Changes in the upper level need to be delayed 2.5 To 3.5 A system clock can make the counter update.

Disable or enable clock input must be T0/T1/T3 Maintaining stability requires at least one system clock cycle before proceeding, otherwise there is a possibility of incorrect counting clock pulses.

In order to ensure correct sampling, the external clock pulse width must be greater than one system clock cycle, and the duty cycle is 50% The external clock frequency must be less than half of the system clock frequency. Due to the difference in system clock frequency and duty cycle caused by the error of the oscillator itself, it is recommended that the maximum frequency of the external clock should not be greater thanf_{sys}/2.5.

Register definition

GTCCR **- General purpose timer counter control register**

GTCCR **- General purpose timer counter control register**								
address: 0x43					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	TSM	-	-	-	-	-	PSRASY	PSRSYNC
R/W	R/W	-	-	-	-	-	W	W
Bit	Name	description						
7	TSM	Timer counter synchronization mode control bit. When set TSMBit is"1"When, it is the timer counter synchronization mode. In synchronous mode, writePSRASY Bit sum PSRSYNC The value of the bit is maintained, so that the corresponding prescaler is always reset. This ensures that the corresponding timer counter is stopped and configured to the same value. When setTSM Bit is"0"Time, PSRASY Bit sum PSRSYNC The value of the bit will be cleared by hardware, and the timer counter will start working at the same time.							
6:2	-	Reserved.						
1	PSRASY	See timer TC2 Register description.						
0	PSRSYNC	Prescaler CPS310 Reset the control bit. When set PSRSYNC Bit is"1"Time, prescaler CPS310 Will be reset. whenTSM When the bit is not set, the hardware will be cleared after reset PSRSYNC Bit. When setPSRSYNC Bit is"0"When the setting is invalid. In multiplexing mode,TC0/TC1/TC3 Shared prescaler, reset will affect these three timers. In standalone mode, reset will only affect TC0. Reading the value of this bit will always be"0".						

PSSR **- Prescaler selection register**

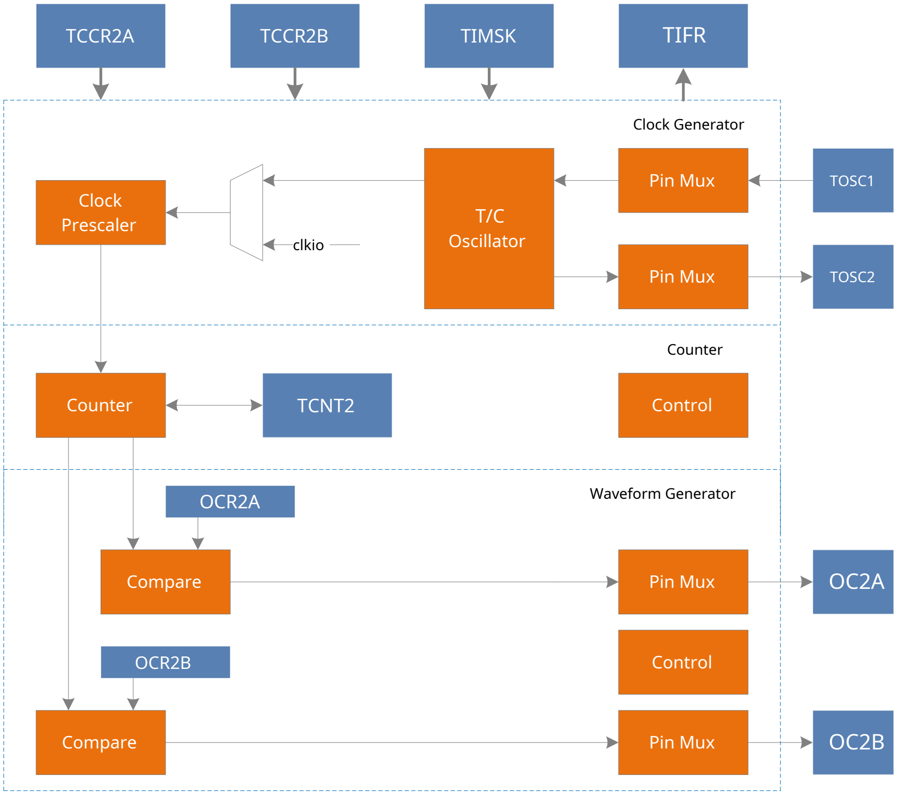
PSSR **- Prescaler selection register**								
address: 0xE2					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	PSS1	PSS3	-	-	-	-	PSR3	PSR1
R/W	R/W	R/W	-	-	-	-	R/W	R/W
Bit	Name	description						
7	PSS1	Prescaler selection control bit. When set PSS1 Bit is"1"Time,TC1 Use prescaler alone CPS1. When setPSS1 Bit is "0"When, it is the prescaler multiplexing mode.TC0 with TC1 Shared prescaler CPS310. PrescalerCPS1 If it is invalid, it will always be reset. IfPSS3 Bit at the same time"0",TC3 with TC0,TC1 Shared prescaler CPS310. Prescaler CPS1 with CPS3 Both are invalid and will always be reset.							
6	PSS3	Prescaler selection control bit. When set PSS3 Bit is"1"Time,TC3 Use prescaler alone CPS3.						

		When set PSS3 Bit is"0"When, it is the prescaler multiplexing mode.TC0 with TC3 Shared prescaler CPS310. PrescalerCPS3 If it is invalid, it will always be reset. IfPSS1 Bit at the same time"0",TC1 with TC0,TC3 Shared prescaler CPS310. Prescaler CPS1 with CPS3 Both are invalid and will always be reset.
5:2	-	Reserved.
1	PSR3	Prescaler CPS3 Reset the control bit. PSR3 Only in TC3 Effective in single use mode. When setPSR3 Bit is"1"Time, prescaler CPS3 Will be reset. The hardware will be cleared after resetPSR3 Bit. When setPSR3 Bit is"0"When the setting is invalid. Reading the value of this bit will always be"0".
0	PSR1	Prescaler CPS1 Reset the control bit. PSR1 Only in TC1 Effective in single use mode. When setPSR1 Bit is"1"Time, prescaler CPS1 Will be reset. The hardware will be cleared after resetPSR1 Bit. When setPSR1 Bit is"0"When the setting is invalid. Reading the value of this bit will always be"0".

Timer/counter 2 (TMR2)

- 8 Bit counter
- Two independent comparison units
- When a comparison match occurs, the counter is automatically cleared and automatically loaded
- Phase-corrected without interference pulses PWMOutput
- Frequency generator
- External event counter
- 10 Bit clock prescaler
- Overflow and compare match interrupt
- Allow the use of external 32.768KHz of RTC Crystal count

Overview



TC2 Structure chart

TC2 Is a universal 8 Bit timer counter module, support PWMOutput, can accurately generate waveform.TC2 contain 1 A 8 Bit counter, waveform generation mode control unit and 2 Output comparison unit. The waveform generation mode control unit controls the working mode of the counter and the generation of the comparison output waveform. According to different working modes, the counter counts each clockClkt2 Realize the operation of clearing, adding one or subtracting one.Clkt2 It can be generated by an internal clock source or an external clock source. When using external32.768KHz When the crystal oscillator counts,TC2 Can be used as RTC counter. When the count value of the counterTCNT2

Reach the maximum value (equal to the maximum value 0xFF Or output compare register OCR2A,defined as TOP, The maximum value is defined as MAX To show the difference), the counter will be cleared or decremented by one. When the count value of the counterTCNT2 Reached the minimum value (equal to 0x00,defined as BOTTOM), the counter will increment by one. When the count value of the counterTCNT2 Arrivals OCR2A/OCR2B When a comparison match occurs, it will be cleared or set to output a comparison signal OC2A/OCR2BTo produce PWMWaveform.

Operating mode

Timer counter 2 There are four different working modes, including normal mode (Normal), cleared on comparison match (CTC) Mode, fast pulse width modulation (FPWM) Mode and phase correction pulse width modulation (PCPWM) Mode, which is controlled by the waveform generation modeWGM2[2:0]To choose. The four modes are described in detail below. Since there are two independent output comparison units, use"A"with"B"To indicate, in lowercase"X"To represent these two output comparison unit channels.

Normal mode

Normal mode is the simplest working mode of the timer counter. At this time, the waveform generation mode control bitWGM2[2:0]=0, The maximum value of the count TOP for MAX(0xFF). In this mode, the counting method increases by one for each counting clock. When the counter reachesTOP Back after overflow BOTTOM Start accumulation again. Count valueTCNT2 The timer counter overflow flag is set in the same counting clock that becomes zero TOV2. In this modeTOV2 The sign is like the first 9 The counting bit is only set but not cleared. Overflow interrupt service routine will be automatically clearedTOV2 Mark, the software can use it to improve the resolution of the timer counter. There is no special situation to consider in the normal mode, and a new count value can be written at any time. Set upOC2xThe output comparison signal can only be obtained when the data direction register of the pin is outputOC2x的waveform. whenCOM2x=1 When a comparison match occurs, it will be flipped OC2x Signal, the frequency of the waveform in this case can be calculated with the following formula:

$$f_{oc2xnormal} = f_{sys}/(2*N*256)$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

The output compare unit can be used to generate interrupts, but it is not recommended to use interrupts in normal mode, which will take up too much CPU time.

CTC mode

Set up WGM2[2:0]=2 Time counter 2 enter CTC Mode, the maximum value of the count TOP for OCR2A. In this mode, the counting method is each counting clock plus one increment, when the counter valueTCNT2 equal TOP The hour counter is cleared. OCR2A Defines the maximum value of the count, that is, the resolution of the counter. This mode allows users to easily control the frequency of the compare match output, and also simplifies the operation of counting external events.

When the counter reaches the maximum count, the compare match flag is output OCF2 Is set, an interrupt will be generated when the corresponding interrupt enable bit is set. Can be updated in the interrupt service routineOCR2A The register is the maximum value of the count. In this mode OCR2A Double buffering is not used. Be careful when updating the maximum value to close to the minimum value when the counter is working with no prescaler or a very low prescaler. If writeOCR2A Is less than the value at the time TCNT2 Value, the counter will lose a comparison match. Before the next comparison match occurs, the counter has to count toTOP, And then from BOTTOM Start counting toOCR2Avalue. Same as normal mode, Count value back toBOTTOMSet in the count clock TOV2Sign.

Set upOC2xThe output comparison signal can only be obtained when the data direction register of the pin is outputOC2x的waveform. whenCOM2x=1 When a comparison match occurs, it will be flipped OC2x Signal, the frequency of the waveform in this case can be calculated with the following formula:

$$f_{oc2xctc} = f_{sys}/(2*N*(1+OCR2A))$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024). It can be seen from the formula that when settingOCR2x for 0x0 And without prescaler, the maximum frequency that can be obtained is $f_{sys}/2$ The output waveform.

fast PWMmode

Set up WGM2[2:0]=3 or 7 Time counter 2 Enter fast PWM Mode, which can be used to generate high frequency PWMWaveform, maximum count TOP Respectively MAX(0xFF)or OCR2A. fastPWMMode and others PWMThe mode is different in that it is a one-way operation. Counter from minimum0x00 Accumulate to TOP Back again BOTTOM Recount. When the count valueTCNT2 Arrivals OCR2x or BOTTOM When, output the comparison signal OC2x Will be set or cleared, depending on the comparison output mode COM2x See the register description for details. Due to the one-way operation, fastPWMThe operating frequency of the mode is phase correction with bidirectional operation PWM Twice the pattern. High frequency characteristics make fastPWM Mode is suitable for power regulation, rectification and DAC application. High-frequency signals can reduce the size of external components (inductance, capacitance, etc.), thereby reducing system costs.

When the count value reaches the maximum value, the timer counter overflow flag TOV2 Will be set, and the value of the compare buffer will be updated to the compare value. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routineOCR2x register. Set upOC2x The output comparison signal can only be obtained when the data direction register of the pin is output OC2x 的waveform. The frequency of the waveform can be calculated with the following formula:

$f_{oc2xfpwm} = f_{sys/(}N*(1+TOP))$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

when TCNT2 with OCR2x When a comparison match occurs, the waveform generator is set (cleared)OC2x Signal when TCNT2 When cleared, the waveform generator is cleared (set)OC2x Signal to generate PWMwave. thusOCR2x Extreme values will produce special PWM Waveform. whenOCR2x Set as 0x00 When the output PWM For every (1+TOP)There is a narrow spike in each count clock. when OCR2x When set to the maximum value, the output waveform is a continuous high or low level.

Phase correction PWMmode

When set WGM2[2:0]=1 or 5 Time counter 2 Enter phase correction PWM Mode, the maximum value of the count TOP Respectively MAX(0xFF)or OCR2A. The counter adopts two-way operation, byBOTTOM Increment to TOP, And then decrease to BOTTOM, Repeat this operation again. Count arrivalTOP with BOTTOMChange the counting direction every time, the count value is in TOP or BOTTOM There is only one counting clock on each. In the process of increment or decrement, the count valueTCNT2 versus OCR2x When matching, output comparison signal OC2x Will be cleared or set, depending on the comparison output mode COM2x setting. Compared with unidirectional operation, the maximum frequency that can be obtained by bidirectional operation is smaller, but its excellent symmetry is more suitable for motor control. Phase correction PWM Mode, when the count reaches BOTTOM Set when TOV2 Sign when the count arrives TOP Update the value of the comparison buffer to the comparison value at the time. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routineOCR2x register.

Set up OC2x The output comparison signal can only be obtained when the data direction register of the pin is output OC2x 的waveform. The frequency of the waveform can be calculated with the following formula:

$f_{oc2xpcpwm} = f_{sys/(}N*TOP*2)$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

During the count-up process, when TCNT2 versus OCR2x When there is a match, the waveform generator is cleared (set)OC2x signal. In the process of counting down, whenTCNT2 versus OCR2x When a match, the waveform generator is set (cleared)OC2x signal. thus OCR2x The extreme value of will produce a special PWMwave. whenOCR2x When set to the maximum or minimum value,OC2x The signal output will always remain low or high.

In order to ensure the output PWMThe symmetry of the wave on both sides of the minimum value will be reversed in two cases when there is no comparison match. OC2x signal. The first situation is whenOCR2x Maximum value 0xFF When changing to other data. whenOCR2x Is the maximum value, when the count value reaches the maximum,OC2x The output of is the same as the result of the comparison match when counting in descending order, that is, keep OC2x constant. At this time, the comparison value will be updated to the new oneOCR2x Value (not 0xFF),OC2x The value will always be maintained, straight

When the count is in the ascending order, a comparison match occurs and flips. at this timeOC2x The signal is not symmetrical about the minimum value, so it needs to be TCNT2 Flip when reaching maximum OC2x Signal, that is, there is no rollover when a comparison match occurs OC2x The first case of the signal. The second situation is whenTCNT2 From OCR2x When the high value starts to count, a comparison match will be lost, which will cause asymmetry. Also need to flipOC2x The signal to achieve symmetry on both sides of the minimum.

TC2 Asynchronous operation

When at ASSR Register AS2 Bit is"1"Time,TC2 Working in asynchronous mode, the clock source of the counter comes from the oscillator of the external timer counter. In asynchronous modeTC2 The operation should consider the following points.

?? Switching between synchronous and asynchronous modes may cause TCNT2,OCR2A,OCR2B,TCCR2A with TCCR2B Corruption of data. The safe operation steps are as follows:

1. Cleared OCIE2A,TOIE2 with OCIE2B Register bit to turn off TC2 Interruption
2. Position AS2 Select the appropriate clock source;
3. Correct TCNT2,OCR2A,TCCR2A,OCR2B with TCCR2B Write new data into the register;
4. When switching to asynchronous mode, wait TCN2UB,OCR2AUB,TCR2AUB,OCR2BUB with TCR2BUB Bit cleared
5. Cleared TC2 The interrupt flag bit;
6. Enable the interrupts that need to be used.

?? Oscillator is best to use 32.768KHz Watch crystal oscillator. The system clock frequency must be higher than the crystal frequency4 Times more.

?? CPU write TCNT2,OCR2A,TCCR2A,OCR2B with TCCR2B Time, the hardware will put the data into the scratchpad first, two TOSC1 It is latched into the corresponding register after the rising edge of the clock. Before the data is latched from the scratchpad to the destination register, no new data write operation can be performed. Each register has its own independent scratchpad, so writeTCNT2 Does not interfere with writing OCR2. Asynchronous Status RegisterASSR Used to check whether the data has been written to the destination register. If usingTC2

?? As MCU The wake-up condition of the sleep mode, the sleep mode cannot be entered before the end of each register update, otherwise MCU May be in TC2 Enter the sleep mode before the setting takes effect, thereby TC2 Unable to wake up the system.

?? If using TC2 As MCU In the wake-up condition of sleep mode, attention must be paid to the process of re-entering sleep mode. Interrupt logic needs oneTOSC1 The clock cycle is reset, if the time from wake-up to re-entry to sleep is less than one TOSC1 In the clock cycle, the interrupt will no longer occur, and the device cannot wake up. The following methods of operation are recommended:

1. Write appropriate data to each register;
2. wait ASSR The corresponding update busy flag is cleared;
3. Enter sleep mode.

?? If the asynchronous working mode is selected,TC2 The oscillator will always work unless it enters power-down mode. The user must note that the stable time of this oscillator may be as long as1 Seconds, therefore, it is recommended that the user TC2 Wait at least after the oscillator

1 Use again in seconds TC2 Asynchronous working mode. The wake-up process in sleep mode in asynchronous working mode: After the

?? interrupt condition is met, the wake-up process is started on the next timer clock. In other words, the counter accumulates at least one more clock before the processor can read the value of the counter. After wake upMCU Execute interrupt service routine, then start execution SLEEP The program after the statement. Read within a short time after waking up from sleep modeTCNT2 The value of may return incorrect data.

?? becauseTCNT2 Is made asynchronous TOSC1 Clock-driven, while reading TCNT2 It must be done through a register synchronized with the internal system clock. Synchronization occurs at each TOSC1 The rising edge. The system clock is reactivated after waking up from sleep mode, and the readTCNT2 The value is the value before entering sleep mode until the next TOSC1 It will be updated only when the rising edge arrives. When waking up from sleep modeTOSC1 The phase of is completely unpredictable and is related to the wake-up time. Therefore, readTCNT2

The recommended sequence of values is:

1. Write an arbitrary value to OCR2A or TCCR2A；

2. Wait for the corresponding update busy flag to be cleared;

3. Read TCNT2.

?? In asynchronous mode, the synchronization of the interrupt flag bit is required 3 System clock cycles plus 1 Timer cycles. inMCU The counter has accumulated at least one more clock before the counter value that caused the interrupt flag to be set can be read. The change of the output compare signal is synchronized with the timer clock, not the system clock.

TC2 Prescaler

TC2 The input clock of the prescaler is called clkt2s, Located by ASSR Register AS2 Bit to select the internal system clock clkio Or external TOSC1 Clock source, the default is the system clock clkio Connected. IfAS2 Position,TC2 Will be by TOSC1 Asynchronous drive. whenTOSC1 Pin and TOSC2 An external pin 32.768KHz Clock crystal oscillator,TC2 Can be used as RTC counter. Not recommended inTOSC1 The external clock signal is directly applied to the pin.

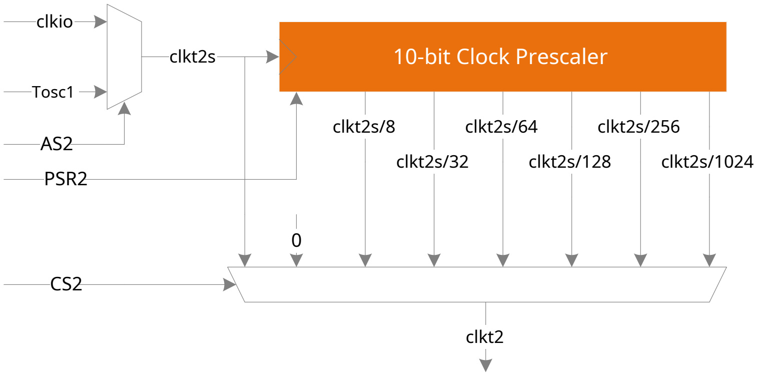


Figure 5 TC2 Prescaler structure diagram

The picture above is TC2 Prescaler, as shown in the figure, the possible prescaler options are:clkt2s/8, clkt2s/32, clkt2s/64, clkt2s/128, clkt2s/256,with clkt2s/1024. You can also chooseclkt2s with 0(Stop counting). PositionSFIOR Register PSR2 The bit resets the prescaler, allowing the user to start from a predictable prescaler.

Register definition

TC2 Register list

register	address	Defaults	description
TCCR2A	0xB0	0x00	TC2 Control register A
TCCR2B	0xB1	0x00	TC2 Control register B
TCNT2	0xB2	0x00	TC2 Count value register
OCR2A	0xB3	0x00	TC2 Output compare register A
OCR2B	0xB4	0x00	TC2 Output compare register B
ASSR	0xB6	0x00	TC2 Asynchronous Status Register
TIMSK2	0x70	0x00	Timer counter interrupt mask register
TIFR2	0x37	0x00	Timer counter interrupt flag register

TCCR2A–TC2 Control register A

TCCR2 A-TC2 Control register A								
address: 0xB0					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	COM2A1	COM2A0	COM2B1	COM2B0	-	-	WGM21	WGM20
R/W	W	R/W	R/W	R/W	-	-	R/W	R/W

Bit	Name	description
7	COM2A1	TC2 Compare match output A Mode control high. COM2A1 with COM2A0 Together to form output comparison mode control COM2A[1:0],control OC2A The output waveform. in caseCOM2A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC2A Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM2A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
6	COM2A0	TC2 Compare match output A Mode control low. COM2A0 with COM2A1 Together to form output comparison mode control COM2A[1:0],control OC2A The output waveform. in caseCOM2A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC2A Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM2A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
5	COM2B1	TC2 Compare match output B Mode control high. COM2B1 with COM2B0 Together to form output comparison mode control COM2B[1:0],control OC2B The output waveform. in caseCOM2B of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC2B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM2B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
4	COM2B0	TC2 Compare match output B Mode control low. COM2B0 with COM2B1 Together to form output comparison mode control COM2B[1:0],control OC2B The output waveform. in caseCOM2B of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC2B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM2B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details. Reserved.
3:2	-	
1	WGM21	TC2 The waveform generation mode controls the high bit. WGM20 with WGM21,WGM22 Form the waveform generation mode control together WGM2[2:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
0	WGM20	TC2 Waveform generation mode control low bit. WGM21 with WGM20,WGM22 Form the waveform generation mode control together WGM2[2:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.

TCCR2B –TC2 Control register B

TCCR2B -TC2 Control register B								
address: 0xB1					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	FOC2A	FOC2B	-	-	WGM22	CS22	CS21	CS20
R/W	W	W	-	-	R/W	R/W	R/W	R/W

Bit	Name	description
7	FOC2A	TC2 Force output comparison A Control bit. Work in nonPWM Mode, you can compulsorily output the comparison bit FOC2A write"1"Way to generate comparison matches. Force compare match will not be setOCF2A Flag, it will not reload or clear the timer, but the output pin OC2A Will be followed COM2A The settings are updated accordingly, just as if a comparison match actually occurred. Read FOC2A The return value of is always zero.
6	FOC2B	TC2 Force output comparison B Control bit. Work in nonPWM Mode, you can compulsorily output the comparison bit FOC2B write"1"Way to generate comparison matches. Force compare match will not be setOCF2B Flag, it will not reload or clear the timer, but the output pin OC2B Will be followed COM2B The settings are updated accordingly, just as if a comparison match actually occurred. Read FOC2B The return value of is always zero.
5:4	-	Reserved.
3	WGM22	TC2 The waveform generation mode controls the high bit. WGM22 withWGM20,WGM21 Form the waveform generation mode control togetherWGM2[2:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
2	CS22	TC2 The clock selection controls the high bit. Used to select the timer counter 2 Clock source.
1	CS21	TC2 Clock selection control center position. Used to select the timer counter 2 Clock source.
0	CS20	TC2 Clock selection control low bit. Used to select the timer counter 2 Clock source.

CS2[2:0]	description
0	No clock source, stop counting
1	clk_{t2s}
2	clk_{t2s}/8, From the prescaler
3	clk_{t2s}/32, From the prescaler
4	clk_{t2s}/64, From the prescaler
5	clk_{t2s}/128, From the prescaler
6	clk_{t2s}/256, From the prescaler
7	clk_{t2s}/1024, From the prescaler

The following table is not PWMMode (i.e. normal mode and CTC Mode), the comparison output mode controls the output comparison waveform.

Table 1 non- PWMMode OC2x Comparison output mode control

COM2x[1:0]	description
0	OC2x Disconnect, general IO Flip when the port
1	operation compares match OC2x signal
2	Clear on comparison match OC2x signal
3	Set on compare match OC2x signal

The following table is fast PWMThe comparison output mode in the mode controls the output comparison waveform.

Table 2 fast PWMMode OC2x Comparison output mode control

COM2x[1:0]	description
0	OC2x Disconnect, general IO Port operation
1	reservation
2	Clear on comparison match OC2x Signal, set when the maximum value matches OC2x Set when the
3	signal compare matches OC2x Signal, cleared when the maximum value matches OC2x signal

The following table shows the control of the output comparison waveform in the comparison output mode in the phase correction mode.

Table 3 Phase correction PWMMode OC2x Comparison output mode control

COM2x[1:0]	description
0	OC2x Disconnect, general IO Port operation
1	reservation
2	Cleared when compare matches in ascending order OC2x Signal, set when compare match in descending count OC2x signal
3	Set when compare match in ascending count OC2x Signal, compare match in descending count Time clear OC2x signal

The following table shows the waveform generation mode control.

Table 4 Waveform generation mode control

WGM2[2:0]	Operating mode	TOP value	Update OCR2x time	Position TOV2 time
0	Normal	0xFF	immediately	MAX
1	PCPWM	0xFF	TOP	BOTTOM
2	CTC	OCR2A	immediately	MAX
3	FPWM	0xFF	TOP	MAX
4	Keep	-	-	-
5	PCPWM	OCR2A	TOP	BOTTOM
6	Keep	-	-	-
7	FPWM	OCR2A	TOP	TOP

TCNT2 –TC2 Count value register

TCNT2 -TC2 Count value register								
address: 0xB2					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	TCNT27	TCNT26	TCNT25	TCNT24	TCNT23	TCNT22	TCNT21	TCNT20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	TCNT2	TC2 Count value register. by TCNT2 The register can directly 8 Read and write access for the count value. CPU Correct TCNT2 The write operation of the register will prevent the comparison match from occurring in the next timer clock cycle, even if the timer has stopped. This allows initializationTCNT2 The value of the register is the same as OCR2 The values of are consistent without causing interruption. If write TCNT2 Equal to or bypassed OCR2 Value, the comparison match will be lost, resulting in incorrect waveforms. The timer stops counting when the clock source is not selected, but CPU Still accessible TCNT2.CPU The write counter has a higher priority than clearing or adding and subtracting operations.

OCR2A – TC2 Output compare register A

***OCR2A* – TC2 Output compare register A**								
address: 0xB3					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR2A7	OCR2A6	OCR2A5	OCR2A4	OCR2A3	OCR2A2	OCR2A1	OCR2A0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	OCR2A	TC2 Output compare register A. OCR2A Contains a 8 Bit data, uninterruptedly with the counter value TCNT2 Compare. Compare match can be used to generate output compare interrupt, or used toOC2A A waveform is generated on the pin. When using PWMMode,OCR2A The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can update OCR2A The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR2A Buffer register, when double buffering function is disabled CPU Visiting is OCR2A itself.

OCR2B – TC2 Output compare register B

***OCR2B* – TC2 Output compare register B**								
address: 0xB4					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR2B7	OCR2B6 OCR2B5 OCR2B4 OCR2B3 OCR2B2 OCR2B1 OCR2B0 R/WR/						
R/W	R/W	WR/WR/WR/WR/WR/W						

Bit	Name	description
7:0	OCR2B	TC2 Output comparison B register. OCR2B Contains a 8 Bit data, uninterruptedly with the counter value TCNT2 Compare. Compare match can be used to generate output compare interrupt, or used toOC2B A waveform is generated on the pin. When using PWMMode,OCR2B The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR2B The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWM Pulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR2B Buffer register, when double buffering function is disabled CPU Visiting is OCR2B itself.

TIMSK2 – TC2 Interrupt mask register

TIMSK2 – TC2 Interrupt mask register								
address: 0x70					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCIE2B	OCIE2A	TOIE2
R/W	-	-	-	-	-	R/W	R/W	R/W

Bit	Name	description
7:3		Reserved.
2	OCIE2B	TC2 Output comparison B Match interrupt enable bit. when OCIE2B Bit is"1", And the global interrupt is set,TC2 Output comparison B Match interrupt is enabled. When a comparison match occurs, that isTIFR2 in OCF2B When the bit is set, an interrupt is generated. whenOCIE2B Bit is"0"Time,TC2 Output comparison B The match interrupt is disabled.
1	OCIE2A	TC2 Output comparison A Match interrupt enable bit. whenOCIE2A Bit is"1", And the global interrupt is set,TC2 Output comparison A Match interrupt is enabled. When a comparison match occurs, that isTIFR2 in OCF2A When the bit is set, an interrupt is generated. whenOCIE2A Bit is"0"Time,TC2 Output comparison A The match interrupt is disabled.
0	TOIE2	TC2 Overflow interrupt enable bit. when TOIE2 Bit is"1", And the global interrupt is set,TC2 The overflow interrupt is enabled. whenTC2 Overflow occurs, i.e. TIFR2 middle TOV2 When the bit is set, an interrupt is generated. whenTOIE2 Bit is"0"Time, TC2 Overflow interrupts are disabled.

TIFR2 – TC2 Interrupt flag register

TIFR2 – TC2 Interrupt flag register								
address: 0x37					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	-	-	-	-	-	OCF2B	OCF2A	TOV2
R/W	-	-	-	-	-	R/W	R/W	R/W

Bit	Name	description
7:3	-	Reserved.
2	OCF2B	TC2 Output comparison B Match flag bit.

		when TCNT2 equal OCR2B When, the comparison unit gives a match signal and sets the comparison flag OCF2B. If the output is compared at this timeB Interrupt enable OCIE2B for"1"And the global interrupt flag is set, an output comparison will be generated B Interrupted. When this interrupt service routine is executedOCF2B Will be automatically cleared, or OCF2B Bit write"1"This bit can also be cleared.
1	OCF2A	TC2 Output comparison A Match flag bit. whenTCNT2 equal OCR2A When, the comparison unit gives a match signal and sets the comparison flag OCF2A. If the output is compared at this timeA Interrupt enable OCIE2A for"1"And the global interrupt flag is set, an output comparison will be generated A Interrupted. When this interrupt service routine is executedOCF2A Will be automatically cleared, or OCF2A Bit write"1"This bit can also be cleared.
0	TOV2	TC2 Overflow flag. When the counter overflows, the overflow flag is set TOV2. If overflow interrupt is enabled at this timeTOIE2 for"1"And the global interrupt flag is set, an overflow interrupt will be generated. When this interrupt service routine is executedTOV2 Will be automatically cleared, or TOV2 Bit write"1"This bit can also be cleared.

ASSR – Asynchronous interface status register

ASSR– **TC2 Asynchronous interface status register**								
address: 0xB6					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	INTCK	-	AS2	TCN2UB	OCR2AUB	OCR2BUB	TCR2AUB	TCR2BUB
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	INTCK	Asynchronous clock selection control bit. When set INTCK Bit is 1 , Select internal RC32K As an asynchronous clock source. When setINTCK Bit is 0 When, select the external crystal oscillator clock as the asynchronous clock source. Reserved.
6	-	
5	AS2	Timer 2 Asynchronous mode selection control bit. When set AS2 Bit is 1 Time, timer 2 It works in asynchronous mode, and its clock source is INTCK Bit to choose. When set AS2 Bit is 0 Time, timer 2 Working in synchronous mode, its clock source is Clk_{io}. whenAS2 When the value of is changed,TCNT2,OCR2A,OCR2B,TCCR2A with TCCR2B The value of the register may be incorrect and needs to be reconfigured.
4	TCN2UB	TCNT2 Register update flag. When the timer 2 Working in asynchronous mode, right TCNT2 When performing a write operation, TCN2UB The bit will be set. whenTCNT2 After the value of is updated, the hardware will clear it to zero TCN2UB Bit. Only ifTCN2UB Bit is 0 When TCNT2 Update.
3	OCR2AUB	OCR2A Register update flag. When the timer 2 Working in asynchronous mode, right OCR2A When performing a write operation,OCR2AUB The bit will be set. whenOCR2A After the value of is updated, the hardware will clear it to zero OCR2AUB Bit. Only ifOCR2AUB Bit is 0 When OCR2A Update.
2	OCR2BUB	OCR2B Register update flag. When the timer 2 Working in asynchronous mode, right OCR2B When performing a write operation,OCR2BUB The bit will be set. whenOCR2B After the value of is updated, the hardware will clear it to zero OCR2BUB Bit. Only ifOCR2BUB Bit is 0 When OCR2B Update.

1	TCR2AUB	TCCR2A Register update flag. When the timer 2 Working in asynchronous mode, right TCCR2A When performing a write operation,TCR2AUB The bit will be set. whenTCCR2A After the value of is updated, the hardware will clear it to zero TCR2AUB Bit. Only ifTCR2AUB Bit is 0 When TCCR2A Update.
0	TCR2BUB	TCCR2B Register update flag. When the timer 2 Working in asynchronous mode, right TCCR2B When performing a write operation,TCR2BUB The bit will be set. whenTCCR2B After the value of is updated, the hardware will clear it to zero TCR2BUB Bit. Only ifTCR2BUB Bit is 0 When TCCR2B Update.

Timer/counter 3 (TMR3)

- Z truly 16 Bit design, allowing 16 Bit PWM 3
- Z Independent output comparison unit
- Z Double-buffered output compare register
- Z 1 Input capture unit
- Z Input capture noise suppressor
- Z Automatically clear the counter and load automatically when compare matches
- Z Phase-corrected without interference pulses PWM
- Z Variable PWMcycle
- Z Frequency generator
- Z External event counter
- Z 5 Independent interrupt source
- Z With dead time control
- Z 6 Selectable trigger sources are automatically turned off PWMOutput

Overview

TC3 Is a universal 16 Bit timer counter module, support PWMOutput, can accurately generate waveform.TC3 contain 1 A 16 Bit counter, waveform generation mode control unit,2 Independent output comparison unit and 1 Input capture unit. The waveform generation mode control unit controls the working mode of the counter and the generation of the comparison output waveform. According to different working modes, the counter counts each clockClkt3 Realize the operation of clearing, adding one or subtracting one.Clkt3 It can be generated by an internal clock source or an external clock source. When the count value of the counterTCNT3 Reach the maximum value (equal to the maximum value 0xFFFF Or fixed value or output compare register OCR3A Or input capture register ICR3,defined as TOP, The maximum value is defined as MAX To show the difference), the counter will be cleared or decremented by one. When the count value of the counterTCNT3 Reached the minimum value (equal to 0x0000,defined as BOTTOM), the counter will increment by one. When the count value of the counterTCNT3 Arrivals OCR3A or OCR3B or OCR3C When a comparison match occurs, it will be cleared or set to output a comparison signal OC3A or OC3B or OC3CTo produce PWMWaveform. When the input capture function is turned on, the counter is triggered to start or stop counting,ICR3 The register will record the count value in the trigger period of the capture signal.

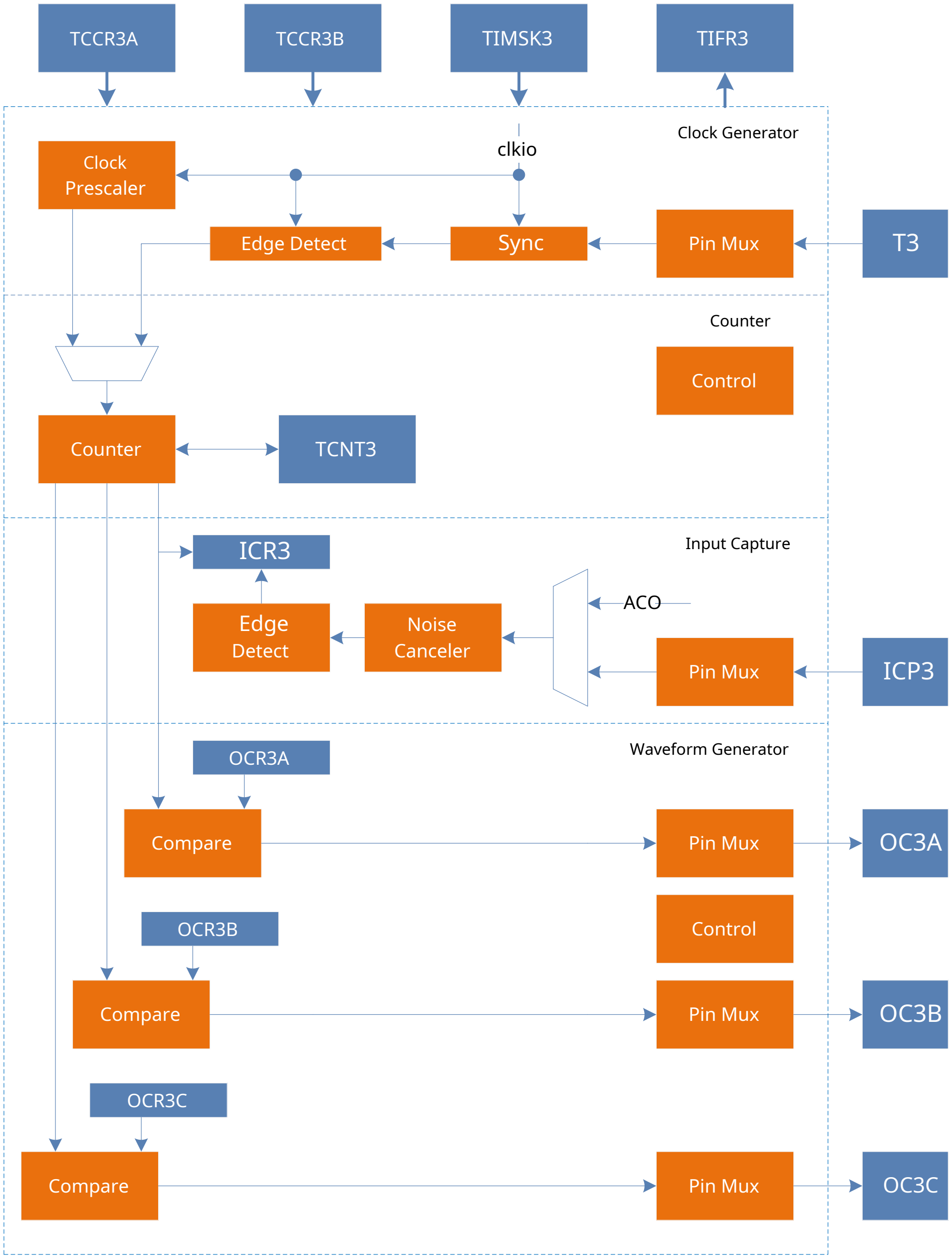


Figure 6 TC3 Structure chart

Operating mode

Timer counter 1 There are six different working modes, including normal mode (Normal), cleared on comparison match (CTC) Mode, fast pulse width modulation (FPWM) Mode, phase correction pulse width modulation (PCPWM) Mode, phase frequency correction pulse width modulation (PFCPWM) Mode, and input capture (ICP)mode. Control bit by waveform generation mode WGM3[3:0]To choose. The six modes are described in detail below. Since there are three independent output comparison units, use "A","B"with"C"To indicate, in lowercase"X"To represent these two output comparison unit channels.

Normal mode

Normal mode is the simplest working mode of the timer counter. At this time, the waveform generation mode control bitWGM3[3:0]=0, The maximum value of the count TOP for MAX(0xFFFF). In this mode, the counting method increases by one for each counting clock. When the counter reachesTOP Back after overflow BOTTOM Start accumulation again. Count valueTCNT3 The timer counter overflow flag is set in the same counting clock that becomes zero TOV3. In this modeTOV3 The sign is like the first 17 The counting bit is only set but not cleared. Overflow interrupt service routine will be automatically clearedTOV3 Mark, the software can use it to improve the resolution of the timer counter. There is no special situation to consider in the normal mode, and a new count value can be written at any time.

Set up OC3x The output comparison signal can only be obtained when the data direction register of the pin is output OC3x 的waveform. when COM3x=1 When a comparison match occurs, it will be flipped OC3x Signal, the frequency of the waveform in this case can be calculated with the following formula:

$$f_{OC3xnormal} = f_{sys/}(2*N*65536)$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

The output compare unit can be used to generate interrupts, but it is not recommended to use interrupts in normal mode, which will take up too much CPU time.

CTC mode

Set up WGM3[3:0]=4 or 12 Time counter 1 enter CTC mode. whenWGM3[3]=0 , The maximum value of the count TOP for OCR3A, whenWGM3[3]=1 , The maximum value of the count TOP for ICR3. The following isWGM3[3:0]=4 As an example to describe CTC In this mode, the counting method is incremented by one for each counting clock. When the counter value TCNT3 equal TOP The hour counter is cleared. This mode allows users to easily control the frequency of the compare match output, and also simplifies the operation of counting external events.

When the counter arrives TOP=OCR3A When, output compare match flag OCF3A Is set when the counter reaches TOP=ICR3 When, output compare match flag ICF3 Is set, an interrupt will be generated when the corresponding interrupt enable bit is set. Can be updated in the interrupt service routineOCR3A register. In this modeOCR3A Double buffering is not used. Be careful when updating the maximum value to close to the minimum value when the counter is working with no prescaler or a very low prescaler. If writeOCR3A Is less than the value at the time TCNT3 Value, the counter will lose a comparison match. Before the next comparison match occurs, the counter has to count toMAX, And then from BOTTOMStart counting to OCR3A. Like normal mode, the count value returns to0x0 Set in the count clock TOV3 Sign.

Set up OC3x The output comparison signal can only be obtained when the data direction register of the pin is output OC3x 的waveform. The frequency of the waveform can be calculated with the following formula:

$$f_{OC3xctc} = f_{sys/}(2*N*(1+OCR3A))$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

It can be seen from the formula that when setting OCR3A for 0x0 And without prescaler, the maximum frequency that can be obtained is $f_{sys/}2$ The output waveform.

whenWGM3[3:0]=12 Time andWGM3[3:0]=4 Similar, just put OCR3A Related to ICR3 That's it.

fast PWMmode

Set upWGM3[3:0]=5,6,7,14 or 15 Time counter 1 Enter fast PWMMode, maximum count TOP Respectively 0xFF,0x1FF,0x3FF,ICR3 or OCR3A, Can be used to generate high frequency PWMWaveform. fastPWM

Mode and others PWM The mode is different in that it is a one-way operation. Counter fromBOTTOM Accumulate to TOP Back again BOTTOM Recount. When the count valueTCNT3 Arrivals TOP or BOTTOM When, output the comparison signal OC3x Will be set or cleared, depending on the comparison output mode COM3 See the register description for details. Due to the one-way operation, fast PWMThe operating frequency of the mode is phase correction with bidirectional operationPWMTwice the pattern. High frequency characteristics make fastPWM Mode is suitable for power regulation, rectification and DAC application. High-frequency signals can reduce the size of external components (inductance, capacitance, etc.), thereby reducing system costs.

When the count value reaches TOP Time, timer counter overflow flag TOV3 Will be set, and the value of the compare buffer will be updated to the compare value. If the interrupt is enabled, it can be updated in the interrupt service routineOCR3A register.

Set up OC3x The output comparison signal can only be obtained when the data direction register of the pin is output OC3x 的waveform. The frequency of the waveform can be calculated with the following formula:

$f_{OC3xfpwm} = f_{sys/(}N*(1+TOP))$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

when TCNT3 with OCR3x When a comparison match occurs, the waveform generator is set (cleared)OC3x Signal when TCNT3 When cleared, the waveform generator is cleared (set)OC3x Signal to generate PWMwave. thusOCR3x Extreme values will produce special PWM Waveform. whenOCR3x Set as 0x00 When the output PWM For every (1+TOP)There is a narrow spike in each count clock. whenOCR3x Set as TOP When, the output waveform is a continuous high level or low level. If you use OCR3A As TOP And set COM3A=1, Output comparison signal OC3A Will produce a duty cycle of 50%of PWM wave.

Phase correction PWMmode

When setWGM3[3:0]=1,2,3,10 or 11 Time counter 1 Enter phase correction PWMMode, the maximum value of the count TOP Respectively 0xFF,0x1FF,0x3FF,ICR3 or OCR3A. The counter adopts two-way operation, byBOTTOM Increment to TOP, And then decrease to BOTTOM, Repeat this operation again. Count arrivalTOP with BOTTOM Change the counting direction every time, the count value is in TOP or BOTTOMThere is only one counting clock on each. In the process of increment or decrement, the count value TCNT3 versus OCR3x When matching, output comparison signal OC3x Will be cleared or set, depending on the comparison output mode COM3 setting. Compared with unidirectional operation, the maximum frequency that can be obtained by bidirectional operation is smaller, but its excellent symmetry is more suitable for motor control.

Phase correction PWM Mode, when the count reaches BOTTOM Set when TOV3 Sign when the count arrives TOP Update the value of the comparison buffer to the comparison value at the time. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routine OCR3x Memory.

Set up OC3x The output comparison signal can only be obtained when the data direction register of the pin is output OC3x Waveform. The frequency of the waveform can be calculated with the following formula:

$f_{OC3xcpcpwm} = f_{sys/(}N*TOP*2)$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

During the count-up process, when TCNT3 versus OCR3x When there is a match, the waveform generator is cleared (set)OC3x signal. In the process of counting down, whenTCNT3 versus OCR3x When a match, the waveform generator is set (cleared)OC3x signal. thusOCR3x The extreme value of will produce a special PWM wave. whenOCR3x Set as TOP or BOTTOM Time,OC3x Signal input

The output will always remain low or high. If you useOCR3A As TOP And set COM3A=1, Output comparison signal OC3A Will produce a duty cycle of 50%of PWMwave.

In order to ensure the output PWMWave in BOTTOM Symmetry on both sides, when there is no comparison match, there are two cases will be flipped OC3x signal. The first situation is whenOCR3x The value is determined by TOP When changing to other data. whenOCR3x for TOP, The count reaches TOP Time,OC3x The output of is the same as the result of the comparison match when counting in descending order, that is, keep OC3x constant. At this time, the comparison value will be updated to the new oneOCR3x Value (not TOP),OC3x The value of will be maintained until a comparison match occurs during ascending counting and it is reversed. at this timeOC3x The signal is not symmetrical about the minimum value, so it needs to be TCNT3 Flip when reaching maximum OC3x Signal, that is, there is no rollover when a comparison match occurs OC3x The first case of the signal. The second situation is whenTCNT3 From OCR3x When a high value starts to count, a comparison match will be lost, which will cause asymmetry. Also need to flipOC3x The signal to achieve symmetry on both sides of the minimum.

Phase frequency correction PWMmode

When setWGM3[3:0]=8 or 9 Time counter 1 Enter phase frequency correction PWMMode, the maximum value of the count TOP Respectively ICR3 or OCR3A. The counter adopts two-way operation, byBOTTOM Increment to TOP, And then decrease to BOTTOM, Repeat this operation again. Count arrivalTOP with BOTTOM Change the counting direction every time, the count value is in TOP or BOTTOM There is only one counting clock on each. In the process of increment or decrement, the count valueTCNT3 versus OCR3x When matching, output comparison signal OC3x Will be cleared or set, depending on the comparison output mode COM3 setting. Compared with unidirectional operation, the maximum frequency that can be obtained by bidirectional operation is smaller, but its excellent symmetry is more suitable for motor control.

Phase frequency correction PWM Mode, when the count reaches BOTTOM Set when TOV3 Mark and update the value of the comparison buffer to the comparison value. The time to update the comparison value is the phase frequency correction PWM Mode and phase correction PWM The biggest difference in the model. If the interrupt is enabled, the compare buffer can be updated in the interrupt service routineOCR3x Memory. whenCPU change TOP Value is OCR3A or ICR3 Value, you must ensure that the new TOP The value is not less than the one already in use TOP Value, otherwise the comparison match will not happen again.

Set up OC3x The output comparison signal can only be obtained when the data direction register of the pin is output OC3x Waveform. The frequency of the waveform can be calculated with the following formula:

$$f_{OC3xcpfcpwm} = f_{sys/(}N*TOP*2)$$

among them,N Represents the prescaler factor (1,8,64,256 or 1024) .

During the count-up process, when TCNT3 versus OCR3x When there is a match, the waveform generator is cleared (set)OC3x signal. In the process of counting down, whenTCNT3 versus OCR3x When a match, the waveform generator is set (cleared)OC3x signal. thusOCR3x The extreme value of will produce a special PWM wave. whenOCR3x Set as TOP or BOTTOM Time,OC3x The signal output will always remain low or high. If you useOCR3A As TOP And set COM3A=1, Output comparison signal OC3A Will produce a duty cycle of 50%of PWMwave.

because OCR3x The register is in BOTTOM Updated all the time, so TOP The count length of the ascending order and the descending order is the same on both sides of the value, and a symmetrical waveform with the correct frequency and phase is generated.

When using fixed TOP Value, it's best to use ICR3 Register as TOP Value, i.e. settingWGM3[3:0]=8,at this time OCR3A Registers only need to be used to generate PWMOutput. If you want to produce frequency changesPWMWave, must be changed TOP value, OCR3A The double-buffering characteristics of the "will be more suitable for this application.

Input capture mode

Input capture is used to capture external events and give them a time stamp to indicate the moment when this event occurs. It can be performed in the previous counting mode, but it is necessary to remove the use ICR3 Value as count TOP Value waveform generation mode.

The trigger signal of an external event is determined by the pin ICP3 The input can also be realized by an analog comparator unit. When the pin ICP3 The logic level on the change, or the output of the analog comparator ACO The level changes, and this level change is captured by the input capture unit, the input capture is triggered, at this time 16 Bit count TCNT3 Data is copied to the input capture register ICR3, And enter the capture flag at the same time ICF3 Set if ICIE1 Bit is"1", The input capture flag will generate an input capture interrupt.

By setting analog comparison control and status registerACSRAnalog compare input capture control bitACICTo select the input capture trigger source ICP3 or ACO. It should be noted that changing the trigger source may cause an input capture, so after changing the trigger source, you mustICF3 Perform a clear operation to avoid false results.

The input capture signal is sent to the edge detector after an optional noise suppressor, and the control bit is selected according to the input capture ICES1 To see if the detected edge meets the trigger condition. The noise suppressor is a simple digital filter that performs4 Sub-sampling, only when 4 When the sub-sampled values are equal, its output will be sent to the edge detector. Noise suppressor by TCCR3B Register ICNC1 The bit controls its enabling or disabling.

When using the input capture function, when ICF3 After being set, it should be read as early as possible ICR3 The value of the register, because after the next capture event occurs ICR3 The value of will be updated. It is recommended to enable the input capture interrupt. In any input capture mode, it is not recommended to change the count during operationTOP value.

The time stamp captured by the input can be used to calculate the frequency, duty cycle, and other characteristics of the signal, as well as to create a log for trigger events. When measuring the duty cycle of an external signal, it is required to change the trigger edge after each capture, so readICR3 The signal edge of the trigger must be changed as soon as possible after the value.

PWMAutomatic shutdown and restart of output

When set TCCR3C Register DOC3x When the bit is high,PWMThe automatic shutdown function of the output will be enabled. When the trigger condition is met, the hardware will clear the corresponding COM3x Bit, will PWM output signal OC3x Disconnect from its output pin and switch to general purpose IO Output, achieve PWM The output is automatically turned off. At this time, the state of the output pin can beIO To control the output of the port.

PWMAfter the automatic shutdown of the output is enabled, the trigger conditions need to be set. TCCR3D Register DSX3n Bit to select the trigger source. Trigger sources include analog comparator interrupt, external interrupt, pin level change interrupt and timer overflow interrupt. For details, please refer toTCCR3D Register description. When one or some trigger sources are selected as the trigger condition, the hardware will clear these interrupt flag bits at the same time as they are set.COM3x Bit to close PWMOutput.

Close when a trigger event occurs PWMAfter the output, the timer module does not have the corresponding interrupt flag bit, and the software needs to know the trigger condition and trigger event by reading the interrupt flag bit of the trigger source.

when PWMWhen the output is automatically turned off and the output needs to be restarted again, the software only needs to be reset COM3x Bit to switch OC3x The signal is output to the corresponding pin. It should be noted that after the automatic shutdown occurs, the timer does not stop working.

OC3x The status of the signal is also constantly being updated. Software can set after timer overflow or compare match COM3x Bit to output OC3x Signal so that you can get a clear PWMOutput status.

Dead time control

Set up DTEN3 Bit is"1"When the function of inserting the dead time is enabled,OC3A with OC3B The output waveform will be in B The set dead time is inserted on the basis of the waveform generated by the channel comparison output. The length of the time is DTR3 The time value corresponding to the count clock number of the register. As shown below,OC3A with OC3B The dead time insertion is based on the channel B The comparison output waveform is the reference. whenCOM3A with COM3B Same as"2"or"3"Time,OC3A The polarity of the waveform and OC3B The polarity of the waveform is the same, when COM3A with COM3B Respectively"2"or"3"Time,OC3A The waveform and OC3B The polarity of the waveform is opposite.

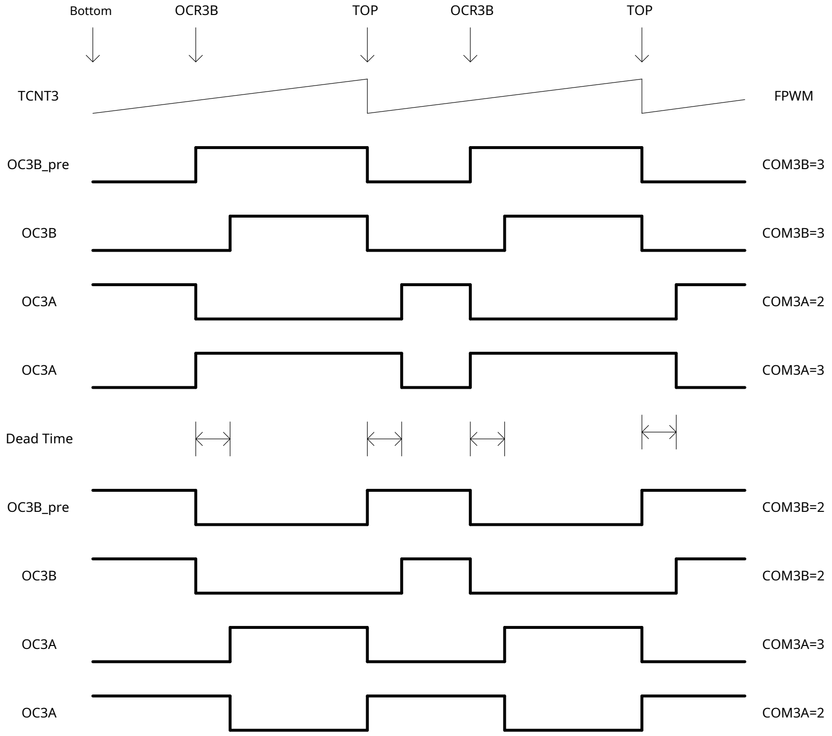


Figure 7 FPWMMode TC3 Dead time control

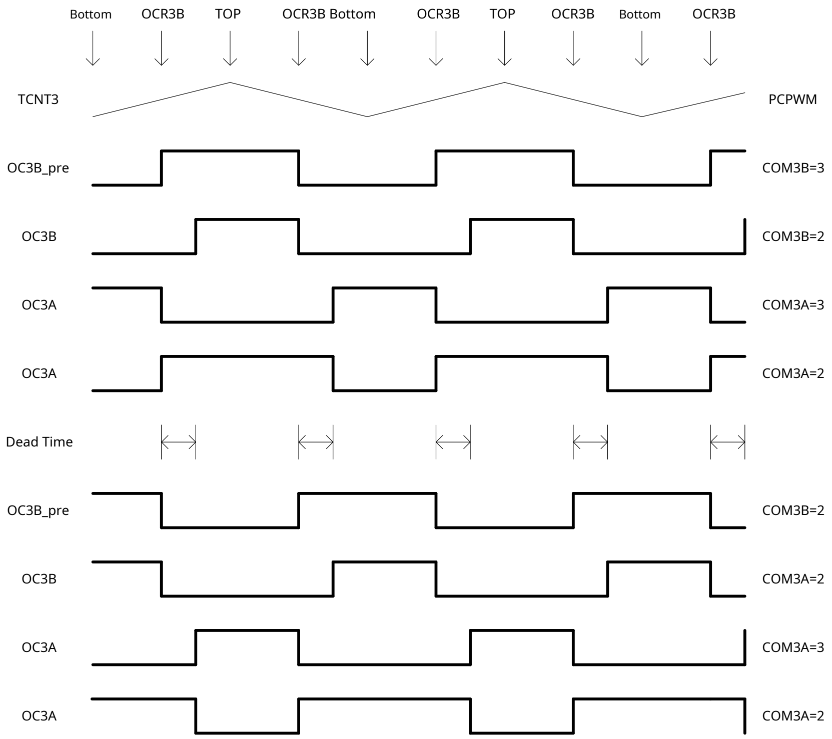


Figure 8 PCPWMMode TC3 Dead time control

Set up DTEN3 Bit is"0"Time, the function of inserting the dead time is disabled,OC3A with OC3B The output waveform of each

Compare the waveforms generated by the output.

Register definition

TC3 Register list

register	address	Defaults	description
TCCR3A	0x90	0x00	TC3 Control register A
TCCR3B	0x91	0x00	TC3 Control register B
TCCR3C	0x92	0x00	TC3 Control register C
TCCR3D	0x93	0x00	TC3 Control register D
TCNT3L	0x94	0x00	TC3 Count value register low byte
TCNT3H	0x95	0x00	TC3 Count value register high byte
ICR3L	0x96	0x00	TC3 Input capture register low byte
ICR3H	0x97	0x00	TC3 Input capture register high byte
OCR3AL	0x98	0x00	TC3 Output compare register A Low byte
OCR3AH	0x99	0x00	TC3 Output compare register A High byte
OCR3BL	0x9A	0x00	TC3 Output compare register B Low byte
OCR3BH	0x9B	0x00	TC3 Output compare register B High byte
DTR3L	0x9C	0x00	TC3 Dead time register low byte
DTR3H	0x9D	0x00	TC3 Dead time register high byte
OCR3CL	0x9E	0x00	TC3 Output compare register C Low byte

OCR3CH	0x9F	0x00	TC3 Output compare register C High byte
TIMSK3	0x71	0x00	Timer counter interrupt mask register
TIFR3	0x38	0x00	Timer counter interrupt flag register

TCCR3A–TC3 Control register A

TCCR3A -TC3 Control register A								
address: 0x90					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	COM3A1	COM3A0	COM3B1	COM3B0	COM3C1	COM3C0	WGM31	WGM30
R/W	R/W	R/W	R/W	R/W	WWR/W			R/W

Bit	Name	description
7	COM3A1	Compare match output A Mode control high. COM3A1 with COM3A0 composition COM3A[1:0]To control the output comparison waveform OC3A. in case COM3A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC3A Pin, but The data direction register of this pin must be set high to output this waveform. In different working modes,COM3A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
6	COM3A0	Compare match output A Mode control low. COM3A1 with COM3A0 composition COM3A[1:0]To control the output comparison waveform OC3A. in case COM3A of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC3A Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM3A The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
5	COM3B1	Compare match output B Mode control high. COM3B1 with COM3B0 composition COM3B[1:0]To control the output comparison waveform OC3B. in case COM3B of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC3B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM3B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
4	COM3B0	Compare match output B Mode control low. COM3B1 with COM3B0 composition COM3B[1:0]To control the output comparison waveform OC3B. in case COM3B of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC3B Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM3B The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
3	COM3C1	Compare match output C Mode control high. COM3C1 with COM3C0 composition COM3C[1:0]To control the output comparison waveform OC3C. in case COM3C of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC3C Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM3C The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
2	COM3C0	Compare match output C Mode control low. COM3C1 with COM3C0 composition COM3C[1:0]To control the output comparison waveform OC3C. in case

		COM3C of 1 Bit or 2 Bits are set, the output comparison waveform occupies OC3C Pin, but the data direction register of this pin must be set high to output this waveform. In different working modes,COM3C The control of the output comparison waveform is also different, see the description of the comparison output mode control table for details.
1	WGM31	The waveform generation mode controls the next low bit. WGM31 with WGM33, WGM32, WGM30 Form the waveform generation mode control together WGM3[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
0	WGM30	The waveform generation mode controls the lowest bit. WGM30 with WGM33, WGM32, WGM31 Form the waveform generation mode control together WGM3[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.

The following table is not PWMMode (i.e. normal mode and CTC Mode), the comparison output mode controls the output comparison waveform.

non- PWMComparison output mode control in mode

COM3x[1:0]	description
0	OC3x Disconnect, general IO Flip when the port
1	operation compares match OC3x signal
2	Clear on comparison match OC3x signal
3	Set on compare match OC3x signal

The following table is fast PWMThe comparison output mode in the mode controls the output comparison waveform.

fast PWMComparison output mode control in mode

COM3x[1:0]	description
0	OC3x Disconnect, general IO Mouth operation
1	WGM3 for 15 Time: Flip when compare matches OC3A signal, OC3B disconnect WGM3 For other values:OC3x Disconnect, general IO Cleared when port operation compare match
2	OC3x Signal, set when the maximum value matches OC3x Set when the signal compare matches OC3x
3	Signal, cleared when the maximum value matches OC3x signal

The following table shows the control of the output comparison waveform in the comparison output mode in the phase correction mode.

Phase correction and phase frequency correction PWMComparison output mode control in mode

COM3x[1:0]	description
0	OC3x Disconnect, general IO Mouth operation
1	WGM3 for 9 or 11 Time: Flip when compare matches OC3A signal, OC3B disconnect WGM3 For other values:OC3x Disconnect, general IO Compare match cleared under ascending count
2	of port operation OC3x Signal, compare match set in descending count OC3x signal
3	Compare match set in ascending count OC3x Signal, compare match cleared under descending count OC3x signal

TCCR3B–TC3 Control register B

TCCR3B -TC3 Control register B								
address: 0x91					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	ICNC3	ICES3	-	WGM33	WGM32	CS32	CS31	CS30
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	ICNC3	Input capture noise suppressor enable control bit. When set ICNC3 Bit is"1"When the input is enabled to capture the noise suppressor, the external pin ICP3 Input is filtered, continuous 4 The input signal is valid only when the sampling values are equal. This function delays the input capture 4 Clock cycles. When set ICNC3 Bit is"0"When the input capture noise suppressor is prohibited, the external pin ICP3 The input is directly valid.
6	ICES3	Input capture trigger edge selection control bit. When set ICES3 Bit is"1"Select the rising edge of the level to trigger the input capture; when set ICES3 Bit is"0"When, select the falling edge of the level to trigger the input capture. When an event is captured, the value of the counter is copied toICR3 Register, and set the input capture flag at the same time ICF3. If the interrupt is enabled, an input capture interrupt is generated. Reserved.
5	-	
4	WGM33	The waveform generation mode controls the high bit. WGM33 with WGM32, WGM31, WGM30 Form the waveform generation mode control together WGM3[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
3	WGM32	The waveform generation mode controls the second highest bit. WGM32 with WGM33, WGM31, WGM30 Form the waveform generation mode control together WGM3[3:0], Control the counting method and waveform generation method of the counter, see the description of the waveform generation mode table for details.
2	CS32	The clock selection controls the high bit. Used to select the timer counter 3 Clock source.
1	CS31	Clock selection control center position. Used to select the timer counter 3 Clock source.
0	CS30	Clock selection control low bit. Used to select the timer counter 3 Clock source.

CS3[2:0]	description
0	No clock source, stop counting
1	clk_{sys}
2	$clk_{sys/}$8, From the prescaler
3	$clk_{sys/}$64, From the prescaler
4	$clk_{sys/}$256, From the prescaler
5	$clk_{sys/}$1024, From the external clock of the
6	prescaler T3 Pin, falling edge trigger
7	External clock T3 Pin, rising edge trigger

The following table shows the waveform generation mode control.

Table 5 Waveform generation mode control

WGM3[3:0]	Operating mode	TOP value	Update OCR1A time	Position TOV3 time
0	Normal	0xFFFF	immediately	MAX
1	8 Bit PCPWM	0x00FF	TOP	BOTTOM
2	9 Bit PCPWM	0x01FF	TOP	BOTTOM
3	10 Bit PCPWM	0x03FF	TOP	BOTTOM
4	CTC	OCR3A	immediately	MAX
5	8 Bit FPWM	0x00FF	BOTTOM	TOP
6	9 Bit FPWM	0x01FF	BOTTOM	TOP
7	10 Bit FPWM	0x03FF	BOTTOM	TOP
8	PFCPWM	ICR3	BOTTOM	BOTTOM
9	PFCPWM	OCR3A	BOTTOM	BOTTOM
10	PCPWM	ICR3	TOP	BOTTOM
11	PCPWM	OCR3A	TOP	BOTTOM
12	CTC	ICR3	immediately	MAX
13	Keep	-	-	-
14	FPWM	ICR3	TOP	TOP
15	FPWM	OCR3A	TOP	TOP

TCCR3C–TC3 Control register C

TCCR3C -TC3 Control register C								
address: 0x92					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	FOC3A	FOC3B	DOC3B	DOC3A	DTEN3	-	DOC3C	FOC3C
R/W	W	W	-	-	-	-	-	-

Bit	Name	description
7	FOC3A	Force output comparison A. Work in non PWM Mode, you can compulsorily output the comparison bit FOC3A write"1"Way to generate comparison matches. Force compare match will not be setOCF3A Flag, it will not reload or clear the timer, but the output pin OC3A Will be followed COM3A The settings are updated accordingly, just as if a comparison match actually occurred. Work on PWMMode, write TCCR3A It must be cleared when registering. Read FOC3A The return value of is always zero.
6	FOC3B	Force output comparison B. Work in non PWM Mode, you can compulsorily output the comparison bit FOC3B write"1"Way to generate comparison matches. Force compare match will not be setOCF3B Flag, it will not reload or clear the timer, but the output pin OC3B Will be followed COM3B The settings are updated accordingly, just as if a comparison match actually occurred. Work on PWMMode, write TCCR3A It must be cleared when registering. Read FOC3B The return value of is always zero.
5	DOC3B	Disable output comparison B Enable control bit.

		when DOC3B When the bit is high, the hardware prohibits the output comparison B Is enabled, after satisfying the condition of prohibiting output,COM3B The bit will be cleared and the output pin OC3B Disconnected, the pin becomes general purpose IO operating. whenDOC3B When the bit is low, the hardware prohibits output comparison B Function is invalid. Disable output comparisonA Enable control bit.
4	DOC3A	when DOC3A When the bit is high, the hardware prohibits the output comparison A Is enabled, after satisfying the condition of prohibiting output,COM3A The bit will be cleared and the output pin OC3A Disconnected, the pin becomes general purpose IO operating. whenDOC3A When the bit is low, the hardware prohibits output comparison A Function is invalid. Dead time enable control bit.
3	DTEN3	when DTEN3 When the bit is high, the dead time is enabled,OC3A with OC3B Become complementary output, and press DTR3L with DTR3H It is set to insert the dead time. whenDTEN3 When the bit is low, the dead time is prohibited.OC3A with OC3B All are single output.
2	-	
1	DOC3C	Disable output comparison C Enable control bit. when DOC3C When the bit is high, the hardware prohibits the output comparison C Is enabled, after satisfying the condition of prohibiting output,COM3C The bit will be cleared and the output pin OC3C Disconnected, the pin becomes general purpose IO operating. whenDOC3C When the bit is low, the hardware prohibits output comparison C Function is invalid. Force output comparisonC.
0	FOC3C	Work in non PWM Mode, you can compulsorily output the comparison bit FOC3C write"1"Way to generate comparison matches. Force compare match will not be setOCF3C Flag, it will not reload or clear the timer, but the output pin OC3C Will be followed COM3C The settings are updated accordingly, just as if a comparison match actually occurred. Work on PWMMode, write TCCR3A It must be cleared when registering. Read FOC3C The return value of is always zero.

TCCR3D–TC3 Control register D

TCCR3D -TC3 Control register D								
address: 0x93					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	DSX37	DSX36	DSX35	DSX34	-	-	DSX31	DSX30
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W

Bit	Name	description
7	DSX37	TC3 Trigger source selection control enable 7 Bit. When setDSX37 Bit is"1"Time,TC0 Overflow is used to turn off the output comparison signal waveform OC3x The trigger source of is enabled. whenDOC3x Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC3x Waveform output. When setDSX37 Bit is"0"Time,TC0 Overflow is used to turn off the output comparison signal waveform OC3x The trigger source of is disabled.
6	DSX36	TC3 Trigger source selection control enable 6 Bit. When setDSX36 Bit is"1"Time,TC2 Overflow is used to turn off the output comparison signal waveform OC3x The trigger source of is enabled. whenDOC3x Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC3x Waveform output. When setDSX36 Bit is"0"Time,TC2 Overflow is used to turn off the output comparison signal waveform OC3x The trigger source of is disabled.

5	DSX35	TC3 Trigger source selection control enable 5 Bit. When setDSX35 Bit is"1"When the pin level changes 1 As a comparison signal waveform for turning off the output OC3x The trigger source of is enabled. whenDOC3x Bit is "1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC3x Waveform output. When setDSX35 Bit is"0"When the pin level changes 1 As a comparison signal waveform for turning off the output OC3x The trigger source of is disabled.
4	DSX34	TC3 Trigger source selection control enable 4 Bit. When setDSX34 Bit is"1"External interrupt 1 As a comparison signal waveform for turning off the output OC3x The trigger source of is enabled. whenDOC3x Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC3x Waveform output. When setDSX34 Bit is"0"External interrupt 1 As a comparison signal waveform for turning off the output OC3x The trigger source of is disabled.
3:2	-	Reserved.
1	DSX31	TC3 Trigger source selection control enable 1 Bit. When setDSX31 Bit is"1"When, analog comparator 1 As a comparison signal waveform for turning off the output OC3x The trigger source of is enabled. whenDOC3x Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC3x Waveform output. When setDSX31 Bit is"0"When, analog comparator 1 As a comparison signal waveform for turning off the output OC3x The trigger source of is disabled.
0	DSX30	TC3 Trigger source selection control enable 0 Bit. When setDSX30 Bit is"1"When, analog comparator 0 As a comparison signal waveform for turning off the output OC3x The trigger source of is enabled. whenDOC3x Bit is"1"When, the rising edge of the interrupt flag register bit of the selected trigger source will be automatically turned off OC3x Waveform output. When setDSX30 Bit is"0"When, analog comparator 0 As a comparison signal waveform for turning off the output OC3x The trigger source of is disabled.

The following table shows the selection and control of the trigger source of the waveform output.

shut down OC3x Trigger source selection control of waveform output

DOC3x	DSX3n=1	Trigger source	description
0	-	-	DOC3x Bit is"0", The trigger source is turned off and the waveform output function is disabled
1	0	Analog comparator 0	ACIF0 The rising edge will turn off OC3x Wave output
1	1	Analog comparator 1	ACIF1 The rising edge will turn off OC3x Wave output
1	4	External Interrupt 1	INTF1 The rising edge will turn off OC3x Wave output
1	5	Pin level change 1	PCIF1 The rising edge will turn off OC3x Wave output
1	6	TC2 overflow	TOV2 The rising edge will turn off OC3x Wave output
1	7	TC0 overflow	TOV0 The rising edge will turn off OC3x Wave output

note:

2)DSX3n=1 Means TCCR1D Register n Bit is 1 When, each register bit can be set at the same time.

TCNT3L–TC3 Counter register low byte

TCNT3L -TC3 Count value register low byte								
address: 0x94					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0

Name	TCNT3L7	TCNT3L6	TCNT3L5	TCNT3L4	TCNT3L3	TCNT3L2	TCNT3L1	TCNT3L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	TCNT3L	TC3 The low byte of the count value. TCNT3H with TCNT3L Combined together to form TCNT3,by TCNT3 The register can directly 16 The bit count value is accessed for read and write. Read and write16The bit register requires two operations. write16BitTCNT3Should be written firstTCNT3H. read16 Bit TCNT3 Should be read first TCNT3L. CPU Correct TCNT3 The write operation of the register will prevent the comparison match from occurring in the next timer clock cycle, even if the timer has stopped. This allows initializationTCNT3 The value of the register is the same as OCR3x The values of are consistent without causing interruption. If writeTCNT3 Equal to or bypassed OCR3x When the value is set, the comparison match will be lost, resulting in incorrect waveforms. The timer stops counting when the clock source is not selected, but CPU Still accessible TCNT3.CPU Write counter ratio to clear or add The priority of the minus operation is high.

TCNT3H–TC3 Counter register high byte

TCNT3H -TC3 Count value register high byte								
address: 0x95					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	TCNT3H7	TCNT3H6	TCNT3H5	TCNT3H4	TCNT3H3	TCNT3H2	TCNT3H1	TCNT3H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	TCNT3H	TC3 The high byte of the count value. TCNT3H with TCNT3L Combined together to form TCNT3,by TCNT3 The register can directly 16 The bit count value is accessed for read and write. Read and write16 The bit register requires two operations. write16 Bit TCNT3 Should be written first TCNT3H. read16 Bit TCNT3 Should be read first TCNT3L. CPU Correct TCNT3 The write operation of the register will prevent the comparison match from occurring in the next timer clock cycle, even if the timer has stopped. This allows initializationTCNT3 The value of the register is the same as OCR3x The values of are consistent without causing interruption. If writeTCNT3 Equal to or bypassed OCR3x Value, the comparison match will be lost, resulting in incorrect waveform Health results. The timer stops counting when the clock source is not selected, but CPU Still accessible TCNT3.CPU Write counter ratio Clear or add The priority of the minus operation is high.

ICR3L–TC3 Capture register low byte

ICR3L -TC3 Input capture register low byte								
address: 0x96					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	ICR3L7	ICR3L6	ICR3L5	ICR3L4	ICR3L3	ICR3L2	ICR3L1	ICR3L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	ICR3L	TC3 Enter the low byte of the captured value. ICR3H with ICR3L Combined together to form 16 Bit ICR3. Read and write16 The bit register requires two operations. write16 Bit ICR3 Should be written first ICR3H. read16 Bit ICR3 Should be read first ICR3L.

		When the input capture is triggered, the count value TCNT3 Will update and copy to ICR3 In the register.ICR3 Registers can also be used Defined counted TOP value.

ICR3H–TC3 Capture register high byte

ICR3H -TC3 Input capture register high byte								
address: 0x97					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	ICR3H7	ICR3H6	ICR3H5	ICR3H4	ICR3H3	ICR3H2	ICR3H1	ICR3H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	ICR3H	TC3 Enter the high byte of the captured value. ICR3H with ICR3L Combined together to form 16 Bit ICR3. Read and write16 The bit register requires two operations. write16 Bit ICR3 Should be written first ICR3H. read16 Bit ICR3 Should be read first ICR3L. When the input capture is triggered, the count value TCNT3 Will update and copy to ICR3 In the register.ICR3 Registers can also be used Defined counted TOP value.

OCR3AL–TC3 Output compare register A Low byte

OCR3AL -TC3 Output compare register A Low byte								
address: 0x98					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OCR3AL7	OCR3AL6	OCR3AL5	OCR3AL4	OCR3AL3	OCR3AL2	OCR3AL1	OCR3AL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	OCR3AL	Output compare register A The low byte. OCR3AL with OCR3AH Combined together to form 16 Bit OCR3A. Read and write16 The bit register requires two operations. write 16 Bit OCR3A Should be written first OCR3AH. read16 Bit OCR3A Should be read first OCR3AL. OCR3A Uninterruptedly with the counter value TCNT3 Compare. Compare match can be used to generate output compare interrupt, or used toOC3A A waveform is generated on the pin. When using PWMMode,OCR3A The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR3A The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR3A Buffer register, when double buffering function is disabled CPU Visiting is OCR3A itself.

OCR3AH–TC3 Output compare register A High byte

OCR3AH -TC3 Output compare register A High byte								
address: 0x99					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
	OCR3AH7	OCR3AH6	OCR3AH5	OCR3AH4	OCR3AH3	OCR3AH2	OCR3AH1	OCR3AH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	OCR3AH	Output compare register A The high byte.

		OCR3AL with OCR3AH Combined together to form 16 Bit OCR3A. Read and write16 The bit register requires two operations. write 16 Bit OCR3A Should be written first OCR3AH. read16 Bit OCR3A Should be read first OCR3AL. OCR3A Uninterruptedly with the counter value TCNT3 Compare. Compare match can be used to generate output compare interrupt, or used toOC3A A waveform is generated on the pin. When using PWMMode,OCR3A The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR3A The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR3A Buffer register, when double buffering function is disabled CPU Visiting is OCR3A itself.

OCR3BL–TC3 Output compare register B Low byte

OCR3BL -TC3 Output compare register B Low byte								
address: 0x9A					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OCR3BL7	OCR3BL6	OCR3BL5	OCR3BL4	OCR3BL3	OCR3BL2	OCR3BL1	OCR3BL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR3BL	Output compare register B The low byte. OCR3BL with OCR3BH Combined together to form 16 Bit OCR3B. Read and write16 The bit register requires two operations. write 16 Bit OCR3B Should be written first OCR3BH. read16 Bit OCR3B Should be read first OCR3BL. OCR3B Uninterruptedly with the counter value TCNT3 Compare. Compare match can be used to generate output compare interrupt, or used toOC3B A waveform is generated on the pin. When using PWMMode,OCR3B The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR3B The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR3B Buffer register, when double buffering function is disabled CPU Visiting is OCR3B itself.						

OCR3BH–TC3 Output compare register B High byte

OCR3BH -TC3 Output compare register B High byte								
address: 0x9B					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OCR3BH7	OCR3BH6	OCR3BH5	OCR3BH4	OCR3BH3	OCR3BH2	OCR3BH1	OCR3BH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR3BH	Output compare register B The high byte. OCR3BL with OCR3BH Combined together to form 16 Bit OCR3B. Read and write16 The bit register requires two operations. write 16 Bit OCR3B Should be written first OCR3BH. read16 Bit OCR3B Should be read first OCR3BL. OCR3B Uninterruptedly with the counter value TCNT3 Compare. Compare match can be used to generate output compare interrupt, or used toOC3B A waveform is generated on the pin. When using PWMMode,OCR3B The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR3B The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse.						

		When using the double buffering function,CPU Visiting is OCR3B Buffer register, when double buffering function is disabled CPU Visiting is OCR3B itself.

OCR3CL–TC3 Output compare register C Low byte

OCR3CL **-**TC3 Output compare register C Low byte								
address: 0x9E					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OCR3CL7	OCR3CL6	OCR3CL5	OCR3CL4	OCR3CL3	OCR3CL2	OCR3CL1	OCR3CL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR3CL	Output compare register C The low byte. OCR3CL with OCR3CH Combined together to form 16 Bit OCR3C. Read and write16 The bit register requires two operations. write16 Bit OCR3C Should be written first OCR3CH. read16 Bit OCR3C Should be read first OCR3CL. OCR3C Uninterruptedly with the counter value TCNT3 Compare. Compare match can be used to generate output compare interrupt, or used toOC3C A waveform is generated on the pin. When using PWMMode,OCR3C The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR3C The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPUVisiting isOCR3CBuffer register, When double buffering is disabledCPUVisiting isOCR3C itself.							

OCR3CH–TC3 Output compare register C High byte

OCR3CH **-**TC3 Output compare register C High byte								
address: 0x9F					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OCR3CH7	OCR3CH6	OCR3CH5	OCR3CH4	OCR3CH3	OCR3CH2	OCR3CH1	OCR3CH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	OCR3CH	Output compare register C The high byte. OCR3CL with OCR3CH Combined together to form 16 16 Bit OCR3C. Read and write16 The bit register requires two operations. write Bit OCR3C Should be written first OCR3CH. read 16 Bit OCR3C Should be read first OCR3CL. OCR3C Uninterruptedly with the counter value TCNT3 Compare. Compare match can be used to generate output compare interrupt, or used toOC3C A waveform is generated on the pin. When using PWMMode,OCR3C The register uses double-buffered registers. In normal working mode and match clear mode, the double buffering function is prohibited. Double buffering can updateOCR3C The register is synchronized with the maximum or minimum count at all times to prevent asymmetry PWMPulse, eliminating the interference pulse. When using the double buffering function,CPU Visiting is OCR3C Buffer register, when double buffering function is disabled CPU Visiting is OCR3C itself.							

DTR3L–TC3 Dead time register low byte

DTR3L **-**TC3 Dead time register low byte	
address: 0x9C	Defaults: 0x00

Bit	7	6	5	4	3	2	1	0
Name	DTR3L7	DTR3L6	DTR3L5	DTR3L4	DTR3L3	DTR3L2	DTR3L1	DTR3L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	DTR3L	The low byte of the dead time register. when DTEN3 When the bit is high,OC3A with OC3B Is complementary output,OC3A The dead time inserted on the output is determined by DTR3L It is determined by a count clock.						

DTR3H–TC3 Dead time register high byte

DTR3H -TC3 Dead time register high byte								
address: 0x9D					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	DTR3H7	DTR3H6	DTR3H5	DTR3H4	DTR3H3	DTR3H2	DTR3H1	DTR3H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	DTR3H	Dead time register high byte. when DTEN3 When the bit is high,OC3A with OC3B Is complementary output,OC3B The dead time inserted on the output is determined by DTR3H It is determined by a count clock.						

TIMSK3–TC3 Interrupt mask register

TIMSK3 **– TC3 Interrupt mask register**								
address: 0x71					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	-	-	ICIE3	-	OCIE3C	OCIE3B	OCIE3A	TOIE3
R/W	-	-	R/W	-	R/W	R/W	R/W	R/W
Bit	Name	description						
7:6	-	Reserved.						
5	ICIE3	TC3 Input capture interrupt enable control bit. when ICIE3 Bit is"1"Time, and the global interrupt is set,TC3 The input capture interrupt is enabled. When the input capture is triggered, that isTIFR3 of ICF3 The flag is set and an interrupt occurs. whenICIE3 Bit is"0"Time,TC3 The input capture interrupt is disabled. Reserved.						
4	-							
3	OCIE3C	TC3 Output comparison C Match interrupt enable bit. whenOCIE3C Bit is"1", And the global interrupt is set,TC3 Output comparison C Match interrupt is enabled. When a comparison match occurs When TIFR3 in OCF3C When the bit is set, an interrupt is generated. whenOCIE3C Bit is"0"Time,TC3 Output comparison C The match interrupt is disabled.						
2	OCIE3B	TC3 Output comparison B Match interrupt enable bit. whenOCIE3B Bit is"1", And the global interrupt is set,TC3 Output comparison B Match interrupt is enabled. When a comparison match occurs, that isTIFR3 in OCF3B When the bit is set, an interrupt is generated. whenOCIE3B Bit is"0" Time,TC3 Output comparison B The match interrupt is disabled.						
1	OCIE3A	TC3 Output comparison A Match interrupt enable bit. whenOCIE3A Bit is"1", And the global interrupt is set,TC3 Output comparison A Match interrupt is enabled. When a comparison match occurs						

		When TIFR3 in OCF3A When the bit is set, an interrupt is generated. When the match OCIE3A Bit is"0"Time,TC3 Output comparison A interrupt is disabled.
0	TOIE3	TC3 Overflow interrupt enable bit. when TOIE3 Bit is"1", And the global interrupt is set,TC3 The overflow interrupt is enabled. whenTC3 Overflow occurs, i.e. TIFR3 middle TOV3 When the bit is set, an interrupt is generated. whenTOIE3 Bit is"0"Time,TC3 Overflow interrupts are disabled.

TIFR3–TC3 Interrupt flag register

TIFR3 **– TC3 Interrupt flag register**								
address: 0x38					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	-	-	ICF3	-	-	OCF3B	OCF3A	TOV3
R/W	-	-	R/W	-	-	R/W	R/W	R/W

Bit	Name	description
7:6	-	Reserved.
5	ICF3	Input capture flag. When an input capture event occurs,ICF3 The flag is set. whenICR3 Used for counting TOP Value, and the count value reaches TOP Value,ICF3 The flag is set. IfICIE1 for"1"And the global interrupt flag is set, an input capture interrupt will be generated.ICF3 The flag bit will not be cleared automatically, it needs software to ICF3 Bit write"1"Cleared.
4	-	Reserved.
3	OCF3C	Output comparison C Match flag bit. when TCNT3 equal OCR3C When, the comparison unit gives a match signal and sets the comparison flag OCF3C. If the output compare interrupt is enabled at this timeOCIE3C for"1"And the global interrupt flag is set, an output compare interrupt will be generated.OCF3 The flag will not be cleared automatically, software is required OCF3C Bit write"1"Clear this bit.
2	OCF3B	Output comparison B Match flag bit. when TCNT3 equal OCR3B When, the comparison unit gives a match signal and sets the comparison flag OCF3B. If the output compare interrupt is enabled at this timeOCIE3B for"1"And the global interrupt flag is set, an output compare interrupt will be generated.OCF3B The flag bit will not be cleared automatically, it needs software to OCF3B Bit write"1"Cleared.
1	OCF3A	Output comparison A Match flag bit. when TCNT3 equal OCR3A When, the comparison unit gives a match signal and sets the comparison flag OCF3A. If the output compare interrupt is enabled at this timeOCIE3A for"1"And the global interrupt flag is set, an output compare interrupt will be generated.OCF3A The flag bit will not be cleared automatically, it needs software to OCF3A Bit write"1"Cleared.
0	TOV3	Overflow flag. When the counter overflows, the overflow flag is set TOV3. If overflow interrupt is enabled at this timeTOIE3 for"1" And the global interrupt flag is set, an overflow interrupt will be generated. TOV3 The flag bit will not be automatically cleared, it needs software to TOV3 Bit write"1"Cleared.

Synchronous Serial Peripheral Interface (SPI)

Z Full duplex, three-wire synchronous data transmission

Z Master or slave operation

Z Lowest bit or highest bit is transmitted first

Z 7 Programmable bit rate

Z Send end interrupt flag

Z Write conflict flag protection mechanism

Z Can wake up from idle mode

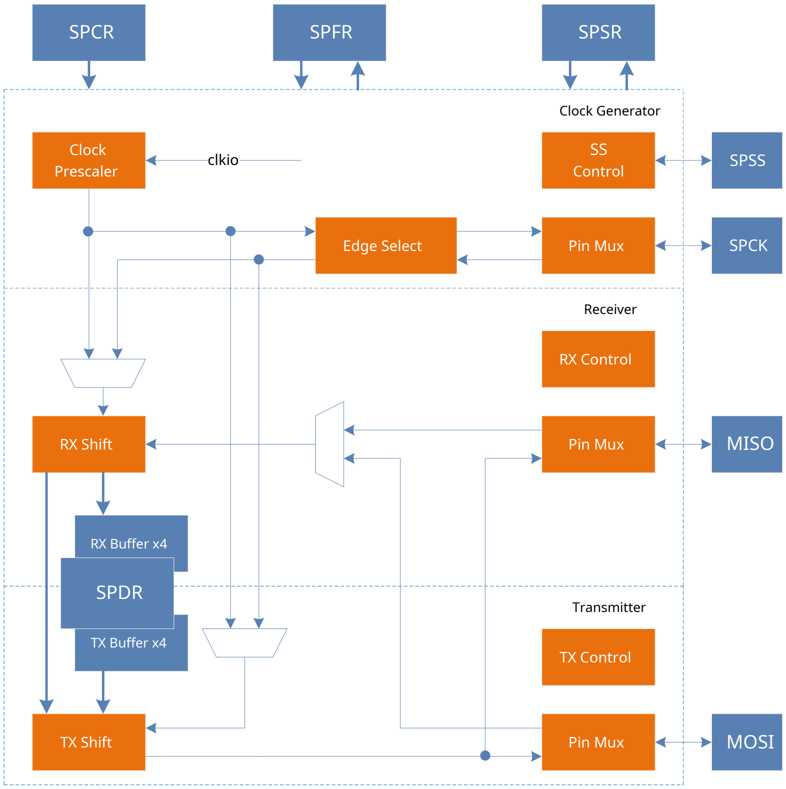
Z Double speed mode when the host is operating

Z Support host two-wire input mode

Z Both input and output 4 Buffer register

Summary

SPI It mainly includes three parts: clock prescaler, clock detector, slave select detector, transmitter and receiver Device.



SPI Structure chart

The control and status registers are shared by these three parts. The clock prescaler only works in the host operating mode, and the frequency division coefficient is selected by the bit rate control bit to generate the corresponding frequency division clock and output toSPCK Pin on. The clock detector only works in the slave operation mode.SPCK The input clock edge on the pin, according to SPI In the data transfer mode, shift operations are performed on the transmit and receive shift registers. Slave selection detector to slave selection signalSPSS To test and get

The status of the transmission controls the operation of the transmitter and receiver. The transmitter consists of a shift register and transmission control logic. The receiver consists of a shift register, four receiving buffers and receiving control logic.

Clock generation

The clock generation logic is divided into a master clock prescaler and a slave clock detector, which work in the master operation and slave operation modes respectively. The clock prescaler selects the frequency division coefficient by the bit rate control bit and the double speed control bit, and generates the corresponding frequency division clock (total7 An optional frequency division coefficient, see the register description for details), output to SPCK The pins provide a clock for communication, and at the same time provide a shift clock for the internal transmit and receive shift registers. Clock detector to input clockSPCK Perform edge detection, according to SPI The data transmission mode shifts the transmitter and receiver. To ensure the correct sampling of the clock signal, SPCK The width of the clock's high level and low level must be greater than 2 System clock cycles.

Send and receive

SPI The module supports simultaneous transmission and reception in single-wire mode, and only supports dual-wire reception from the host in dual-wire mode.

Single wire transmission and reception

SPI The master will need to communicate with the slave selection signal SPSS Pull down to start a transmission process. The master and slave prepare the data to be transmitted, and the master is in the clock signalSPCK To generate clock pulses to exchange data, the data from the host MOSI Move out from MISO Move in, the data from the slave MISO Move out from MOSI Move in, the host will be pulled up after the data is exchanged SPSS The signal can complete the communication.

When configured as a host,SPI Module does not control SPSS Pins must be handled by user software. Software pull downSPSS Pin, select the slave to be communicated, and start the transmission. The software writes the data that needs to be transmittedSPDR The register will start the clock generator, the hardware will generate the communication clock, and the 8 The bit data is shifted out to the slave, and the data of the slave is shifted in at the same time. After shifting one byte of data, stop the clock generator and set the transmission complete flagSPIF. The software can write data to SPDR Register to continue to transmit the next byte, or it can be pulled high SPSS Signal to end the current transmission. The last incoming data will be stored in the receive buffer.

When configured as a slave, as long as SPSS The signal is always high,SPI The module will stay asleep and keep MISO The pins are tri-stated. Software can be updated at this timeSPDR The contents of the register. Even nowSPCK There is an input clock pulse on the pin, SPDR Data will not be removed until SPSS The signal is pulled low. When one byte of data transmission is completed, the hardware sets the transmission complete flagSPIF. At this time, the software can continue to go toSPDR The register writes data, and the last incoming data will be stored in the receive buffer.

SPI The module has only four buffers in the sending direction and four buffers in the receiving direction. When sending data, when the sending buffer is not full (that is, the sending buffer full flag bitWRFULL Bit is low), you can SPDR The register is written. When receiving data, when the receiving buffer is in a non-empty state (that is, the receiving buffer empty flag bitRDEMPT Bit is low), you can access SPDR The register reads the characters that have been received.

Host two-wire reception

SPI The two-wire mode of the module is only valid in the host operating mode, and the difference from the single-wire mode isMOSI with MISO Are used for the host to receive data, each SPCK Simultaneous clock pulse reception 2 Bits of data (MISO Online data is in before,MOSI The data on the line is behind), after receiving two bytes of data, the hardware sets the transmission completion flag SPIF, The data is saved in the receive buffer and shift register. At this time the software must readSPDR Register twice to get the two bytes of data received. It should be noted that although the host does not send data to the slave in the two-wire mode, the software still needs to send data to theSPDR Register write data to start the clock generator to generate communication clock, write once SPDR The register can receive two bytes of data.

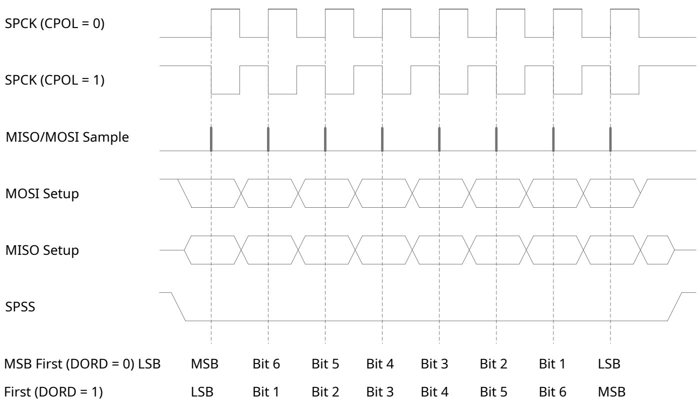
Data pattern

In single-wire mode, relative to serial data,SPI Have 4 Species SPCK The combination of phase and polarity is determined by CPHA with CPOL To control, as shown in the following table.

CPHA with CPOL Select data transfer mode

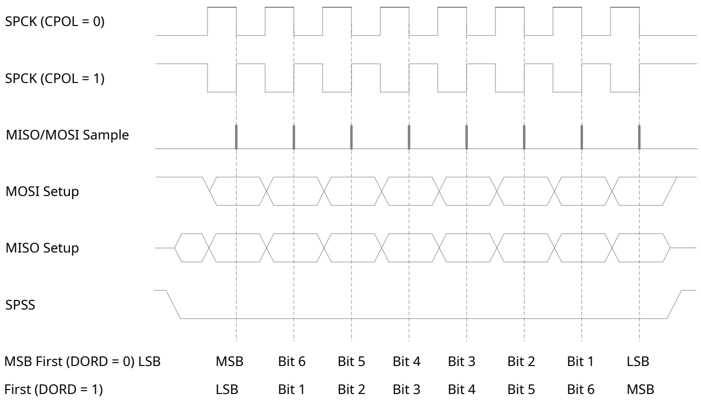
CPOL	CPHA	Starting edge	End edge	SPI mode
0	0	Sampling (rising edge)	Setting (falling edge)	0
0	1	Setting (rising edge)	Sampling (falling edge)	1
1	0	Sampling (falling edge)	Setting (rising edge)	2
1	1	Setting (falling edge)	Sampling (rising edge)	3

when CPHA = 0 The clock edge of data sampling and setting is shown in the figure below:



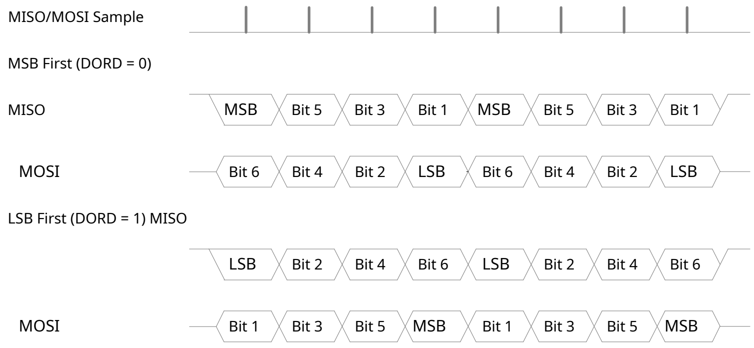
CPHA for"0"Time SPI Data transfer mode

when CPHA = 1 The clock edge of data sampling and setting is shown in the figure below:



CPHA for"1"Time SPI Data transfer mode

In two-line mode,MISO with MISO Both are used as the input of the host, and the time of data sampling is still determined by the data transmission mode. The way is shown in the figure below:


MISO/MOSI Sample
MSB First (DORD = 0)
MISO
MSB Bit 5 Bit 3 Bit 1 MSB Bit 5 Bit 3 Bit 1
MOSI
Bit 6 Bit 4 Bit 2 LSB Bit 6 Bit 4 Bit 2 LSB
LSB First (DORD = 1) MISO
LSB Bit 2 Bit 4 Bit 6 LSB Bit 2 Bit 4 Bit 6
MOSI
Bit 1 Bit 3 Bit 5 MSB Bit 1 Bit 3 Bit 5 MSB

Host mode DUALfor"1"Time SPI Data sampling mode

SPSS Pin function

When configured as a slave, the slave selects the signal SPSS The pin is always used as an input. whenSPSS When the pin remains low,SPI The interface is activated,MISO Pin becomes output pin (software carries out corresponding port configuration), Other pins are inputs. whenSPSS When the pin remains high,SPI The module is reset and no longer receives data.SPSS The pin is very useful for data packet/byte synchronization, which can synchronize the bit counter of the slave and the clock generator of the master. whenSPSS When pulled up,SPI The slave resets the receiving and sending logic immediately, and discards the incomplete data in the shift register.

When configured as a host, the user software can decide SPSS The direction of the pin. IfSPSS Configured as an output, it can be used to drive the slave SPSS Pin. IfSPSS It is configured as an input and must be kept high to ensure the normal operation of the host. When configured as a host andSPSS Pin is input, external circuit pulls low SPSS Pin,SPI The module will think that another master chooses itself as a slave and starts to transmit data. In order to prevent bus conflicts,

SPI The module will perform the following actions:

1. Clear at SPCR Register MSTR Bit, converted to slave, thus MOSI with SPCK Become input
2. Set at SPSR Register SPIF Bit, generated if the interrupt is enabled SPI Interrupted. Therefore, use interrupt mode processingSPI Host data transfer, and there is SPSS When the possibility of being pulled down, the interrupt service routine should check MSTR Whether the bit is"1". If it is cleared, the software must set it to re-enable itSPI Host mode.

SPI initialization

Before communicating SPI Initialize. The initialization process usually includes the selection of the operation of the host and the slave, the setting of the data transmission mode, the selection of the bit rate, and the direction control of each pin. Among them, the pin side under the operation of the master and slave The control of each direction is different, as shown in the following table:

Pin direction control

Pin	Orientation in host mode	Orientation in slave mode
MOSI	User software definition	enter
MISO	enter	User software definition
SPCK	User software definition	enter
SPSS	User software definition	enter

SPI Host initialization

SPI The initialization process of the host mode is as follows:

1. Position MSTR Bit, set the bit rate selection control bit, data transmission mode, data transmission sequence, interrupt enable or not, and dual-wire enable or not;
2. Set up MOSI with SPCK Pin is output;
3. Position SPE Bit.

In host mode, when you don't want SPI When the module is selected as a slave by another host, it can be set SPSS The pin is an output.

SPI Slave initialization

SPI The initialization process of slave mode is as follows:

1. Cleared MSTR Bit, set data transmission mode, data transmission sequence, interrupt enable or not;
2. Set up MISO Pin is output;
3. Position SPE Bit.

SPI Interrupt

When one or more of the following events occur,SPI Interrupt flag SPIF Will be set:

1. When configured as a host and SPSS Pin is input, external circuit pulls low SPSS Pin
2. When the sending buffer status is full, the software continues to SPDR Register write data;
3. When the receiving buffer status is full;
4. When the data written in the sending buffer has been sent out, the sending buffer status is empty.

when SPIF Bit is set, and SPI Interrupt enable bit SPIE And the global interrupt enable bit are both high, it will generate SPI Interrupted. After entering the interrupt service routine, the hardware willSPIF Perform clearing. IfSPIF Bit is determined by the above event 1 with 2 To set, SPIF Will be cleared; if SPIF Bit is determined by the above event 3 with 4 To set,SPIF It will not be cleared, because it will still be set when the status of the receive or transmit buffer has not changed SPIF Bit, need to be cleared by software operation at this time.

SPI In the interrupt service routine, the software is cleared SPIF The sequence of bit operations is as follows:

1) Read SPIF The status of the bit, if it is low, it indicates SPIF The bit has been cleared by hardware and does not need to be cleared again by software; if it is high, Continue the operation;

2) Read SPFR Register, if RDFULL Bit is high, indicating that the current receiving buffer status is full, read SPDR Deposit The receiver gets the received data,RDFULL The bit will go low and the software can continue to read SPDR The register gets the received data until RDEMPT Bit is high

3) Read SPFR Register, if RDFULL Bit is low, andWREMPT The bit is high, indicating that the current receive buffer status is Not full, and the sending buffer status is empty, software can read SPDR The register gets the received data until RDEMPT Bit is high

4) After the software obtains the received data, perform resetting SPIF Bit. becauseSPIF The bit is read-only and cannot be directly SPIF Bit is cleared, and needs to be read first SPSR Register, then access SPDR(Read or write SPDR Register) way to clear SPIF Bit.

Register definition

SPI Register list

register	address	Defaults	description
SPCR	0x4C	0x00	SPI Control register
SPSR	0x4D	0x00	SPI Status register
SPDR	0x4E	0x00	SPI Data register
SDFR	0x39	0x00	SPI Buffer register

SPCR-SPI Control register

SPCR – SPI Control register								
address: 0x4C					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

<table>
<tr><th>Bit</th><th colspan="4">Name description</th></tr>
<tr><td>7</td><td>SPIE</td><td colspan="3">SPI Interrupt enable bit.
When set SPIE Bit is"1"Time,SPI The interrupt is enabled. When atSPSR In the register SPIF
When the bit is set and the global interrupt is enabled, it generates SPI
Interrupted. When setSPIE Bit is"0"Time,SPI Interrupts are disabled.</td></tr>
<tr><td>6</td><td>SPE</td><td colspan="3">SPI Enable bit.
When set SPE Bit is"1"Time,SPI The module is enabled. Carry on anySPI Must be set before operation
Bit SPE.
When set SPE Bit is"0"Time,SPI The module is disabled. Data</td></tr>
<tr><td>5</td><td>DORD</td><td colspan="3">sequence control bit.
When set DORD Bit is"1"When the data LSB Send it first.
When set DORD Bit is"0"When the data MSB Send it first. The master selects</td></tr>
<tr><td>4</td><td>MSTR</td><td colspan="3">the control bit from the slave.
When set MSTR Bit is"1"When, select the host mode.
When set MSTR Bit is"0"When, select the slave mode.
In host mode,SPSS When the pin is configured as an input and is pulled low,MSTR Bit will be cleared, bit
in SPSR Register SPIF Is set, the user must resetMSTR Enter host mode. Clock</td></tr>
<tr><td rowspan="4">3</td><td rowspan="4">CPOL</td><td colspan="3">polarity control bit.
When set CPOL Bit is"1"When idle SPCK It is high level.
When setCPOL Bit is"0"When idle SPCK It is low level.</td></tr>
<tr><td>CPOL</td><td>Starting edge</td><td>End edge</td></tr>
<tr><td>0</td><td>Rising edge</td><td>Falling edge</td></tr>
<tr><td>1</td><td>Falling edge</td><td>Rising edge</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">CPHA</td><td colspan="3">Clock phase control bit.
When set CPHA Bit is"1"When the data is set at the start edge, data is sampled at the end edge.
When setCPHA Bit is"0"When the data is sampled at the start edge, data is set at the end edge.</td></tr>
<tr><td>CPHA</td><td colspan="2">Start edge end edge</td></tr>
</table>

		0	sampling	Set up
		1	Set up	sampling
1	SPR1	Clock rate selection bit 1. SPR1 with SPR0 Used to choose SPI The clock rate of the transmission. For specific control methods, seeSPCK Table of the relationship with the system clock.		
0	SPR0	Clock rate selection bit 0. SPR1 with SPR0 Used to choose SPI The clock rate of the transmission. For specific control methods, seeSPCK Table of the relationship with the system clock.		

SPSR-SPI Status register

SPSR **– SPI Status register**

address: 0x4D					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	SPIF	WCOL	-	-	-	DUAL	-	SPI2X
R/W	R	R	R	R	R	R/W	R	R/W
Initial	0	0	0	0	0	0	0	0

Bit	Name	description
7	SPIF	SPI Interrupt flag bit. Set after the end of the serial transfer SPIF Logo, host mode, configuration SPSS When the pin is an input and is pulled low,SPIF Will also be set. If this timeSPCR Register SPIE Bit and global interrupt enable bit are set,SPI An interrupt is generated. After entering the interrupt service routineSPIF Bits are automatically cleared, or by first reading SPSR Register re-access SPDR Register to zero SPIF Bit. Write
6	WCOL	conflict flag. Write during data transfer SPDR Register will be set WCOL Bit.WCOL Bit can be connected Read first SPSR Register re-access SPDR Register to clear.
5	-	Reserved.
4	-	Reserved.
3	-	Reserved.
2	DUAL	Two-wire mode control bit. When set DUAL Bit is"1"When, enable SPI Two-wire transmission mode. When set DUAL Bit is"0"When, prohibit SPI Two-wire transmission mode. Two-wire transmission mode is only available in SPI Effective in host mode,MISO with MOSI Both are used as host data input, and the data transmission method is described in the chapters of host two-wire reception and data mode. Reserved.
1	-	
0	SPI2X	SPI Double speed control bit. When set SPI2X Bit is"1"Time,SPI The transmission speed is doubled. When set SPI2X Bit is"0"Time,SPI The transmission speed is not doubled. For specific control methods, seeSPCK Table of the relationship with the system clock.

The following table is SPCK The relationship with the system clock.

SPCK Relationship with system clock

SPI2X	SPR1	SPR0	SPCK Frequency of
0	0	0	$f_{sys}/4$
0	0	1	$f_{sys}/16$
0	1	0	$f_{sys}/64$
0	1	1	$f_{sys}/128$
1	0	0	$f_{sys}/2$
1	0	1	$f_{sys}/8$
1	1	0	$f_{sys}/32$
1	1	1	$f_{sys}/64$

SPDR – SPI Data register

<table>
<tr><th colspan="9"><i>SPDR</i> – SPI Data register</th></tr>
<tr><td colspan="5">address: 0x4E</td><td colspan="4">Defaults: 0x00</td></tr>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Name</td><td>SPDR7</td><td>SPDR6</td><td>SPDR5</td><td>SPDR4</td><td>SPDR3</td><td>SPDR2</td><td>SPDR1</td><td>SPDR0</td></tr>
<tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr>
<tr><td>Bit</td><td>Name</td><td colspan="7">description</td></tr>
<tr><td>7:0</td><td>SPDR</td><td colspan="7">SPI Data sent and received.
SPI Send data and receive data sharing SPI Data register SPDR. Write dataSPDR
That is, write to the transmit data shift register, from SPDR Reading data means reading the receiving data buffer
Device.</td></tr>
</table>

SPFR-SPI Buffer register

<table>
<tr><th colspan="9"><i>SPFR</i> – SPI Buffer register</th></tr>
<tr><td colspan="5">address: 0x39</td><td colspan="4">Defaults: 0x00</td></tr>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Name</td><td>RDFULI</td><td>RDEMPT</td><td>RDPTR1</td><td>RDPTR0</td><td>WRFULL</td><td>WREMPT</td><td>WRPTR1</td><td>WRPTR0</td></tr>
<tr><td>R/W</td><td>R</td><td>R/W</td><td>R</td><td>R</td><td>R</td><td>R/W</td><td>R</td><td>R</td></tr>
<tr><td>Bit</td><td>Name</td><td colspan="7">description</td></tr>
<tr><td>7</td><td>RDFULL</td><td colspan="7">Receive buffer full flag bit.
When the data in the receive buffer reaches four bytes,RDFULL The bit is high, indicating that the receive buffer is full, and the interrupt flag bit will be set at the same time. If the software does not read the data in the receive buffer in time, when the data is received again, the receive buffer overflows, and the previous data will be overwritten by the new data.
When the data in the receive buffer is less than four bytes,RDFULL The bit is low, indicating that the receiving buffer is not full and data can still be received.
When at the same time RDEMPT Bit sumWREMPT When the bit is set, the receive and transmit buffer addresses and SPI The shift register pointers will all return to zero,RDFULL Bit is low. Receive buffer empty</td></tr>
<tr><td>6</td><td>RDEMPT</td><td colspan="7">flag.
When no data is received,RDEMPT The bit is high, indicating that the receive buffer is empty.
When data is received, it will be stored in the receive buffer,RDEMPT The bit is low, indicating that the receive buffer is not empty, at this time MCU Can be accessed via SPDR Register to read the number in the receive buffer</td></tr>
</table>

		according to. In order to ensure that the received data will not be lost, the software can be used when the receiving buffer is not empty.RDEMPT Read the data in the receive buffer when the bit is low. RightRDEMPT Bit set operation (write 1), the receive buffer address will be reset to zero. When at the same timeRDEMPT Bit sumWREMPT When the bit is set, the receive and transmit buffer addresses and SPI The shift register pointers will all return to zero,RDEMPT Bit is high. The high bit of the receive buffer address.
5	RDPTR1	
4	RDPTR0	The low bit of the receive buffer address. Right SPDR When the register is read,MCU The received data will be read from the receiving buffer, and the address of the receiving buffer will be accumulated. Right RDEMPT Bit set operation (write 1), the receive buffer address will be reset to zero.
3	WRFULL	Send buffer full flag bit. When the data in the transmit buffer reaches four bytes,WRFULL The bit is high, indicating that the transmit buffer is full. When the data in the transmit buffer is less than four bytes,WRFULL The bit is low, indicating that the transmit buffer is not full. If you want to increase the transmission speed, the software can set the transmit buffer to a non-full state. WRFULL Write data when the bit is low,SPI The controller will send out the data in turn.
2	WREMPT	Send buffer empty flag bit. When the data written into the sending buffer has been sent,WREMPT Bit is high, indicating that the transmit buffer is empty, and the interrupt flag bit will be set at the same time SPIF. RightSPDR After the register is written, the sending buffer address will accumulate. When the data written in the sending buffer is not all sent, there is at least one byte of data in the receiving buffer. WREMPT The bit is low, indicating that the transmit buffer is not empty. RightWREMPT Bit set operation (write 1), the send buffer address will be reset to zero. When at the same time RDEMPT Bit sumWREMPT When the bit is set, the receive and transmit buffer addresses and SPI The shift register pointers will all return to zero,WREMPT Bit is high. The high bit of the send buffer address.
1	WRPTR1	
0	WRPTR0	The low bit of the transmit buffer address. Right SPDR When the register is written,SPDR The data in will be written into the sending buffer, and the sending buffer address will be accumulated. RightWREMPT Bit set operation (write 1), the send buffer address will be reset to zero.

USART0- *Universal synchronous/asynchronous serial transceiver*

Z Full-duplex operation (independent serial receive and transmit registers)

Z Asynchronous or synchronous operation

Z Master or slave operation

Z High-precision baud rate generator

Z stand by 5,6,7,8,or 9 Data bit sum 1,or 2 Parity generation and check

Z mechanism supported by hardware of one stop bit

Z Data overspeed detection

Z Frame error detection

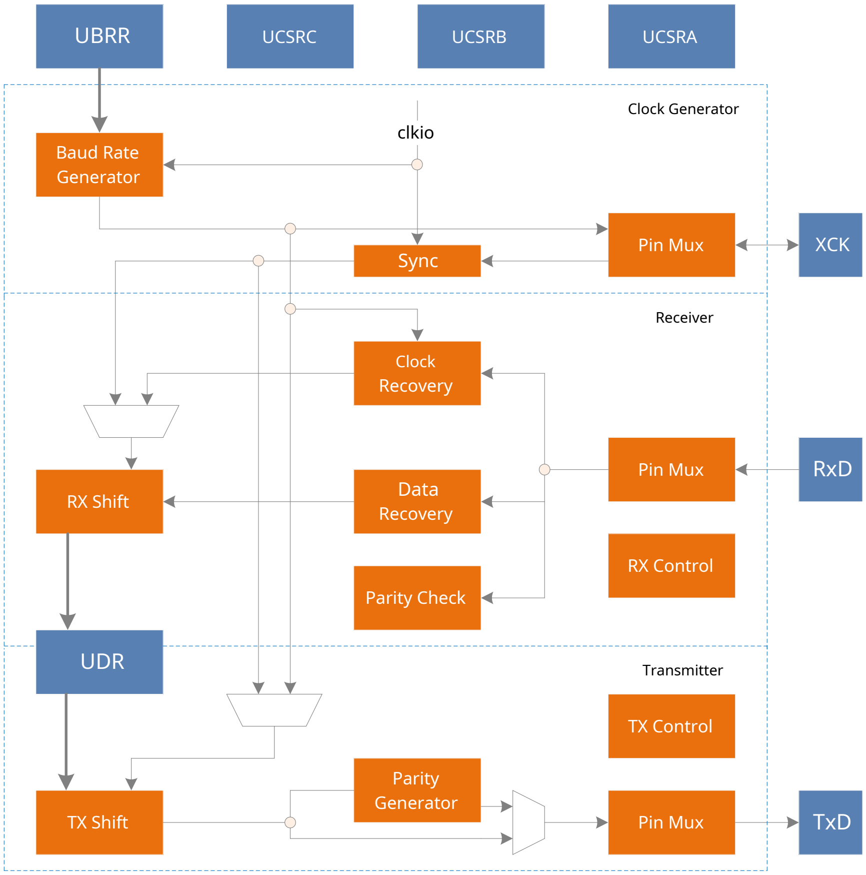
Z Noise filtering, including error start bit detection and digital low-pass filter three independent interrupts: send

Z end interrupt, send data register empty interrupt and receive end interrupt

Z Multi-processor communication mode

Z Double-speed asynchronous communication mode

Summary



USART Structure chart

USART It mainly includes three parts: clock generator, transmitter and receiver. The control and status registers are shared by these three parts. The clock generator is composed of the baud rate generator and the synchronization logic of the external input clock in the synchronous slave operation mode.

XCK The pin is only used in synchronous transfer mode. The transmitter includes a write data buffer, serial shift register, parity generator, and control logic needed to process different frame formats. The write data buffer allows data to be sent continuously without introducing delay between data frames. The receiver has a clock and data recovery unit for asynchronous data reception. In addition to the recovery unit, the receiver also includes parity, control logic, serial shift registers and a two-stage receive bufferUDR. The receiver supports the same frame format as the transmitter, and can detect frame errors, data overspeed and parity errors.

Clock generation

The clock generation logic generates basic clocks for the transmitter and receiver.USART stand by 4 Clocks in two modes: normal asynchronous mode, double-speed asynchronous mode, master synchronous mode, and slave synchronous mode.USCRC of UMSEL Bits are used to select synchronous or asynchronous mode.USCRA of U2X Bit controls the double-speed enable in asynchronous mode. Only valid in synchronous modeXCK The data direction register of the pin (and IO Multiplexing) determines whether the clock source is generated internally (master mode) or externally (slave mode).

Baud rate generator

Baud rate register UBRR And the descending counter are connected together as USART Programmable prescaler or baud rate generator. The descending counter works on the system clock ($f_{sys)}$Down, when it counts to zero or UBRRL When the register is written, it will be automatically loaded UBRR The value of the register. When the count reaches zero, a clock is generated, which is used as the output clock of the baud rate generator, and the frequency is$f_{sys/(}$UBRR+1).

The following table shows the calculated baud rate (bits per second) and UBRR Value formula.

Operating mode	Baud rate calculation formula (1)	UBRR Value calculation formula
Asynchronous normal mode	BAUD = $f_{sys/(}$16*(UBRR+1))	UBRR = $f_{sys/(}$16*BAUD)-1
Asynchronous double speed mode	BAUD = $f_{sys/(}$8*(UBRR+1))	UBRR = $f_{sys/(}$8*BAUD)-1
Synchronous host mode	BAUD = $f_{sys/(}$2*(UBRR+1))	UBRR = $f_{sys/(}$2*BAUD)-1

Description:

1. The baud rate is defined as the bit transmission speed per second (bps);

2. BUAD Is the baud rate,f_{sys}Is the system clock,UBRR Baud rate register UBRRH with UBRRL The combined value of.

Double speed working mode

By setting UCSRA Register U2X The bit can double the transmission rate. This bit is only valid in asynchronous working mode. Set this bit to"0".

Setting this bit will halve the frequency division value of the baud rate divider, effectively doubling the transmission rate of asynchronous communication. In this case, the receiver only uses half the number of samples to sample and clock the data, so a more accurate baud rate setting and system clock are required. The transmitter is unchanged.

External clock

The synchronous slave operation mode is driven by an external clock. The external clock is sent by the transmitter after passing through the synchronization register and the edge detector

And the receiver, this process will introduce a delay of two system clocks, so the external XCK The maximum clock frequency of is limited by the following formula:

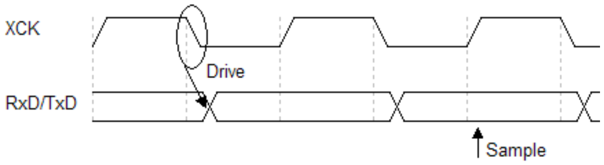
$$f_{XCK} < f_{sys}/4$$

pay attention fsys It is determined by the stability of the system clock. In order to prevent data loss due to frequency drift, it is recommended to reserve sufficient margin.

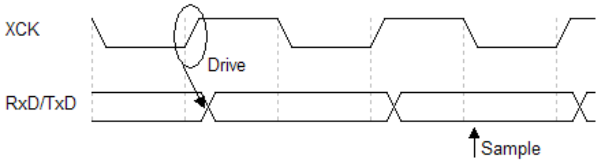
Synchronous clock operation

In synchronous mode,XCK The pins are used for clock input (slave mode) or clock output (master mode). The basic law of the relationship between the edge of the clock and data sampling and data change is: to the data input terminal (RxD) The clock edge used for sampling is opposite to the clock edge used for data output changes.

UCPOL = 1



UCPOL = 0



Synchronous mode XCK Timing

As shown in the figure above, when UCPOL Value is"1"When XCK The falling edge of changes the data output, at XCK Data sampling on the rising edge of UCPOL Value is"0"When XCK The rising edge of changes the data output, at XCK Data sampling is performed on the falling edge of.

Frame format

A serial data frame is composed of data words plus synchronization bits (start bit and stop bit) and parity bits for error correction.

USART Accept the following 30 A combined data frame format:

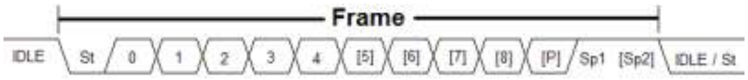
?? 1 Start bit

?? 5,6,7,8 or 9 No parity bit, odd parity bit or

?? even parity bit for each data bit

?? 1 or 2 Stop bits

The data frame starts with the start bit, followed by the lowest bit of the data word, followed by other data bits, and ends with the highest bit of the data word, at most successfully transmitted 9 Bit data. If the check is enabled, the check bit will follow the data word, and finally the stop bit. After a complete data frame is transmitted, the next new data frame can be transmitted immediately, or the transmission line can be in an idle (high level) state. The following figure shows a possible data frame structure, and the bits in square brackets are optional.



USART Frame structure diagram

Description:

1)IDLE Communication line (RxD or TxD) There is no data transmission, it must be high when the line is idle

2)St Start bit, always low

3)0-8 Data bit

4)P Parity, odd or even parity

5)Sp The stop bit is always high. The structure of the data frame is determined by UCSRB with UCSRC In the register UCSZ[2:0],UPM[1:0] with USBS set up. The same settings are used for receiving and sending. Any changes to the settings may disrupt the ongoing data transfer. among them,UCSZ[2:0]Determine the number of data bits in the data frame,UPM[1:0]Used to enable and determine the type of verification,USBS Set the frame to have one or two end bits. The receiver ignores the second stop bit, so the frame error is only in the first stop bit"0"Was detected.

Check digit calculation

The calculation of the check digit is to perform an exclusive OR operation on each bit of the data. If odd parity is selected, the XOR result needs to be inverted. The relationship between the check bit and the data bit is as follows:

$$P_{even} = d_{n-1} \oplus \ldots \oplus d_3 \oplus d_2 \oplus d_1 \oplus d_0 \oplus 0 \ P$$

$$odd = d_{n-1} \oplus \ldots \oplus d_3 \oplus d_2 \oplus d_1 \oplus d_0 \oplus 1$$

Description:

1)P_{even} Even parity result

2)P_{odd} Odd check result

3)d_n First n Data bits

USART initialization

Before communicating USART Initialize. The initialization process usually includes the setting of the baud rate, the setting of the frame structure, and enabling the receiver or transmitter as needed. For interrupt drivenUSART Operation, the global interrupt flag must be cleared and prohibited during initialization USART All interrupts.

When performing re-initialization, such as changing the baud rate or frame structure, it must be ensured that no data is transmitted.TXC The flag bit can be used to detect whether the transmitter has completed all transmissions,RXC The flag bit can be used to detect whether there is still data in the receive buffer that has not been read. in caseTXC The flag bit is used for this purpose, before each data transmission (write UDR Before register) must be cleared TXC Flag bit.

Transmitter

Position UCSRB Register TXEN Bit will enable USART The data is sent. After enablingTxD Pin common IO Function is USART The function is replaced and becomes the serial output of the transmitter. Before sending data, set the baud rate, working mode and frame format. If using synchronous transmission mode, apply toXCK The clock signal on the pin is the clock for data transmission.

send 5 To 8 Frame of data

Load the data to be sent into the sending buffer to start data sending.CPU By writing UDR Register to load data. When the sending shift register can send a new frame of data, the data in the buffer will be transferred to the shift register. When the shift register is in an idle state (no data transmission is in progress), or the last stop bit of the previous frame of data has been sent, it will load new data. Once the shift register is loaded with new data, it will be transferred according to the established settings

Lose a complete frame.

send 9 Frame of bit data

If send 9 For the frame of bit data, the first 9 Bit write register UCSRB of TXB8 Bit and then lower 8 Write bit data to send data register UDR. First9 Bit data is used to represent address frames in multi-machine communication, and can be used for protocol processing in synchronous communication.

Send parity bit

The parity check generating circuit generates the corresponding check bit for the serial data frame. When the check bit is enabled (UPM1 = 1), the transmission control logic circuit will insert a parity bit between the last bit of the data word and the first stop bit.

Send flag bit and interrupt handling

USART The transmitter has two flag bits:USART Data register empty flag UDRE And end of transmission flag TXC, Both flags can generate interrupts.

Data register empty flag UDRE It is used to indicate whether the sending buffer can write a new data. This bit is set when the transmit buffer is empty"1", Set when full"0". whenUDRE Bit is"1"Time,CPU Data register UDR Write new data, but not vice versa.

when UCSRB Data register empty interrupt enable bit in the register UDRIE for"1"When, as long as UDRE Is set (and the global interrupt is enabled),Will produceUSARTData register empty interrupt request. Pair registerUDRPerforming a write operation will clearUDRE.

When using interrupt mode to transmit data, a new data must be written in the data register empty interrupt service routine. UDR To clear UDRE, Or disable the data register null interrupt. Otherwise, once the interrupt service routine ends, a new interrupt will be generated again.

When the entire data frame is moved out of the transmission shift register and there is no new data in the transmission register, the transmission end flag TXC Will be set. whenUCSRB End of transmission interrupt enable bit on TXCIE(And global interrupt enable) set"1"When TXC The flag is set,USART The transmission end interrupt will be executed. Once in the interrupt service routine,TXC The flag bit is automatically cleared,CPU You can also write to this bit"1"Come to zero.

Disable transmitter

when TXEN After clearing, the transmitter can be truly disabled only after all the data has been sent, that is, there is no data to be transmitted in the transmission shift register and the transmission buffer register. After the transmitter is disabled,TxD The pin resumes its general purpose IO Features.

receiver

Position UCSRB The receive enable bit of the register (RXEN) To start USART receiver. After enablingRxD Pin common IO Function is USART The function is replaced and becomes the serial input port of the receiver. Before data reception, the baud rate, operation mode and frame format must be set first. If you use synchronous reception mode,XCK The clock on the pin is used as the transmission clock.

receive 5 To 8 Frame of bit data

Once the receiver detects a valid start bit, it begins to receive data. Each bit of data after the start bit will be at the set baud rate orXCK Clock to receive, until the first stop bit of a frame of data is received, the second stop bit will be

The receiver ignores it. Each bit of data received is sent to the receiving shift register. After receiving the first stop bit, the receiver sets the bit atUCSRA Register's receive data completion flag RXC Bit and transfer the complete data frame in the shift register to the receive buffer,CPU By reading UDR The register can get the received data.

receive 9 Frame of bit data

If set 9 Data frame of bit data, in the slave UDR Read low 8 The register must be read before bit data UCSRB of RXB8 To get the first 9 Bit data. This rule also applies to status flagsFE,DOR as well as PE. ReadUDR The storage unit will change the state of the receiving buffer, thereby changing the same stored in the buffer TXB8,FE,DOR and PE Bit.

Receive end flag and interrupt processing

USART The receiver has a flag bit: receiving end flag RXC, Used to indicate whether there is unread data in the receiving buffer. When there is unread data in the receiving buffer, this bit is"1", And vice versa"0". If the receiver is disabled, the receive buffer will be refreshed,RXC It will also be cleared.

Position UCSRB End of reception interrupt enable bit RXCIE After, as long as RXC The flag is set (and the global interrupt is enabled), it will generate USART The reception ends interrupt. When using the interrupt mode for data reception, the interrupt service program must start fromUDR Read data to clear RXC Flag, otherwise as long as the interrupt handler ends, a new interrupt will be generated.

Receive error flag

USART The receiver has three error flags: frame error FE, Data overflow DOR Parity error PE. They are all located UCSRA register. The error flag is stored in the receive buffer together with the data frame. All error flags cannot generate interrupts.

Frame error flag FE Indicates the state of the first stop bit of the next readable frame stored in the receive buffer. Stop bit is correct (value is"1")then FE Marked as"0",otherwise FE Marked as"1". This flag can be used to detect synchronization loss, transmission interruption, and can also be used for protocol processing.

Data overflow flag DOR Indicates that data was lost due to the full receiving buffer. When the receiving buffer is full and there is data in the receiving shift register, if a new start bit is detected at this time, data overflow occurs.DOR The flag is set to indicate that the last read UDR And the next read UDR One or more data frames were lost in between. When the data frame is successfully transferred from the shift register to the receive buffer,DOR The flag is cleared. Parity error flagPE Indicates that the next frame of data in the receive buffer has a parity error during reception. If parity check is not enabled,PE It is cleared.

Parity checker

Set the parity mode bit UPM1 The parity checker will be activated. The mode of parity (even parity or odd parity) is determined byUPM0 Decided. After the parity check is enabled, the checker will calculate the parity of the input data and compare the result with the parity of the data frame. The check result will be stored in the receive buffer along with the data and stop bits.CPU By reading PE Bit to check whether there is a parity error in the received frame. If the next data read from the receive buffer has a parity error and the parity check is enabled, thenUPE Is set, has been valid until the receive buffer UDR Is read.

Disable receiver

Compared with the transmitter, the receiver is prohibited from acting immediately. The data being received will be lost. Disable the receiver (RXEN After clearing), the receiver will no longer be occupied RxD Pin, the receive buffer will also be refreshed.

Asynchronous data reception

USART There is a clock recovery unit and data recovery unit to handle asynchronous data reception. Clock recovery logic is used to synchronize slave RxD The asynchronous serial data input by the pin and the internal baud rate clock. The data recovery logic is used to collect data and filter each bit of data input through a low-pass filter, thereby improving the anti-interference performance of the receiver. The working range of asynchronous reception depends on the accuracy of the internal baud rate clock, the frame input rate and the number of data bits contained in a frame.

Asynchronous working range

The receiver's operating range depends on the degree of mismatch between the received data rate and the internal baud rate. If the transmitter transmits data at a bit rate that is too fast or too slow, or the baud rate generated inside the receiver does not have the same frequency, then the receiver cannot synchronize with the start bit. In order to ensure that the receiver does not miss the sampling of the start bit of the next frame, the data input rate and the internal receiver baud rate cannot be too different, and the ratio between them is used to describe the error range of the baud rate. The following two tables respectively give the maximum allowable baud rate error range in normal mode and double speed mode.

Maximum receiver baud rate error range in normal mode

Data bit + parity bit length and	Maximum error range (%)	Recommended error range (%)
5	+ 6.7/-6.8	±3.0
6	+ 5.8/-5.9	±2.5
7	+ 5.1/-5.2	±2.0
8	+ 4.6/-4.5	±3.0
9	+ 4.1/-4.2	±1.5
10	+ 3.8/-3.8	±1.5

Maximum receiver baud rate error range in double speed mode

Data bit + parity bit length and	Maximum error range (%) Recommended error range (%)	
5	+ 5.7/-5.9 ±2.5	
6	+ 4.9/-5.1 ±2.0	
7	+ 4.4/-4.5 ±1.5	
8	+ 3.9/-4.0 ±1.5	
9	+ 3.5/-3.6 ±1.0	
10	+ 3.2/-3.3 ±1.0	

It can be seen from the table that the baud rate allows a larger range of changes in the normal mode. The above recommended baud rate error range is based on the assumption that the receiver and transmitter have equal contributions to the maximum total error. There are two possible reasons for the receiver baud rate error. First of all, the stability of the receiver system clock is related to the operating voltage and temperature. Generally, this problem does not occur when the crystal oscillator is used to generate the system clock, but when the internal oscillator is used, the system clock may be deviated. The second reason is that the baud rate generator may not be able to get the baud rate just desired by dividing the system clock. You can adjust

UBRR The value of makes the error as low as acceptable.

Baud rate setting and introducing errors

For standard crystal and resonator frequencies, the actual communication baud rate in asynchronous mode can be obtained by the baud rate calculation formula, and the error between it and the common communication baud rate can be calculated by the following formula:

$$\text{Error}[\%] = (\text{Baud}_{real/}\text{Baud} - 1)*100\%$$

among them,Baud Is the commonly used communication baud rate,Baud$_{real}$In order to calculate the baud rate by the calculation formula, enter the baud rate calculation formula to get the baud rate error and the system clock f_{sys}And baud rate register UBRR The relationship between the values is as follows: Normal mode:

$$\text{Error}[\%] = (f_{sys/(}16*(\text{UBRR}+1))/\text{Baud} - 1)*100\%$$

Double speed mode:

$$\text{Error}[\%] = (f_{sys/(}8*(\text{UBRR}+1))/\text{Baud} - 1)*100\%$$

When the clock error on both sides of the communication is not considered, that is, the system clock f_{sys} When it is a standard clock, the baud rate error can be obtained UBRR The relationship between values. The following table is16MHz Different under system clock UBRR The baud rate error under the value setting.

16MHz Settings under the system clock UBRR Value error

Baud rate (bps)	f_{sys} = 16.000MHz			
	Normal mode (U2X = 0)		Double speed mode (U2X = 1)	
	UBRR	error	UBRR	error
2400	416	-0.1%	832 0.0%	
4800	207	0.2%	416-0.1%	
9600	103	0.2%	207 0.2%	
14.4K	68	0.6%	138-0.1%	
19.2K	51	0.2%	103 0.2%	
28.8K	34	-0.8%	68 0.6%	
38.4K	25	2.1%	34-0.8%	
57.6K	16	0.2%	51 0.2%	
76.8K	12	0.2%	25 0.2%	
115.2K	8	-3.5%	16 2.1%	
230.4K	3	8.5%	8-3.5%	
250K	3	0%	70%	
0.5M	1	0%	30%	
1M	0	0%	1 0%	

Multi-processor communication mode

PositionUCSRAMulti-processor communication mode (MPCM) Bit can be rightUSART The data frames received by the receiver are filtered. Those frames without address information will be ignored and will not be stored in the receive buffer. In a multi-processor system, each processor communicates through the same serial bus. This filtering effectively reduces the needCPU The number of data frames processed.

MPCMThe setting of the bit does not affect the work of the transmitter, but in a multi-processor communication system, its use will be different.

If the length of the data frame received by the receiver is 5 To 8 Bit, then the first stop bit will be used to indicate whether the current frame contains data or address information. If the length of the data frame received by the receiver is9 Position, then by the first 9 Bit to determine whether it is data or address information. If the frame type flag is"1", Then this is an address frame, otherwise it is a data frame.

In the multi-processor communication mode, multiple slave processors are allowed to receive data from one master processor. First, determine which slave processor is addressed by decoding the address frame. The addressed slave processor will normally receive subsequent data, while other slave processors will ignore these data frames until the next address frame is received.

For a processor as a host, it can use 9 Bit data frame format, and use the first 9 Bit data to identify the frame format. In this communication mode, the slave processor must also work in9 Bit data frame format. The following are the steps for data exchange in multi-processor communication mode:

1. All slave processors work in multi-processor communication mode (setMPCM);
2. The master processor sends an address frame, and all slave processors receive this frame. Slave processorUCSRA Register RXC Bit Normally set;
3. Every slave readsUDR The contents of the register, decode the address frame to determine whether it is selected. If selected, Clear UCSRA Register MPCMBit, unchecked, keep MPCMfor"1"And wait for the arrival of the next address frame;
4. The addressed slave processor receives all data frames until a new address frame is received. Unaddressed slave The processor ignores these data frames;
5. Set after the addressed slave processor receives the last data frameMPCMBit and wait for the next address frame arrival. Then repeat from the second step.

use 5 To 8 The frame format of bit data is possible, but it is impractical because the receiver must be using n with n+1 Switch between frame formats. Since the receiver and transmitter use the same character length setting, this setting makes full-duplex operation difficult. If using5 To 8 For the frame format of bit data, the transmitter should set two stop bits, of which the first stop bit is used to determine the frame type.

Register definition

UCSRA – USART Control and status registers A

UCSRA – USART Control and status registers A								
address: 0xC0					Defaults: 0x20			
Bit	7	6	5	4	3 2		1	0
Name	RXC	TXC	UDRE	FE	DOR PE		U2X	MPME
R/W	R	R/W	R	R	RR		R/W	R/W

Bit	Name description	
7	RXC	Receive end flag bit. when RXC The value is"1"Indicates that there is unread data in the receiving buffer. whenRXC The value is"0" When, it indicates that there is no unread data in the receiving buffer. When the receiver is disabled, the receive buffer is refreshed, resulting inRXC It is cleared. When the reception is over, the interrupt enable bitRXCIE for"1" Time,RXC Can be used to generate a reception end interrupt.
6	TXC	Send end flag bit. When the data in the transmit shift register is sent out and the transmit buffer is empty TXC Position. When the transmission end interrupt is executedTXC Automatically clear, or through the TXC write"1"To perform resetting. When the transmission is over, the interrupt enable bitTXCIE for"1"Time,TXC Can be used to generate a transmission end interrupt.

5	UDRE	Data register empty flag. when UDRE for"1"When, show USART The send data buffer is empty and data can be written. when UDRE for"0"When, show USART The send data buffer is full and data cannot be written. When the data register is empty, the interrupt enable bitUDRIE for"1"Time,UDRE Can be used to generate data register empty interrupt.
4	FE	Frame error flag. when FE for"1"When, it indicates that the data received by the receive data buffer has a frame error, that is, the first stop bit is"0". whenFE for"0"When, it indicates that the data received by the receive data buffer has no frame error, that is, the first stop bit is"1".FE After being set, it will remain valid until UDR Is read. CorrectUCSRA When writing,FE This one is going to write"0". Data overflow flag.
3	DOR	When the receive buffer is full (contains two data), there is data in the receive shift register. If a new start bit is detected at this time, data overflow occurs.DOR Is set and is valid until UDR Is read. CorrectUCSRA When writing,DOR This one is going to
2	PE	write"0". Parity error flag. When parity check is enabled (UPM1 for"1"), and the received data frame in the receive buffer has a parity error,PE Is set and is valid until UDR Is read. CorrectUCSRA When writing,PE This one is going to write"0". Double-speed transmission enable bit.
1	U2X	when U2X for"1"When the transmission rate of asynchronous communication mode is doubled. whenU2X for"0"When the transmission rate of asynchronous communication mode is normal rate. This bit is only valid in asynchronous operation mode. Clear this bit when using synchronous operation mode.
0	MPCM	Multiprocessor communication mode enable bit. Set up MPCMThe bit will activate the multiprocessor communication mode.MPCMAfter setting,USART Those input frames that do not contain address information received by the receiver will be ignored. The transmitter is not affected byMPCM The impact of settings.

UCSRB – USART Control and status registers B

UCSRB – USART Control and status registers B								
address: 0xC1					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name	RXCIE	TXCIE	UDRIE	RXEN	TXEN UCSZ2		RXB8	TXB8
R/W	R/W	R/W	R/W	R/W	R/WR/W		R	R/W

Bit	Name description	
7	RXCIE	Receive end interrupt enable bit. Enable after set RXC Interrupt, disable after clearing RXC Interrupted. whenRXCIE for"1", Global interrupt Enable,UCSRA Register RXC for"1"Can produce USART The reception ends interrupt. End of
6	TXCIE	transmission interrupt enable bit. Enable after set TXC Interrupt, disable after clearing TXC Interrupted. whenTXCIE for"1", Global interrupt Enable,UCSRA Register TXC for"1"Can produce USART End of transmission interrupted. Data
5	UDRIE	register empty interrupt enable bit. Enable after set UDRE Interrupt, disable after clearing UDRE Interrupted. whenUDRIE for"1", Global Interrupt enable,UCSRA Register UDRE for"1"Can produce USART Data register empty

		Interrupted.
4	RXEN	Receive enable bit. Start after setting USART receiver.RxD Pin common IO Function is USART Receive replaced. Disabling the receiver will refresh the receive buffer and enableFE,DOR and PE The flag is
3	TXEN	invalid. Send enable bit. Start after setting USART Transmitter.TxD Pin common IO Function is USART Replaced by sending.TXEN After clearing, only after all data transmission is completed can it be truly prohibited USART send.
2	UCSZ2	Character length control 2 Bit. UCSZ2 versus UCSRC Register UCSZ1:0 Combine them to set the number of data bits contained in the data frame.
1	RXB8	Receive data 8 Bit. When the data frame length is 9 Position,RXB8 It is the highest bit of received data. Read UDR Included low 8 Read the bit data before RXB8. Send data8 Bit.
0	TXB8	When the data frame length is 9 Position,TXB8 It is the highest bit of the sent data. WriteUDR Included The low 8 Bit data must be written before TXB8.

UCSRC– USART Control and status registers C

<table>
<tr><th colspan="9"><i>UCSRC</i>– USART Control and status registers C</th></tr>
<tr><td colspan="5">address: 0xC2</td><td colspan="4">Defaults: 0x06</td></tr>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Name</td><td>UMSEL1</td><td>UMSEL0</td><td>UPM1</td><td>UPM0</td><td>USBS</td><td>UCSZ1</td><td>UCSZ0</td><td>UCPOL</td></tr>
<tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr>
</table>

<table>
<tr><th>Bit</th><th>Name</th><th colspan="2">description</th></tr>
<tr><td rowspan="6">7:6</td><td rowspan="6">UMSEL1:0</td><td colspan="2">USART Mode selection bit.
UMSEL Select synchronous or asynchronous operation mode.</td></tr>
<tr><td>UMSEL</td><td>mode</td></tr>
<tr><td>0</td><td>USART Asynchronous operation mode</td></tr>
<tr><td>1</td><td>USART Synchronous operation mode</td></tr>
<tr><td>2</td><td>SPI Slave operation mode</td></tr>
<tr><td>3</td><td>SPI Host operating mode</td></tr>
<tr><td rowspan="6">5:4</td><td rowspan="6">UPM1:0</td><td colspan="2">Parity mode selection bit.
High position UPM1 Choose to enable or disable parity check, low order UPM0 Select odd or even check.</td></tr>
<tr><td>UPM1:0</td><td>mode</td></tr>
<tr><td>0</td><td>Prohibit parity</td></tr>
<tr><td>1</td><td>Keep</td></tr>
<tr><td>2</td><td>Enable even parity</td></tr>
<tr><td>3</td><td>Enable odd parity</td></tr>
<tr><td rowspan="3">3</td><td rowspan="3">USBS</td><td colspan="2">Stop bit selection bit. Select the number of stop bits.</td></tr>
<tr><td>USBS</td><td>Number of stop bits</td></tr>
<tr><td>0</td><td>1</td></tr>
</table>

		1	2	
2:1	UCSZ1:0	Data frame character length selection bit. UCSZ1:0versusUCSRBRegisterUCSZ2Combine to set the data contained in the data frame Number of digits.		
		UCSZ2:0	Data frame length	
		0	5 Bit	
		1	6 Bit	
		2	7 Bit	
		3	8 Bit	
		4	Keep	
		5	Keep	
		6	Keep	
		7	9 Bit	
0	UCPOL	Clock polarity selection bit. in USART In synchronous working mode,UCPOL Set the output data change and input data sampling and synchronization clock XCK The relationship between. Use asynchronous working mode with UCPOL Don't care, clear this bit		
		UCPOL	Send data changes	Receive data sampling
		0	XCK Rising edge	XCK Falling edge
		1	XCK Falling edge	XCK Rising edge

UBRRL – USART Baud rate register low byte

UBRRL – USART Baud rate register low byte								
address: 0xC4					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	UBRR7	UBRR6	UBRR5	UBRR4	UBRR3	UBRR2	UBRR1	UBRR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	Name	description						
7:0	UBRR[7:0]	USART The low byte of the baud rate register. USART The baud rate register contains UBRRL with UBRRH Two parts, used together To set the baud rate of communication.						

UBRRH – USART Baud rate register high byte

UBRRH – USART Baud rate register high byte								
address: 0xC5					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name	-	-	-	-	UBRR11	UBRR10	UBRR9	UBRR8
R/W	-	-	-	-	R/W	R/W	R/W	R/W
Bit	Name	description						
7:4	-	Reserved.						

<table>
<tr><td rowspan="5">3:0</td><td rowspan="5">UBRR[11:8]</td><td colspan="2">USART The high byte of the baud rate register.
USART The baud rate register contains UBRRL with UBRRH Two parts, combined
Used to set the baud rate of communication.
UBRR = {UBRR[11:8], UBRRL}</td></tr>
<tr><td>Operating mode</td><td>Baud rate calculation formula</td></tr>
<tr><td>Asynchronous normal mode</td><td>BAUD = $f_{sys/(}$16*(UBRR+1))</td></tr>
<tr><td>Asynchronous double speed mode</td><td>BAUD = $f_{sys/(}$8*(UBRR+1))</td></tr>
<tr><td>Synchronous host mode</td><td>BAUD = $f_{sys/(}$2*(UBRR+1))</td></tr>
</table>

UDR – USART Data register

<table>
<tr><th colspan="9"><i>UDR</i> – USART Data register</th></tr>
<tr><td colspan="5">address: 0xC6</td><td colspan="4">Defaults: 0x00</td></tr>
<tr><td>Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Name</td><td>UDR7</td><td>UDR6</td><td>UDR5</td><td>UDR4</td><td>UDR3</td><td>UDR2</td><td>UDR1</td><td>UDR0</td></tr>
<tr><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td></tr>
<tr><td>Bit</td><td>Name</td><td colspan="7">description</td></tr>
<tr><td>7:0</td><td>UDR</td><td colspan="7">USART Data sent and received.
USARTThe transmit data buffer and the receive data buffer are sharedUSARTData registerUDR. Write dataUDR That is, write to the send data buffer, from UDR Reading data means reading the receive data buffer.
in 5 To 8 In bit data frame mode, the unused first 9 Bits are ignored by the transmitter, and the receiver sets them to 0.
Only if UCSRA Register UDRE Marked as"1"Write to the transmit buffer only when, otherwise, the operation of the transmitter will be wrong. When the transmit shift register is empty, the transmitter will load the data in the transmit buffer into the transmit shift register,Then data serially from TxD Pin output.
The receive buffer contains a two-stage FIFO, Once the receive buffer is read,FIFO Will change its state.</td></tr>
</table>

USART0-SPI Operating mode

Z Full-duplex operation, three-wire synchronous data transmission

Z Master or slave operation

Z Support all four working modes (mode 0,1,2 with 3)

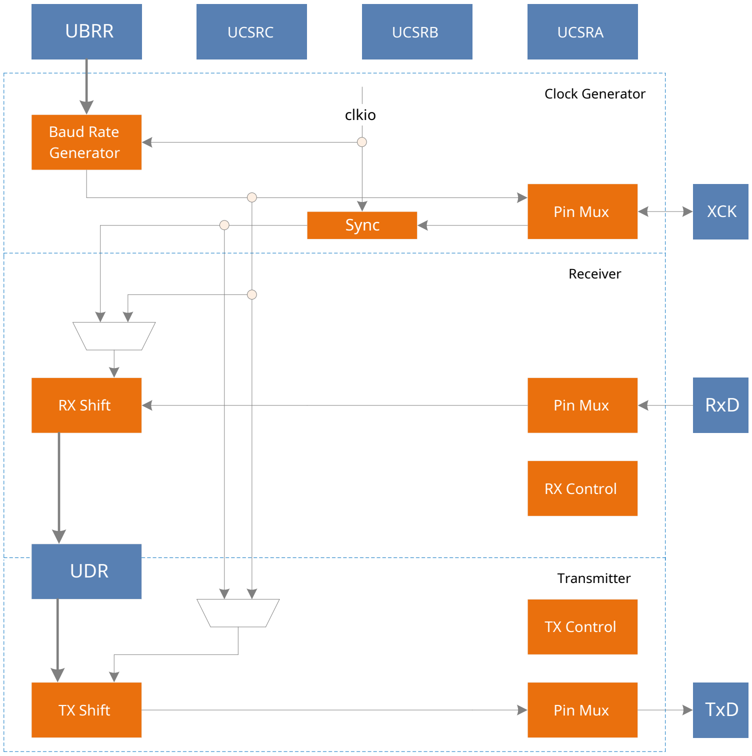
Z Low bit or high bit is transmitted first (configurable data transmission sequence)

Z Queue operation (double buffer)

Z High resolution baud rate generator

Summary

When set USCRC of UMSEL1 Bit is"1"When, enable SPI Working mode, use USPI To represent. thisSPI The module is three-wire SPI Working mode, with four-wire SPI Compared with the mode, the slave selection line is missing, and the other three lines are the same.USPI Occupy USART The resources include transmit and receive shift registers and buffers, and baud rate generators. Parity generation and check logic, data and clock recovery logic are all invalid. The address of the control and status register is the same, but the register The function of the device will follow SPI The work model needs to change.



USART in SPI Structure chart

Clock generation

when SPI When working in the host mode, it is necessary to provide a clock for communication and multiplexing USART The baud rate generator is used to generate this clock. The clock fromXCK Pin output, so XCK The data direction register of the pin (DDR_XCK) Must be set to

"1".

The clock frequency is determined by the following calculation formula:

BAUD = $f_{sys/(}$2*(UBRR+1))

when SPI When working in slave mode, the communication clock is provided by the external host. XCK Pin input, so XCK The data direction register of the pin (DDR_XCK) Must be set to"0".

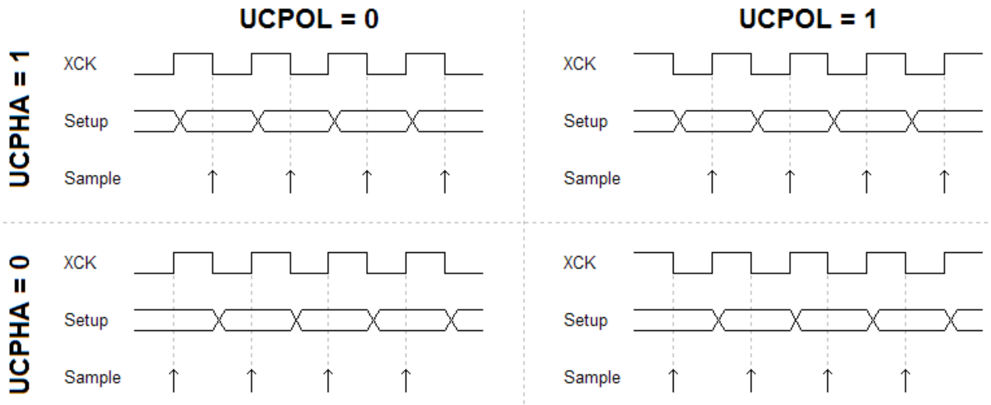
SPI Data pattern and timing

SPI There are four combinations of clock phase and polarity, with control bits UCPHA with UCPOL To decide, specific controls such as

As shown in the following table and figure below:

SPI Operating mode

SPI mode	UCPOL	UCPHA	Starting edge	End edge
0	0	0	Rising edge sampling	Falling edge setting
1	0	1	Rising edge setting	Falling edge sampling
2	1	0	Falling edge sampling	Rising edge setting
3	1	1	Falling edge setting	Rising edge sampling



SPI Working mode icon

Frame format

SPI A serial frame can start from the lowest bit or the highest bit and end with the highest bit or the lowest bit. In total 8 Bit data. After the end of a frame, a new frame can be transmitted immediately, and the data line can be pulled up to an idle state after the transmission is over.

data transmission

SPI Set UCSRB Register TXEN Bit is"1"To enable the transmitter,TxD The pin is occupied by the transmitter to send serial output data. The receiver can be disabled at this time.

SPI Set UCSRB Register RXEN Bit is"1"To enable the receiver,RxD The pin is occupied by the receiver to receive serial input data. The transmitter must be enabled at this time.

SPI Both send and receive XCK Used as a transmission clock.

Before communicating SPI Initialize. The initialization process usually includes the setting of the baud rate, the setting of the frame data bit transmission sequence, and enabling the receiver or transmitter as needed. For interrupt drivenSPI Operation, in initialization

When you want to clear the global interrupt flag and disable it SPI All interrupts. When performing re-initialization, such as changing the baud rate or frame structure, it must be ensured that no data is transmitted.TXC The flag bit can be used to detect whether the transmitter has completed all transmissions,RXC The flag bit can be used to detect whether there is still data in the receive buffer that has not been read. in caseTXC The flag bit is used for this purpose, before each data transmission (write UDR Before register) must be cleared TXC Flag bit.

initialization SPI In the future, to UDR Data transfer can be started by writing data into the register. Since the transmitter controls the transmission clock, this is the case for both sending and receiving data. When the transmit shift register is ready to transmit a new frame of data, the transmitter will write toUDR The data of the register is moved from the sending buffer to the sending shift register and sent out. In order to ensure the synchronization of the input buffer and the sending data, it must be read every time after sending a byte of dataUDR register. When data overflow occurs, the most recently received data will be lost instead of the earliest received data.

Send flag and interrupt

SPI The transmitter has two flag bits:SPI Data register empty flag UDRE And end of transmission flag TXC, Both flags can generate interrupts.

Data register empty flag UDRE It is used to indicate whether the sending buffer can write a new data. This bit is set when the transmit buffer is empty"1", Set when full"0". whenUDRE Bit is"1"Time,CPU Data register UDR Write new data, but not vice versa.

when UCSRB Data register empty interrupt enable bit in the register UDRIE for"1"When, as long as UDRE Is set (and the global interrupt is enabled), it will generate SPI Data register empty interrupt request. Pair registerUDR Performing a write operation will clear
UDRE. When using interrupt mode to transmit data, a new data must be written in the data register empty interrupt service routine. UDR To clear UDRE, Or disable the data register null interrupt. Otherwise, once the interrupt service routine ends, a new interrupt will be generated again.

When the entire data frame is moved out of the transmission shift register and there is no new data in the transmission register, the transmission end flag TXC Will be set. whenUCSRB End of transmission interrupt enable bit on TXCIE(And global interrupt enable) set"1"When TXC The flag is set,SPI The transmission end interrupt will be executed. Once in the interrupt service routine,TXC The flag bit is automatically cleared,CPU You can also write to this bit"1"Come to zero.

Disable transmitter

when TXEN After clearing, the transmitter can be truly disabled only after all the data has been sent, that is, there is no data to be transmitted in the transmission shift register and the transmission buffer register. After the transmitter is disabled,TxD The pin resumes its general purpose
IO Features.

Receive end flag and interrupt

SPI The receiver has a flag bit: receiving end flag RXC, Used to indicate whether there is unread data in the receiving buffer. When there is unread data in the receiving buffer, this bit is"1", And vice versa"0". If the receiver is disabled, the receive buffer will be refreshed,RXC It will also be cleared. PositionUCSRB End of reception interrupt enable bit RXCIE After, as long as RXC The flag is set (and the global interrupt is enabled), it will generate SPI The reception ends interrupt. When using the interrupt mode for data reception, the interrupt service program must start fromUDR Read data to clear RXC Sign, otherwise as long as

As soon as the interrupt handler ends, a new interrupt will be generated.

Disable receiver

Compared with the transmitter, the receiver is prohibited from acting immediately. The data being received will be lost. Disable the receiver (RXEN After clearing), the receiver will no longer be occupied RxD Pin, the receive buffer will also be refreshed.

Register definition

USART Register list

register	address	Defaults	description
UCSRA	0xC0	0x20	USPI Control and status registers A
UCSRB	0xC1	0x00	USPI Control and status registers B
UCSRC	0xC2	0x06	USPI Control and status registers C
UBRRL	0xC4	0x0	USPI Baud rate register low byte
UBRRH	0xC5	0x0	USPI Baud rate register high byte
UDR	0xC6	0x0	USPI Data register

UCSRA – USPI Control and status registers A

UCSRA – USPI Control and status registers A								
address: 0xC0					Defaults: 0x20			
Bit	7	6	5	4	3 2		1	0
Name	RXC	TXC	UDRE	-	-	-	-	-
R/W	R	R/W	R	-	-	-	-	-

Bit	Name description	
7	RXC	Receive end flag bit. when RXC The value is"1"Indicates that there is unread data in the receiving buffer. whenRXC The value is"0" When, it indicates that there is no unread data in the receiving buffer. When the receiver is disabled, the receive buffer is refreshed, resulting inRXC It is cleared. When the reception is over, the interrupt enable bitRXCIE for"1" Time,RXC Can be used to generate a reception end interrupt.
6	TXC	Send end flag bit. When the data in the transmit shift register is sent out and the transmit buffer is empty TXC Position. When the transmission end interrupt is executedTXC Automatically clear, or through the TXC write"1"To perform resetting. When the transmission is over, the interrupt enable bitTXCIE for"1"Time,TXC Can be used to generate a transmission end interrupt. Data register empty
5	UDRE	flag. when UDRE for"1"When, show USPI The send data buffer is empty and data can be written. when UDRE for"0"When, show USPI The send data buffer is full and data cannot be written. When the number According to register empty interrupt enable bit UDRIE for"1"Time,UDRE Can be used to generate data register empty interrupt.
4:0	-	USPI Next reservation.

UCSRB – USPI Control and status registers B

UCSRB – USPI Control and status registers B								
address: 0xC1					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name	RXCIE	TXCIE	UDRIE	RXEN	TXEN-		-	-
R/W	R/W	R/W	R/W	R/W	R/W-		-	-

Bit	Name description	
7	RXCIE	Receive end interrupt enable bit. Enable after set RXC Interrupt, disable after clearing RXC Interrupted. whenRXCIE for"1", In the global Disable enable,UCSRA Register RXC for"1"Can produce USPI The reception ends interrupt. End of
6	TXCIE	transmission interrupt enable bit. Enable after set TXC Interrupt, disable after clearing TXC Interrupted. whenTXCIE for"1", In the global Disable enable,UCSRA Register TXC for"1"Can produce USPI End of transmission interrupted. Data
5	UDRIE	register empty interrupt enable bit. Enable after set UDRE Interrupt, disable after clearing UDRE Interrupted. whenUDRIE for"1", Global interrupt enable,UCSRA Register UDRE for"1"Can produce USPI Data register empty interrupt.
4	RXEN	Receive enable bit. Start after setting USPI receiver.RxD Pin common IO Function is USPI Receive replaced. Disabling the receiver will refresh the receive buffer.
3	TXEN	Send enable bit. Start after setting USPI Transmitter.TxD Pin common IO Function is USPI Replaced by sending. TXEN After clearing, only after all data transmission is completed can it be truly prohibited USART send.
2:0	-	USPI Next reservation.

UCSRC– USART Control and status registers C

UCSRC– USART Control and status registers C								
address: 0xC2					Defaults: 0x86			
Bit	7	6	5	4	3	2	1	0
Name	UMSEL1	UMSEL0	-	-	-	DORD	UCPHA	UCPOL
R/W	R/W	R/W	-	-	-	R/W	R/W	R/W

Bit	Name	description	
7:6	UMSEL1:0	USART Mode selection bit. UMSEL Select synchronous or asynchronous operation mode.	
		UMSEL	mode
		0	USART Asynchronous operation mode
		1	USART Synchronous operation mode
		2	SPI Slave operation mode
		3	SPI Host operating mode
5:3	-	USPI Next reservation.	

2	DORD	Data transmission sequence selection bit.		
		DORD	Data sequence	
		0	High bit first	
		1	Low order first	
1	UCPHA	Clock phase selection. UCPHA Select data sampling to occur at the start or end edge.		
		UCPHA	Sampling moment	
		0	Starting edge	
		1	End edge	
0	UCPOL	Clock polarity selection. UCPOL Select the data change and sampling to occur on the rising edge or the falling edge.		
		UCPOL	Send data changes	Sampling of received data
		0	XCK Rising edge	XCK Falling edge
		1	XCK Falling edge	XCK Rising edge

UBRRL – USPI Baud rate register low byte

UBRRL **– USPI Baud rate register low byte**								
address: 0xC4					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	UBRR7	UBRR6	UBRR5	UBRR4	UBRR3	UBRR2	UBRR1	UBRR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	UBRR[7:0]	USPI The low byte of the baud rate register. USPI The baud rate register contains and UBRRL UBRRH The two parts are combined to set the baud rate of communication.

UBRRH – USPI Baud rate register high byte

UBRRH **– USPI Baud rate register high byte**								
address: 0xC5					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	UBRR11	UBRR10	UBRR9	UBRR8
R/W	-	-	-	-	R/W	R/W	R/W	R/W

Bit	Name	description
7:4	-	USPI Next reservation.
3:0	UBRR[11:8]	USPI The high byte of the baud rate register. USPI The baud rate register contains UBRRL with UBRRH The two parts are combined to set the baud rate of communication. UBRR = {UBRR[11:8], UBRRL}

		Operating mode	Baud rate calculation formula
		Slave mode	The baud rate is determined by the external host
		Host mode	$BAUD = f_{sys/(}2*(UBRR+1))$

UDR – USPI Data register

UDR **– USPI Data register**								
address: 0xC6					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	UDR7	UDR6	UDR5	UDR4	UDR3	UDR2	UDR1	UDR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	UDR	USPI Data sent and received. USPI The transmit data buffer and the receive data buffer are shared USPI Data register UDR. Write data UDR That is, write to the send data buffer, from UDR Reading data means reading the receive data buffer. in 5 To 8 In bit data frame mode, the unused first 9 Bits are ignored by the transmitter, and the receiver sets them to 0. Only if UCSRA Register UDRE Marked as"1"Write to the transmit buffer only when, otherwise, the operation of the transmitter will be wrong. When the transmit shift register is empty, the transmitter will load the data in the transmit buffer into the transmit shift register, and then the data will be serially transferred from theTxD Pin output. The receive buffer contains a two-stage FIFO, Once the receive buffer is read,FIFO Will change its state.

TWI – Two-wire serial bus (I2C)

- Simple, powerful and flexible communication interface, only need 2 line
- Support host and slave operation
- The device can work in transmitter mode or receiver mode
- 7 Bit address space allows 128 Each slave supports
- multi-master arbitration
- Up to 400Kbps Data transfer rate
- Fully programmable slave address and public address
- Can wake up when the address matches in sleep mode

TWI Bus introduction

Two-wire serial interface TWI Very suitable for typical processor applications.TWI The protocol allows system designers to use only two bidirectional transmission lines to connect 128 Different devices are interconnected together. These two wires are the clockSCL And data SDA. The external hardware only needs to connect two pull-up resistors on each wire. All devices connected to the bus have their own addresses.TWI The protocol solves the problem of bus arbitration.

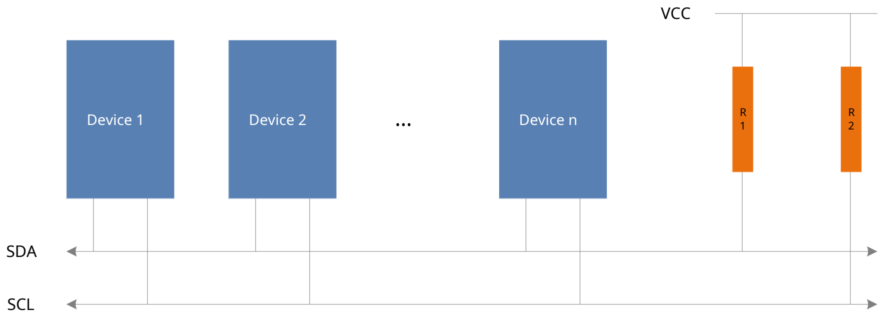
TWI the term

The following defined terms will appear frequently in this section.

the term	description
Host	Devices that start and stop transmissions. The host is also responsible for generatingSCL clock.
Slave	Device addressed by the host
Transmitter	Device that puts data on the bus
receiver	Device that receives data from the bus

Electrical connections

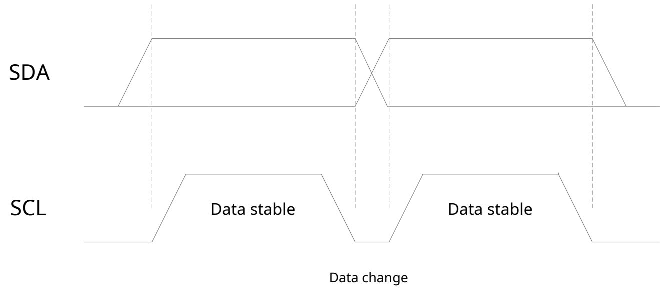
As shown below,TWI Both wires of the interface are connected to the positive power supply through a pull-up resistor. allTWI The bus drivers of compatible devices are all open-drain or open-collector, so that the line and function of the interface operation are realized. whenTWI The device output is"0"Time,TWI The bus will generate a low level. When allTWI When the device output is tri-stated, the bus allows the pull-up resistor to pull the voltage high. In order to ensure all bus operations, usually withTWI All devices connected to the bus must be powered on.



TWI Bus interconnection diagram

Data transmission and frame structure

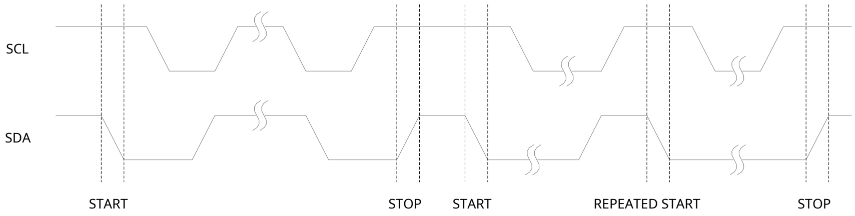
TWI Every data transmission on the bus is synchronized with the clock. When the clock line is high, the level on the data line must Keep it stable, unless it is to produce a start or stop state.



TWI Data validity graph

Start and stop state

TWI The transmission is started and stopped by the host. The host sends out on the busSTART State to send data transmission, send out STOP Status to stop data transmission. inSTART with STOP Between states, the bus is considered busy, and other hosts are not allowed to try to occupy control of the bus. There is a special situation that is only allowed to occur inSTART with STOP A new START State, this is called REPEATED START Status, suitable for the current host to start a new transmission without giving up bus control.REPEATED START After that until next STOP Before, the bus was still considered busy. This is the same asSTART Is consistent, so in this document, if there is no special instructions, all use START To express START with REPEATED START. As shown below,START with STOP Condition is on SCL When the line is high, change SDA Line of electricity Ping state.



START,REPEATED START with STOP State diagram

Address packet format

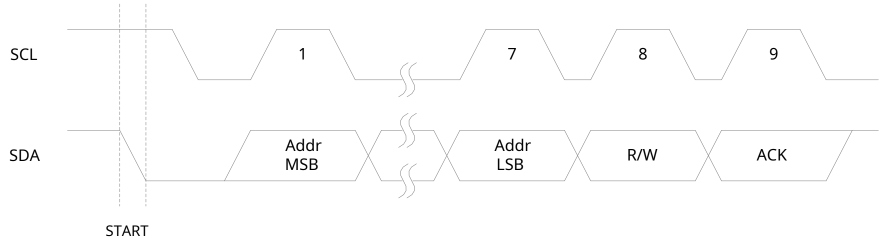
all TWI The address packets transmitted on the bus are 9 Bit data length, by 7 Bit address,1 Bit READ/WRITE Control bits and 1 Bit response bit composition. whenREAD/WRITE Bit is"1", The read operation is performed; when READ/WRITE Bit is"0"When the write operation is performed. After the slave is addressed, it must be9 A SCL(ACK) Cycle through pull down SDA Line to respond. If the slave is busy or cannot respond to the master for other reasons, it should beACK Cycle maintenance SDA The line is high. Then the host can issueSTOP State or REPEATED START The status resumes sending.

The address packet includes a slave address and a read or write control bit, which are used respectively SLA+R or SLA+WTo represent.

Address byte MSB Bit happens first. In addition to reserved addresses"00000000"Is reserved for broadcast calls and all forms "1111xxxx"The format address needs to be reserved for future use, other slave addresses can be freely allocated by the designer.

When a broadcast call occurs, all slaves should be ACK Cycle through pull down SDA Line to respond. The broadcast function can be used when the master needs to send the same information to multiple slaves. When the general call address is addedWRITE After the bit is sent to the bus, all slaves that need to respond to the broadcast call will be ACK Cycle down SDA line. All these slaves that responded to the broadcast call will receive the following data packets. It should be noted that the send broadcast call address plusREAD Bit is
It doesn't make sense, because if several slaves send different data at the same time, it will cause bus conflicts.

The format of the address packet is shown in the figure below:

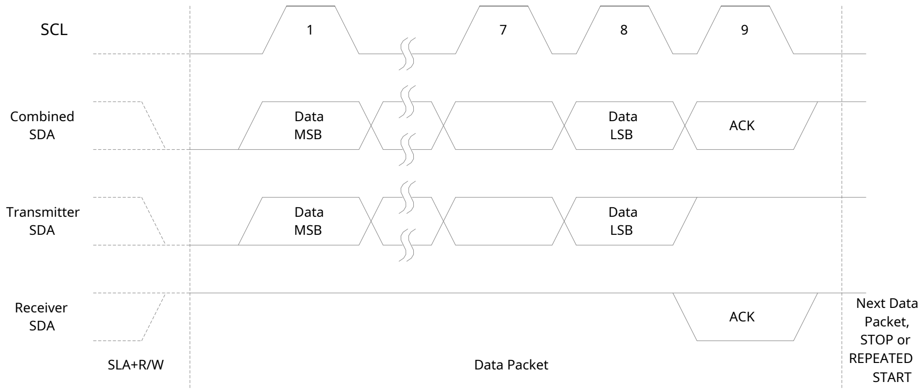


TWI Address packet format diagram

Packet format

all TWI The data packets transmitted on the bus are 9 Bit data length, by 1 Data bytes and 1 Bit response bit composition. During data transmission, the host is responsible for generating the transmission clockSCL with START and STOP Status, the transmitter sends the byte data to be transmitted, and the receiver generates a receive response. Confirmation signalACK Is the receiver in the 9 A SCL(ACK) Cycle through pull down SDA Line to produce. If the receiver isACK Cycle maintenance SDA If the line is high, an unconfirmed signal is sent NACK. When the receiver has received the last byte, or cannot receive any more data for some reason, it should be receiving After the last byte is sent NACK To inform the sender. Data byteMSB Bits are transmitted first.

The data packet format is shown in the figure below:

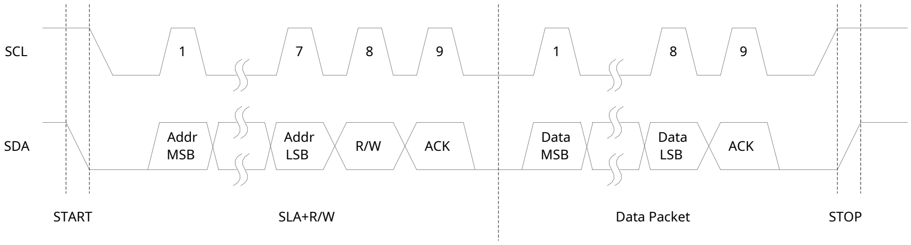


TWI Data packet format diagram

Combined address and data packet transmission, one transmission basically consists of 1 A START,1 A SLA+R/W,1 One or more

Data package and 1 A STOP composition. onlySTART with STOP The empty message is illegal. can useSCL The line and function of the line realize the handshake between the master and the slave. The slave can be pulled downSCL Line to extend SCL The ground level period. This feature is very useful when the clock speed set by the master is much faster than that of the slave, or when the slave needs extra time to process data. Slave extensionSCL The low-level period does not affect SCL It is still determined by the host. It can be seen that the slave can be changed by changingSCL Duty cycle to reduce TWI Data transfer speed.

The figure below shows a typical data transmission. noteSLA+R/Wversus STOP Multiple bytes can be transferred between, take Depends on the implementation protocol of the application software.



typical TWI transmission

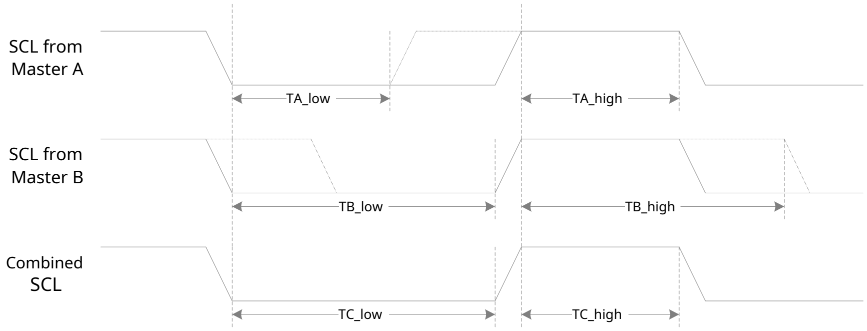
Multi-master system and its arbitration and synchronization

TWI The protocol allows multiple hosts on the bus, and special measures are adopted to ensure that even if two or more hosts start the transmission at the same time, it can be processed like a normal transmission. There are two problems with multi-host systems:

1. The implemented algorithm allows only one of the multiple hosts to complete the transmission. When other hosts find that they have lost the right to choose, they must stop their transmission. This selection process is called arbitration. When the competing master finds that its arbitration has failed, it should immediately switch to the slave mode to check whether it is addressed by the master that has obtained bus control. In fact, when multiple masters start transmitting at the same time, they should not be detected by the slave, that is, it is not allowed to destroy the data being transmitted on the bus.
2. Different hosts may use different SCL frequency. In order to ensure the consistency of transmission, a scheme for synchronizing the serial clock of the host must be designed. This will simplify the arbitration process.

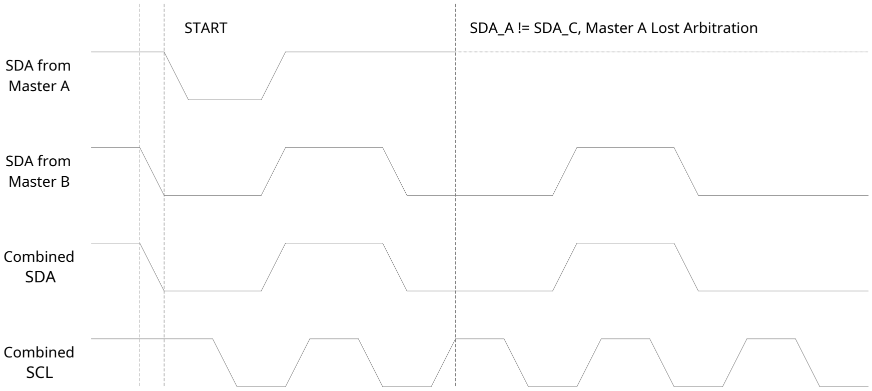
The line and function of the bus are used to solve the above-mentioned problems. The serial clocks of all hosts will be line-and-connected to generate a combined clock whose high level time is equal to the shortest of all host clocks, and its low level is equal to the longest of all host clocks. All hosts are listeningSCL, When combined SCL When the clock goes high or low, they can effectively separate
Start calculating each SCL High-level and low-level time-out periods.

Multi-hosted SCL The clock synchronization mechanism is shown in the figure below:



Multi-host SCL Clock synchronization timing diagram

All hosts continue to monitor after outputting data SDA Line to achieve arbitration. If fromSDA If the value read back does not match the value output by the host, the host loses arbitration. It should be noted that the host output high levelSDA, And another host outputs low-level SDA Arbitration will only be lost. The master that loses arbitration should immediately switch to slave mode and check whether it is addressed. The host that loses arbitration must changeSDA The line is set high, but the clock signal can be generated before the end of the current data or address packet. The arbitration will continue until there is only one host left in the system, which may take up multiple bits. If multiple masters address the same slave, the arbitration will continue until the packet.



Arbitration between two hosts

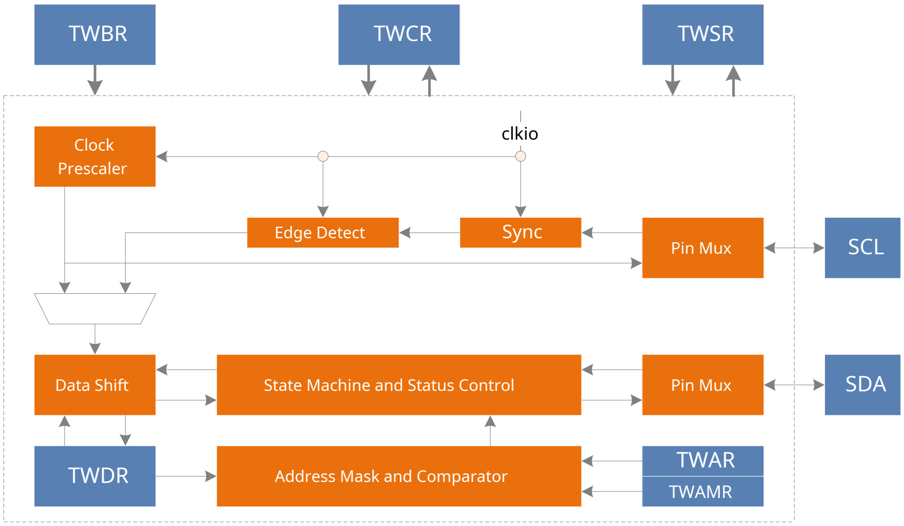
Note that arbitration is not allowed in the following situations:

- ?? One REPEATED START Between status and one data bit;
- ?? one STOP Between status and a data bit;
- ?? One REPEATED START State and STOP Between states

The application software must consider the above situations to ensure that these illegal arbitration situations will not occur. This means that in a multi-master system, all data transfers must be made by the sameSLA+R/W Composed with data packets. In other words, all transmissions must contain the same number of data packets, otherwise the arbitration result cannot be defined.

TWI Module overview

TWI The structure diagram of the module is shown in the figure below.



TWI Block Structure chart

TWI The module mainly includes bit rate generator, bus interface unit, address comparator and control unit, etc. See the detailed description below for details.

Bit rate generator unit

The bit rate generator unit mainly controls the SCL Clock cycle.SCL The clock cycle is determined by TWI Bit rate register TWBR with TWI Status register TWSR The pre-scaling control bits in are jointly determined. The operation of the slave is not affected by the bit rate or prescaler setting, but it is necessary to ensure that the working clock of the slave is at leastSCL Frequency 16 Times. Note that the slave may be extendedSCL Low-level period, thereby reducing TWI The average clock frequency of the bus.SCL The clock frequency is generated by the following calculation formula:

$$f_{scl} = f_{sys/(}16 + 2*TWBR*4_{TWPS)}$$

among them,TWBR for TWI The value of the bit rate register,TWPS for TWI The prescaler control bit in the status register.

Bus interface unit

The bus interface unit includes data and address shift registers TWDR,START/STOP Controller and arbitration decision hardware circuit.

TWDRContains the address or data byte to be sent,Or the received address or data byte. In addition to containing8BitTWDR, The bus interface unit also includes sending or receiving ACK/NACK register. This oneACK/NACK The register cannot be directly accessed by application software. When receiving data, it can passTWI Control register TWCR To set or clear. When sending data, the receivedACK/NACK Value by TWI Status register TWSR middle TWS Value to reflect.

START/STOP The controller is responsible for generating and detecting START,REPEATED START with STOP status. whenMCU When in some sleep modes,START/STOP The controller can still detect START with STOP State when being TWI The host on the bus will MCU Wake up from sleep mode.

in case TWI The data transmission is started in the master mode, and the arbitration detection circuit will continue to monitor the bus to determine whether it still has the right to control the bus. whenTWI After the module loses control of the bus, the control unit will perform the correct action and generate an appropriate status code to notify MCU.

Address matching unit

The address matching unit is used to check whether the received address byte matches TWI In the address register 7 The bit addresses match. when TWAR In the register TWI Broadcast call identification enable bit (TWGCE) Is set, the address received from the bus will also be compared with the broadcast address. Once the address is matched successfully, the control unit will perform the correct action.TWI The module can respond or not respond to the host's addressing, depending on TWCR Register settings. Even in sleep mode, the address matching unit can compare addresses. If it is addressed by the host on the bus, it willMCU Wake up from sleep mode.

control unit

The control unit is responsible for monitoring the bus and according to TWCR The setting produces a corresponding response. when TWI When an event that requires application software participation occurs on the bus,TWI Interrupt flag TWINT Will be set. In the next clock cycle,TWI Status register TWSR Will be updated to indicate the status code of the event. inTWINT When set,TWSR Contains exact status information. At other times, TWSRIt is a special status code, which means that there is no exact status information. onceTWINT The flag is set,SCL Keeps the low level all the time, suspending the TWI Transmission, let the application software handle the event.

In the following situations,TWINT The flag bit will be set:

?? TWI Send START/REPEATED START Post-state

?? TWI Send SLA+R/WRear

?? TWI After transmitting an address byte

?? TWI After bus arbitration fails

?? TWI After being addressed by the master (slave address matching or broadcast mode),

?? when it is addressed and working as a slave, it receives STOP or REPEATED START Illegal

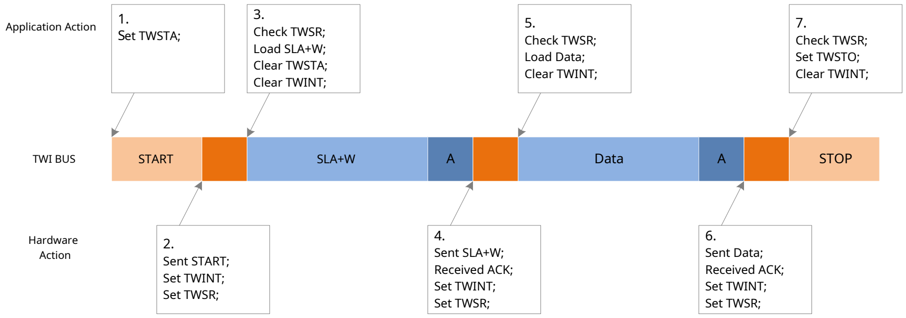
?? START or STOP When the bus error is caused by the status

TWI usage of

TWI The interface is byte-oriented and interrupt-based. All bus events, such as receiving a byte or sending aSTART Signal, etc., will generate a TWI Interrupted. due toTWI Is based on interrupts, so in TWI During the byte transfer, the application software can freely perform other operations.TWCR In the register TWI Interrupt enable bit TWIE Together with the global interrupt enable bit to control the TWINT Whether to generate when the flag is set TWI Interrupted. in caseTWIE Bit is cleared, application software must use query TWINT Flag bit method to detect TWI Action on the bus.

when TWINT When the flag is set, it means TWI The interface has completed the current operation and is waiting for a response from the application software. under these circumstances,TWI Status register TWSR It contains the status code reflecting the current bus status. Application software can be setTWCR with TWDR Register to determine the next TWI Bus cycle TWI How the interface works.

The following figure shows the application and TWI Example of interface connection. In this example, the master expects to send one byte of data to the slave. The description here is very simple and will be shown in more detail in the following chapters.



TWI Typical transmission process diagram

Shown in the picture TWI The transmission process is:

1. TWI The first step of the transmission is to send START. Through toTWCR Write a specific value to the register, indicating TWI Hardware send START signal. The written value will be explained in detail later. To be set in the written valueTWINT, This is very important, to TWINTBit write"1"This bit will be cleared.TWCRRegisterTWINTSet periodTWINo action will be initiated. Once the software is clearedTWINT Bit,TWI Module starts immediately START Signal transmission.

2. when START The status is sent,TWCR of TWINT The flag bit will be set,TWSR The new status code updated to indicate START The signal was sent successfully.

3. Application View TWSR The value of, OK START The status has been successfully sent. in caseTWSR Displayed as other values, the application can perform some special operations, such as calling an error handler. When it is determined that the status code is consistent with expectations, the program willSLA+W The value of is loaded into TWDR In the register.TWDR Registers can be used in address and data at the same time. Then the software goesTWCR Write a specific value to the register, indicating TWI Hardware send TWDR middle SLA+W Value. The written value will be explained in detail later. To be set in the written valueTWINT, To clear TWINT Flag bit.TWCR Register TWINT Set period TWI No action will be initiated. Once the software is clearedTWINT Bit,TWI The module immediately starts the transmission of the address packet.

4. When the address packet is sent,TWCR of TWINT The flag bit will be set,TWSR Update to a new status code, indicating that the address packet was successfully sent. The status code will also reflect whether the slave responds to the address packet.

5. Application View TWSR Value, to confirm that the address packet has been successfully sent, and the received ACK For the expected value. in caseTWSR Displayed as other values, the application can perform some special operations, such as calling an error handler. When it is determined that the status code is consistent with expectations, the program willData The value of is loaded into TWDR In the register. Then the software goesTWCR Write a specific value to the register, indicating TWI Hardware send TWDR middle Data Value. The written value will be explained in detail later. To be set in the written valueTWINT, To clear TWINT Flag bit.TWCR Register TWINT Set period TWI No action will be initiated. Once the software is clearedTWINT Bit,TWI The module immediately starts the transmission of the data packet.

6. When the data packet is sent,TWCR of TWINT The flag bit will be set,TWSR Update to a new status code, indicating that the data packet was successfully sent. The status code will also reflect whether the slave responds to the data packet.

7. Application View TWSR Value, to determine that the data packet has been successfully sent, the received ACK For the expected value. in caseTWSR Displayed as other values, the application can perform some special operations, such as calling an error handler. When it is determined that the status code is consistent with the expectation, the software goesTWCR Write a specific value to the register, indicating TWI Hardware send STOP signal. The written value will be explained in detail later. To be set in the written valueTWINT, To clear TWINT Flag bit.TWCR Register TWINT Set period TWI No action will be initiated. Once the software is clearedTWINT Bit,TWI Module starts immediately STOP Signal transmission. It’s important to note that inSTOP After the signal is sent TWINT Will not be set.

Although the example is relatively simple, it contains TWI All rules in the data transfer process. Summarized as follows:

?? when TWI When you complete an operation and wait for feedback from the application,TWINT The flag is set.SCL The clock line will be kept

Pull down until TWINT Cleared

?? when TWINT The flag is set, the user must update all TWI The value of the register is the same as the next TWI The value related to the bus cycle. E.g,TWDR The register must be loaded with the value to be sent in the next bus cycle. When all the

?? registers are updated and other necessary operations are completed at the same time, the application program writesTWCR register. WritingTWCR Time,TWINT Bit must be set to clear TWINT Sign.TWINT After being cleared,TWI Start execution by TWCR Set operation.

Transmission mode

TWI Can work below 4 The main mode: host transmitter (MT), host receiver (MR), slave transmitter (ST) And slave receiver (SR). Multiple modes can be used in the same application. E.g,TWI can use MT Pattern towards TWI EEPROMWrite data, use MR Pattern from EEPROMRead the data. If there are other hosts on the system, some may also go toTWI To send data, it will use SR mode. It is up to the application software to decide which mode to adopt.

These modes will be explained in detail below. In the data transmission in each mode, pictures are combined to describe possible status codes. These pictures contain the following abbreviations:

S: Start status

Rs: REPEATED START status

R: Read operation flag (SDA Is high)

W: Write operation flag (SDA Low level)

A: Response bit (SDA Low level)

NA: No response bit (SDA Is high)

Data:8 Bit data byte

P: STOP status

SLA: Slave address

The circle in the picture is used to indicate TWINT The flag is set, and the number in the circle indicates TWSR The status code in the register, where the prescaler control bit is masked as"0". In these places, the application must perform corresponding actions to continue or completeTWI transmission.TWI The transfer will be suspended until TWINT The flag bit is cleared.

when TWINT The flag is set,TWSR The status code in is used to determine the appropriate software operation. The details of the software operation and subsequent serial transmission required under each status code are given in each table. Note in the tableTWSR The prescaler control bit in is masked as"0".

Host sending mode

In the host sending mode,TWI A certain number of data bytes will be sent to the slave receiver. In order to enter host mode, you must sendSTART signal. The next address packet format is determinedTWI Whether to enter host transmitter mode or host receiver mode. If sendSLA+W, Then enter the host sending mode. If sendSLA+R, Then enter the host receiving mode. This

The status codes mentioned in the chapter assume that the prescaler control bit is"0".

Through to TWCR Write the following values to the register to send START signal:

TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
1	x	1	0	x	1	0	x

TWEN Position must be set"1"To enable TWI interface,TWSTA Set"1"To send START signal,TWINT Set"1"To clear

TWINT Flag bit.TWI The module detects the bus status and sends it immediately when the bus is idle START signal. When sentSTART After the hardware is set TWINT Flag bit, updated at the same time TWSR The status code is 0x08.

In order to enter the host sending mode, you must send SLA+W. This can be done by the following operations. Go firstTWDR Register write Enter SLA+WAnd then to TWINT Bit write"1"Cleared TWINT Flag bit to continue transmission, that is, to TWCR Register write down Column value to send SLA+W:

TWINT TWEA		TWSTA	TWSTO	TWWC	TWEN	-	TWIE
1 x		0	0	x	1	0	x

when SLA+W After the transmission is completed and the response signal is received,TWINT Is set again, and at the same time TWSR The status code is updated. The possible status codes are0x18,0x20 or 0x38. The appropriate response under each status code will be described in detail in the status code table.

when SLA+W After sending successfully, you can start sending data packets. This can be done by going toTWDR The register writes data to complete. TWDR only at TWINT It can be written only when the flag bit is high. Otherwise, the access is ignored and the conflict flag bit is written at the same timeTWWC Will be set. UpdatedTWDR Afterwards TWINT Bit write"1"Cleared TWINT Flag bit to continue transmission. On the goTWCR Write the following values to the register to send data:

TWINT TWEA TWSTA			TWSTO	TWWC	TWEN	-	TWIE
1 x 0			0	x	1	0	x

When the data packet is sent and the response signal is received,TWINT Is set again, and at the same time TWSR The status code is updated. The possible status codes are0x28 or 0x30. The appropriate response under each status code will be described in detail in the status code table.

When the data is sent successfully, you can continue to send data packets. This process is repeated until the last byte is sent. Host generated STOP Signal or REPEATED START The entire transmission of the signal ends.

Through to TWCR Write the following values to the register to send STOP signal:

TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
1	x	0	1	x	1	0	x

Through to TWCR Write the following values to the register to send REPEATED START signal:

TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
1	x	1	0	x	1	0	x

Sending REPEATED START(The status code is 0x10)after that,TWI The interface can visit the same slave again, or visit Ask the new slave without sending STOP signal.REPEATED START This allows the master to switch between different slaves, master transmitter and master receiver modes without losing control of the bus.

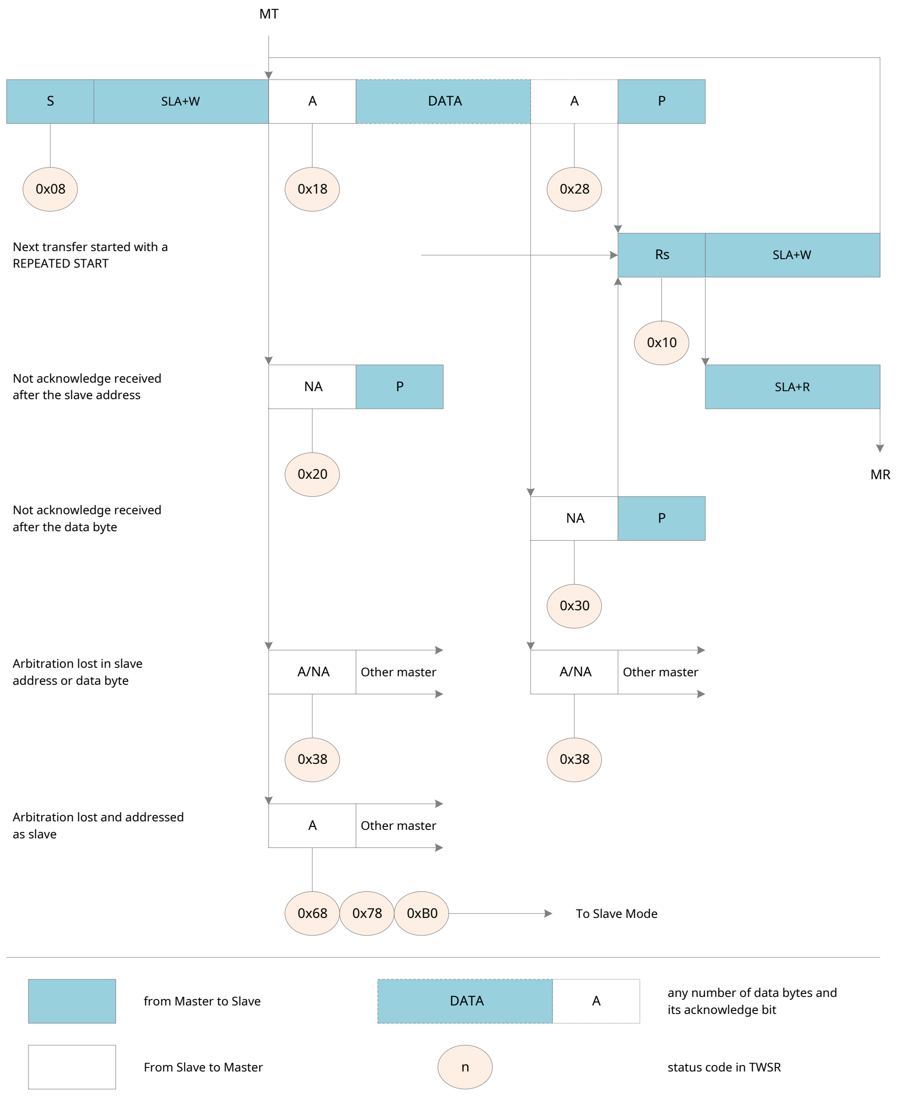
The status codes and corresponding operations in the host sending mode are shown in the following table:

Status code table of host sending mode

status code	Bus and hard Piece status	Application software response					Hardware's next move
		Read/write	Correct TWCR Operation				
		TWDR	STA STO TWINT TWEA				
0x08	START sent give away	load SLA+W	0 0 1 x				Will send SLA+W; Will receive ACK or NACK
0x10	REPEATED START sent	load SLA+W	0	0	1	x	Will send SLA+W; Will receive ACK or NACK

	give away	load SLA+R	0	0	1	x	Will send SLA+R； Will receive ACK or NACK; Will switch to MR mode
0x18	SLA+W Already send; received ACK	load data	0	0	1	x	Will send data; Will receive ACK or NACK
		No fuck Make	1	0	1	x	Will send REPEATED START
		No fuck Make	0	1	1	x	Will send STOP； Will reset TWSTO Sign
		No fuck Make	1	1	1	x	Will send STOP； Will reset TWSTO Sign Will send START
0x20	SLA+W Already send; received NACK	load data	0	0	1	x	Will send data; Will receive ACK or NACK
		No fuck Make	1	0	1	x	Will send REPEATED START
		No fuck Make	0	1	1	x	Will send STOP； Will reset TWSTO Sign
		No fuck Make	1	1	1	x	Will send STOP； Will reset TWSTO Sign Will send START
0x28	Data byte Sent; received Roger that ACK	load data	0	0	1	x	Will send data; Will receive ACK or NACK
		No fuck Make	1	0	1	x	Will send REPEATED START
		No fuck Make	0	1	1	x	Will send STOP； Will reset TWSTO Sign
		No fuck Make	1	1	1	x	Will send STOP； Will reset TWSTO Sign Will send START
0x30	Data byte Sent; received Roger that NACK	load data	0	0	1	x	Will send data; Will receive ACK or NACK
		No fuck Make	1	0	1	x	Will send REPEATED START
		No fuck Make	0	1	1	x	Will send STOP； Will reset TWSTO Sign
		No fuck Make	1	1	1	x	Will send STOP； Will reset TWSTO Sign Will send START
0x38	SLA+W or Data arbitration failure	No fuck Make	0	0	1	x	Will release the bus; Unaddressed slave mode
		No fuck Make	1	0	1	x	Will be sent when free START

The format and status of the host sending mode are shown in the figure below:


MT
S
SLA+W
A
DATA
A
P
0x08
0x18
0x28
Next transfer started with a REPEATED START
Rs
SLA+W
0x10
Not acknowledge received after the slave address
NA
P
SLA+R
0x20
MR
Not acknowledge received after the data byte
NA
P
0x30
Arbitration lost in slave address or data byte
A/NA
Other master
A/NA
Other master
0x38
0x38
Arbitration lost and addressed as slave
A
Other master
0x68
0x78
0xB0
To Slave Mode
from Master to Slave
DATA
A
any number of data bytes and its acknowledge bit
From Slave to Master
n
status code in TWSR

Format and state diagram of host sending mode

Host receiving mode

In the host receiving mode,TWI A certain number of data bytes will be received from the slave transmitter. In order to enter host mode, you must sendSTART signal. The next address packet format is determinedTWI Whether to enter host transmitter mode or host receiver mode. If sendSLA+W, Then enter the host sending mode. If sendSLA+R, Then enter the host receiving mode. This

The status codes mentioned in the chapter assume that the prescaler control bit is"0".

Through to TWCR Write the following values to the register to send START signal:

TWINT TWEA TWSTA TWSTO TWWC					TWEN	-	TWIE
1 x 1 0 x					1	0	x

TWEN Position must be set"1"To enable TWI interface,TWSTA Set"1"To send START signal,TWINT Set"1"To clear TWINT Flag bit.TWI The module detects the bus status and sends it immediately when the bus is idle START signal. When sentSTART After the hardware is set TWINT Flag bit, updated at the same time TWSR The status code is 0x08.

In order to enter the host receiving mode, you must send SLA+R. This can be done by the following operations. Go firstTWDR Register write Enter SLA+RAnd then to TWINT Bit write"1"Cleared TWINT Flag bit to continue transmission, that is, to TWCR Register write down Column value to send SLA+R:

TWINT TWEA		TWSTA	TWSTO	TWWC	TWEN	-	TWIE
1 x		0	0	x	1	0	x

when SLA+R After the transmission is completed and the response signal is received,TWINT Is set again, and at the same time TWSR The status code is updated. possible The status code is 0x38,0x40 or 0x48. The appropriate response under each status code will be described in detail in the status code table.

when SLA+R After sending successfully, you can start to receive data packets. Through toTWINT Bit write"1"Cleared TWINT Flag bit Continue to receive. On the goTWCR Write the following values to the register to start receiving:

TWINT TWEA TWSTA TWSTO TWWC					TWEN	-	TWIE
1 x 0 0 x					1	0	x

When the data packet is received and the response signal is sent,TWINT Is set again, and at the same time TWSR The status code is updated. The possible status codes are0x50 or 0x58. The appropriate response under each status code will be described in detail in the status code table.

When the data is successfully received, you can continue to receive data packets. This process is repeated until the last byte is received. After the host receives the last byte, it must sendNACK The response signal is sent to the slave transmitter. Host generatedSTOP signal or REPEATED START The entire reception of the signal ends.

Through to TWCR Write the following values to the register to send STOP signal:

TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
1	x	0	1	x	1	0	x

Through to TWCR Write the following values to the register to send REPEATED START signal:

TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
1	x	1	0	x	1	0	x

Sending REPEATED START(The status code is 0x10)after that,TWI The interface can visit the same host again, or visit Ask the new host without sending STOP signal.REPEATED START This allows the master to switch between different slaves, master transmitter and master receiver modes without losing control of the bus.

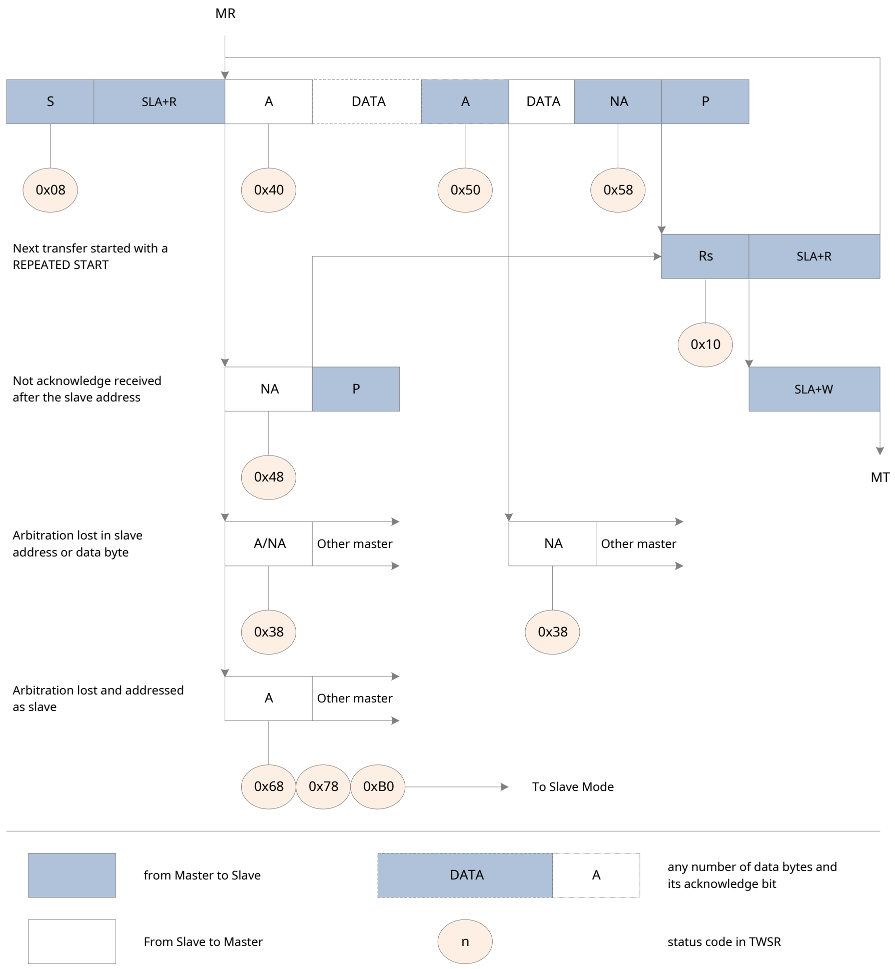
The status codes and corresponding operations in the host receiving mode are shown in the following table:

Status code table of host receiving mode

status code	Bus and hard Piece status	Application software response					Hardware's next move
		Read/write	Correct TWCR Operation				
		TWDR	STA	STO	TWINT TWEA		
0x08	START Already send	load SLA+R	0	0	1 x		Will send SLA+R； Will receive ACK or NACK
0x10	REPEATED START Already	load SLA+R	0	0	1	x	Will send SLA+R； Will receive ACK or NACK

	send	load SLA+W	0	0	1	x	Will send SLA+W； Will receive ACK or NACK; Will switch to MT mode
0x38	SLA+R or Data arbitration failure	No fuck Make	0	0	1	x	Will release the bus; Unaddressed slave mode
		No fuck Make	1	0	1	x	Will be sent when free START
0x40	SLA+R Already send; received ACK	No fuck Make	0	0	1	0	Will receive data; Will send NACK
		No fuck Make	0	0	1	1	Will receive data; Will send ACK
0x48	SLA+R Already send; received NACK	No fuck Make	1	0	1	x	Will send REPEATED START
		No fuck Make	0	1	1	x	Will send STOP； Will reset TWSTO Sign
		No fuck Make	1	1	1	x	Will send STOP； Will reset TWSTO Sign Will send START
0x50	Data byte Received; ACK sent give away	Read data	0	0	1	0	Will receive data; Will send NACK
		Read data	0	0	1	1	Will receive data; Will send ACK
0x58	Data byte Received; NACK Already send	Read data	1	0	1	x	Will send REPEATED START
		Read data	0	1	1	x	Will send STOP； Will reset TWSTO Sign
		Read data	1	1	1	x	Will send STOP； Will reset TWSTO Sign Will send START

The format and status of the host receiving mode are shown in the figure below:


MR
S
SLA+R
A
DATA
A
DATA
NA
P
0x08
0x40
0x50
0x58
Next transfer started with a REPEATED START
Rs
SLA+R
0x10
Not acknowledge received after the slave address
NA
P
SLA+W
0x48
MT
Arbitration lost in slave address or data byte
A/NA
Other master
NA
Other master
0x38
0x38
Arbitration lost and addressed as slave
A
Other master
0x68
0x78
0xB0
To Slave Mode
from Master to Slave
DATA
A
any number of data bytes and its acknowledge bit
From Slave to Master
n
status code in TWSR

Format and state diagram of host receiving mode

Slave receiving mode

In the slave receiving mode, a certain number of data bytes can be received from the master transmitter. The status codes mentioned in this chapter

It is assumed that the prescaler control bit is"0".

To start the slave receiving mode, you need to set TWAR with TWCR register.

TWAR Need to be set as follows:

TWA6 TWA5		TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
Device slave address							

TWAR height of 7 When the bit is addressed by the host TWI The slave address to which the interface will respond. IfLSB Position,TWI Will respond to broadcast calls

Call address (0x00), otherwise ignore the broadcast call address.

TWCR Need to be set as follows:

TWINT TWEA		TWSTA	TWSTO	TWWC	TWEN	-	TWIE
0 1		0	0	0	1	0	x

TWEN Must be set to enable TWI interface,TWEA Must be set to make the master address (slave address or general call) to return confirmation message when it reaches itself ACK.TWSTA with TWSTO Must be cleared.

initialization TWAR with TWCR after that,TWI The interface waits until its own slave address (or broadcast address) is addressed. When the data direction bit following the slave address is"0"(Indicating a write operation),TWI Enter the slave receiving mode. When the data direction bit is"1" (Representing a read operation),TWI Enter the slave sending mode. After receiving its own slave address and write operation flag,TWINT The flag bit is set, and the valid status code is also updated to TWSR in. The appropriate response under each status code will be described in detail in the status code table. It should be noted that when the host modeTWI After the arbitration fails, you can also enter the slave receiving mode (see status code 0x68 with 0x78).

If during transmission TWEA Bit is reset,TWI Will return after receiving a byte NACK(High level) to SDA on-line. This can be used to indicate that the slave cannot receive more data. whenTWEA Bit is"0"Time,TWI It will not respond to its own slave address. butTWI Will still monitor the bus, once TWEA After being set, the address recognition and response can be restored. In other words, you can useTWEA Temporarily TWI The interface is isolated from the bus.

In sleep modes other than idle mode,TWI The clock of the interface can be turned off. If the slave receiver mode is enabled, the interface will use the bus clock to continue responding to the slave address or broadcast address. Address match will wake upMCU. During the wake-up period, TWI Interface will remain SCL Is low until TWINT The flag is cleared. whenTWI After the interface clock is restored to normal Receive more data.

The status codes of the slave receiving mode are shown in the following table:

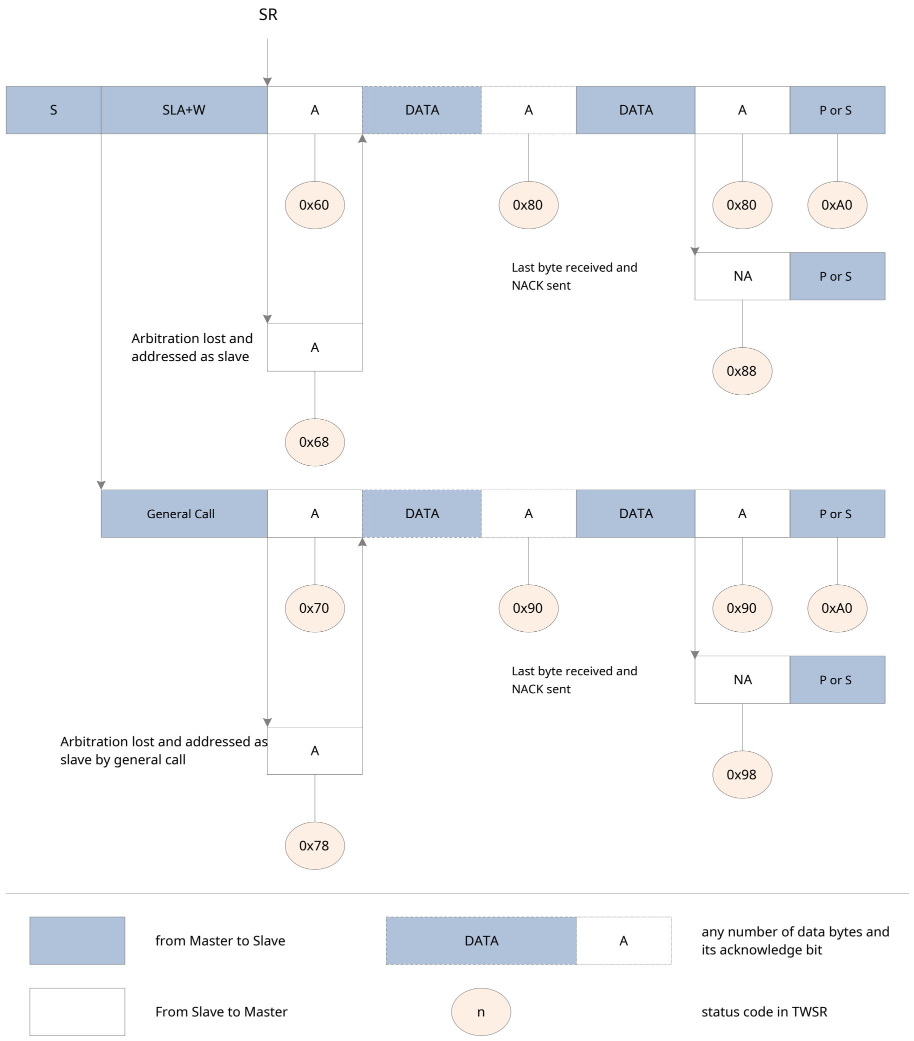
Status code table of slave receiving mode

status code	Bus and hardware status	Application software response					Hardware's next move
		Read/write	Correct TWCR Operation				
		TWDR	STA	STO	TWINT	TWEA	
0x60	SLA+WReceived; ACK Has been sent	No fuck Make	x	0	1	0	Will receive data; Will send NACK
		No fuck Make	x	0	1	1	Will receive data; Will send ACK
0x68	send SLA+R/W Time Arbitration failed SLA+WReceived; ACK Has been sent	No fuck Make	x	0	1	0	Will receive data; Will send NACK
		No fuck Make	x	0	1	1	Will receive data; Will send ACK
0x70	Broadcast address has been received; ACK Has been sent	no operation Make	x	0	1	0	Will receive data; Will send NACK
		No fuck Make	x	0	1	1	Will receive data; Will send ACK
0x78	send SLA+R/W Time Arbitration failed SLA+WReceived; ACK Has been sent	No fuck Make	x	0	1	0	Will receive data; Will send NACK
		No fuck Make	x	0	1	1	Will receive data; Will send ACK
0x80	Own data has been received; ACK Has been sent	read data	x	0	1	0	Will receive data; Will send NACK
		Read data	x	0	1	1	Will receive data; Will send ACK
0x88	Own data has been received;	read	0	0	1	0	Will switch to unaddressed

	NACK Has been sent	data					Slave mode Will not respond to slave ground Address and broadcast
		Read data	0	0	1	1	Will switch to unaddressed Slave mode Will respond to slave ground site; TWGCE=1 Time will ring Should be broadcast
		Read data	1	0	1	0	Will switch to unaddressed Slave mode Will not respond to slave ground Address and broadcast; Will send when the bus is idle give away START
		Read data	1	0	1	1	Will switch to unaddressed Slave mode Will respond to slave ground site; TWGCE=1 Time will ring Should be broadcast Will send when the bus is idle give away START
0x90	Broadcast data has been received; read ACK Has been sent	data	x	0	1	0	Will receive data; Will send NACK
		Read data	x	0	1	1	Will receive data; Will send ACK
0x98	Broadcast data has been received; read NACK Has been sent	data	0	0	1	0	Will switch to unaddressed Slave mode Will not respond to slave ground Address and broadcast
		Read data	0	0	1	1	Will switch to unaddressed Slave mode Will respond to slave ground site; TWGCE=1 Time will ring Should be broadcast
		Read data	1	0	1	0	Will switch to unaddressed Slave mode Will not respond to slave ground Address and broadcast; Will send when the bus is idle give away START
		Read	1	0	1	1	Will switch to unaddressed

		data					Slave mode Will respond to slave ground site; TWGCE=1 Time will ring Should be broadcast Will send when the bus is idle give away START
0xA0	Receive when the slave is working To STOP or REPEATED START	No fuck Make	0	0	1	0	Will switch to unaddressed Slave mode Will not respond to slave ground Address and broadcast
		No fuck Make	0	0	1	1	Will switch to unaddressed Slave mode Will respond to slave ground site; TWGCE=1 Time will ring Should be broadcast
		No fuck Make	1	0	1	0	Will switch to unaddressed Slave mode Will not respond to slave ground Address and broadcast; Will send when the bus is idle give away START
		No fuck Make	1	0	1	1	Will switch to unaddressed Slave mode Will respond to slave ground site; TWGCE=1 Time will ring Should be broadcast Will send when the bus is idle give away START

The format and state diagram of the slave receiving mode are as follows:


SR
S
SLA+W
A
DATA
A
DATA
A
P or S
0x60
0x80
0x80
0xA0
Last byte received and NACK sent
NA
P or S
Arbitration lost and addressed as slave
A
0x88
0x68
General Call
A
DATA
A
DATA
A
P or S
0x70
0x90
0x90
0xA0
Last byte received and NACK sent
NA
P or S
Arbitration lost and addressed as slave by general call
A
0x98
0x78
from Master to Slave
DATA
A
any number of data bytes and its acknowledge bit
From Slave to Master
n
status code in TWSR

Format and state diagram of slave receiving mode

Slave sending mode

In the slave sending mode, a certain number of data bytes can be sent to the host receiver. The status codes mentioned in this chapter

It is assumed that the prescaler control bit is"0".

To start the slave receiving mode, you need to set TWAR with TWCR register.

TWAR Need to be set as follows:

TWA6 TWA5		TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
Device slave address							

TWAR height of 7 When the bit is addressed by the host TWI The slave address to which the interface will respond. IfLSB Position,TWI Will respond to broadcast calls

Call address (0x00), otherwise ignore the broadcast call address.

TWCR Need to be set as follows:

TWINT TWEA		TWSTA	TWSTO	TWWC	TWEN	-	TWIE
0 1		0	0	0	1	0	x

TWEN Must be set to enable TWI interface,TWEA Must be set to make the master address (slave address or general call) to return confirmation message when it reaches itself ACK.TWSTA with TWSTO Must be cleared.

initialization TWAR with TWCR after that,TWI The interface waits until its own slave address (or broadcast address) is addressed. When the data direction bit following the slave address is"0"(Indicating a write operation),TWI Enter the slave receiving mode. When the data direction bit is"1" (Representing a read operation),TWI Enter the slave sending mode. After receiving its own slave address and read operation flag,TWINT The flag bit is set, and the valid status code is also updated to TWSR in. The appropriate response under each status code will be described in detail in the status code table. It should be noted that when the host modeTWI After the arbitration fails, you can also enter the slave transmission mode (see status code 0xB0).

If during transmission TWEA Bit is reset,TWI Will switch to unaddressed slave mode after sending the last byte. The host receiver gives the last byte of transmissionNACK or ACK Rear,TWSR The status code in the register will be updated to 0xC0 or 0xC8. If the master receiver continues to transmit, the slave transmitter will not respond, and the master will receive all "1"Data (ie 0xFF). When the slave has sent the last byte of data (TWEA Is cleared) and expect to get NACK Response, and the host wants to receive more data and send ACK In response,TWSR Will be updated to 0xC8.

when TWEA Bit is"0"Time,TWI It will not respond to its own slave address. butTWI Will still monitor the bus, once TWEA After being set, the address recognition and response can be restored. In other words, you can useTWEA Temporarily TWI The interface is isolated from the bus.

In sleep modes other than idle mode,TWI The clock of the interface can be turned off. If the slave receiver mode is enabled, the interface will use the bus clock to continue responding to the slave address or broadcast address. Address match will wake upMCU. During the wake-up period, TWI Interface will remain SCL Is low until TWINT The flag is cleared. whenTWI After the interface clock is restored to normal Receive more data.

The status code of the slave sending mode is shown in the following table:

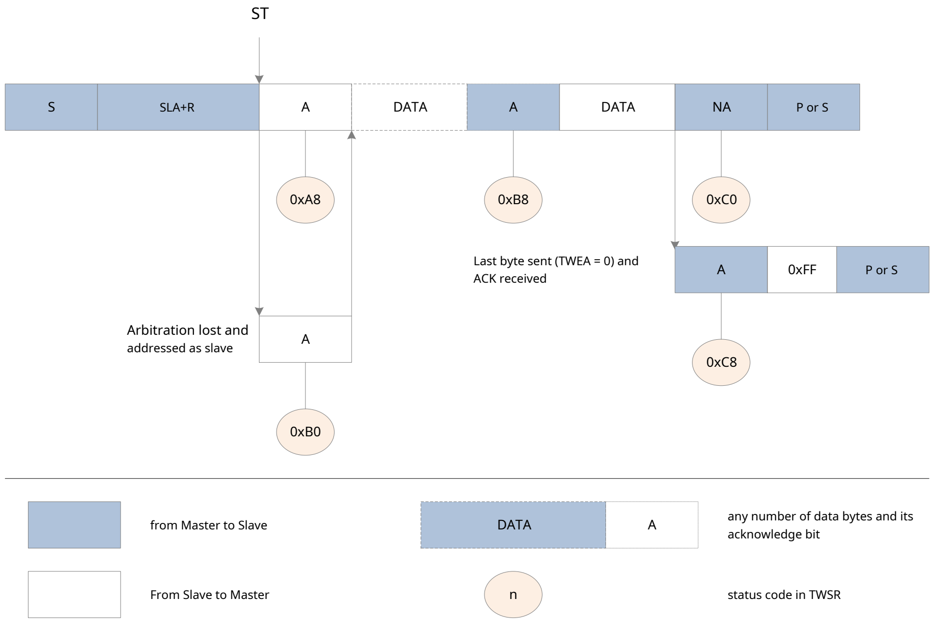
Status code table of slave sending mode

status code	Bus and hard Piece status	Application software response					Hardware's next move
		Read/write	Correct TWCR Operation				
		TWDR	STA	STO	TWINT	TWEA	
0xA8	SLA+R Already receive; ACK sent give away	Download Data	x	0	1	0	Will send the last number according to; Expect to receive NACK
		Download Data	x	0	1	1	Will send data; Will receive ACK
0xB0	hair give away SLA+R/W Arbitration defeat; SLA+R Already receive; ACK sent give away	Download Data	x	0	1	0	Will send the last number according to; Expect to receive NACK
		Download Data	x	0	1	1	Will send data; Will receive ACK

0xB8	Data has been sent give away; ACK Received	Download Data	x	0	1	0	Will send the last number according to; Expect to receive NACK
		Download Data	x	0	1	1	Will send data; Will receive ACK
0xC0	Data has been sent give away; NACK Already receive	No action	0	0	1	0	Will switch to unaddressed slave mode; Will not respond to the slave address and broadcast
		No action	0	0	1	1	Will switch to unaddressed slave mode; Will respond to the slave address; TWGCE=1 Will respond widely broadcast
		No action	1	0	1	0	Will switch to unaddressed slave mode; Will not respond to the slave address and broadcast; Will be sent when the bus is free START
		No action	1	0	1	1	Will switch to unaddressed slave mode; Will respond to the slave address; TWGCE=1 Will respond widely broadcast; Will be sent when the bus is free START
0xC8	the last one Data has been sent give away; ACK picked up Receive	No action	0	0	1	0	Will switch to unaddressed slave mode; Will not respond to the slave address and broadcast
		No action	0	0	1	1	Will switch to unaddressed slave mode; Will respond to the slave address; TWGCE=1 Will respond widely broadcast
		No action	1	0	1	0	Will switch to unaddressed slave mode; Will not respond to the slave address and broadcast; Will be sent when the bus is free START
		No action	1	0	1	1	Will switch to unaddressed slave mode;

							Will respond to the slave address; TWGCE=1 Will respond widely broadcast; Will be sent when the bus is free START

The format and status of the slave sending mode are shown in the figure below:


ST
S
SLA+R
A
DATA
A
DATA
NA
P or S
0xA8
0xB8
0xC0
Last byte sent (TWEA = 0) and ACK received
A
0xFF
P or S
Arbitration lost and addressed as slave
A
0xC8
0xB0
from Master to Slave
DATA
A
any number of data bytes and its acknowledge bit
From Slave to Master
n
status code in TWSR

Format and state diagram of slave sending mode

Other status

There are two status codes without corresponding TWI The status definition is shown in the following table:

Other status code tables

status code	Bus and hardware status state	Application software response					Hardware's next move
		Read/write	Correct TWCR Operation				
		TWDR	STA	STO	TWINT	TWEA	
0xF8	No state information; no operation TWINT = 0		Do not operate TWCR				Wait or perform current operation Make
0x00	Unlawful START or STOP cause Bus error	No action	0	1	1	x	Only affect internal hardware; Will not send STOP To On the bus; bus release And clear TWSTO Bit

status code 0xF8 Indicates that there is currently no relevant information because TWINT Marked as"0". This state may occur inTWI The interface is not involved in serial transmission or the current transmission has not been completed.

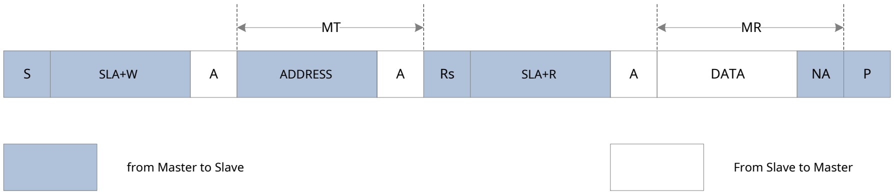
status 0x00 Indicates that a bus error occurred during serial transmission. Be illegalSTART or STOP Bus errors will occur when they occur. For example, in address and data, address andACK Appeared in between START or STOP. Bus error will be set TWINT. In order to recover from the error, it must be setTWSTOAnd by writing"1"To clear TWINT. This will makeTWI The interface enters the unaddressed slave mode without generating STOP, And release SCL with SDA, And cleared TWSTO Bit. Combination mode

In some cases, in order to complete the desired work, several types of TWI Combine the patterns. For example, from serialEEPROM To read data, a typical transmission includes the following steps:

1. The transmission must be started;
2. Must tell EEPROMThe location where the data should be read;
3. Must complete the read operation;
4. The transfer must end.

Note that data can be transferred from the master to the slave, and vice versa. The host tells the slave where to read the data, using the host sending mode. Next, read data from the slave, using the master receiving mode. The direction of transmission will change. The host must maintain bus control at all stages, and all steps are uninterrupted operations. If in a multi-host system, in step2 with 3 In the meantime, another host changes the location of reading data, which breaks this principle. The location of the host reading data Would be wrong. Change the direction of data transmission by sending between the transmission address byte and the received data REPEATED START To achieve. sendREPEATED START After that, the host still has control of the bus.

The following figure describes this transfer process:



Combine multiple TWI Mode to access serial EEPROMFigure

Multi-host system and arbitration

If there are multiple hosts connected to the same TWI On the bus, one or more of them may start data transmission at the same time. TWI The protocol ensures that in this case, through an arbitration process, one of the hosts is allowed to transmit without losing data. Let's take two masters trying to send data to the slave as an example to describe the process of bus arbitration.

There are several different situations that will produce the bus arbitration process:

?? Two or more masters communicate with one slave at the same time. In this case, neither the master nor the slave knows that there is competition on the bus;

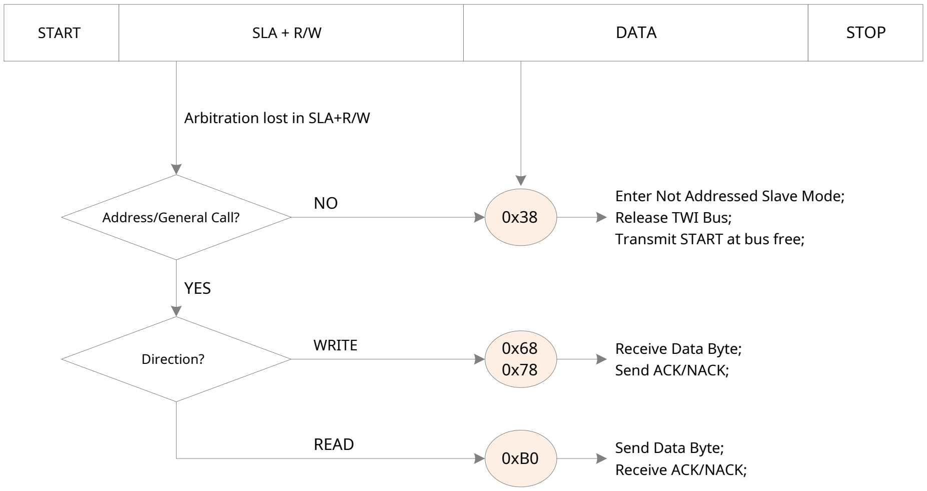
?? Two or more masters simultaneously access the same slave with different data or operating directions. In this case, arbitration will occur.READ/WRITE Bit or data bit. When other hosts go toSDA Send online"0"When, to SDA Send online "1"The host will lose arbitration. The failed master will switch to the unaddressed slave mode, or wait for the bus to be free to send a newSTART Signal, it all depends on the operation of the application software. Two or more masters

?? access different slaves. In this case, the bus arbitration occurs atSLA stage. When other hosts go toSDA Send online"0" When, to SDA Send online"1"The host will lose arbitration. inSLA The host that fails during bus arbitration will switch to slave mode and check whether it is addressed by the host that has obtained bus control. If it is addressed, it will enterSR or ST Mode, it depends on SLA Back READ/WRITE Bit. If not found

Address, it will switch to the unaddressed slave mode, or wait for the bus to be free to send a new START signal,

It depends on the operation of the application software.

The following figure describes the process of bus arbitration:



Bus arbitration process diagram

Register definition

TWI Register list

register	address	Defaults	description
TWBR	0x B8	0x00	TWI Bit rate register
TWSR	0xB9	0x00	TWI Status register
TWAR	0xBA	0x00	TWI Address register
TWDR	0xBB	0x00	TWI Data register
TWCR	0xBC	0x00	TWI Control register
TWAMR	0xBD	0x00	TWI Address mask register

TWBR – TWI Bit rate register

TWBR – TWI Bit rate register								
address: 0xB8					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3 TWBR2 TWBR1 TWBR0 R/WR/			
R/W	R/W	R/W	R/W	R/W	WR/WR/W			

Bit	Name	description
7:0	TWBR[7:0]	TWI Bit rate selection control bit. TWBR Is the frequency division factor of the bit rate generator. The bit rate generator is a frequency divider used to generateSCL clock. The formula for calculating the bit rate is as follows: $f_{scl} = f_{sys}/(16 + 2*TWBR*4_{TWPS})$.

TWSR – TWI Status register

TWSR – TWI Status register								
address: 0xB9					Defaults: 0xF8			
Bit	7	6	5	4	3 2		1	0
Name	TWS7	TWS6	TWS5	TWS4	TWS3-		TWPS1 TWPS0	
R/W	R/W	R/W	R/W	R/W	R/WR/W		R/WR/W	

Bit	Name	description
7:3	TWS[7:3]	TWI Status flag bit. 5 Bit TWS reaction TWI The state of the logic and the bus. Different status values are different The meaning of TWI The description of the working mode. FromTWSR The values read include 5 The state value of the bit and 2 The pre-scaling control bit of the bit, the pre-scaling bit should be masked when detecting the state"0". This is the setting of the status detection independent of the prescaler. Reserved.
2	-	
1	TWPS1	TWI The prescaler controls the high bit. TWPS1 with TWPS0 Make up together TWPS[1:0], Used to control the bit rate prescaler factor, and TWBR Control the bit rate together.
0	TWPS0	TWI The low bit of prescaler control. TWPS0 with TWPS1 Make up together TWPS[1:0], Used to control the bit rate prescaler Factor, and TWBR Control the bit rate together.

TWPS[1:0]	Prescaler factor
0	1
1	4
2	16
3	64

TWAR – TWI Address register

TWAR – TWI Address register								
address: 0xBA					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name	TWAR6	TWAR5 TWAR4 TWAR3			TWAR2 TWAR1 TWAR0 TWGCE R/			
R/W	R/W	R/WR/WR/W			WR/WR/WR/W			

Bit	Name	description
7:1	TWA[6:0]	TWI Slave address bits. TWA for TWI Slave address. whenTWI When working in slave mode,TWI Will be based on This address responds. This address is not required for host mode. But in a multi-master system, it is also necessary to set the slave address for other masters to access.
0	TWGCE	TWI Broadcast recognition enable control bit. When set TWGCE Bit is"1"When, enable TWI Bus broadcast recognition. When setTWGCE Bit is "0"When, prohibit TWI Bus broadcast recognition. whenTWGCE Set and the received address frame is 0x00 Time,TWI The module will respond to this bus broadcast.

TWDR – TWI Data register

TWDR – TWI Data register								
address: 0xBB					Defaults: 0xFF			
Bit	7	6	5	4	3	2	1	0
Name	TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7:0	TWD[7:0]	TWI Data register. TWD Is the next byte on the bus about to be transmitted, or just received from the bus The last byte reached.

TWCR – TWI Control register

TWCR – TWI Control register								
address: 0xBC					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
R/W	R/W	R/W	R/W	R/W	R	R/W	-	R/W

Bit	Name	description
7	TWINT	TWI Interrupt flag bit. when TWI Finish the current work and hope that the hardware will be set when the application software intervenes TWINT Bit. If the global interrupt is set andTWIE When the bit is set, it will produce TWI Interrupt,MCU Will execute TWI Interrupt service routine. whenTWINT When the flag is set,SCL The low level of the signal will be extended. TWINT The flag bit can only be written to this bit"1"Way to clear. Even if the interrupt service routine is executed, the hardware will not automatically clear this bit. At the same time, please note that clearing this bit will turn it on immediatelyTWI Operation. therefore, Clearing TWINTBefore the position, you must first complete the TWAR, TWAMR,TWSR with TWDR Register access.
6	TWEA	TWI Enable the response control bit. TWEA Bit control response pulse generation. When setTWEA Bit is"1", And one of the following conditions is met, it will be TWI A response pulse is generated on the bus: 1) Receive the slave address of the device; 2)TWGCE Receive a broadcast call when set; 3) One byte of data is received in the master receiving or slave receiving mode. When setTWEA Bit is"0" Time, the device temporarily and TWI The bus is disconnected. After the bit is set, the device resumes address recognition.
5	TWSTA	TWI Start state control bit. when CPU Wish to be TWI Need to be set when the host on the bus TWSTA Bit. The hardware will detect whether the bus is available, and when the bus is free, it will generate an initial state on the bus. When the bus is not idle,TWI Will wait until the stop state is detected, and then generate an initial state to declare that it wants to become the host. The software must be cleared after sending the initial stateTWSTA Bit.

4	TWSTO	TWI Stop status control bit. When in host mode TWSTO Bit is"1"Time,TWI A stop state will be generated on the bus, and then automatically cleared TWSTO Bit. In slave mode, setTWSTO Bit can make TWI Recover from the error state. At this time, the stop state will not be generated, it will only letTWI Return to a defined unaddressed slave mode and release at the same time SCL with SDA The signal line is in a high impedance state.
3	TWWC	TWI Write conflict flag. when TWINT When the flag bit is low, write TWDR Register will be set TWWC Flag bit. when TWINT When the flag bit is high, write TWDR Register will be cleared TWWC Flag bit.
2	TWEN	TWI Enable control bit. TWEN Bit enable TWI Operate and activate TWI interface. When setTWEN Bit is"1"Time, TWI control IO Pin is connected to SCL with SDA Pin. When setTWEN Bit is"0"Time, TWI The interface module is closed and all transmissions are terminated, including ongoing operations.
1	-	Reserved.
0	TWIE	TWI Interrupt enable control bit. When set TWIE Bit is"1", And when the global interrupt is set, as long as TWINT If the flag is high, it will activate TWI Interrupt request.

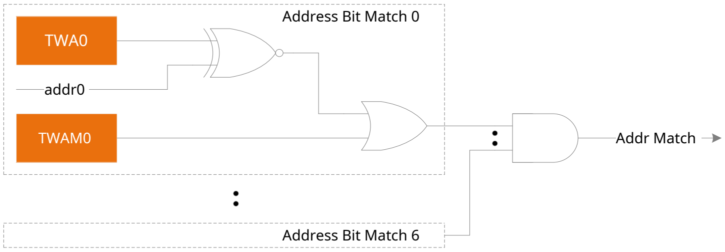
TWAMR – TWI Address mask register

TWAMR – TWI Address mask register								
address: 0xBD					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name	TWAR6	TWAR5 TWAR4 TWAR3			TWAR2 TWAR1 TWAR0 TWGCE R/			
R/W	R/W	R/WR/WR/W			WR/WR/WR/W			

Bit	Name	description
7:1	TWAM[6:0]	TWI Address mask control bit. TWAMfor 7 Bit TWI Slave address mask control. TWAMEach bit of is used to shield (prohibit)TWAR In the corresponding address bit. When the mask bit is set, the address matching logic will ignore the received address bit andTWA The comparison result of the corresponding bit. The following figure gives Detailed information about the address matching logic.
0	-	Reserved.

TWI Address matching logic

The picture below is TWI Address matching logic block diagram:



Analog comparator 0 (AC0)

Z 10mV Comparison accuracy

Z Factory offset calibration

Z stand by 3 Off-chip analog input

Z stand by ADC Multiplexed input (ADMUX)

Z Supports internal differential amplifier input (DFFO)

Z Support internal 8 Bit DAC enter(DAO)

Z Programmable output digital filter control

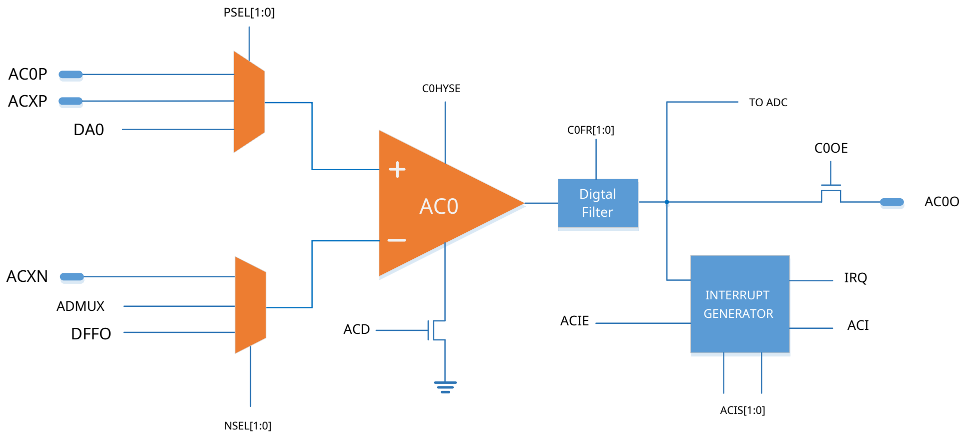
Summary

The analog comparator compares the level of the positive terminal and the negative terminal of the input comparator. When the voltage at the positive terminal is higher than the voltage at the negative terminal, the output of the analog comparator ACO Is set. whenACO When the level changes, the edge of the signal can be used to trigger an interrupt. output signalACO Can also be used to trigger a timer counter 1 Input capture and the timer generated PWM The output is controlled.

LGT8FX8P Integrated analog comparator AC0, Including a multi-channel analog input selector, the positive and negative input source of the comparator can be selected from an external port or from a variety of internally generated reference sources. The analog comparator itself supports offset calibration, which can ensure the consistency of the comparator's work. The comparator supports an optional hardware hysteresis function to improve the stability of the comparator output. At the same time, the output terminal of the comparator integrates a hardware that can be programmed with a digital filter, and the appropriate filter setting can be selected according to application requirements to obtain a more stable comparison output.

The output status of the comparator can be directly read through the register, or an interrupt request can be generated to achieve more efficient real-time events Capture function. The output of the comparator can also be directly output to the outsideIO port.

Op Amp/Analog Comparator 0 The structure diagram is shown in the figure below.



Analog comparator 0 Function diagram

Analog comparator input

Both input terminals of the analog comparator support a variety of optional input sources. Three inputs for the positive end are optional:

1. External independent analog input AC0P
2. Analog comparator 0/1 Common analog input ACXP
3. internal 8 Bit DAC Output DAO

The selection of the input source is controlled by the status register C0SR middle C0BGBit and C0XR Register C0PS0Bit common control, please refer to the register description part of this chapter for details.

AC0P for AC0 Dedicated positive mode input channel. Note that in different packagesAC0P The pin position is slightly different.QFP48 Encapsulated AC0P It is an independent port.QFP32 Encapsulate this AC0P Port and PD6 Parallel to one port.

ACXP Comparator 0/1 Common positive input.LGT8FX8P There are two analog comparators inside,ACXP The positive terminal multiplexing selector connected to the two comparators at the same time facilitates the cooperative work of the two comparators.

DAO From the inside 8 Bit DAC Output.DAC The reference source can be selected from system power, internal reference or input from external reference.DAC Please refer to the configuration DAC Related chapters.

C0BG	C0PS0	AC0 Positive input source
0	0	AC0P
0	1	ACXP
1	0	DAO
1	1	Turn off the positive input channel of the comparator

The negative input can also choose three different analog inputs:

1. Comparators 0/1 Common analog input ACXN
2. ADC Multiplexer output ADMUX
3. Internal differential amplifier output DFFO

The input channel of the negative terminal of the comparator is selected by ADC Modular ADCSRB In the register CME00/01 Position control. When the comparator negative input is selected asADMUX , Need to pass ADC Modular ADMUX register CHMUX Bit selection analog input channel, in this mode, the input of the comparator can be expanded more flexibly.

ACXN Comparator 0/1 Common negative input, easy to realize comparator 0/1 Collaborative work;

DFFO From the internal differential amplifier output. Differential amplifier optionalx1/x8/x16/x32 Gain control, can achieve small Signal detection and measurement.

CME01	CME00	AC0 Negative input source
0	0	ACXN
0	1	ADMUX
1	0	DFFO
1	1	Close the negative input channel of the comparator

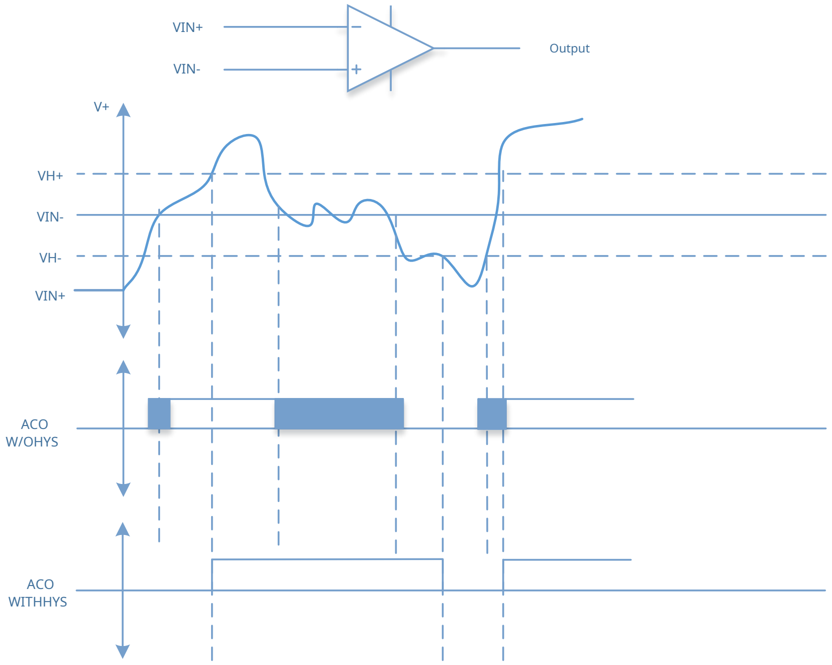
Comparator output filtering

The output of the comparator supports a controllable hysteresis. Users can passC0XR Register C0HYSE Bit enable hysteresis circuit. The hysteresis circuit can eliminate the unstable state of the comparator state change process and achieve the output filtering function.

It is recommended that users turn on the hysteresis circuit when using the comparator to obtain a stable comparator output. As shown in the figure below, the hysteresis circuit is located between the analog output and digital output of the comparator. When the input voltage of the positive terminal of the comparator

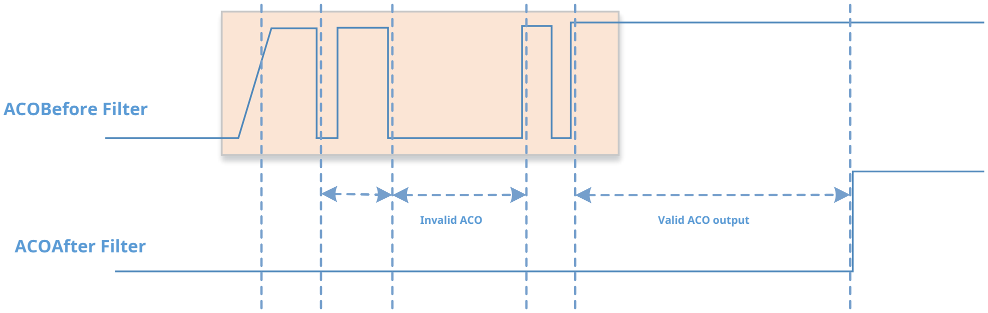
V_{IN+}more than the(V_{IN-} + V_{H+})When the comparator COUT Output is high; when V_{IN+}The voltage is less than (V_{IN-}- V_{H-})When the comparator output is low. The hysteresis circuit avoids the jitter caused by the circuit itself when the voltage at the positive terminal of the comparator is close to the voltage at the negative terminal.

The relationship between the hysteresis voltage of the comparator and the output of the comparator:



Although the hysteresis circuit is very effective in suppressing the voltage ripple close to the threshold of the comparator, in the actual application environment, the input The signal will be disturbed by different intensities. Strong interference may cause the input level to rise instantaneously, which exceeds the threshold range of the hysteresis circuit and cannot be effectively suppressed.LGT8FX8P A programmable digital filter is integrated at the output of the comparator, which can filter out the effect of instantaneous interference on the output of the comparator. The digital filter can select the appropriate filter time width according to the application requirements. Only when the output of the comparator is stable and continues to meet the filter time limit, the filter circuit will update the output of the comparator.

So as to achieve a more stable output result.



Comparator output filter timing

AC0 Digital filtering through C0XR Register C0FEN as well as C0FS Bit control, please refer to this chapter for specific setting methods Register definition part.

Comparator output and PWMcontrol

LGT8FX8P Support multi-channel PWMOutput,PWMThe signal can be used in conjunction with the comparator module. The output of the comparator can be used for direct shutdownPWMSignal to achieve a more flexible PWMProtection scheme.

versus PWMFor output related control, please refer to the relevant part of the timer chapter.

Register definition

C0SR – AC0 Control and status registers

C0SR **– AC0 Control and status registers**								
address: 0x50					Defaults: 0x80			
Bit	7	6	5	4	3	2	1	0
Name	C0D	C0BG	C0O	C0I	C0IE	C0IC	C0IS1	C0IS0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	C0D	Analog comparator disable bit. When set C0D Bit is"1"When, the analog comparator is turned off. When setC0D Bit is"0"When the analog comparator is turned on.
6	C0BG	Analog comparator 0 Positive input source selection bit.C0BG versus C0XR Register C0PS0 Bit common setting AC0 Positive input source, {C0BG, C0PS0} = 00 = AC0P As positive input 01 = ACXP As positive input 10 = internal DAC As the positive input 11 = shut down AC0 The positive input source
5	C0O	of the analog comparator output status bit. The output of the analog comparator is directly connected to the C0O Bit. Software can readC0O The value of the bit is used to obtain the output value of the analog comparator.
4	C0I	The interrupt flag of the analog comparator. When the output event of the analog comparator is triggered by C0IS In the bit-defined interrupt mode,C0I The bit is set. When the interrupt enable bitC0IE for"1"And when the global interrupt is set, an interrupt is generated. When executing the analog comparator interrupt service routine,C0I Will be automatically cleared, or C0I Bit write"1" This bit can also be cleared.
3	C0IE	The interrupt enable bit of the analog comparator. When set C0IE Bit is 1, And enable the global interrupt,AC0 The interrupt is enabled. When setC0IE Bit is 0,AC0 The interrupt is disabled. Analog comparator input capture
2	C0IC	enable bit C0IC = 1, Timer counter 1 The input capture source comes from the output of the analog comparator. C0IC = 0, Timer counter 1 The input capture source comes from an external pin ICP1. Analog
1	C0IS1	comparator interrupt mode control high bit.
0	C0IS0	Analog comparator interrupt mode control low bit.C0IS0 with C0IS1 Make up together C0IS[1:0], Used to control the interrupt trigger mode of the analog comparator.

C0IS[1:0]	Interrupt mode
00	ACO Trigger on rising or falling edge
01	Reserved.
10	ACO Falling edge trigger
11	ACO Trigger on the rising edge of

ADCSRB – ADC Control and status registers B

ADCSRB – ADC Control and status registers B								
address: 0x7B	Defaults: 0x00							
Bit	7	6	5	4	3	2	1	0
Name	CME01	CME00	CME11	CME10	ACTS	ADTS2	ADTS1	ADTS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	CME01	AC0 Negative input selection, CME0 = {CME01, CME00}
6	CME00	00: External port ACXN As AC0 Negative input 01: ADC Multiplexed output as AC0 Negative input 10: Differential amplifier output as AC0 Negative input 11: shut down AC0 Negative input source
5	CME11	AC1 Negative input selection, CME1 = {CME11, CME10}
4	CME10	00: External port ACXN As AC1 Negative input 01: External port AC1N As AC1 Negative input 10: ADC internal 1/5 Partial pressure as AC1 Negative input 11: The output of the differential op amp is as AC1 Negative input
3	ACHS	AC Trigger source channel selection 0 – AC0 Output as ADC Automatic conversion trigger source 1 – AC1 Output as ADC Automatic conversion trigger source
2:0	ADTS	see ADC Register description.

C0XR – AC0 Auxiliary control register

C0XR – AC0 Auxiliary control register								
address: 0x51	Defaults: 0x00							
Bit	7	6	5	4	3	2	1	0
Name	-	C0OE	C0HYSE	C0PS0	C0WKE	C0FEN	C0FS1	C0FS0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	-	Keep
6	C0OE	AC0 Enable control of comparator output to external port C0OE = 1, AC0 The comparator output to the external port PD2 C0OE = 0, Disable the output of the comparator to the external port
5	C0HYSE	AC0 The output hysteresis function enables control. 1 = Enable output hysteresis 0 = Disable output hysteresis
4	C0PS0	AC0 The positive input source selects the low bit. C0PS0 versus C0BG Joint control AC0 For the positive input source, please refer to C0SR Register definition
3	COWKE	AC0 Enable control for wake-up from sleep. 1 = Enable the wake-up function of the comparator output

		0 = Turn off the wake-up function of the comparator output
2	C0FEN	Comparator digital filter enable control. 1 = Enable digital filter 0 = Disable digital filter
1:0	C0FS[1:0]	Comparator digital filter width setting 00 = shut down 01 = 32us 10 = 64us 11 = 96us

Analog comparator 1 (AC1)

- Z 10mV Comparison accuracy
- Z Factory offset calibration
- Z stand by 4 Off-chip analog input
- Z Support internal 1/5 Voltage divider input (VDO)
- Z Supports internal differential amplifier input (DFFO)
- Z Support internal 8 Bit DAC enter(DAO)
- Z Programmable output filter control

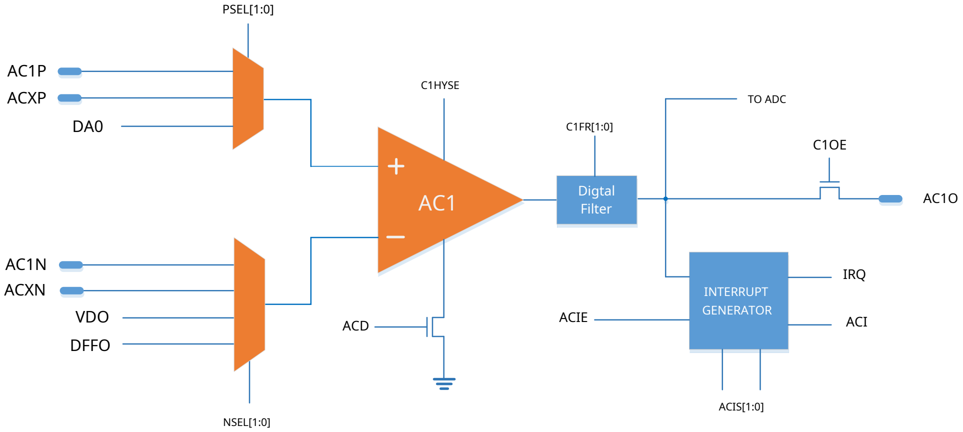
Summary

The analog comparator compares the level of the positive terminal and the negative terminal of the input comparator. When the voltage at the positive terminal is higher than the voltage at the negative terminal, the output of the analog comparator ACO Is set. whenACO When the level changes, the edge of the signal can be used to trigger an interrupt. output signalACO Can also be used to trigger a timer counter 1 Input capture and the timer generated PWM The output is controlled.

LGT8FX8P Integrated analog comparator AC1, Including a multi-channel analog input selector, the positive and negative input source of the comparator can be selected from an external port or from a variety of internally generated reference sources. The analog comparator itself supports offset calibration, which can ensure the consistency of the comparator's work. The comparator supports an optional hardware hysteresis function to improve the stability of the comparator output. At the same time, the output terminal of the comparator integrates a hardware that can be programmed with a digital filter, and the appropriate filter setting can be selected according to application requirements to obtain a more stable comparison output.

The output status of the comparator can be directly read through the register, or an interrupt request can be generated to achieve more efficient real-time events Capture function. The output of the comparator can also be directly output to the outsideIO port.

Analog comparator 1 The structure diagram is shown in the figure below.



Analog comparator 1 Module structure diagram

Analog comparator input

Both input terminals of the analog comparator support a variety of optional input sources. Three inputs for the positive end are optional:

1. External independent analog input AC1P
2. Analog comparator 0/1 Common analog input ACXP

3. internal 8 Bit DAC Output DAO

The selection of the input source is controlled by the status register C1SR middle C1BGBit and C1XR Register C1PS0Bit common control, please refer to the register description part of this chapter for details.

AC1P for AC1 Dedicated positive mode input channel.

ACXP Comparator 0/1 Common positive input.LGT8FX8P There are two analog comparators inside,ACXP The positive terminal multiplexing selector connected to the two comparators at the same time facilitates the cooperative work of the two comparators.

DAO From the inside 8 Bit DAC Output.DAC The reference source can be selected from system power, internal reference or input from external reference.DAC Please refer to the configuration DAC Related chapters.

C1BG	C1PS0	AC1 Positive input
0	0	ACXP
0	1	AC1P
1	0	DAO
1	1	Turn off the positive input channel of the comparator

Negative input can also be selected 4 Different analog inputs:

1. External analog input AC1N As AC1 Negative input
2. Comparators 0/1 Common negative input ACXN
3. ADC internal 1/5 Voltage divider output as AC1 Negative input
4. Internal differential amplifier output DFFO As AC1 Negative input

The input channel of the negative terminal of the comparator is selected by ADC Modular ADCSRB In the register CME11/10 Position control. When the comparator negative input is selected asADC When the internal multiplexer output, it needs to pass ADC Modular ADCSRC register VDS Bit selects the input reference source of the multi-channel voltage division.

ACXN Comparator 0/1 Common negative input, easy to realize comparator 0/1 Collaborative work;

DFFO From the internal differential amplifier output. Differential amplifier optionalx1/x8/x16/x32 Gain control, can achieve small Signal detection and measurement.

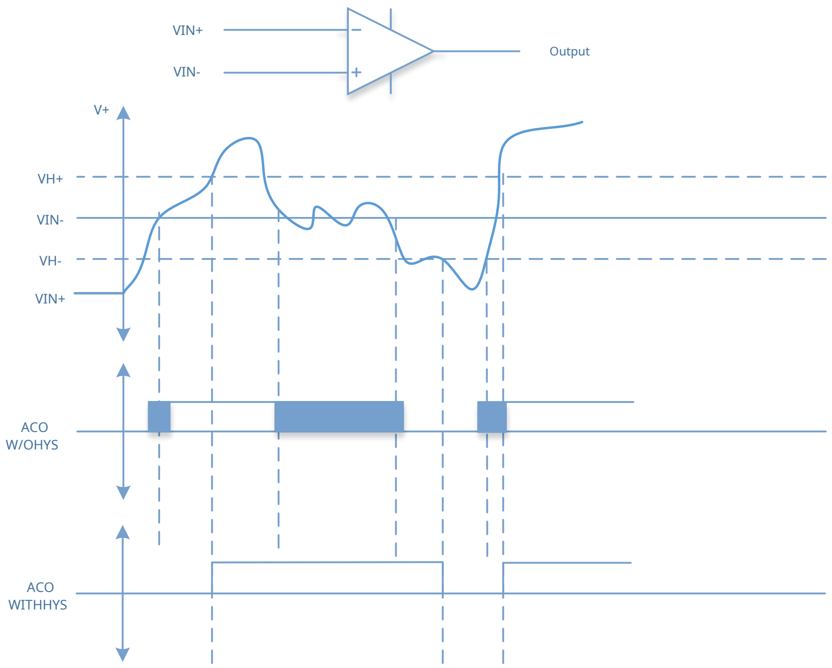
CME11	CME10	AC1 Negative input
0	0	ACXN
0	1	AC1N
1	0	VDO
1	1	DFFO

Comparator output filtering

The output of the comparator supports a controllable hysteresis. Users can passC1XR Register C1HYSE Bit enable hysteresis circuit. The hysteresis circuit can eliminate the unstable state of the comparator state change process and achieve the output filtering function.

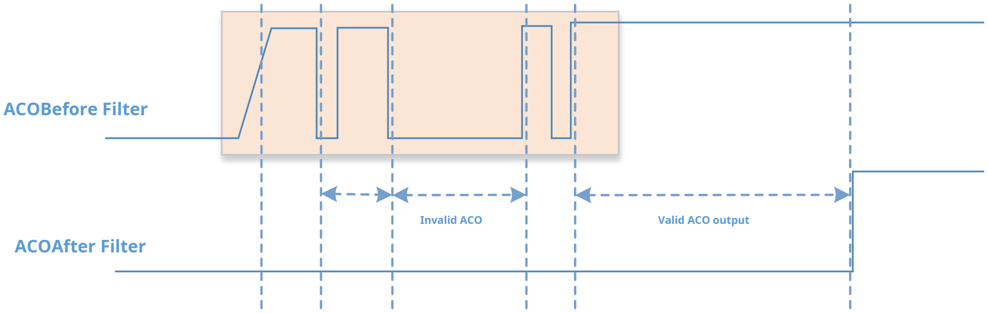
It is recommended that users turn on the hysteresis circuit when using the comparator to obtain a stable comparator output. As shown in the figure below, the hysteresis circuit is located between the analog output and digital output of the comparator. When the input voltage of the positive terminal of the comparator V_{IN+}more than the(V_{IN-} + V_{H+})When the comparator COUT Output is high; when V_{IN+}The voltage is less than (V_{IN-}- V_{H-})When the comparator output is low. The hysteresis circuit avoids the jitter caused by the circuit itself when the voltage at the positive terminal of the comparator is close to the voltage at the negative terminal.

The relationship between the hysteresis voltage of the comparator and the output of the comparator:



Although the hysteresis circuit is very effective in suppressing the voltage ripple close to the threshold of the comparator, in the actual application environment, the input The signal will be disturbed by different intensities. Strong interference may cause the input level to rise instantaneously, which exceeds the threshold range of the hysteresis circuit and cannot be effectively suppressed.LGT8FX8P A programmable digital filter is integrated at the output of the comparator, which can filter out the effect of instantaneous interference on the output of the comparator. The digital filter can select the appropriate filter time width according to the application requirements. Only when the output of the comparator is stable and continues to meet the filter time limit, the filter circuit will update the output of the comparator.

So as to achieve a more stable output result.



Comparator output filter timing

AC1 Digital filtering through C1XR Register C0FEN as well as C1FS Bit control, please refer to this chapter for specific setting methods Register definition part.

Comparator output and PWMcontrol

LGT8FX8P Support multi-channel PWMOutput,PWMThe signal can be used in conjunction with the comparator module. The output of the comparator can be used for direct shutdownPWMSignal to achieve a more flexible PWMProtection scheme.

versus PWMFor output related control, please refer to the relevant part of the timer chapter.

Register definition

C1SR – AC1 Control and status registers

C1SR – AC1 Control and status registers								
address: 0x2F					Defaults: 0x80			
Bit	7	6	5	4	3	2	1	0
Name	C1D	C1BG	C1O	C1I	C1IE	C1IC	C1IS1	C1IS0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	C1D	Analog comparator disable bit. When set C1D Bit is"1"When, the analog comparator is turned off. When setC1D Bit is"0"When the analog comparator is turned on.
6	C1BG	Analog comparator 1 Positive input source selection bit.C1BG versus C1XR Register C1PS0 Bit common setting AC1 Positive input source, {C1BG, C1PS0} = 00 = ACXP As positive input 01 = AC1P As positive input 10 = internal DAC As the positive input 11 = shut down AC1 The positive input source
5	C1O	of the analog comparator output status bit. The output of the analog comparator is directly connected to the C1O Bit. Software can readC1O The value of the bit is used to obtain the output value of the analog comparator.
4	C1I	The interrupt flag of the analog comparator. When the output event of the analog comparator is triggered by C1IS In the bit-defined interrupt mode,C1I The bit is set. When the interrupt enable bitC1IE for"1"And when the global interrupt is set, an interrupt is generated. When executing the analog comparator interrupt service routine,C1I Will be automatically cleared, or C1I Bit write"1" This bit can also be cleared.
3	C1IE	The interrupt enable bit of the analog comparator. When set C1IE Bit is 1, And enable the global interrupt,AC1 The interrupt is enabled. When setC1IE Bit is 0,AC1 The interrupt is disabled. Analog comparator input capture
2	C1IC	enable bit C1IC = 1, Timer counter 1 The input capture source comes from the output of the analog comparator. C1IC = 0, Timer counter 1 The input capture source comes from an external pin ICP1. Analog
1	C1IS1	comparator interrupt mode control high bit.
0	C1IS0	Analog comparator interrupt mode control low bit.C1IS0 with C1IS1 Make up together C1PS[1:0], Used to control the interrupt trigger mode of the analog comparator.

C1IS[1:0]	Interrupt mode
00	AC1 Trigger on rising or falling edge
01	Reserved.
10	AC1 Falling edge trigger
11	AC1 Trigger on the rising edge of

ADCSRB – ADC Control and status registers B

ADCSRB **– ADC Control and status registers B**								
address: 0x7B	Defaults: 0x00							
Bit	7	6	5	4	3	2	1	0
Name	CME01	CME00	CME11	CME10	ACTS	ADTS2	ADTS1	ADTS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	CME01	AC0 Negative input selection, CME0 = {CME01, CME00}
6	CME00	00: External port ACXN As AC0 Negative input 01: ADC Multiplexed output as AC0 Negative input 10: Differential amplifier output as AC0 Negative input 11: shut down AC0 Negative input source
5	CME11	AC1 Negative input selection, CME1 = {CME11, CME10}
4	CME10	00: External port ACXN As AC1 Negative input 01: External port AC1N As AC1 Negative input 10: ADC internal 1/5 Partial pressure as AC1 Negative input 11: The output of the differential op amp is as AC1 Negative input
3	ACHS	AC Trigger source channel selection 0 – AC0 Output as ADC Automatic conversion trigger source 1 – AC1 Output as ADC Automatic conversion trigger source
2:0	ADTS	see ADC Register description.

C1XR – AC1 Auxiliary control register

C1XR **– AC1 Auxiliary control register**								
address: 0x3A	Defaults: 0x00							
Bit	7	6	5	4	3	2	1	0
Name	-	C1OE	C1HYSE	C1PS0	C1WKE	C1FEN	C1FS1	C1FS0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	-	Keep
6	C1OE	AC1 Enable control of comparator output to external port C1OE = 1, AC1 The comparator output to the external port PE5 C1OE = 0, Disable the output of the comparator to the external port
5	C1HYSE	AC1 The output hysteresis function enables control. 1 = Enable output hysteresis 0 = Disable output hysteresis
4	C1PS0	AC1 The positive input source selects the low bit. C1PS0 versus C1BG Joint control AC1 For the positive input source, please refer to C1SR Register definition
3	C1WKE	AC1 Enable control for wake-up from sleep. 1 = Enable the wake-up function of the comparator output

		0 = Turn off the wake-up function of the comparator output
2	C1FEN	Comparator digital filter enable control. 1 = Enable digital filter 0 = Disable digital filter
1:0	C1FS[1:0]	Comparator digital filter width setting 00 = shut down 01 = 32us 10 = 64us 11 = 96us

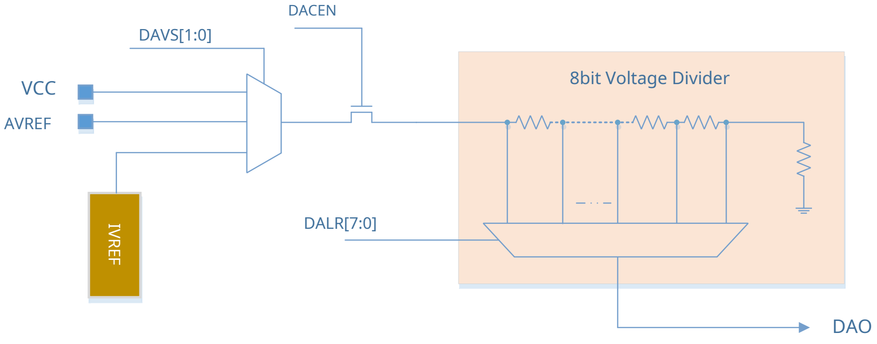
Digital-to-analog converter (DAC)

- Z 8 Digital-to-analog conversion output
- Z DAC The output can be used as an analog comparator reference input
- Z stand by DAC Output to external port (DAO)
- Z Optional VCC/AVREF/IVREF Voltage divider power supply

Summary

LGT8FX8P Internally integrated one 8 Bit Programmable Digital-to-Analog Converter (DAC).DAC The reference power input can be selected from the system working power supply, the internal reference voltage source or from the external port of the chip AVREF enter.DAC The output can be selected as an internal comparator AC0/1 The input source can also be directly output to the external pins of the chip as an external reference. whenDAC When outputting to an external pin, it cannot be directly used to drive the load. It needs to be driven by a voltage follower or other similar drive.

Moving circuit.DAC The internal structure is shown in the figure below:



Register definition

DACON-DAC Control register

DACON– DAC Control register								
address: 0xA0					0000_0000			
Bit	7	6	5	4	3	2	1	0
	-	-	-	-	DACEN	DAOE	DAVS1	DAVS0
R/W	-	-	-	-	R/W	W/R	R/W	W/R

Bit	Name	description
7:4	-	Keep
3	DACEN	DAC Enable control bit 1 = Enable DAC Module 0 = Disable DAC Module
2	DAOE	DAC Output to external port enable control 1 = Enable DAC Output to external PD4 0 = Prohibit DAC Output to external port
1	DAVS1	DAC Reference voltage source selection bit 1
0	DAVS0	DAC Reference voltage source selection bit 0. [DVS1, DVS0] =

		00: Voltage source to select system operating voltage VCC 01: Voltage source is selected as external input AVREF 10: The voltage source is selected as the internal reference voltage 11: shut down DAC The reference source will be closed at the same time DAC Module

DALR-DAC Data register

VRCON1– DAC1 Control register								
address: 0xA1					0000_0000			
Bit	7	6	5	4	3	2	1	0
	DALR[7:0]							
R/W	W/R							
Bit	Name	description						
7:0	DALR	DAC Data register, set DAC Mode output voltage DAC Output voltage and DALR Relationship: $V_{DAO} = V_{REF}*(DALR + 1)/256$ among them: $V_{DAO}$for DAC Output analog voltage $V_{REF}$for DAC Reference voltage source, by DACON Register DAVS Bit selection						

12 Bit analog-to-digital converter (ADC)

Z 12 Bit resolution,DNL for±1LSB,INL for±1.5LSB

Z At the highest resolution, the sampling rate is up to 500KSPS

Z 12 Multiplexed single-ended input channels

Z Multiple input programmable gain differential amplifier channels

Z The input voltage range is 0-VCC

Z internal 1.024V/2.048V/4.096V Reference voltage

Z support AVCC And external reference voltage input

Z Internal multiple input 1/5,4/5 Voltage divider circuit supports offset

Z calibration in positive and negative directions

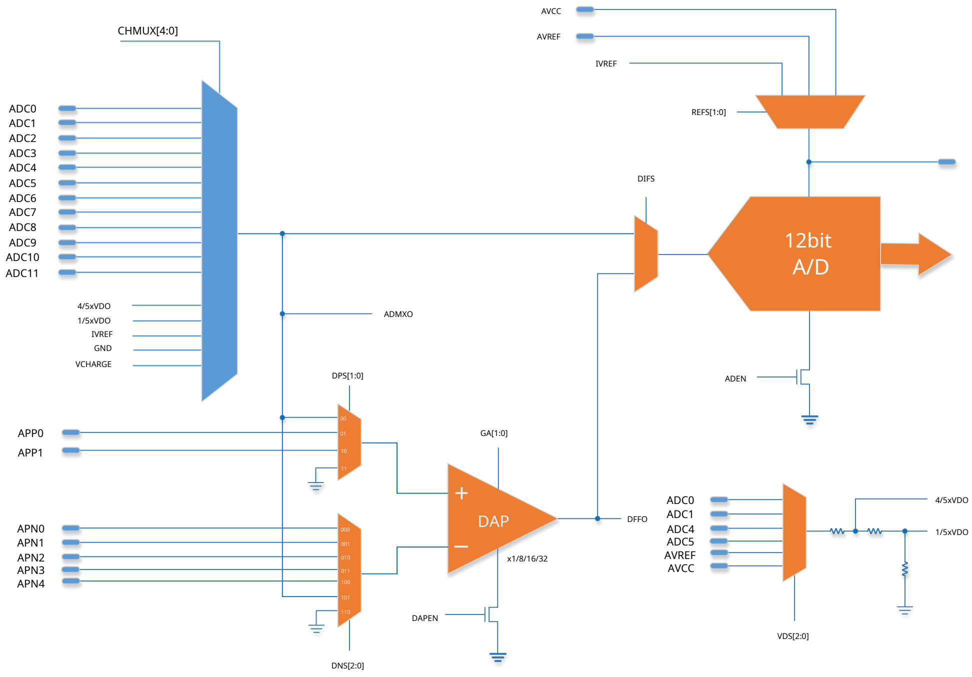
Z Automatic start conversion trigger mode based on interrupt source

Z Support automatic channel monitoring of up/down overflow

Z The conversion result supports optional alignment mode

Z Conversion end interrupt request

Overview



ADCStructure chart

The analog-to-digital converter is one 12 Successive approximation ADC.ADC With a 17 Channel analog multiplexer connection Connect to the external port of the chip 12 Analog inputs and 5 The internal voltage source of the channel performs sampling and conversion.ADC A programmable gain is integrated inside x1/x8/x16/x32 Differential operational amplifier, the amplifier input can come from an external port or ADC The output of the multiplexer. The result of the differential op amp can be used asADC Analog input.

ADC The internal analog input source includes from ADC Internal multiple input voltage divider; internal reference voltage source; internal analog reference ground and analog output from the touch button module. Simultaneous output of internal multiple input voltage divider4/5,1/5 Two way

Voltage; the input of the voltage divider can choose the level from the external port or from the system power supply.

ADC Support offset calibration. The process of offset calibration is controlled by software. The offset calibration includes the calibration amount in both positive and negative directions. After the offset calibration is enabled,ADC The controller will automatically use the positive and negative calibration value pairs ADC Sampling results are calibrated.

For the method of offset calibration, please refer to the relevant part of this chapter.

ADC Operation

ADC Convert the input analog voltage into a 12 The number of bits. Minimum representativeGND, The maximum value represents the reference voltage minus 1LSB. The reference voltage source can beADC Power supply voltage AVCC, External reference voltage AVREF Or internal 1.024V/2.048V Reference voltage by writing ADMUX Register REFS Bit to choose.

The analog input channel can be written by ADMUX Register CHMUX Bit to choose. anyADC The input pins, external reference voltage pins, and internal reference voltage sources can all be used as ADC Single-ended input. Set byADTMR Register DIFS Can be ADC The input channel is switched to the internal differential amplifier. The input source and gain of the differential amplifier can be passed DAPCR Register settings.

By setting ADCSRA Register ADEN Bit to start ADC,ADEN When cleared ADC Does not consume power, so it is recommended to turn off before entering sleep mode ADC.

ADC The result of the conversion is 12 Bit, store and ADC Data register ADCH and ADCL in. The conversion result is right-aligned by default, but it can be set byADMUX Register ADLAR The bits become left-justified.

If it is set to left-align the conversion result, and the highest only needs8Bit conversion accuracy, Then just readADCHWill suffice. Otherwise, read it first ADCL, Then read ADCH, To ensure that the content in the data register is the result of the same conversion. Once readADCL After the data register ADCL with ADCH Latched, read ADCH After the conversion result can be updated to the data register ADCL with ADCH.

ADC The end of the conversion can trigger an interrupt. Even if the end of the conversion occurs atADCL versus ADCH In the meantime, the interrupt will still trigger.

Start a conversion

to ADC Start conversion bit ADSC Bit write"1"You can start a single conversion. This bit remains high during the conversion process and is cleared by hardware until the conversion is completed. If the channel is changed during the conversion, thenADC This conversion will be completed before changing the channel.

ADC There are different trigger sources for conversion. Set upADCSRA Register ADC Auto trigger enable bit ADATE You can enable automatic triggering. Set upADCSRB Register ADC Trigger selection bit ADTS You can select the trigger source. When the selected trigger signal generates a rising edge,ADC The prescaler resets and starts conversion. This provides a way to initiate conversion at fixed intervals. Even if the trigger signal still exists after the conversion, a new conversion will not be started. If the trigger signal generates a rising edge during the conversion process, this rising edge will also be ignored. Even if the specific interrupt is disabled or the global interrupt enable bit is"0", Its interrupt flag will still be set. This can trigger a conversion without generating an interrupt. But in order to trigger a new conversion when the next interrupt event occurs, the interrupt flag must be cleared.

use ADC The interrupt flag is used as the trigger source, which can start the next time after the current conversion ends ADC Conversion. after that ADC It works in continuous conversion mode, continuously sampling and ADC The data register is updated. First turn

Change is through to ADCSRA Register ADSC Bit write"1"To start. In this mode, subsequentADC Conversion does not depend on ADC Interrupt flag ADIF Whether to set.

If auto trigger is enabled, set ADCSRA Register ADSC A single conversion will be initiated.ADSC The flag can also be used to detect whether the conversion is in progress. No matter how the conversion is initiated, during the conversion processADSC Has always been"1".

Prescaler and ADC Conversion timing

Under default conditions, the successive approximation circuit requires a slave 300KHz To 3MHz Input clock for maximum accuracy. If the required conversion accuracy is lower than12 Bit, then the frequency of the input clock can be higher than 3MHz, In order to achieve a higher sampling rate.

ADC The module includes a prescaler, which can generate acceptable ADC Input clock. Prescaler pass ADCSRA Register ADPS Bit to be set. PositionADCSRA Register ADEN Will enable ADC, The prescaler starts counting. as long asADEN Bit is"1", The prescaler keeps counting until ADEN It is cleared.

ADCSRA Register ADSC After being set, the single-ended conversion is in the next ADC Start on the rising edge of the clock cycle. Required for normal conversion15 A ADC Clock cycle.ADC Enable(ADCSRA Register ADEN Set) after need 50 A ADC The input clock cycle initializes the analog circuit before the first conversion can be effectively performed.

in ADC During the conversion process, the sampling and holding is at the beginning of the conversion 1.5 A ADC The input clock starts, and the first time ADC The result of the conversion is output after the start 14.5 A ADC Input clock. After the conversion,ADC The result is sent to ADC Data register, and ADIF The flag bit is set.ADSC It is cleared at the same time. The software can be set again afterwardsADSC Flag or automatically trigger to start a new conversion.

Sampling channel and reference voltage

ADMUX In the register MUX and REFS Single buffering is realized through temporary registers.CPU Random access to temporary registers is possible. Before the conversion starts,CPU The choice of channel and reference source can be configured at any time. to ensure thatADC There is sufficient sampling time, once the conversion starts, the configuration of channel and reference source selection is not allowed. After the conversion is complete (ADCSRA Register ADIF After setting), the channel and reference source selection will be updated. The start time of the conversion isADSC Next after set ADC The rising edge of the input clock. Therefore, it is recommended that the user setADSC The one after ADC Do not operate during the input clock cycle ADMUX To select a new channel and reference source.

When using automatic triggering, the time when the trigger event occurs is uncertain. In order to control the impact of the new settings on the conversion, the update ADMUX Be especially careful when registering. IfADATE and ADEN Both are set, the interruption time can occur at any time, so as to automatically trigger and start ADC Conversion. If it changes during this periodADMUX The contents of the register, then the user cannot distinguish whether the next conversion is based on the old configuration or the new configuration. It is recommended that users checkADMUX Update:

1)ADATE or ADEN Bit is"0"；

2) During the conversion process, but at least one ADC Input clock cycle;

3) After the conversion ends, but before the interrupt flag of the trigger source is cleared. If you update in any of the cases mentioned aboveADMUX , Then the new configuration will take effect before the next conversion. selectADC Note that when inputting channels, select the channel before starting the conversion. ADSC The one after the set ADC After the clock cycle, a new analog input channel can be selected, but the easiest way is to wait until the conversion is complete before changing the channel.

ADC Reference voltage source V_{ref} Reflects ADC The conversion range. If the single-ended channel level exceedsV_{ref}, The conversion result will be close to the maximum 0xFFF.V_{ref}Can be AVCC, External AREF The voltage of the pin, the internal reference voltage source.

Use internal benchmarks (1.024V/2.048V/4.096V)Precautions:

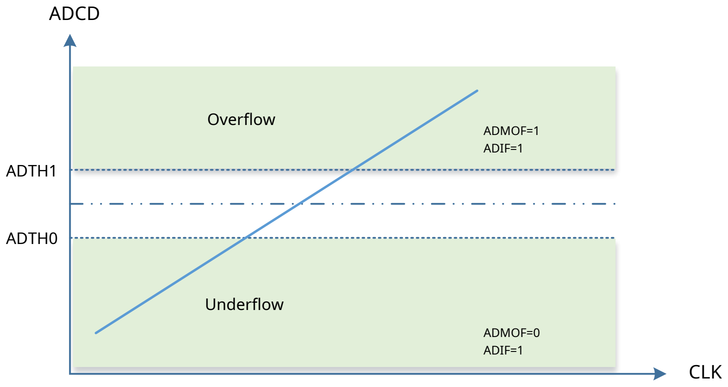
After the chip is powered on, the internal reference is calibrated to 1.024V, If the user uses 1.024V The internal benchmark can be used directly without other operations. But if you need to use2.048V or 4.096V It is necessary to update the calibration value of the internal reference by itself.2.048V/4.096V The calibration value of is loaded into the register after power-on VCAL2/3(0xCE/0xCC), When the program is initialized, the VCAL2/3 The value of is read in and written to VCAL(0XC8)The register is now calibrated.

Automatic channel monitoring

Automatic channel monitoring mode is used for real-time monitoring and selection ADC The voltage change of the input channel. Software passADCSRC Register AMEN Bit enable automatic channel monitoring function,ADC Automatically convert the voltage of the selected channel, when the conversion result is outside the given overflow range, it will be set ADC Interrupt flag bit (ADIF), And stop automatic monitoring at the same time. The software can respond to the overflow event by interrupt or query.ADMSC Register AMOF Bits are used to indicate the type of overflow event.ADIF

The flag bit is automatically cleared by hardware after responding to the interrupt reset; in query mode, it can be written by software 1 Cleared. Only ifADIF

Cleared and set by ADCSRC Register AMEN Bit to re-enable the auto-monitoring mode.



To overcome a single ADC Unstable conversion results, automatic detection supports a configurable digital filtering function. Digital filter The wave detects the continuous conversion results, and only when a consistent result is obtained within a limited number of consecutive conversions, the overflow event is triggered. The number of consecutive conversions can be passedADMSC Register AMFC[3:0]Bit setting.

Automatic channel monitoring function passed ADCSRC Register AMEN Position control. registerADT0 Used to set the threshold of underflow;ADT1 Used to set the overflow threshold.ADT0/1 for 16 Bit register. Software setAMEN After the position, it will stop immediately ADC The current conversion action, and reset ADC Control state, and then enter automatic conversion mode.

Before starting the automatic channel detection mode, you need to set the detected channel and other related configurations. Software can be cleared at any timeAMEN Register, disable automatic detection mode.

Multiple input voltage divider circuit (VDS)

ADC It contains a multi-input voltage divider module inside. Voltage divider input voltage source can be selected from externalADC Input channel (ADC0/1/4/5), external reference AVREF Or simulate working power. Simultaneous output of voltage divider module4/5 as well as 1/5 The two voltages are to ADC internal 12,13 Input channel. among them4/5 This way is mostly used ADC Offset calibration1/5 Except for internal offset calibration, it is mostly used for power supply voltage detection and similar applications. The related functions of the voltage divider circuit are mainly composed ofADCSRD Register control is realized.

ADC Offset calibration

Due to the deviation of the manufacturing process and the inherent characteristics of the circuit structure, it will cause ADC The internal comparator circuit produces different degrees of offset errors. Therefore, the offset voltage is compensated to produce high-precisionADC The conversion structure is very critical.

LGT8FX8P Inside the chip ADC Support the related interface of offset voltage test, can complete the offset measurement and calibration with the cooperation of software.

Principle of offset calibration:

Offset calibration is mainly done by changing the input polarity of the internal comparator, testing in both positive and negative directions ADC Conversion result. Since the offset voltage in the positive and negative directions is also expressed in two polarities, an intermediate offset error value can be obtained by subtracting the results of these two conversions. In normal application, the conversion result can be adjusted accordingly according to this offset voltage.

Offset calibration process:

1. Configuration VDS Module, will VDS The input source is selected as the analog power supply (AVCC)
2. ADC The reference voltage is selected as the analog power supply (AVCC)
3. ADCSRC[SPN] = 0, ADC Read 4/5VDO Channel, the conversion value is recorded as PVAL
4. ADCSRC[SPN] = 1, ADC Read 4/5VDO Channel, conversion value record bit NVAL
5. Will value (NVAL – PVAL) >>1 Store to OFR0 register
6. ADCSRC[SPN] = 1, ADC Read 1/5VDO Channel, the conversion result is recorded as NVAL
7. ADCSRC[SPN] = 0, ADC Read 1/5VDO Channel, conversion result record bit PVAL
8. Will value (NVAL – PVAL) >> 1 Store to OFR1 register
9. Set up ADCSRC[OFEN]=1 Enable offset compensation function

Special attention: Since the offset error has positive and negative directions, the above data and calculations are all signed operations.

Need to change during offset calibration ADC Related configuration, so it is recommended that the offset calibration be completed before the configuration used normally to make. In order to improve the calibration accuracy, it is recommendedADC Sampling and filtering multiple times when reading the channel conversion.

Offset calibration OFR0/1 After the configuration is complete, pass OFEN Bit enables automatic offset compensation. After normal conversion in the future, ADC Control will be based on ADC Conversion results, used automatically OFR0/1 Make compensation.

ADC Dynamic calibration

The offset calibration method described above is based on the offset in a test environment and test input. When the system environment changes,ADC The imbalance will change accordingly. Therefore, if real-time calibration compensation can be achieved, it will be helpful to overcome the performance difference caused by the change of the device with the working environment, and improveADC Measurement accuracy is very important.

Here is a suggested algorithm. Based on the principle of the offset calibration algorithm, it can dynamically compensate the offset error caused by the working environment and obtain consistent and accurate test results.

This method does not need to calculate the offset voltage, nor does it need to enable offset compensation (OFEN). The algorithm only needs to passSPN control ADC The polarity of the switch is different SPN Down-sampling the two measurement results, the errors introduced by the offset in the two results appear in both positive and negative directions, so we can simply offset the errors caused by the disappearance by adding and averaging.

We assume that when ADC When converting, the test error introduced by the offset is VOFSAnd therefore control SPN Do it twice in a row ADC Convert, get ADC The conversion result can be expressed as:

SPN = 1 Time, $V_{ADC1} = V_{REL} + V_{OFS1}$

SPN = 0 Time, $V_{ADC0} = V_{REL} – V_{OFS0}$

We add the two measurement results to eliminate V_{OFS}Input to the actual sample V_{REL}The impact. Due to the circuit Matching characteristics,$V_{OFS1}$with V_{OFS0}It may not be exactly the same, but overall the effect of compensating for the offset error can still be achieved.

Dynamic offset compensation algorithm flow:

1. Initialize according to application needs ADC Conversion parameters

2. Set up SPN=1,start up ADC Sample, record ADC The sampling result is VADC1
3. Set up SPN=0,start up ADC Sample, record ADC The sampling result is VADC2
4. (VADC1 + VADC2) >> 1 This time ADC Conversion result

In practical applications, this algorithm can be combined with the sampling average algorithm to get a more ideal effect.

Register definition

ADC Register list

register	address	Defaults	description
ADCL	0x78	0x00	ADC Data low byte register
ADCH	0x79	0x00	ADC Data high byte register
ADCSRA	0x7A	0x00	ADC Control and status registers A
ADCSRB	0x7B	0x00	ADC Control and status registers B
ADMUX	0x7C	0x00	ADC Multiplex control register
ADCSRC	0x7D	0x01	ADC Control and status registers C
DIDR0	0x7E	0x00	Digital input inhibit control register 0
DIDR1	0x7F	0x00	Digital input inhibit control register 0
DAPCR	0xDC	0x00	Differential amplifier control register
OFR0	0xA3	0x00	Offset compensation register 0
OFR1	0xA4	0x00	Offset compensation register 1
ADT0L	0xA5	0x00	Automatic monitoring of low underflow threshold 8 Bit
ADT0H	0xA6	0x00	Automatic monitoring of underflow threshold is high 8 Bit
ADT1L	0xAA	0x00	Automatic monitoring of low overflow threshold 8 Bit
ADT1H	0xAB	0x00	Automatic monitoring of high overflow threshold 8 Bit
ADMSC	0xAC	0x01	Automatic monitoring of status and control registers
ADCSRD	0xAD	0x00	ADC Control and status registers D

ADCL – ADC Data low byte register

ADCL – ADC Data low byte register								
address: 0x78					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name0	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0
Name1	ADC3	ADC2	ADC1	ADC0	-	-	-	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial	0	0	0	0	0	0	0	0

Bit	Name	description
7:0	ADC[7:0]/ ADC[3:0]	ADC Data low byte register. when ADLAR Bit is"0"Time,ADC The storage of output data in the register is aligned with the low bit, that is ADCL for ADC[7:0],Such as Name0 Shown; when ADLAR Bit is"1"Time, ADC The storage of output data in the register is aligned with the high order, that is ADCL height of 4 Bit is ADC[3:0],low 4 Bit meaningless, such as Name1 Shown.

ADCH – ADC Data high byte register

ADCH – ADC Data high byte register								
address: 0x79					Defaults: 0x00			
Bit	7	6	5	4	3 2		1	0
Name0	-	-	-	-	ADC11 ADC10		ADC9	ADC8
Name1	ADC11	ADC10	ADC9	ADC8	ADC7	ADC6	ADC5	ADC4
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Initial	0	0	0	0	0	0	0	0

Bit	Name	description
7:0	ADC[11:8]/ ADC[11:4]	ADC Data low byte register. when ADLAR Bit is"0"Time,ADC The storage of output data in the register is aligned with the low bit, that is ADCH The low 4 Bit is ADC[11:8],high 4 Bit meaningless, such as Name0 Shown; when ADLAR Bit is"1"Time,ADC The storage of output data in the register is aligned with the high order, that is ADCH for ADC[11:4],Such as Name1 Shown.

ADCSRA – ADC Control and status registers A

ADCSRA – ADC Control and status registers A								
address: 0x7A					Defaults: 0x05			
Bit	7	6	5	4	3 2		1	0
Name	ADEN	ADSC	ADATE	ADIF	ADIE ADPS2		ADPS1	ADPS0
R/W	R/W	R/W	R/W	R/W	R/WR/W		R/W	R/W
Initial	0	0	0	0	0 0		1	0

Bit	Name	description
7	ADEN	ADC Enable control bit. When set ADEN Bit is"1"Time,ADC Is enabled. When setADEN Bit is"0"Time,ADC banned.
6	ADSC	ADC Start the conversion. In single conversion mode,ADSC Setting this bit will initiate a conversion. In continuous conversion mode,ADSC Setting this bit will start the first conversion.
5	ADATE	ADC The enable control bit is automatically triggered. When set ADATE Bit is"1"When, the automatic trigger function is enabled. The rising edge of the selected trigger signal starts a conversion. The choice of trigger source is determined byADCSRB Register ADTS To control. When set ADATE Bit is"0"When, the automatic trigger function is disabled.
4	ADIF	ADC Interrupt flag bit. when ADC Set after completing a conversion and updating the data register ADIF. IfADC Interrupt enable bit ADIE for"1"And the global interrupt is set,ADC An interrupt is generated. carried outADC Interrupt will be cleared ADIF Bit, you can also write to this bit"1"Come to zero.
3	ADIE	ADC Interrupt enable control bit. When set ADIE Bit is"1"And when the global interrupt is set,ADC The interrupt is enabled. When setADIE Bit is"0"Time,ADC Interrupts are disabled.

2:0	ADPS[2:0] ADC Prescaler selection control bit. ADPS Select system clock generation ADC The prescaler factor of the clock.

ADPS[2:0]	Prescaler factor
0	2
1	2
2	4
3	8
4	16
5	32 (default)
6	64
7	128

ADCSRB – ADC Control and status registers B

ADCSRB – ADC Control and status registers B								
address: 0x7B						Defaults: 0x00		
Bit	7	6	5	4	3	2	1	0
Name	ACME01	ACME00	ACME11	ACME10	ACTS	ADTS2	ADTS1	ADTS0
R/W	R/W	R/W	R/W	R/W	W/O	R/W	R/W	R/W
Initial	0	0	0	0	0	0	0	0

Bit	Name	description
7	ACME01	Comparators 0 Negative input selection 00：The negative terminal selects the external input ACIN0 01: Negative end selection ADC Multiplexed output 1X：The negative terminal selects the op amp 0 Output
6	ACME00	
5	ACME11	comparator 1 Negative input selection 00：The negative terminal selects the external input ACIN2 01: Negative end selection ADC Multiplexed output 1X: Negative terminal selection op amp 1 Output
4	ACME10	
3	ACTS	AC Trigger source channel selection 0 – AC0 Output as ADC Automatic conversion trigger source 1 – AC1 Output as ADC Automatic conversion trigger source
2:0	ADTS[2:0] ADC Automatic trigger source selection control bit.	When set ADATE Bit is"1", The automatic trigger function is enabled, and the trigger source is selected by ADTS To control. When setADATE Bit is"0"Time,ADTS The settings are invalid. The rising edge of the selected trigger signal interrupt flag starts a conversion. When switching from a trigger source with an interrupt flag cleared to a trigger source with an interrupt flag set, the trigger signal will generate a rising edge.ADEN Position,ADC A conversion will also be started. When switching to continuous conversion mode (ADTS=0), the automatic trigger function is disabled.

ADTS[2:0]	Trigger source
0	Continuous conversion mode
1	Comparators 0/1

		2	External Interrupt 0
		3	Timer counter 0 Comparison match
		4	Timer counter 0 overflow
		5	Timer counter 1 Comparison match B
		6	Timer counter 1 overflow
		7	Timer counter 1 Input capture event

ADMUX – ADC Multiplex control register

ADMUX – ADC Multiplex control register								
address: 0x7C					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	REFS1	REFS0	ADLAR CHMUX4 CHMUX3 CHMUX2 CHMUX1 CHMUX0 R/					
R/W	R/W	R/W	WR/WR/WR/WR/WR/W					
Initial	0	0	0 0 0 0 0 0					

Bit	Name	description		
7:6	REFS[1:0]	versus ADCSRD Register REFS2 Cooperate for selection ADC The reference voltage source is set by REFS Control bit to select the reference voltage, if it is changed during the conversion REFS The setting will only take effect after the current conversion is over.		
		REFS2, REFS[1:0]	Reference voltage selection	
		0_00	AREF	
		0_01	AVCC	
		0_10	On-chip 2.048V Reference voltage source	
		0_11	On-chip 1.024V Reference voltage source	
		1_00	On-chip 4.096V Reference voltage source	
5	ADLAR	The left-justified enable control bit of the conversion result. When set ADLAR Bit is"1", The conversion result is ADC The data register is left-justified. When set ADLAR Bit is"0", The conversion result is ADC Right pair in the data register Qi.		
4:0	CHMUX[4:0]	ADC Input source selection control bit.		
		CHMUX[4:0]	Single-ended input source	description
		0_0000	PC0	External port input
		0_0001	PC1	
		0_0010	PC2	
		0_0011	PC3	
		0_0100	PC4	
		0_0101	PC5	
		0_0110	PE1	
		0_0111	PE3	
		0_1001	PC7	

		0_1010	PF0	
		0_1011	PE6	
		0_1100	PE7	
		0_1110	4/5VDO	Internal voltage divider circuit
		0_1000	1/5VDO	
		0_1101	IVREF	Internal reference
		0_1111	AGND	Analog ground
		1_XXXX	DACO	internal DAC Output

ADCSRC – ADC Control status register C

***ADCSRC*– ADC Control status register C**								
address: 0x7D				Defaults: 0x00				
Bit	7	6	5	4	3	2	1	0
Name	OFEN	-	SPN	AMEN	-	SPD	DIFS	ADTM
R/W	R/W	-	R/W	R/W	-	R/W	R/W	R/W

Bit	Name	description
7	OFEN	1=Enable offset compensation;0=Turn off offset compensation
6	-	Unimplemented
5	SPN	ADC Switch input polarity control, only used for offset calibration process. Must be cleared when normal
4	AMEN	Channel automatic monitoring enable; 1：Enable automatic channel monitoring function 0：Disable automatic channel monitoring function
3	-	Unimplemented
2	SPD	0=ADC Low speed conversion mode 1=ADC High-speed conversion mode, only used for low-impedance analog input
1	DIFS	0 = ADC Conversion from ADC Multiplexer 1 = ADC The conversion comes from the internal differential amplifier test mode,
0	ADTM	from AVREF Output internal reference voltage on the port

DIDR0 – Digital input inhibit control register 0

***DIDR0* - Digital input inhibit control register 0**								
address: 0x7E					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	PE3D	PE1D	PC5D	PC4D	PC3D	PC2D	PC1D	PC0D
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
7	PE3D	1=shut down PE3 Digital input function
6	PE1D	1=shut down PE1 Digital input function
5	PC5D	1=shut down PC5 Digital input function
4	PC4D	1=shut down PC4 Digital input function

3	PC3D	1=shut down PC3 Digital input function
2	PC2D	1=shut down PC2 Digital input function
1	PC1D	1=shut down PC1 Digital input function
0	PC0D	1=shut down PC0 Digital input function

DIDR1 – Digital input inhibit control register 1

DIDR1 - Digital input inhibit control register 1								
address: 0x7F					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	PE7D	PE6D	PE0D	C0PD	PF0D	PC7D	PD7D	PD6D
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Name	description
0	PD6D	1=shut down PD6 Digital input function
1	PD7D	1=shut down PD7 Digital input function
2	PC7D	1=shut down PC7 Digital input function
3	PF0D	1=shut down PF0 Digital input function
4	C0PD	1=shut down AC0P Digital input function (LQFP48) 1=
5	PE0D	shut down PE0 Digital input function
6	PE6D	1=shut down PE6 Digital input function
7	PE7D	1=shut down PE7 Digital input function

ADCSRD – ADC Control register D

ADCSRD – ADC Control register D								
address: 0xAD					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	BGEN	REFS2	IVSEL1	IVSEL0	-	VDS2	VDS1	VDS0
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W

Bit	Name	description
7	BGEN	Internal reference global enable control, 1=Enable
6	REFS2	versus ADMUX Register REFS Combination for selection ADC Please refer to the converted reference voltage ADMUX Register REFS The definition of when ADC The reference voltage is selected as VCC or AVREF,IVSEL
5:4	IVSEL	Used to control the output voltage of the internal reference: 00 = 1.024V 01 = 2.048V 1x = 4.096V
3	-	Keep
2:0	VDS[2:0] Voltage	divider circuit input source selection 000/111 = Close the voltage divider circuit module 001 = ADC0 010 = ADC1 011 = ADC4

		100 = ADC5 101 = External reference input (AVREF) 110 = System power

DAPCR – Differential Op Amp Control Register

DAPCR **- Differential Op Amp Control Register**								
address: 0xDC					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	DAPEN	GA1	GA0	DNS2	DNS1	DNS0	DPS1	DPS0
R/W	W/R	W/R	W/R	W/R	W/R	R/W	R/W	R/W

Bit	Name	description
7	DAPEN	1 = Enable the differential amplifier;0 = Turn off the differential amplifier
6:5	GA[1:0]	differential amplifier gain control 00 = x1 01 = x8 10 = x16 11 = x32
4:2		DNS[2:0] Input source selection bit for the inverting input of the differential amplifier 000 = ADC2/APN0 001 = ADC3/APN1 010 = ADC8/APN2 011 = ADC9/APN3 100 = PE0/APN4 101 = ADC Multiplexing 110 = AGND 111 = Turn off the reverse input of the differential amplifier
1:0		DPS[1:0] Input source selection bit for the positive input of the differential amplifier 00 = ADC Multiplexing 01 = ADC0/APP0 10 = ADC1/APP1 11 = AGND

OFR0 – Offset compensation register 0

OFR0 **- Offset compensation register 0**								
address: 0xA3					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OFR0[7:0]							
R/W	W/R							

Bit	Name	description
7:0	OFR0	Offset compensation register 0；OFR0 Is a signed number. Store in twos complement format

OFR1 – Offset compensation register 1

OFR1 - Offset compensation register 1								
address: 0xA4					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	OFR1[7:0]							
R/W	W/R							

Bit	Name	description
7:0	OFR1	Offset compensation register 1；OFR1 Is a signed number. Store in twos complement format

ADMSC – ADC Channel monitoring status control register

ADMSC – ADC Channel monitoring status control register								
address: 0xAC					Defaults: 0x01			
Bit	7	6	5	4	3	2	1	0
Name	AMOF	-	-	-	AMFC3	AMFC2	AMFC1	AMFC0
R/W	-	-	-	-	R/W	R/W	R/W	R/W

Bit	Name	description
7	AMOF	Automatically monitor the flag bit of the overflow event type;1=Overflow, 0=Underflow
6:4	-	Unimplemented
3:0	AMFC	Automatic monitoring digital filter control bit: 0000 = Disable configuration 0001 = One conversion, no filtering 0010 = Two consecutive times 0011 = Three consecutive times .... 1110 = 14 Consecutive 1111 = 15 Consecutive

ADT0L – Automatic monitoring of low underflow threshold 8 Bit

ADT0L - Automatic monitoring of low underflow threshold 8 Bit								
address: 0xA5					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	ADT0L[7:0]							
R/W	W/R							

Bit	Name	description
7:0	ADT0L	Automatic monitoring of underflow threshold register low 8 Bit

ADT0H – Automatic monitoring of underflow threshold is high 8 Bit

ADT0H - Automatic monitoring of underflow threshold is high 8 Bit								
address: 0xA6					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	ADT0H[7:0]							
R/W	W/R							

Bit	Name	description
7:0	ADT0H	Automatic monitoring of underflow threshold register high 8 Bit

ADT1L – Automatic monitoring of low overflow threshold 8 Bit

ADT0L - Automatic monitoring of low overflow threshold 8 Bit								
address: 0xAA					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	ADT1L[7:0]							
R/W	W/R							

Bit	Name	description
7:0	ADT1L	Automatic detection of overflow threshold register low 8 Bit

ADT1H – Automatic monitoring of high overflow threshold 8 Bit

ADT1H - Automatic monitoring of high overflow threshold 8 Bit								
address: 0xAB					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	ADT1H[7:0]							
R/W	W/R							

Bit	Name	description
7:0	ADT1H	Automatic detection of overflow threshold register high 8 Bit

VCAL – Internal reference calibration register

VCAL - Internal reference calibration register								
address: 0xC8					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	VCAL[7:0]							
R/W	W/R							

Bit	Name	description
7:0	VCAL	Internal reference calibration register. Loaded by default after power-on1.024V Calibration value. Write the calibration values of other reference voltages into this register to realize the calibration of the relevant reference. For example, the reference configuration is2.048V After VCAL2 Write to the change register to complete the 2.048V Calibration of internal references.

VCAL1 – 1.024V Reference Calibration Register

VCAL1 – 1.024V Internal reference calibration register								
address: 0xCD					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	VCAL1[7:0]							
R/W	R/O							

Bit	Name	description
7:0	VCAL1	1.024V Internal reference calibration factor

VCAL2 – 2.048V Reference Calibration Register

VCAL2 – 2.048V Internal reference calibration register								
address: 0xCE					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	VCAL2[7:0]							
R/W	R/O							

Bit	Name	description
7:0	VCAL2	2.048V Internal reference calibration factor

VCAL3– 4.096V Reference Calibration Register

VCAL1 – 4.096V Internal reference calibration register								
address: 0xCC					Defaults: 0x00			
Bit	7	6	5	4	3	2	1	0
Name	VCAL3[7:0]							
R/W	R/O							

Bit	Name	description
7:0	VCAL3	4.096V Internal reference calibration factor

Register Cheat Sheet

Addr	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Extended IO Register									
$F6	GUID3	GUID Byte 3							
$F5	GUID2	GUID Byte 2							
$F4	GUID1	GUID Byte 1							
$F3	GUID0	GUID Byte 0							
$F2	PMCR	PMCE	CLKFS	CLKSS	WCLKS	OSCKEN	OSCMEN	RCKEN	RCMEN
$F0	PMX2	WCE	STOSC1	STOSC0	-	-	XIEN	E6EN	C6EN
$EE	PMX0	PMXCE	C1BF4	C1AF5	C0BF3	C0AC0	SSB1	TXD6	RXD5
$ED	PMX1	-	-	-	-	-	C3AC	C2BF7	C2AF6
$EC	TCKSR	-	F2XEN	TC2XF1	TC2XF0	-	AFCKS	TC2XS1	TC2XS0
$E2	PSSR	PSS1	PSS3	-	-	-	-	PSR3	PSR1
$E1	OCPUE	PUE7	PUE6	PUE5	PUE4	PUE3	PUE2	PUE1	PUE0
$E0	HDR	-	-	HDR5	HDR4	HDR3	HDR2	HDR1	HDR0
$DE	DAPTE	DAPTE	-	-	-	-	-	-	-
$DD	DAPTR	DAPTP	DAP Trimming						
$DC	DAPCR	DAPEN	GA1	GA0	DNS2	DNS1	DNS0	DPS1	DPS0
$D8									
$D7									
$D6									
$D5									
$D4									
$D2									
$D1									
$D0									
$CF	LDOCR	WCE				PDEN	VSEL2	VSEL1	VSEL0
$CE	VCAL2	Calibration value for 2.048V internal reference							
$CD	VCAL1	Calibration value for 1.024V internal reference							
$CC	VCAL3	Calibration value for 4.096V internal reference							
$C8	VCAL	Internal Voltage Reference calibration register							
$C6	UDR	USART Data Register							
$C5	UBRRH	-	-	-	-	USART Baud Rate Register High			
$C4	UBRRL	USART Baud Rate Register Low							
$C2	UCSRC	UMSEL1	UMSEL0	UPM1	UPM0	USBS0	UCSZ01	UCSZ00	UCPOL0
$C1	UCSRB	RXCIE0	TXCIE0	UDRIE0	RXEN0	TXEN0	UCSZ02	RXB80	TXB80
$C0	UCSRA	RXC0	TXC0	UDRE0	FE0	DOR0	UPE0	U2X0	MPCM0
$BD	TWAMR	TWI Address Mask							-
$BC	TWCR	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
$BB	TWDR	TWI Data							
$BA	TWAR	TWI Address							TWGCE

Addr	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
$B9	TWSR	TWI Status bits					-	TWPS	
$B8	TWBR	TWI Bit Rate register							
$B6	ASSR	INTCK	-	AS2	TCN2UB	OCR2AUB	OCR2BUB	TCR2AUB	TCR2BUB
$B4	OCR2B	Timer 2 Output Compare Register B							
$B3	OCR2A	Timer 2 Output Compare Register A							
$B2	TCNT2	Timer 2 Counter Register							
$B1	TCCR2B	FOC2A	FOC2B	-	-	WGM22	CS2		
$B0	TCCR2A	COM2A1	COM2A0	COM2B1	COM2B0	-	-	WGM21	WGM20
$AF	DPS2R	-	-	-	-	DPS2E	LPRCE	TOS1	TOS0
$AE	IOCWK	IOCD7	IOCD6	IOCD5	IOCD4	IOCD3	IOCD2	IOCD1	IOCD0
$AD	ADCSRD	BGEN	REFS2	IVSEL1	IVSEL0	-	VDS2	VDS1	VDS0
$AC	ADMSC	AMOF	-	-	-	AMFC3	AMFC2	AMFC1	AMFC0
$AB	ADT1H	ADC Auto-monitor Overflow threshold high byte							
$AA	ADT1L	ADC Auto-monitor Overflow threshold low byte							
$A9	PORTE	Port Output E (for compatible with LGT8FX8D)							
$A8	DDRE	Data Direction E (for compatible with LGT8FX8D)							
$A7	PINE	Port Input E (for compatible with LGT8FX8D)							
$A6	ADT0H	ADC Auto-monitor Underflow threshold high byte							
$A5	ADT0L	ADC Auto-monitor Underflow threshold low byte							
$A4	OFR1	ADC positive offset trimming							
$A3	OFR0	ADC negative offset trimming							
$A1	DALR	DAC data register							
$A0	DACON	-	-	-	-	DACEN	DAOE	DAVS1	DAVS0
$9F	OCR3CH	Compare output register high byte of Timer3 C channel							
$9E	OCR3CL	Compare output register low byte of Timer3 C channel							
$9D	DTR3H	Dead-band register high byte of Timer3							
$9C	DTR3L	Dead-band register low byte of Timer3							
$9B	OCR3BH	Compare output register high byte of Timer3 B channel							
$9A	OCR3BL	Compare output register low byte of Timer3 B channel							
$99	OCR3AH	Compare output register high byte of Timer3 A channel							
$98	OCR3AL	Compare output register low byte of Timer3 A channel							
$97	ICR3H	Input capture register high byte of Timer3							
$96	ICR3L	Input capture register low byte of Timer3							
$95	TCNT3H	Counter register high byte of Timer3							
$94	TCNT3L	Counter register low byte of Timer3							
$93	TCCR3D	Control register D of Timer3							
$92	TCCR3C	Control register C of Timer3							
$91	TCCR3B	Control register B of Timer3							
$90	TCCR3A	Control register A of Timer3							
$8D	DTR1H	Dead-band register high byte of Timer1							
$8C	DTR1L	Dead-band register low byte of Timer1							
$8B	OCR1BH	Timer 1 Output Compare B High							

Addr	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
$8A	OCR1BL	Timer 1 Output Compare B Low							
$89	OCR1AH	Timer 1 Output Compare A High							
$88	OCR1AL	Timer 1 Output Compare A Low							
$87	ICR1H	Timer 1 Input Capture High							
$86	ICR1L	Timer 1 Input Capture Low							
$85	TCNT1H	Timer 1 Counter High							
$84	TCNT1L	Timer 1 Counter Low							
$83	TCCR1D	DSX17	DSX16	DSX15	DAX14	-	-	DSX11	DSX10
$82	TCCR1C	FOC1A	FOC1B	DOC1B	DOC1A	DTEN1	-	-	-
$81	TCCR1B	ICNC1	ICES1	-	WGM13	WGM12	CS1		
$80	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	-	-	WGM11	WGM10
$7F	DIDR1	PE7D	PE6D	PE0D	C0PD	PF0D	PC7D	PD7D	PD6D
$7E	DIDR0	PE3D	PE1D	PC5D	PC4D	PC3D	PC2D	PC1D	PC0D
$7D	ADCSRC	OFEN	-	SPN	AMEN	-	SPD	DIFS	ADTM
$7C	ADMUX	REFS1	REFS0	ADLAR	CHMUX				
$7B	ADCSRB	CME01	CME00	CME11	CME10	-	ADTS		
$7A	ADCSRA	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS		
$79	ADCH	ADC Data High							
$78	ADCL	ADC Data Low							
$76	DIDR2	-	PB5D	-	-	-	-	-	-
$75	IVBASE	Interrupt Vector Base Address							
$74	PCMSK4								
$73	PCMSK3	PCINT[39:32]							
$71	TIMSK3			ICIE3	-	OCIE3C	OCIE3B	OCIE3A	TOIE3
$70	TIMSK2	-	-	-	-	-	OCIE2B	OCIE2A	TOIE2
$6F	TIMSK1	-	-	ICIE1	-	-	OCIE1B	OCIE1A	TOIE1
$6E	TIMSK0	-	-	-	-	-	OCIE0B	OCIE0A	TOIE0
$6D	PCMSK2	PCINT[23:16]							
$6C	PCMSK1	PCINT[15:8]							
$6B	PCMSK0	PCINT[7:0]							
$69	EICRA	-	-	-	-	ISC11	ISC10	ISC01	ISC00
$68	PCICR	-	-	-	PCIE4	PCIE3	PCIE2	PCIE1	PCIE0
$67	RCKCAL	RC32K Calibration							
$66	RCMCAL	RC32M Calibration							
$65	PRR1	-	-	PRWDT	-	PRTIM3	PREFL	PRPCI	-
$64	PRR/0	PRTWI	PRTIM2	PRTIM0	-	PRTIM1	PRSPI	PRUART0	PRADC
$62	VDTCR	WCE	SWR	-	VDTS			VDREN	VDTEN
$61	CLKPR	WCE	CKOE1	CKOE0	-	CLKPS			
$60	WDTCSR	WDIF	WDIE	WDP3	WDCE	WDE	WDP2	WDP1	WDP0
DirectIO Register									
$5F	SREG	I	T	H	S	V	N	Z	C
$5E	SPH	Stack Point High							

Addr	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
$5D	SPL	Stack Point Low							
$5C	E2PD3	E2PCTL Data register byte 3							
$5B	C1TR	AC1 trimming data							
$5A	E2PD1	E2PCTL Data register byte1							
$59	DSAH	DSA[31:16] access port of uDSC							
$58	DSAL	DSA[15:0] access port of uDSC							
$57	E2PD2	E2PCTL Data register byte 2							
$56	ECCR	WEN	EEN	ERN	SWM	CP1	CP0	ECS1	ECS0
$55	MCUCR	FWKEN	FPDEN	SWR	PUD	IRLD	IFAIL	IVSEL	WCE
$54	MCUSR	SWDD	-	-	OCDRF	WDRF	BORF	EXTRF	PORF
$53	SMCR	-	-	-	-	SM			SE
$52	C0TR	AC0 Trimming register							
$51	C0XR	-	C0OE	C0HYSE	C0PS0	C0WKE	C0FEN	C0FS1	C0FS0
$50	C0SR	C0D	C0BG	C0O	C0I	C0IE	C0IC	C0IS	
$4F	DTR0	TC0 Dead-band timing control register							
$4E	SPDR	SPI Data register							
$4D	SPSR	SPIF	WCOL	-	-	-	DUAL	-	SPI2X
$4C	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR	
$4B	GPIOR2	General Purpose Register 2							
$4A	GPIOR1	General Purpose Register 1							
$49	TCCR0C	DSX07	DSX06	DSX05	DSX04	-	-	DSX01	DSX00
$48	OCR0B	Timer 0 Output Compare Register B							
$47	OCR0A	Timer 0 Output Compare Register A							
$46	TCNT0	Timer 0 Counter							
$45	TCCR0B	FOC0A	FOC0B	OC0AS	DTEN0	WGM02	CS02	CS01	CS00
$44	TCCR0A	COM0A1	COM0A0	COM0B1	COM0B0	DOC0B	DOC0A	WGM01	WGM00
$43	GTCCR	TSM	-	-	-	-	-	PSRASY	PSRSYNC
$42	EEARH	E2PCTL Address High							
$41	EEARL	E2PCTL Address Low							
$40	E2PD0	E2PCTL Data byte 0							
$3F	EECR	EEPM2	EEPM2	EEPM1	EEPM0	EERIE	EEMWE	EEWE	EERE
$3E	GPIOR0	General Purpose Register 0							
$3D	EIMSK	-	-	-	-	-	-	INT1	INT0
$3C	EIFR	-	-	-	-	-	-	INTF1	INTF0
$3B	PCIFR	-	-	-	-	PCIF3	PCIF2	PCIF1	PCIF0
$3A	C1XR	-	C1OE	C1HYSE	C1PS0	C1WKE	C1FEN	C1FS1	C1FS0
$39	SPFR	RDFULL	RDEMPT	RDPTR1	RDPTR0	WRFULL	WREMPT	WRPTR1	WRPTR0
$38	TIFR3	-	-	ICF3	-	-	OCF3B	OCF3A	TOV3
$37	TIFR2	-	-	-	-	-	OCF2B	OCF2A	TOV2
$36	TIFR1	-	-	ICF1	-	-	OCF1B	OCF1A	TOV1
$35	TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0
$34	PORTF	Port Output of Group F							

Addr	Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
$33	DDRF	Data Direction of Group F							
$32	PINF	Port Input of Group F							
$31	DSDY	DSDY access port of uDSC							
$30	DSDX	DSDX access port of uDSC							
$2F	C1SR	C1D	C1BG	C1O	C1I	C1IE	C1IC	C1IS	
$2E	PORTE	Port Output of Group E							
$2D	DDRE	Data Direction of Group E							
$2C	PINE	Port Input of Group E							
$2B	PORTD	Port Output of Group D							
$2A	DDRD	Data Direction of Group D							
$29	PIND	Port Input of Group D							
$28	PORTC	Port Output of Group C							
$27	DDRC	Data Direction of Group C							
$26	PINC	Port Input of Group C							
$25	PORTB	Port Output of Group B							
$24	DDRB	Data Direction of Group B							
$23	PINB	Port Input of Group B							
$22	DSSD	DSSD access port of uDSC							
$21	DSIR	Instruction regiter of uDSC							
$20	DSCR	DSUEN	MM	D1	D0	-	DSN	DSZ	DSC

Instruction set cheat sheet

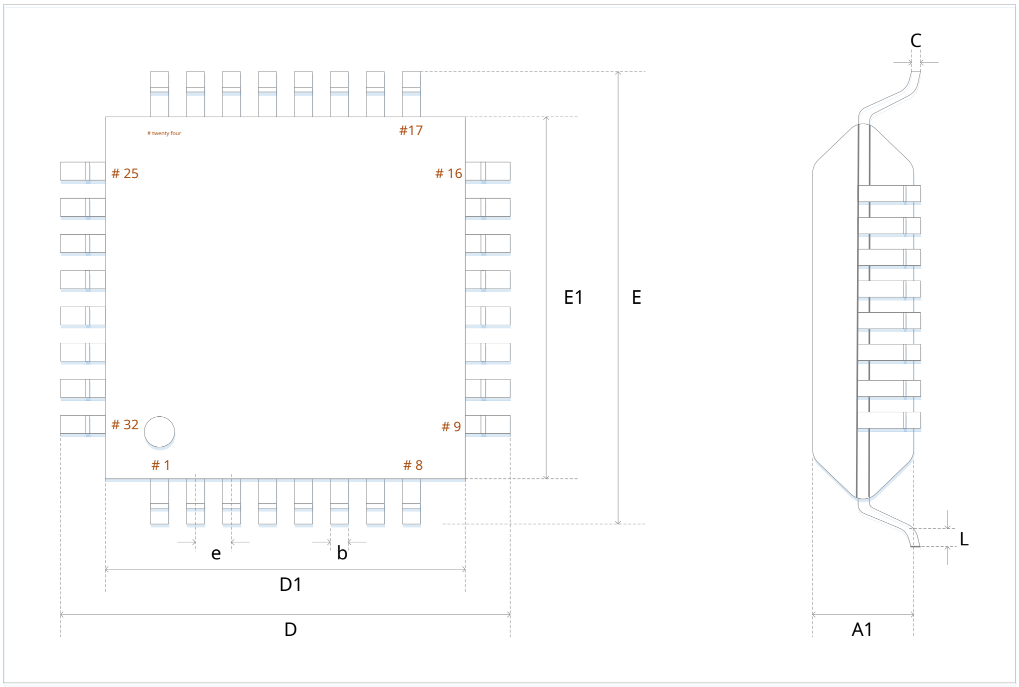
instruction	Operand	description	operating	Mark bit	cycle
Arithmetic and logical operation instructions					
ADD	R_d, R_r	Register addition	$R_d \leftarrow R_d + R_r$	Z,C,N,V,H	1
ADC	R_d, R_r	Add register with carry	$R_d \leftarrow R_d + R_r + CR_{dh}:R_{dl}$	Z,C,N,V,H	1
ADIW	R_{dl}, K	Add immediate numbers and words	$\leftarrow R_{dh}:R_{dl} + KR_d \leftarrow R_{d-}$	Z,C,N,V,S	1
SUB	R_d, R_r	Register addition and subtraction	R_r	Z,C,N,V,H	1
SUBI	R_d, K	Register minus constant	$R_d \leftarrow R_{d-} K$	Z,C,N,V,H	1
SBC	R_d, R_r	Addition and subtraction of registers with borrow	$R_d \leftarrow R_{d-} R_{r-} CR_d \leftarrow R$	Z,C,N,V,H	1
SBCI	R_d, K	Register with borrow to subtract constant	$_d - K\text{-}CR_{dh}:R_{dl} \leftarrow R_{dh:}$	Z,C,N,V,H	1
SBIW	R_{dl}, K	Subtract immediate numbers and words	$R_{dl-} KR_d \leftarrow R_d \& R_r$	Z,C,N,V,S	1
AND	R_d, R_r	Logical and		Z,N,V	1
ANDI	R_d, K	Register logic and constants	$R_d \leftarrow R_d \& K$	Z,N,V	1
OR	R_d, R_r	Logical OR	$R_d \leftarrow R_d \vert R_r$	Z,N,V	1
ORI	R_d, K	Register logic or constant	$R_d \leftarrow R_d \vert K$	Z,N,V	1
EOR	R_d, R_r	Register XOR	$R_d \leftarrow R_d \oplus R_r$	Z,N,V	1
COM	R_d	One's complement	$R_d \leftarrow \$FF\text{-}R_d$	Z,C,N,V	1
NEG	R_d	2 Prohibit complement	$R_d \leftarrow \$00\text{-}R_d$	Z,C,N,V,H	1
SBR	R_d, K	Set the bits in the register	$R_d \leftarrow R_d \vee K$	Z,N,V	1
CBR	R_d, K	Clear bits in the register	$R_d \leftarrow R_d \vee (\$FF - K)$	Z,N,V	1
INC	R_d	Increase	$R_d \leftarrow R_d + 1$	Z,N,V	1
DEC	R_d	Decrease	$R_d \leftarrow R_{d-} 1$	Z,N,V	1
TST	R_d	Test as 0 Or negative	$R_d \leftarrow R_d \& R_d$	Z,N,V	1
CLR	R_d	Clear register	$R_d \leftarrow R_d \oplus R_d$	Z,N,V	1
SER	R_d	The registers are all set to 1	$R_d \leftarrow \$FF$	None	1
MUL	R_d, R_r	Unsigned multiplication	$R_1: R_0 \leftarrow R_d \times R_r$	Z, C	1
MULS	R_d, R_r	Signed multiplication	$R_1: R_0 \leftarrow R_d \times R_r$	Z, C	1
MULSU	R_d, R_r	Signed number multiplied by unsigned number	$R_1: R_0 \leftarrow R_d \times R_r$	Z, C	1
FMUL	R_d, R_r	Unsigned multiplication, shift	$R_1: R_0 \leftarrow (R_d \times R_r) << 1\ R_1:$	Z, C	1
FMULS	R_d, R_r	Signed multiplication, shift	$R_0 \leftarrow (R_d \times R_r) << 1\ R_1: R_0$	Z, C	1
FMULSU	R_d, R_r	Signed number multiplied by unsigned number, shift	$\leftarrow (R_d \times R_r) << 1$	Z, C	1
Jump instruction					
RJMP	K	Relative jump	PC ← PC + K + 1	None	1
IJMP		Indirect jump (to Z Point to address)	PC ← Z	None	2
JMP	K	Jump directly	PC ← K	None	2
RCALL	K	Relative address subroutine call	PC ← PC + K + 1	None	1
ICALL		Indirect subroutine call (Z Point to address)	PC ← Z	None	2
CALL	K	Direct subroutine call	PC ← K	None	2
RET		Subroutine return	PC ← Stack	None	2
RETI		Interrupt return	PC ← Stack	I	2

instruction	Operand	description	operating	Mark bit	cycle
Jump instruction (continued)					
CPSE	R_d, R_r	Equal to jump	If(R_d=R_r) PC ← PC + 2 or 3	None	1/2
CP	R_d, R_r	Compare	R_d- R_r	Z,N,V,C,H	1
CPC	R_d, R_r	Carry comparison	R_d- R_r – C	Z,N,V,C,H	1
CPI	R_d, K	Compare with immediate	R_d- K	Z,N,V,C,H	1
SBRC	R_r, b	Bit is 0 I.e. skip the next instruction	If(R_r(b)=0) PC ← PC + 2 or 3	None	1/2
SBRS	R_r, b	Bit is 1 I.e. skip the next instruction	If(R_r(b)=1) PC ← PC + 2 or 3	None	1/2
SBIC	P, b	I/O Bit is 0 I.e. skip the next instruction	If(P(b)=0) PC ← PC + 2 or 3	None	1/2
SBIS	P, b	I/O Bit is 1 That is, skip the next instruction and	If(P(b)=1) PC ← PC + 2 or 3	None	1/2
BRBS	s, k	mark the status as 1 I.e. jump	If(SREG(S)=1) PC ← PC + K + 1	None	1/2
BRBC	s, k	Status is marked as 0 I.e. jump	If(SREG(S)=0) PC ← PC + K + 1	None	1/2
BREQ	k	Equal to jump	if (Z = 1) then PC ← PC + k + 1 if (Z =	None	1/2
BRNE	k	Jump without waiting	0) then PC ← PC + k + 1 if (C = 1)	None	1/2
BRCS	k	Carry then jump	then PC ← PC + k + 1 if (C = 0) then	None	1/2
BRCC	k	Jump without carry	PC ← PC + k + 1 if (C = 0) then PC ←	None	1/2
BRSH	k	Jump if not less than	PC + k + 1 if (C = 1) then PC ← PC + k	None	1/2
BRLO	k	Jump if less than	+ 1 if (N = 1) then PC ← PC + k + 1 if	None	1/2
BRMI	k	Jump if negative	(N = 0) then PC ← PC + k + 1 if (N⊕	None	1/2
BRPL	k	Regular jump	V = 0) then PC ← PC + k + 1 if (N⊕V	None	1/2
BRGE	k	Signed not less than jump	= 1) then PC ← PC + k + 1 if (H = 1)	None	1/2
BRLT	k	Signed less than 0 I.e. jump	then PC ← PC + k + 1 if (H = 0) then	None	1/2
BRHS	k	Half carry is 1 Then jump	PC ← PC + k + 1	None	1/2
BRHC	k	Half carry is 0 Then jump		None	1/2
BRTS	k	T Jump if set	if (T = 1) then PC ← PC + k + 1 if	None	1/2
BRTC	k	T Jump if cleared	(T = 0) then PC ← PC + k + 1 f (V	None	1/2
BRVS	k	Jump when overflow	= 1) then PC ← PC + k + 1 f (V =	None	1/2
BRVC	k	Jump without overflow	0) then PC ← PC + k + 1 f (I = 1)	None	1/2
BRIE	k	Jump when global interrupt is enabled	then PC ← PC + k + 1 f (I = 0)	None	1/2
BRID	k	Jump if global interrupt is disabled	then PC ← PC + k + 1	None	1/2
Data transfer instructions					
MOV	Rd, Rr	Move data between registers	Rd ← Rr	None	1
MOVW	Rd, Rr	Move one word of data	Rd+1:Rd ← Rr+1:Rr	None	1
LDI	Rd, K	Load immediate	Rd ← K	None	1
LD	Rd, X	Indirect loading	Rd ← (X)	None	1/2
LD	Rd, X+	Indirect loading, address increment	Rd ← (X), X ← X + 1	None	1/2
LD	Rd, -X	Decrease address, load indirectly	X ← X-1, Rd ← (X)	None	1/2
LD	Rd, Y	Indirect loading	Rd ← (Y)	None	1/2
LD	Rd, Y+	Indirect loading, address increment	Rd ← (Y), Y ← Y + 1	None	1/2
LD	Rd, -Y	Decrease address, load indirectly	Y ← Y-1, Rd ← (Y)	None	1/2
LDD	Rd, Y+q	Indirect loading with offset	Rd ← (Y + q)	None	1/2
LD	Rd, Z	Indirect loading	Rd ← (Z)	None	1/2

LD	Rd, Z+	Indirect loading, address increment	Rd ← (Z), Z ← Z+1	None	1/2
LD	Rd, -Z	Decrease address, load indirectly	Z ← Z-1, Rd ← (Z)	None	1/2
LDD	Rd, Z+q	Indirect loading with offset	Rd ← (Z + q)	None	1/2
LDS	Rd, k	Directly from SRAMLoading	Rd ← (k)	None	2
ST	X, Rr	Indirect storage	(X) ← Rr	None	1
ST	X+, Rr	Indirect storage, address increment	(X) ← Rr, X ← X + 1	None	1
ST	-X, Rr	Decrease address, indirect storage	X ← X-1, (X) ← Rr	None	1
ST	Y, Rr	Indirect storage	(Y) ← Rr	None	1
ST	Y+, Rr	Indirect storage, address increment	(Y) ← Rr, Y ← Y + 1	None	1
ST	-Y, Rr	Decrease address, indirect storage	Y ← Y-1, (Y) ← Rr	None	1
STD	Y+q, Rr	Indirect storage with offset	(Y + q) ← Rr	None	1
ST	Z, Rr	Indirect storage	(Z) ← Rr	None	1
ST	Z+, Rr	Indirect storage, address increment	(Z) ← Rr, Z ← Z + 1	None	1
ST	-Z, Rr	Decrease address, indirect storage	Z ← Z-1, (Z) ← Rr	None	1
STD	Z+q, Rr	Indirect storage with offset	(Z + q) ← Rr	None	1
STS	k, Rr	Store directly to SRAMin	(k) ← Rr	None	2
LPM		Load program space data	R0 ← (Z)	None	2
LPM	Rd, Z	Load program space data	Rd ← (Z)	None	2
LPM	Rd, Z+	Load program data, address increment	Rd ← (Z), Z ← Z+1	None	2
LD	Rd, Z+	Indirect loading, address increment	Rd ← (Z), Z ← Z+1	None	1
LD	Rd, -Z	Decrease address, load indirectly	Z ← Z-1, Rd ← (Z)	None	1
LDD	Rd, Z+q	Indirect loading with offset	Rd ← (Z + q)	None	1
LDS	Rd, k	Directly from SRAMLoading	Rd ← (k)	None	2
IN	Rd, P	Read port	Rd ← P	None	1
OUT	P, Rr	Write port	P ← Rr	None	1
PUSH	Rr	Push	STACK ← Rr	None	1
POP	Rd	Pop	Rd ← STACK	None	1/2
SBI	P, b	Set up IO register	I/O(P, b) ← 1	None	1
CBI	P, b	Cleared IO register	I/O(P, b) ← 0	None	1
LSL	Rd	Logical shift left	Rd(n+1) ← Rd(n), Rd(0) ← 0 Rd(n) ←	Z,C,N,V	1
LSR	Rd	Logical shift right	Rd(n+1), Rd(7) ← 0 Rd(0)←C, Rd(n+1)	Z	1
ROL	Rd	Rotate left with carry	← Rd(n), C←Rd(7) Rd(7)←C, Rd(n)	Z	1
ROR	Rd	Rotate right shift including carry	←Rd(n+1), C←Rd(0) Rd(n) ← Rd(n+1),	Z	1
ASR	Rd	Arithmetic shift right	n= 0:6	Z	1
SWAP	Rd	Bit swap	Rd(3:0) ←Rd(7:4), Rd(7:4)←Rd(3:0)	None	1
BSET	s	Set status bit	SREG(s) ← 1	SREG(s)	1
BCLR	s	Clear status bit	SREG(s) ← 0	SREG(s)	1
BST	Rr, b	Store to T Bit	T ← Rr(b)	T	1
BLD	Rd, b	read out T Bit to register	Rd(b) ← T	None	1
SEC		Set carry flag	C ← 1	C	1
CLC		Clear carry flag	C ← 0	C	1

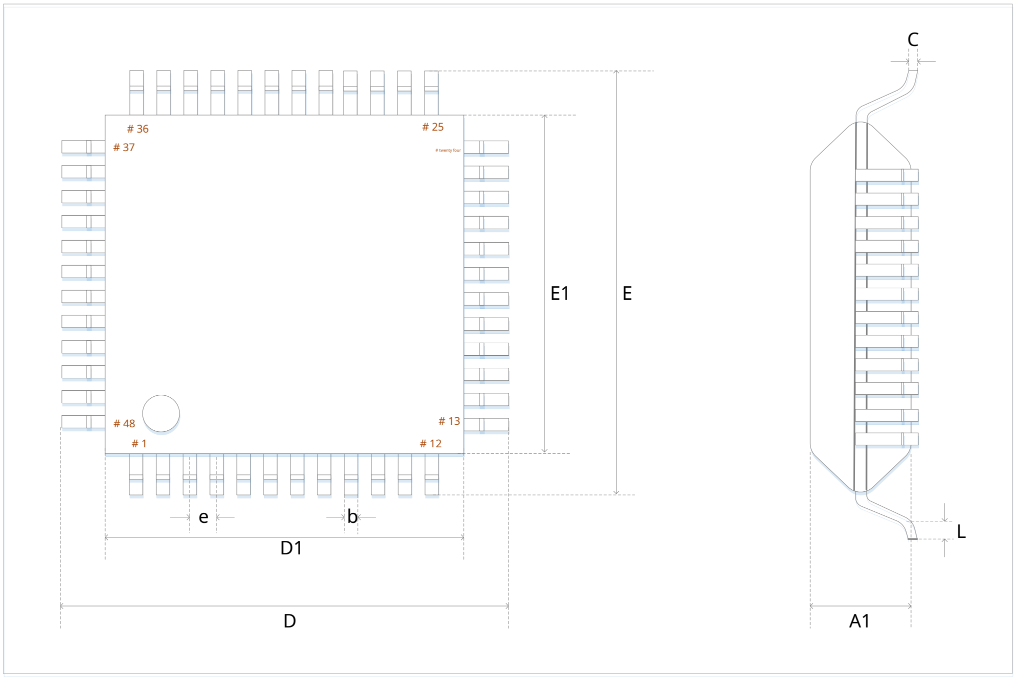
SEN		Set negative sign	N← 1	N	1
CLN		Clear negative sign	N← 0	N	1
SEZ		Set zero flag	Z ← 1	Z	1
CLZ		Clear zero flag	Z ← 0	Z	1
SEI		Enable global interrupt	I ← 1	I	1
CLI		Suppress global interrupts	I ← 0	I	1
SES		Set symbol test flag	S ← 1	S	1
CLS		Clear symbol test flag	S ← 0	S	1
SEV		Set two's complement overflow flag	V ← 1	V	1
CLV		Clear the twos complement overflow flag	V ← 0	V	1
SET		Set up T Bit (SREG)	T ← 1	T	1
CLT		Clear T Bit (SREG)	T ← 0	T	1
MCU Control instruction					
NOP		Null instruction		None	1
SLEEP		Enter sleep mode		None	1
WDR		Watchdog reset		None	1
BREAK		Soft breakpoint	For debugging purposes only	None	N/A
NOP		Null instruction		None	1
SLEEP		Enter sleep mode		None	1

Package parameters



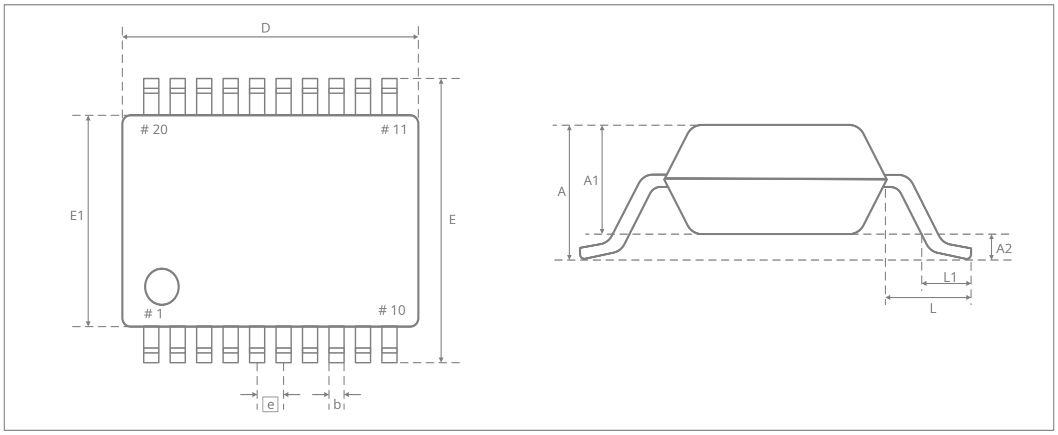
LQFP32 General size definition

Character code	Minimum	Typical value	Max	unit
D	8.90	9.00	9.10	mm
D1	6.90	7.00	7.10	mm
b	0.2	0.30	0.4	mm
e	0.75	0.80	0.85	mm
E	8.90	9.00	9.10	mm
E1	6.90	7.00	7.10	mm
C	-	0.10	-	mm
L	0.55	0.60	0.65	mm
A1	-	1.40	-	mm



LQFP48 General size definition

Character code	Minimum	Typical value	Max	unit
D	8.80	9.00	9.20	mm
D1	6.80	7.00	7.20	mm
b	0.17	0.22	0.27	mm
e	-	0.50BSC	-	mm
E	8.80	9.00	9.20	mm
E1	6.80	7.00	7.20	mm
C	0.09	-	0.2	mm
L	0.45	0.60	0.75	mm
A1	1.35	1.40	1.45	mm



SSOP20L General size definition

Character code	Minimum	Typical value	Max	unit
D	6.90	7.20	7.50	mm
A2	0.03	0.05	0.07	mm
b	0.22	0.30	0.38	mm
e	-	0.65	-	mm
E	7.40	7.80	8.20	mm
E1	5.00	5.30	5.60	mm
L1	0.55	-	0.95	mm
L	-	-	-	mm
A1	-	2.0	-	mm

Version history

V1.0.5 2018/9/26	delete QFP32/PB5 On the pin ADC11 Function correction AC1 On the positive end selectionConfiguration
V1.0.4 2017/11/15	correct SSOP20 PIN8/11 Definition
V1.0.3 2017/6/23	increase SSOP20 Package definition Update TMR3 Interrupt flagInstructions for
V1.0.2 2017/5/15	Update TMR0/TRM1/TMR3 About automatic PWMUpdated instructions for shutdown and restart SPI In the chapter SPI Description of interrupt processingAnd update SPFR Description of the register
V1.0.1 2017/2/13	delete I2C1 Section, this feature is not available Improved the definition of some registers
V1.0.0 2016/12/29	initial version