

MDT10F684

Data Sheet (Version V1.2)

8Bit FLASH

CMOS Microcontroller

1.0 Device Overview

-Memory:

FLASH space: 2K*14 Can withstand 10 Thousands of write operations. EEPROM space: 256 Bytes. 100 Thousands of write operations. SRAM space: 128 byte.

-8Level-deep hardware stack

-I/O Pin Configuration

With independent direction control 14 individual I/O Pin out: PA mouth 6 individual, PC mouth 8 individual.

High current sink/source for direct drive led.

PA Interrupt on port pin change.

PA Port-independent programmable weak pull-ups.

-Timer

Timer0: bring 8Bit programmable prescaler 8Bit timer/counter.

Enhanced Timer1:

- With prescaler 16bit timer/counter.
- external Timer1 Gate control (count enable)
- If you select INTOSC mode, or in LPM mode can be selected to use OSC1 and OSC2 as Timer1 oscillator.

Timer2: bring 8Bit period register, prescaler and postscaler 8bit timer/counter.

-Enhanced capture, comparison and PWM Modules

- 16Bit capture module, maximum resolution is 200ns.
- 16Bit comparison module, the maximum resolution is 200ns.
- With 1, 2 or 4 Output channels with programmable "dead time" 10 Bit PWM Module, the maximum frequency of the output signal is 20kHz.

-8aisle 10Bit ADC

8 External ADC Center.

ADC The reference voltage can be selected as internal or external reference by software. When internal reference is selected, the on-chip 2V, 3V, 4V or VDD as a reference voltage.

-Two analog comparators

On-chip programmable comparator reference voltage (CVREF) module (the reference voltage must be less than VDD, the minimum can be 0V)

The comparator inputs and outputs are directly accessible.

-Dual clock system

External high-speed clock: up to 20MHz.

Internal high speed clock: 8MHz RC (Fcpu Only supports 8MHz, 4MHz, 2MHz, 500KHz, 250KHz, 125KHz)

Internal low speed clock: RC Oscillator 31KHz.

-High performance RISC CPU

Just learn 35 Instructions.

All instructions except jump instructions are single cycle.

Direct, indirect, and relative addressing modes.

-Special Features

- High-precision internal oscillator, factory calibrated to +/-1%.
- The software selectable frequency range is 125kHz arrive 8MHz.
- Software Optional 31kHz Internal oscillator.
- Energy-saving sleep mode.
- Wide operating voltage range (2.0V arrive 5.5V)
- Industrial temperature range.
- Power-On Reset (Power-on Reset, POR)
- Power-On Delay Timer (Power-up Timer, PWRT) and the Oscillator Start-up Timer (Oscillator Start-up Timer, OST)
- With software control option PED Low voltage detection selection (detection voltage has 1.7V, 2.1V, 3.8V Optional)
- With on-chip oscillator (oscillator frequency can be selected by software, when the prescaler ratio is maximum, its nominal value is 268seconds) and software-enabled enhanced low-current watchdog timer (Watchdog Timer, WDT)
- Master reset with pull-up, can be used as input pin.
- Programmable code protection.

-Low power consumption

stand-by current:

- Voltage is 2.0V typical value 50nA.

Working current:

- The frequency is 1MHz, voltage is 2.0V the typical value is 200uA.
 - The frequency is 31kHz, voltage is 2.0V the typical value is 11uA.
- Watchdog timer current:
- Voltage is 2.0V the typical value is 1uA.

-Through two pins (PA0, PA1) Enables in-circuit serial programming and debugging.

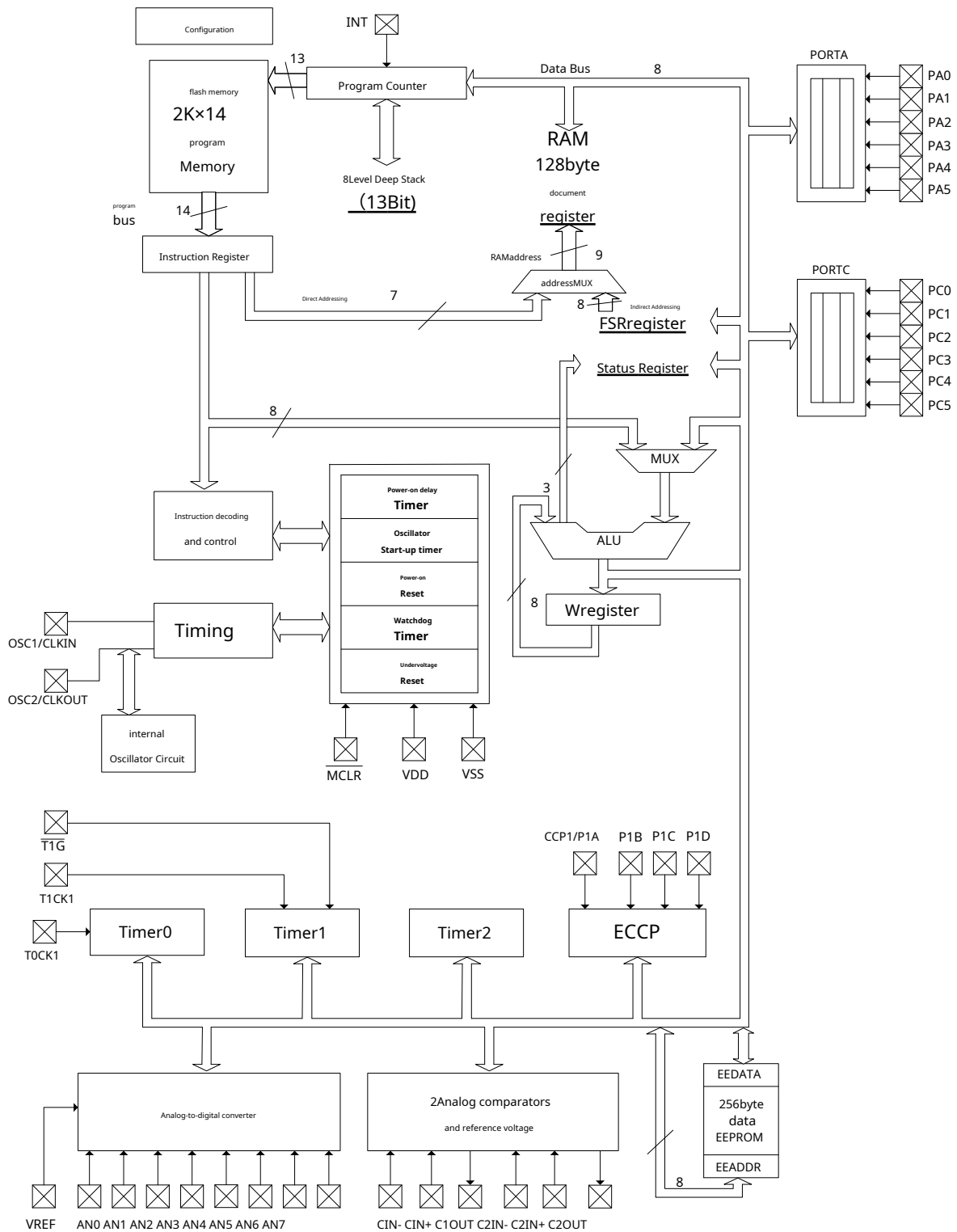
Devices	ROM	RAM		I/O	10Bit A/D Conversion Device (Number of channels)	Timer	PWM	Comparators	Wake-up function Number of pins	Encapsulation
	FLASH (word)	SRAM (word)	EEPROM (word)							
MDT10F684	2048	128	256	14	8	3	4	2	6	PDIP8, SOP8 PDIP14, SOP14, PDIP16, SOP16

1.0 Device Overview.....	1
1.1 System Structure Diagram.....	5
1.2 Package Pinout.....	6
1.3 Pin Description.....	7
2.0 Memory organization.....	8
2.1 Program Memory Organization.....	8
2.2 Data storage device structure.....	9
2.3 dataEEPROMMemory.....	15
2.4 PCLandPCHLAT.....	19
2.5 Indirect addressing,IARandMSRregister.....	20
2.6 Watchdog Timer (WDT)	twenty one
3.0 Reset.....	twenty four
3.1 Overview.....	twenty four
3.2 Power-On Reset.....	26
3.3 MCLR.....	26
3.4 Watchdog Reset.....	27
3.5 Power-On Delay Timer (PWRT)	27
3.6 Brown-out Reset.....	27
3.7 Delay Timing.....	28
3.8 power control(PSTA)register.....	30
4.0 System Clock.....	30
4.1 Overview.....	30
4.2 Oscillator Control.....	31
4.3 Clock Source Mode.....	32
4.4 External Clock Mode.....	32
4.5 Internal Clock Mode.....	35
4.6 Clock Switching.....	38
4.7 Dual-Speed Clock Startup Mode.....	38
4.8 Power-down mode (sleep).....	39
4.9 Fail-Safe Clock Monitor.....	41
5.0 Interrupts.....	42
5.1 Interrupt Special Function Registers.....	44
5.2 PA2/INTInterrupts.....	48
5.3 TIMER0Interrupts.....	48
5.4 PORTAInterrupt on Change.....	48
5.5 EEPROMWrite Data Interrupt.....	48
5.6 A/DInterrupts.....	48
5.7 Interrupted Context Preservation.....	49
6.0 I/Oport.....	49
6.1 PORTAandCPIOAregister.....	49
6.2 PORTCandCPIOCregister.....	51
6.3 Other Pin Functions.....	52

6.4 PORTAPin Description and Pin Diagram.....	54
6.5 PORTCPin Description and Pin Diagram.....	61
7.0Timer.....	65
7.1 TIMER0Modules.....	65
7.2With gateTIMER1Modules.....	67
7.3 TIMER2Modules.....	73
8.0Enhanced Capture/Compare/PWM+(with automatic shutoff and deadband) module.....	76
8.1surface:ECCPMode - Timer Resources Required.....	76
8.2register:CCP1CON: EnhancedCCP1Control Registers.....	76
8.3Capture Mode.....	77
8.4Comparison Mode.....	79
8.5 PWMmodel.....	80
8.6 PWM(Enhanced Mode)	84
9.0Analog-to-digital converter (ADC) Modules.....	94
9.1picture:ADCblock diagram.....	94
9.2 ADCConfiguration.....	94
9.3 ADCWorking Principle.....	97
9.4 ADCRegister Definition.....	99
9.5 A/DCollection time requirements.....	102
10.0Comparator Module.....	104
10.1Comparator Overview.....	104
10.2Comparator Configuration (Figure)	107
10.3Comparator Control.....	108
10.4Comparator Response Time.....	109
10.5How the Comparator Interrupt Works.....	109
10.6How sleep works.....	111
10.7Effects of Reset.....	111
10.8ComparatorsC2GatingTIMER1	112
10.9ComparatorsC2Output andTIMER1Synchronize.....	112
10.10Comparator Reference Voltage.....	113
11.0Configuration Registers.....	115
11.1 CONFIGOPTION.....	115
12.0Instruction List.....	117
13.0Electrical Characteristics.....	118
13.1Absolute Limit Ratings.....	118
13.2DC Electrical Characteristics.....	120
13.3AC Electrical Characteristics.....	121
14.0Development Support.....	122
14.1Simulation Information.....	122
14.2Burning Information.....	124

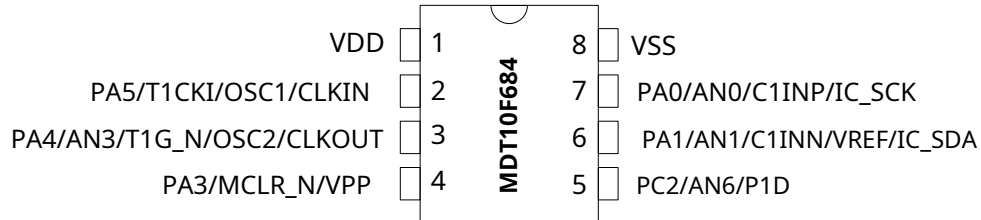
15.0 Packaging Information.....	125
15.1 P-DIP 8 PIN.....	125
15.2 SOP 8 PIN	126
15.3 P-DIP 14 PIN	127
15.4 SOP 14 PIN.....	128
15.5 P-DIP 16 PIN	129
15.6 SOP 16 PIN.....	130

1.1 System structure diagram

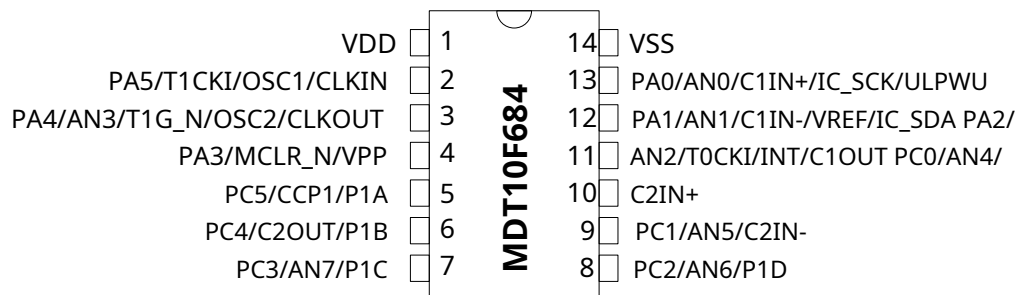


1.2 Package Pinout

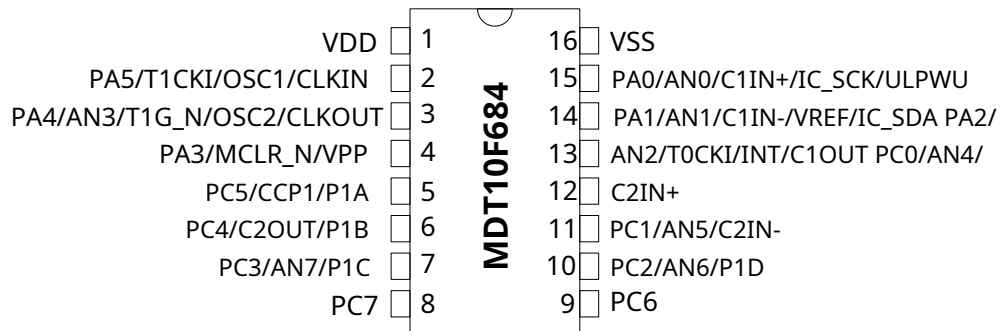
1.2.1 8Pin Diagram (DIP,SOP)



1.2.2 14Pin Diagram (DIP,SOP)



1.2.3 16Pin Diagram (DIP,SOP)



1.3 Pin Description

name	Function	enter type	Output type	illustrate
PA0/AN0/C1INP/IC_SCK/ULPWU	PA0	TTL	CMOS	Programmable pull-up and interrupt-on-changePORTA I/O
	AN0	AN	—	A/Daisle0enter
	C1IN+	AN	—	Comparators1The positive input
	IC_SCK	ST	—	Serial programming and debug clock input
	ULPWU	AN	—	Ultra-low power wake-up input
PA1/AN1/C1INM/VREF/IC_SDA	PA1	TTL	CMOS	Programmable pull-up and interrupt-on-changePORTA I/O
	AN1	AN	—	A/Daisle1enter
	C1IN-	AN	—	Comparators1The inverting input
	VREF	AN	—	A/DExternal reference voltage
	IC_SDA	ST/TTL	CMOS	Serial programming and debug data input and output
PA2/AN2/T0CKI/INT/C1OUT	PA2	ST	CMOS	Programmable pull-up and interrupt-on-changePORTA I/O
	AN2	AN	—	A/Daisle2enter
	T0CKI	ST	—	Timer0Clock Input
	INT	ST	—	External Interrupt
	C1OUT	—	CMOS	Comparators1Output
PA3/ MCLR /VPP	PA3	TTL	—	With interrupt on level changePORTAenter
	MCLR	ST	—	Master reset with internal pull-up
	VPP	HV	—	Programming voltage
PA4/AN3/ T1G /OSC2/CLKOUT	PA4	TTL	CMOS	Programmable pull-up and interrupt-on-changePORTA I/O
	AN3	AN	—	A/Daisle3enter
	T1G	ST	—	Timer1Gate control (count enable)
	OSC2	—	XTAL	Crystal Oscillator/Resonator
	CLKOUT	—	CMOS	Fosc/4Output
PA5/T1CKI/OSC1/CLKIN	PA5	TTL	CMOS	Programmable pull-up and interrupt-on-changePORTA I/O
	T1CKI	ST	—	Timer1clock
	OSC1	XTAL	—	Crystal Oscillator/Resonator
	CLKIN	ST	—	External clock input/RCoscillator Connections
PC0/AN4/C2INP	PC0	TTL	CMOS	PORTC I/O
	AN4	AN	—	A/Daisle4enter
	C2IN+	AN	—	Comparators2The positive input
PC1/AN5/C2INM	PC1	TTL	CMOS	PORTC I/O
	AN5	AN	—	A/Daisle5enter
	C2IN-	AN	—	Comparators2The inverting input
PC2/AN6/P1D	PC2	TTL	CMOS	PORTC I/O
	AN6	AN	—	A/Daisle6enter
	P1D	—	CMOS	PWMOutput
PC3/AN7/P1C	PC3	TTL	CMOS	PORTC I/O
	AN7	AN	—	A/Daisle7enter
	P1C	—	CMOS	PWMOutput
PC4/C2OUT/P1B	PC4	TTL	CMOS	PORTC I/O
	C2OUT	—	CMOS	Comparators2Output
	P1B	—	CMOS	PWMOutput
PC5/CCP1/P1A	PC5	TTL	CMOS	PORTC I/O
	CCP1	ST	CMOS	Capture Input/Compare Output
	P1A	—	CMOS	PWMOutput
PC6	PC6	TTL	CMOS	PORTC I/O
PC7	PC7	TTL	CMOS	PORTC I/O
VDD	VDD	power supply	—	Positive power supply terminal
VSS	VSS	power supply	—	Ground reference terminal

Figure caption:AN =Analog input or output
ST =bringCMOSSchmitt trigger input

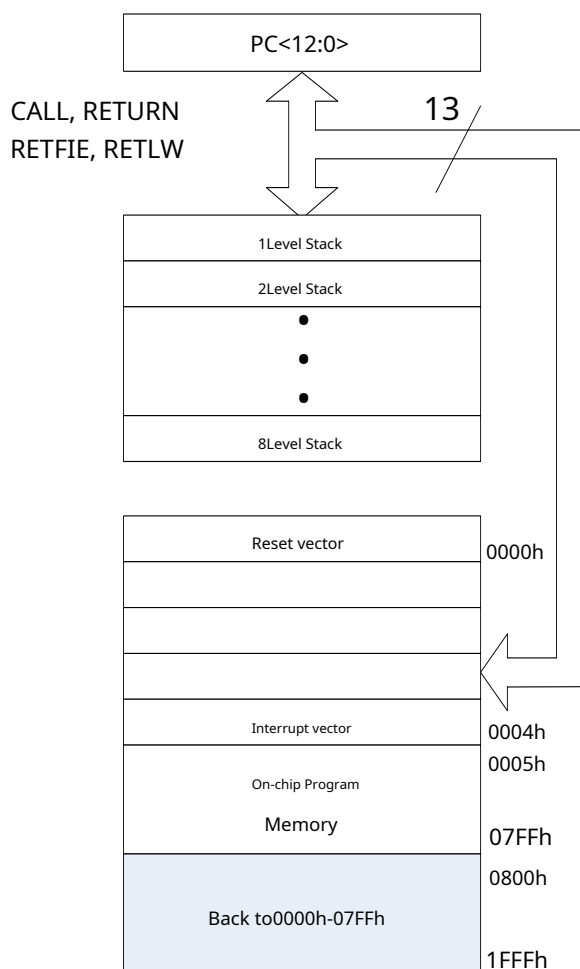
CMOS = CMOSCompatible input or output TTL =
TLLCompatible output

HV =high pressure
XTAL =Crystal Oscillator

2.0 Memory configuration

2.1 Program memory organization

MDT10F684 has a 13-bit program counter, which can be 8Kx14. The program memory space is addressed. MDT10F684 only the first 2Kx14 (0000h-07FFh) memory space. Accessing a location outside this boundary will result in actually accessing the first 2Kx14 storage space. The reset vector address is 0000h, the interrupt vector address is 0004h.



2.1.1 Reset vector (0000H)

A system reset vector with a length of one word (0000H).

- Power-on reset;
- Watchdog reset;
- External reset;

After any of the above resets occurs, the program will start from 0000H. If the system registers are restored to their default values, the execution will be restarted at the same time. STATUS Register TO and PD. The content of the flag bit can determine the system reset mode.

2.2.2.1 Special Function Register Bit Definitions (BANK0)

address	name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	POR and BOR Value
Bank 0										
00H	IAR	useMSRThe contents of the data memory are used to address this location (not a physical register).								xxxx xxxx
01H	RTCC	Timer0Module Registers								xxxx xxxx
02H	PCL	Program Counter (PC)								0000 0000
03H	STATUS	IRP ⁽¹⁾	PR1 ⁽¹⁾	PAGE	\overline{TF}	\overline{PF}	Z	HC	C	0001 1xxx
04H	MSR	Indirect data memory address pointer								xxxx xxxx
05H	PORTA ⁽²⁾	—	—	PA5	PA4	PA3	PA2	PA1	PA0	- -x0x000
06H	—	Not implemented								—
07H	PORTC ⁽²⁾	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	xxxx 0000
08H	—	Not implemented								—
09H	—	Not implemented								—
0AH	PCHLAT	—	—	—	Program Counter High5Write buffer					---0 0000
0BH	INTS	GIE	PE1E	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000
0CH	PIFB1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMRIF	0000 0000
0D	—	Not implemented								—
0E	TMR1L	16BitTMR1Low Byte Holding Register								xxxx xxxx
0FH	TMR1H	16BitTMR1High Byte Holding Register								xxxx xxxx
10H	T1STA	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	0000 0000
11H	TMR2	Timer2Module Registers								0000 0000
12H	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	- 000 0000
13H	CCPR1L	Capture/Compare/PWMregister1The low byte								xxxx xxxx
14H	CCPR1H	Capture/Compare/PWMregister1High Byte								xxxx xxxx
15H	CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000
16H	PWM1CON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000
17H	ECCPAS	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000 0000
18H	WDTCN	—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	---0 1000
19H	CMSTA	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000
1AH	CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	----- 10
1BH	—	Not implemented								—
1CH	—	Not implemented								—
1DH	—	Not implemented								—
1EH	ADRESH	Left-aligned formatA/DThe result is high8Right-aligned orA/DThe result is high2Bit								xxxx xxxx
1FH	ADS0	ADFM	ADVRS1	ADVRS0	CHS2	CHS1	CHS0	$\overline{GO/DONE}$	ADON	0000 0000

Figure caption: -- = Unimplemented cells are read as 0,u = constant,x = unknown,q = Value depends on the situation, shadow = unimplemented

- Note
- 1:IRP and RP1 bits are reserved and should always be kept clear.
 - 2:Depend on ADINSRegister-controlled port pins with analog functions are read as 0, although the data latch is undefined (POR) or unchanged (other resets).

2.2.2.2 Special Function Register Bit Definitions (BANK1)

address	name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	POR and BOR Value
Bank 1										
80H	IAR	useMSRThe contents of the data memory are used to address this location (not a physical register).								xxxx xxxx
81H	TMR	PA PH	IES	TCS	TCE	PSC	PS2	PS1	PS0	1111 1111
82H	PCL	Program Counter (PC)								0000 0000
83H	STATUS	IRP(1)	PR1(1)	PAGE	TF	PF	Z	HC	C	0001 1xxx
84H	MSR	Indirect data memory address pointer								xxxx xxxx
85H	CPIOA	—	—	CPIOA5	CPIOA4	CPIOA3	CPIOA2	CPIOA1	CPIOA0	--11 1111
86H	—	Not implemented								—
87H	CPIOC	CPIOC7	CPIOC6	CPIOC5	CPIOC4	CPIOC3	CPIOC2	CPIOC1	CPIOC0	1111 1111
88H	—	Not implemented								—
89H	—	Not implemented								—
8A	PCHLAT	—	—	—	Program Counter High5Write buffer					---0 0000
8BH	INTS	GIE	PE1E	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000
8CH	PIEB1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000
8DH	—	Not implemented								—
8E	PSTA	—	—	ULPWUE	SBOREN	—	—	POR	BOR	--01 --qq
8F	OSCCON	—	IRCF2	IRCF1	IRCF0	OSTS(2)	HTS	LTS	SCS	-110x000
90H	—	Not implemented								—
91H	ADINS	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111
92H	PR2	Timer2Module Period Register								1111 1111
93H	—	Not implemented								—
94H	—	Not implemented								—
95H	PAPHR(3)	—	—	PHA5	PHA4	—	PHA2	PHA1	PHA0	--11 -111
96H	PAINTR	—	—	PAINTR5	PAINTR4	PAINTR3	PAINTR2	PAINTR1	PAINTR0	--00 0000
97H	—	Not implemented								—
98H	—	Not implemented								—
99H	VRSTA	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0-0000
9AH	EEDATA	EEDATA7	EEDATA6	EEDATA5	EEDATA4	EEDATA3	EEDATA2	EEDATA1	EEDATA0	0000 0000
9BH	EEADR	EEADR7	EEADR6	EEADR5	EEADR4	EEADR3	EEADR2	EEADR1	EEADR0	0000 0000
9CH	EECON1	—	—	—	—	WRERR	WREN	WR	RD	---- x000
9DH	EECON2	EEPROMControl Register2(non-physical register)								-----
9E	ADRESL	Left-aligned formatA/DThe result is low2Right-aligned orA/DThe result is low8Bit								xxxx xxxx
9F	ADS1	ADRVO	ADCS2	ADCS1	ADCS0	—	—	—	—	0000 ----

Figure caption: -- = Unimplemented cells are read as 0, u = constant, x = unknown, q = Value depends on the situation, shadow = unimplemented

Note

- 1:IRP and RP1 bits are reserved and should always be kept clear.
- 2:OSCCON Register OSTS bit reset to 0, with two-speed start and LP, HS or XT is selected as the oscillator. 3: Configuration word register MCLRE for 1 hour RA3 Pull-up enabled.

2.2.2.3 STATUS Status Register

state (STATUS) registers contain:

- ALU Arithmetic status of
- Reset state
- Data storage (SRAM) of the bank select bits

Like any other register, the status register can be used as the destination register for any instruction. Z, H or C An instruction that writes to these three bits with the status register as the destination will disable writing to these three bits. Depending on the device logic, these bits will be set 1. In addition, you cannot write TF and PF. Therefore, when an instruction with the status register as the destination is executed, the result of the status register may be different from what is expected.

For example, executing CLR STATUS The instruction will clear the high 3 Position and Z Location 1. This makes the status register value "000uu1uu" (uu indicates unchanged).

Therefore, it is recommended to use only BCR, BSR, SWAPR and STWR instructions to change the status register, because these instructions do not affect any status bits. For other instructions that do not affect any status bits, see [No.12.0 Section "Instruction Set Summary"](#).

Note: 1: MDT10F684 Do not use STATUS Register IRP and RP1 bits and should be kept clear. It is not recommended to use these bits ,
 Because this may affect the upward compatibility of future products.
 2: In subtraction operation, C and HC The bits are used as borrow and half borrow flags respectively. For specific examples, see SUBWR and SUBWI instruction.

register 03H/83H: Status Register (STATUS)

reserve	reserve	R/W-0	R-1	R-1	R/W	R/W	R/W
IRP	RP1	PAGE	TF	PF	Z	HC	C
bit7							bit0

Figure caption:
 R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORT The value of 1 = Place 1 0 = Clear x = unknown

- bit7 **IRP:** This bit is reserved and should remain as 0

- bit6 **RP1:** This bit is reserved and should remain as 0

- bit5 **PAGE:** Register storage area selection bit (for direct addressing) 1 = Bank1 (80h-FFh) 0 = Bank0 (00h-7Fh)

- bit4 **TF:** Timeout status bit
 1 = After power on, the CLRWT instructions or SLEEP instruction 0 = occur WDT Timeout

- bit3 **PF:** Power-off flag
 1 = After a power-on reset or after CLRWT instruction 0 = Executed SLEEP instruction

- bit2 **Z:** Zero flag
 1 = The result of an arithmetic or logical operation is zero
 0 = The result of an arithmetic or logical operation is not zero



bit1 **HC:** Half carry/borrow bit (ADDWR,ADDWI,SUBWRandSUBWIInstructions). 1 =The result of4A carry occurred from the low position to the high position 0 =The result of4No carry from low bit to high bit occurs

bit0 **C:** Carry/borrow bit₍₁₎ (ADDWR,ADDWI,SUBWRandSUBWIInstruction) 1 =A carry occurred in the highest bit of the result (in subtraction, if no borrow occurred, the value is 1) 0 =No carry occurred in the highest digit of the result

Note 1:borrow **Bit** the polarity of the second operand is opposite. Subtraction is performed by adding the two's complement of the second operand (Two'sComplement) For shift instructions (RRR andRLR), the value of this bit comes from the highest or lowest bit of the source register.

2.2.2.4 TMR Option Register

Options (TMR) registers are readable and writable registers that contain various configuration bits that configure:

- Timer0/WDTPrescaler Assignment Bits
- ExternalPA2/INTInterrupt
- Timer0
- PORTAWeak pull-up on

Note: ToTimer0Specify1:1The prescaler ratio should beTMRRegisterPSCLocation1, to assign the prescaler toWDT.See [No. 7.1.4Section "Software Programmable Prescaler"](#) .

register81H: Option Register (TMR)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PAPH	IES	TCS	TCE	PSC	PS2	PS1	PS0
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORTThe value of 1 =Place1 0 =Clear x =unknown

bit7 **PAPH:**PORTAPull-up enable bit 1 = prohibitPORTApull up
 0 =Enabled by each port latch valuePORTApull up

bit6 **IES:** Interrupt edge selection bit
 1 =Depend onPA2/INTThe rising edge of the pin triggers an interrupt 0 = Depend onPA2/INTThe falling edge of the pin triggers an interrupt

bit5 **TCS:**RTCClock source selection bits
 1 = PA2/T0CKIThe signal transition on the pin 0 =Internal instruction cycle clock (FOSC/4)

bit4 **TCE:**RTCClock source edge select bit
 1 =existPA2/T0CKIIncremental when the pin level changes from high to low 0 =existPA2/T0CKIIncrement when the pin level changes from low to high

bit3 **PSC:** Prescaler allocation bits
 1 =Assign the prescaler toWDT



0 =Assign the prescaler toTimer0Modules

bit2-0

PS<2:0>: Prescaler ratio selection bit

Place value	RTCCFrequency division ratio	WDTFrequency division ratio
000	1:2	1:1
001	1:4	1:2
010	1:8	1:4
011	1:16	1:8
100	1:32	1:16
101	1:64	1:32
110	1:128	1:64
111	1:256	1:128

2.2.2.5 PSTAregister

power control(PSTA) register contains flag bits that distinguish between the following resets:

- Power-on reset (P \bar{O} R)
- Brown-out Reset (B \bar{O} R)
- Watchdog Timer Reset (WDT)
- ExternalMCLRReset

PSTARegisters are also used to control ultra low power wake-up andB \bar{O} R software enabled.

register8E: Power Control Register1 (PSTA)

U-0	U-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W
—	—	ULPWUE	SBOREN	MCR	IER	POR	BOR
bit7							bit0

Figure caption:

R =Readable bit

W =Writable bit

U =Unimplemented bit, read as0

-n = PORThe value of

1 =Place1

0 =Clear

x =unknown

bit7-6

Unimplemented: Read as0

bit5

ULPWUE: Ultra-low power wake-up enable bit

1 =Enable ultra low power wake-up

0 =Disable ultra low power wakeup

bit4

SBOREN: Software undervoltage detection enable bit₁

1 =Enable undervoltage detection

0 =Disable undervoltage detection

bit3

MCR: External reset status bit

1 =Power-on reset value

0 =An external reset has occurred

bit2

IER: Instruction error reset status bit

1 =A command error reset occurred

0 =Power-on reset value

bit1

POR: Power-on reset status bit

1 =No power-on reset occurred

0 =A power-on reset has occurred (must be reset by software after a power-on reset occurs),1)

bit0

BOR: Undervoltage reset status bit

1 =No brownout reset occurred

0 =A brown-out reset has occurred (must be reset by software after a brown-out reset occurs),1)

Note

1: When the configuration word registerLVREN<1:0> = 01This bit is allowed to be used whenBORTake control.

2.3dataEEPROMMemory

dataEEPROMDuring normal working hours (the entireVDDThe memory is not directly mapped to the file register space. Instead, it is indirectly addressed through special function registers.4individualSFRTto read and write this memory:

- EECON1
- EECON2(Writable but not readable)
- EEDATA
- EEADR

EEDATARegister storage8bits of data to be read or written, andEEADRRegisters to be accessedEEPROMThe address of the unit. MDT10F684have256Bytes of dataEEPROM, the address range is from00harriveFWf.

EEPROMThe data memory allows reading and writing in byte units. A byte write operation will automatically erase the target storage unit and write new data (i.e. erase first and then write). .EEPROMThe data memory can be repeatedly erased and written many times. The write time is controlled by an on-chip timer. The write time may vary due to voltage, temperature and chip differences. SeeNo.13.0Section "Electrical Characteristics" Refer to the AC specifications for exact limitations.

If the data memory is code protected,CPUYou can still continue toEEPROMThe memory is read and written. The device programmer can no longer access the dataEEPROMThe data,EEPROMThe cell reads as zero.

register9AH:EEPROMData Register (EEDATA)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEDATA7	EEDATA6	EEDATA5	EEDATA4	EEDATA3	EEDATA2	EEDATA1	EEDATA0
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7~0 **EEDATAn**, from the dataEEPROMRead or send dataEEPROMThe byte value written

register9BH:EEPROMAddress Registers (EEADR)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEADR7	EEADR6	EEADR5	EEADR4	EEADR3	EEADR2	EEADR1	EEADR0
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7~0 **EEADR**: Specify in256Units to be carried outEEPROMA unit of read/write operation



2.3.1 EECON1 and EECON2 register

EECON1 is the control register, its low 4 bits are physically implemented. High 4 bits unimplemented, read as 0.

Control bit RD and WR start read and write operations respectively. Software can only write when the read or write operation is completed, it is cleared by hardware. WR clearing the bit to zero effectively prevents accidental or premature termination of a write operation.

When a normal write operation is performed, WRERR will be placed 1. In these cases, the user can check after reset WRERR bit, clear it to zero and rewrite the corresponding unit. The data and address will be cleared. Therefore, it is necessary to re-EECON1 and EEADR registers are initialized.

At the end of the write operation, PIFB1 Interrupt flag register EEIF will be placed 1. This flag must be cleared by software.

EECON2 is not a physical register. Read EECON2 what you get is the whole 0. EECON2 registers are only in data EEPROM used during writing.

Note: When the data EEPROM performs a write operation (WR = 1) should not be modified EECON1, EECON2, EEADR and EEADRH register.

register 9CH: EEPROM Control Register (EECON1)

U-0	U-0	U-0	U-0	R/W	R/W-0	R/S-0	R/S-0
—	—	—	—	WRERR	WREN	WR	RD
bit7							bit0

Figure caption:
 R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORTThe value of 1 = Place 1 0 = Clear x = unknown

- bit7~4 Unimplemented: Read as 0

- bit3 **WRERR**: EEPROM Error flag
 1 = A write operation is terminated prematurely (any MCLR Reset, WDT reset or brownout detection)
 0 = Write operation completed

- bit2 **WREN**: EEPROM Write Enable Bit 1 =
 Allow write cycles
 0 = Prohibition of data EEPROM write operation

- bit1 **WR**: Write control bit
 1 = Start a write cycle (Hardware will clear this bit once the write operation is complete. Software can only write when the write operation is completed. It cannot be cleared.)
 0 = data input EEPROM The cycle is completed

- bit0 **RD**: Read control bit
 1 = start up EEPROM read operation (read takes one cycle. RD bit is cleared by hardware. Software can only read when the read operation is completed. It cannot be cleared.)
 0 = Not Started EEPROM Read Operation



2.3.2 Read Data EEPROM Memory

To read a data memory location, the user must write the address to EEADR register, and then EECON1 Control bits of registers RDPlace1, like [example2-3-2](#). In the next cycle, EEDATA register has data. Therefore, the data can be read by the next instruction. EEDATA This value will be retained until the next time the user reads from or writes to the cell (during a write operation).

example2-3-2: Read data EEPROM

```
LDR    data_addr,W    ;
BSR    STATUS,PAGE    ;Bank1
STWR   EEADR          ; Send the address to be read to the address register
BSR    EECON1,RD      ;Send read command
LDR    EEDATA,W       ; Read the result and send WRegister Return
```

2.3.3 Writing Data EEPROM Memory

To write EEPROM Data storage unit, The user must first write the address of the unit into EEADR register and write the data into EEDATA. The user must then begin writing each byte in a specific order.

example2-3-3: Write data EEPROM

```
LDR    data_addr,W    ;
BSR    STATUS,PAGE    ;Bank 1
STWR   EEADR          ;Write address to address register
BCR    STATUS,PAGE    ;bank 0
LDR    data_buff,W    ;
BSR    STATUS,PAGE    ;BANK1
STWR   EEDATA         ; Send the data to be written to the data register
BSR    EECON1,WREN    ; Enable write operation
BCR    INTS,GIE       ; Turn off global interrupt
BTSC   INTS,GIE       ;
LJUMP  $-2            ;
LDWI   0X55           ; The following four lines are fixed and must be written like this
STWR   EECON2         ;
LDWI   0XAA           ;
STWR   EECON2         ;
BSR    EECON1,WR      ;Write operation
BTSC   EECON1,WR      ;wait EEPROM operation completed
LJUMP  $-1            ;
BCR    EECON1,WREN    ; Turn off the write operation to prevent accidental writing
BSR    INTS,GIE       ; Enable global interrupt
```

If the above sequence is not followed completely (55h Write EECON2, Will AAh Write EECON2, Then WR Location 1) byte by byte, the write operation will not start. During the execution of this code segment, the four lines of fixed writing must disable interrupts, otherwise unpredictable results may occur. Cycle counting is performed during the execution of the required sequence. When the count value is not equal to the cycle required to execute the required sequence, data cannot be written to EEPROM.

In addition, it is necessary to EECON1 middle WREN Location 1. This mechanism prevents the write operation from being performed due to code execution errors (exceptions). (Right now Program out of control) causes incorrect data writing to EEPROM. Unless updated EEPROM, users should always keep WREN Bit cleared. WREN Bit cannot be cleared by hardware.

2.3.4 Write verification

Depending on the application, data will be written to EEPROM. The actual value is compared with the value to be written (see [example 2-3-4](#)) is a good programming practice used.

example 2-3-4: Write verification

```
BSR    STATUS,PAGE    ;BANK1
LDR    EEDATA,W       ;The last read data is saved to W
BSR    EECON1,RD      ;read EEPROM
XORWR  EEDATA,W       ; Compare the two read results
BCR    STATUS,PAGE
BTSS   STATUS,Z        ; Are they equal?
LJUMP  WRITE_ERR      ; If not equal, error handling
:      ; Equal, continue
```

2.3.4.1 Usage Data EEPROM

Data EEPROM is a high-endurance, byte-addressable array optimized for storing frequently changing information (such as program variables or other frequently updated data). When the variables in one sector change frequently, while the variables in another sector remain unchanged, it is possible to exceed the total number of writes (specified as 124) but does not exceed the total number of writes to a single byte (specified as 120 and 120A). If In this case, the array must be refreshed. For this reason, variables that do not change frequently (such as constants, IDs and calibration values, etc.) are stored in the flash memory.

2.3.5 Avoid accidental write operations

In some cases, users may not want to write data to EEPROM. The memory has various mechanisms to prevent EEPROM writes. When powered on, WREN is cleared. In addition, the power-on delay timer (delay 64ms) can also prevent accidental writing to EEPROM.

Write operation start sequence and WREN bits work together to prevent miswrites in the following situations:

Undervoltage

Power glitch

Software failure

2.3.6 Code protection data EEPROM operation

The data memory can be protected by setting the Configuration Word register (Register CONFIG). This is done by clearing the bit to zero.

When the code protection feature of the data memory is enabled, CPU ability to transfer data from data EEPROM is prevented. If code protection is enabled for data memory, it is recommended that you also enable code protection for program memory. This will cause unused program memory locations to be programmed as 0 (This will serve as a NOP implementation) to prevent someone from accessing the data storage by writing additional programs that can output the contents of the data storage. Programming unused locations in program memory also helps prevent code protection of data memory from being violated.

2.3.7 With dataEEPROMRelated registers/bits

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	PORandBOR The value of	All other Reset value
INTS	GIE	PEIE	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
PIFB1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMRIF	0000 0000	0000 0000
PIEB1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
EEDATA	EEDATA7	EEDATA6	EEDATA5	EEDATA4	EEDATA3	EEDATA2	EEDATA1	EEDATA0	0000 0000	0000 0000
EEADR	EEADR7	EEADR6	EEADR5	EEADR4	EEADR3	EEADR3	EEADR1	EEADR0	0000 0000	0000 0000
EECON1	—	—	—	—	WRERR	WREN	WR	RD	---- x000	---- q000
EECON2	EEPROMControl Register2								-----	-----

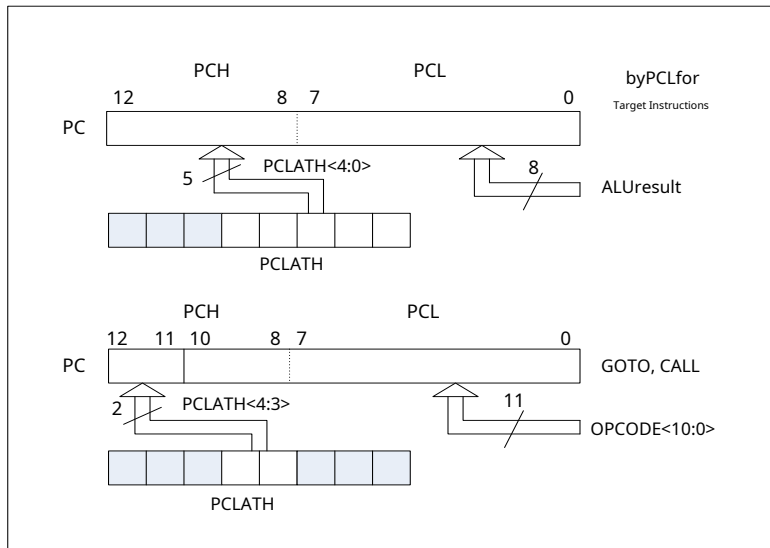
Figure caption: x =unknown,u =unchanged, — = unimplemented (read as0) ,q =The value depends on the situation.
dataEEPROMShaded cells are not used by the module.

Note 1:EECON2Not a physical register.

2.4 PCLandPCHLAT

Program Counter (PC)for13Bit width.8The bits are readable and writablePCLRegister, High5Bit(PC<12:8>) fromPCHLAT, cannot be read or written directly. As long as a reset occurs,PCit will be cleared to zero.[picture2-4](#) Shows loadingPCTwo cases of value.[picture2-4](#) The example above shows that when writingPCL (PCHLAT<4:0>→PCH) is how to loadPCof.[picture2-4](#) The following example shows how to executeLCALLorLJUMP During the instruction period (PCHLAT<4:3>→PCH), how to loadPCof.

picture2-4: Loading in different situationsPC



2.4.1 RevisePCL

Execute anyPCLInstructions that use the register as the destination will also cause the program counter toPC<12:8>Bit(PCH)quiltPCHLAT This can be done by setting the desired high5Bit WritePCHLATregister to change the contents of the program counter.8 Bit WritePCLregister, all the program counter13The bit will becomePCHLATThe values contained in the registers and the values written toPCLThe value in the register.

calculateLJUMPInstructions are written by adding an offset to the program counter (ADDWR PCL). By modifyingPCLRegister jump to lookup table or program branch table (calculationLJUMP) should be used with caution.PCHLATSet to the starting address of the table. If the table length is greater than 255instructions, or if the memory address is low8Located in the middle of the table0xFFReturn to0x00, then every time a rollover occurs between the table start address and the target address in the table,PCHLATMust all be increasing.

2.4.2 Stack

The device has 8-level depth, 13-bit-wide hardware stack (see [picture2-2](#) and [picture2-4](#)). The stack space does not occupy either the program storage area or the data storage area, and the stack pointer is not readable or writable. CALL instruction or when an interrupt causes the program to jump, the value PC will be pushed into (PUSH) stack. RET, RTI or RTFI instruction, the breakpoint address in the stack will be popped from the stack (POP) arrive PC middle. PCHLAT Not affected PUSH or POP Impact of operation.

The stack works like a circular buffer. This means that when the stack is pushed 8 times after the second 9th time the value pushed on the stack for the second time will overwrite the value saved when the stack was pushed for the first time, and the value pushed on the stack for the tenth time will overwrite the value saved when the stack was pushed for the second time, and so on.

Note: 1: There is no status flag to indicate whether the stack overflows or underflows.

2: does not exist is called PUSH or POP. The stack is pushed or popped by executing CALL, RET, RTI and

RTFI instruction, or originates from pointing to an interrupt vector address.

register 02H/82H: Program Counter Low Byte Control Register (PCL)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
bit7							Bit0

Figure caption:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as 0

-n = PORT The value of

1 = Place 1

0 = Clear

x = unknown

bit7-0 **PC<7:0>**: Program counter low 8-bit.

register 0AH/8AH: Program Counter High Byte Control Register (PCHLAT)

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	PC12	PC11	PC10	PC9	PC8
bit7							bit0

Figure caption:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as 0

-n = PORT The value of

1 = Place 1

0 = Clear

x = unknown

bit7-5 Unimplemented: Read as 0

bit4-0 **PC<12:8>**: Program counter high 5-bit.

2.5 Indirect addressing, IAR and MSR register

IAR The register is not a real register. Use IAR register addressing will produce indirect addressing.

use IAR registers can be indirectly addressed. IAR register instructions are actually file select registers (MSR) to access the data pointed to by IAR. A read operation will return 00h. Indirect IAR writing to this register will result in a no-operation (although status flags may be affected). By 8-bit position MSR registers and STATUS register IRP bits can be combined to obtain a valid 9-bit address, such as [picture2-5](#) shown.

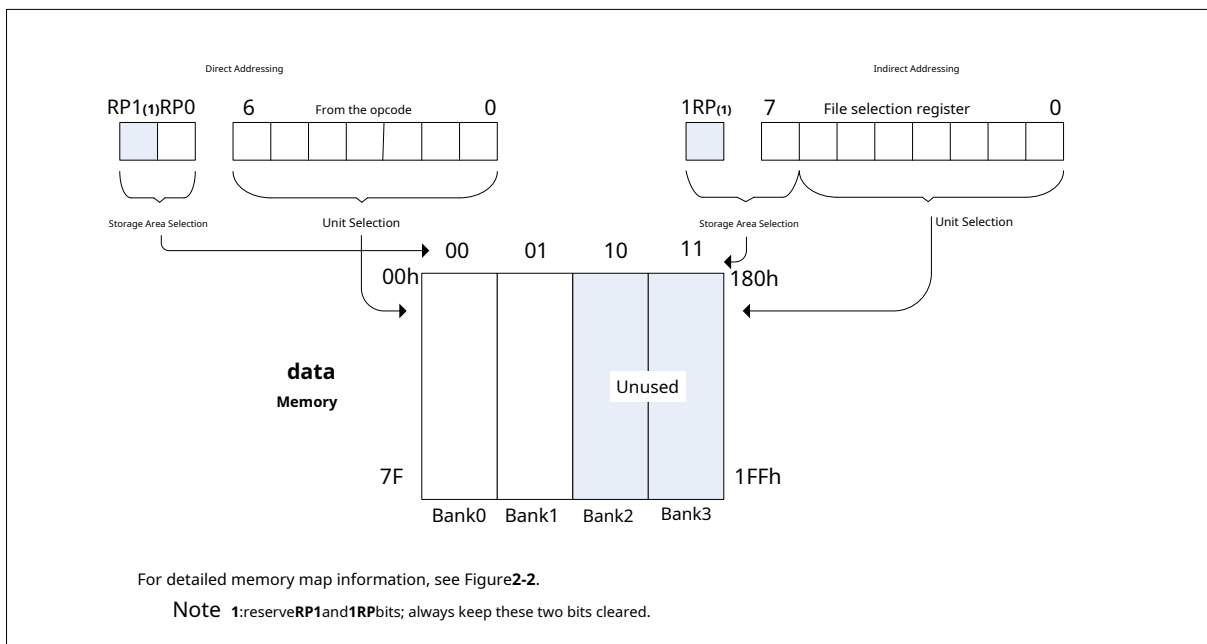
[example2-5](#) Given a method that uses indirect addressing, RAM Address Unit 20h-2Fh simple procedure for clearing.

example2-5: Indirect addressing

```

;Function: Address20Start of continuous16BytesRAMclear0
LDWI      0X20      ;
STWR      MSR       ;Address pointer initial value
NEXT:     CLRR      IAR       ; ClearMSRThe register value corresponding to the value
          INCR      MSR       ;Increment address pointer
          BTSS      MSR,4     ;Are you done?
          LJUMP     NEXT      ; Not finished yet, continue
CONTINUE: ;finished
    
```

picture2-5:MDT10F684Direct/indirect addressing



2.6 Watchdog Timer (WDT)

WDT Features include:

- Work on LFINTOSC (31kHz)
- Include 16Bit Prescaler
- and Timer0 Share 8Bit Prescaler
- Timeout period from 1ms to 268Second

Configuration bits and using software control

WDT in such [surface2-6-1](#) Cleared to zero in the stated case.

2.6.1 WDT Oscillator

WDTThe time base comes from31kHz LFINTOSC.OSCCONRegisterLTSBit does not reflectLFINTOSCWhether it is enabled.

On all resetsWDTCONThe value is "---01000". The nominal time base is thus17ms.

Note: When the Oscillator Start-up Timer (OST)hour,WDTRemains in reset state becauseOSTuseWDTThe ripple counter is used to perform the oscillator delay count.OSTAfter the count expires,WDTWill start counting (if enabled) .

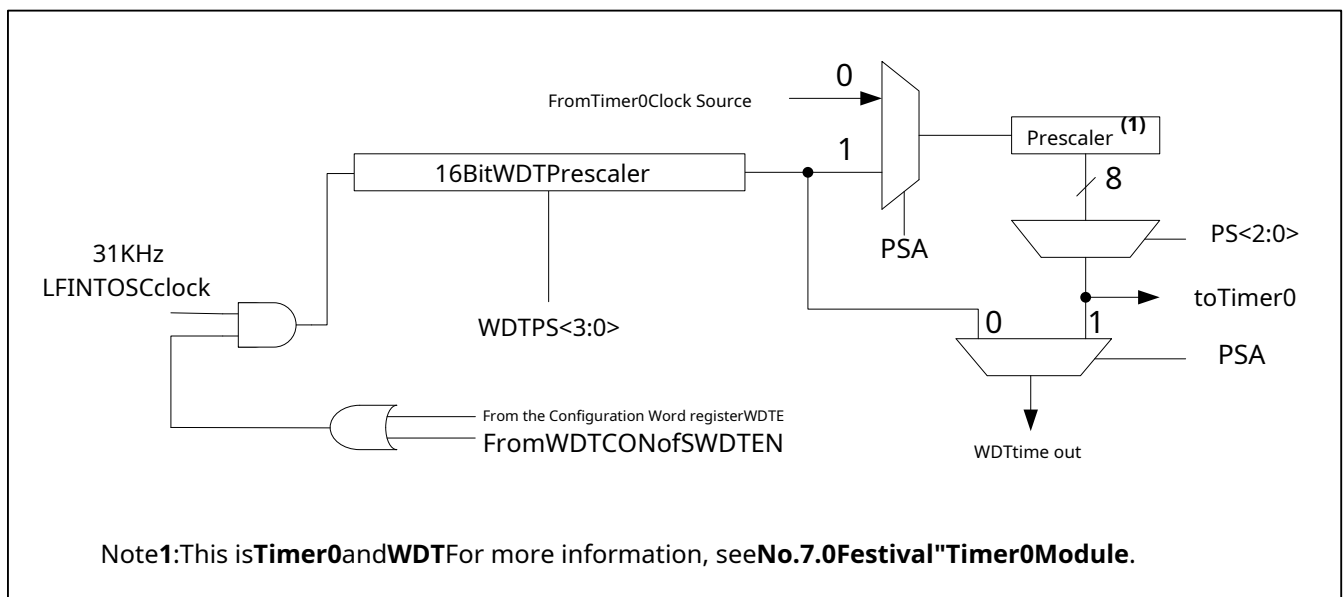
2.6.2 WDT control

WDEbit in the Configuration Word register.1hour,WDTcontinue to operate.

Configuration Word RegisterWDELocation1hour,WDTCONRegisterSWDTENbit has no effect. IfWDEClear, thenSWDTENBits can be used to enable and disableWDT. The location1EnableWDT, clear this bit to disableWDT.

TMRRegisterPSCandPS<2:0>Bit Function ANDMDT10F684For more information, see[No. 7.1 Festival "Timer0 Modules."](#)

picture2-6: Watchdog Timer Block Diagram



register18H: Watchdog Timer Control Register (WDTCON)

U-0	U-0	U-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN
bit7							bit0

Figure caption:

R =Readable bit

W =Writable bit

U =Unimplemented bit, read as0

-n = PORThe value of

1 =Place1

0 =Clear

x =unknown

bit7-5

Unimplemented: Read as0

Bit4-1

WDTPS<3:0>: Watchdog timer period selection bit

value = prescaler

0000 = 1:32

0001 = 1:64

0010 = 1:128

0011 = 1:256

0100 = 1:512(Reset value)

0101 = 1:1024

0110 = 1:2048

0111 = 1:4096

1000 = 1:8192

1001 = 1:16384

1010 = 1:32768

1011 = 1:65536

1100 =reserve

1101 =reserve

1110 =reserve

1111 =reserve

Bit0

SWDTEN: Software enables or disables the watchdog timer bit⁽¹⁾

1 = WDTOpen

0 = WDTOff (Reset value)

Note

1: If the Configuration Word Register (CONFIG)ofWDTEConfiguration bits =1,butWDTis always enabled, and the state of this control bit Status irrelevant.

If the Configuration Word register (CONFIG)ofWDTEConfiguration bits =0, you can use this control bit to turn on/offWDT.

surface2-6-1:WDTstate

condition	WDT
CLRWTOrder	Clear
Oscillator Failure Detection	
Exit Sleep + System Clock =T1OSC,EXTRC,NTRCorEXTCLK	
Exit Sleep + System Clock =XT,HSorLP	Clear untilOSTFinish

surface2-6-2: Summary of registers related to the watchdog timer

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	PORandBOR The value of	All other Reset value
WDTCON	—	—	—	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	---0 0000	---0 0000
TMR	PAPH	IES	TCS	TCE	PSC	PS2	PS1	PS0	1111 1111	1111 1111
CONFIG0	CPDB	CPB	MCLRE	PWRTE	WDTE	FOSC2	FOSC1	FOSC0	—	—

Figure caption: Shaded cells are not used by the watchdog timer.

Note 1For the operation of all bits in the Configuration Word register, see[register1200H](#).

3.0 Reset

3.1 Overview

MDT10F684 There are several different types of reset: a)

Power-on reset (POR)

b) During normal working hours WDT Reset

c) During dormancy WDT Reset

d) During normal working hours MCLR Reset

e) During dormancy MCLR Reset

f) Undervoltage reset (BOR)

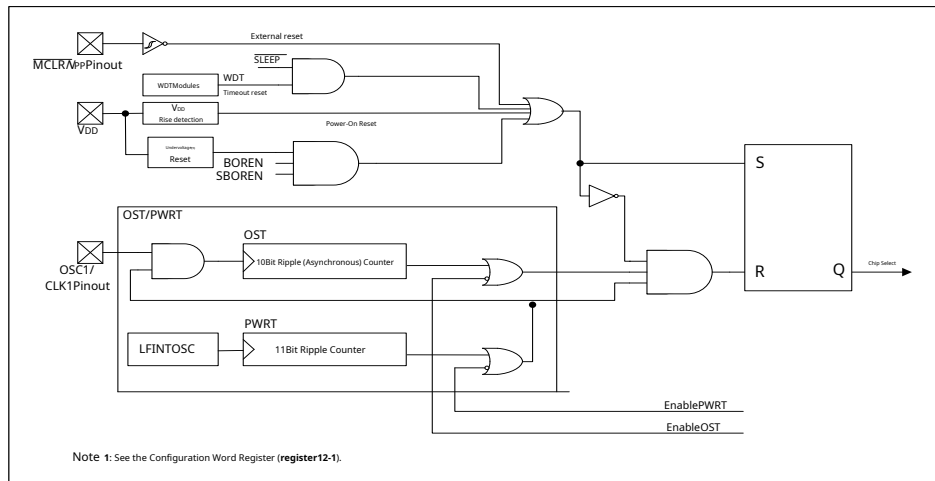
Some registers are not affected by any reset; their state is unknown on a power-on reset, and unchanged on other resets. Most registers reset to their "reset state" on a reset of :

- Power-on reset
- MCLR Reset
- During sleep MCLR Reset
- WDT Reset

WDT Wake-up does not cause registers to WDT This is because wakeup is treated as resuming normal operation. TO and PD The bits are set differently in different reset situations. 1 Or clear, such as [surface3-1](#) The software can use these bits to determine the nature of the reset. The detailed reset status of all registers is shown in [surface3-2](#) shown.

[picture3-1](#) A simplified block diagram of the on-chip reset circuit is given. MCLR There is a noise filter on the reset path to detect and filter out small pulses. For pulse width specifications, see [No.13.0 Section "Electrical Characteristics "](#) .

picture3-1: Simplified block diagram of the on-chip reset circuit



surface3-1: STATUS/PSTABits and their meanings

POR	BOR	TF	PF	condition
0	x	1	1	Power-On Reset
u	0	1	1	Brown-out reset
u	u	0	u	WDT Reset
u	u	0	0	WDT wake
u	u	u	u	During normal working hours MCLR Reset
U	u	1	0	During dormancy MCLR Reset

Figure caption: u = constant, x = unknown

surface3-2: Initial state of the register

register	address	Power-On Reset	MCLRReset WDTReset Brown-out reset ⁽¹⁾	Wake-up from Sleep via interrupt passWDTDelayed wakeup from sleep mode
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
IAR	00h/80h	xxxx xxxx	xxxx xxxx	uuuu uuuu
RTCC	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h	0000 0000	0000 0000	PC + 1 ⁽³⁾
STATUS	03h/83h	0001 1xxx	000q quuu ⁽⁴⁾	uuuq quuu ⁽⁴⁾
MSR	04h/84h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA ⁽²⁾	05h	- -x0x000	- - u0 u000	- - uu uuuu
PORTC ⁽²⁾	07h	- - xx 0000	- - uu 0000	- - uu uuuu
PCHLAT	0Ah/8Ah	- - - 0 0000	- - - 0 0000	- - - u uuuu
INTS	0Bh/8Bh	0000 0000	0000 0000	uuuu uuuu ⁽²⁾
PIFB1	0Ch	0000 0000	0000 0000	uuuu uuuu ⁽²⁾
TMR1L	0eH	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	0F	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1STA	10h	0000 0000	uuuu uuuu	-uuu uuuu
TMR2	11h	0000 0000	0000 0000	uuuu uuuu
T2CON	12h	- 000 0000	- 000 0000	-uuu uuuu
CCPR1L	13h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	14h	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	15h	0000 0000	0000 0000	uuuu uuuu
PWM1CON	16h	0000 0000	0000 0000	uuuu uuuu
ECCPAS	17h	0000 0000	0000 0000	uuuu uuuu
WDTCON	18h	- - - 0 1000	- - - 0 1000	- - - u uuuu
CMSTA	19h	0000 0000	0000 0000	uuuu uuuu
CMCON1	1Ah	- - - - - 10	- - - - - 10	- - - - - uu
ADRESH	1E	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADS0	1F	00-0 0000	00-0 0000	uu-u uuuu
TMR	81h	1111 1111	1111 1111	uuuu uuuu
CPIOA	85h	- - 11 1111	- - 11 1111	- - uu uuuu
CPIOC	87h	- - 11 1111	- - 11 1111	- - uu uuuu
PIEB1	8Ch	0000 0000	0000 0000	uuuu uuuu
PSTA	8E	- - 01 100x	- - 0u 00uq ^(1,5)	- - uu uuuu
OSCCON	8F	-110x000	-110 q000	-uuu uuuu
ADINS	91h	1111 1111	1111 1111	uuuu uuuu
PR2	92h	1111 1111	1111 1111	1111 1111
PAPHR ⁽³⁾	95h	- - 11 -111	- - 11 -111	uuuu uuuu
PAINTR	96h	- - 00 0000	- - 00 0000	- - uu uuuu
VRSTA	99h	0-0-0000	0-0-0000	uu-uuuu
EEDATA	9Ah	0000 0000	0000 0000	uuuu uuuu
EEADR	9B	0000 0000	0000 0000	uuuu uuuu
EECON1	9Ch	- - - - x000	- - - - q000	- - - - uuuu
EECON2	9D	- - - - - - -	- - - - - - -	- - - - - - -
ADRESL	9E	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADS1	9F	0000 ----	0000 ----	uuuu ----

Figure caption: u = constant, x = Unknown, - = unimplemented bit, read as 0, q = The value depends on the specific conditions.

- Note**
- 1: If VDDToo low, the power-on reset will be activated and the registers will be affected differently. 2: INTS and / or PIFB1 Register 1 One or more bits will be affected (cause wake-up). 3: When the device is awakened by an interrupt and GIE Location 1 hour, PCLoad interrupt vector (0004h).
 - 4: For reset values under specific conditions, see [surface3-7-1](#).
 - 5: If the reset is due to brownout, then bit 0 = 0 All other resets will cause bit 0 = u.
 - 6: Depend on ANINS Register-controlled port pins with analog functions are read as 0, even if the data latch is undefined (POR) or unchanged (other resets).

3.2 Power-On Reset

On-chip power-on reset circuitry holds the device in reset until a voltage level is reached sufficient for normal operation. The pin is connected to the VDDA maximum rise time is required to achieve VDD. See [No.13.0 Section "Electrical Characteristics"](#) If Brown-out Reset is enabled, the maximum rise time specification does not apply. The Brown-out Reset circuit will hold the device in Reset until VDD achieve VBOR (See [No.3.6Section "Brownout Reset \(BOR\)"](#)).

Note: When VDD is low, the power-on reset circuit will not generate an internal reset. To re-enable the power-on reset, VDD must maintain at least 100 μ s of Vss Voltage.

When the device starts normal operation (exits reset), the device's operating parameters (i.e. voltage, frequency, temperature, etc.) must be met to ensure normal operation. If these conditions are not met, the device must remain in reset until the operating conditions are met.

3.3 MCLR

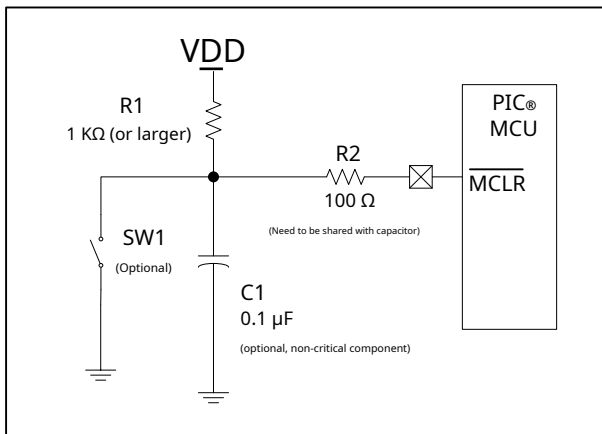
There is a noise filter in the reset path. This filter detects and removes small pulses.

It should be noted that the MCLR pin is driven low.

If the voltage on the pin exceeds the specified value, and an overcurrent exceeding the specified value flows through the device. Therefore, it is recommended not to connect MCLR pins directly to VDD. Recommended use is given in [picture3-3](#) RC network.

By clearing the Configuration Word register MCLR bit, which enables the internal MCLR option. MCLR = 0 When the chip is reset, bit signal. MCLR = 1, PA3/MCLR pin becomes an external reset input. In this mode, PA3/MCLR pins have to VDD Weak pull-up function.

picture3-3:suggested MCLR Circuit



3.4 Watchdog reset

3.5 Power-On Delay Timer (PWRT)

The Power-up Timer provides a time delay only during power-up (power-on reset or brown-out reset). The power-on delay timer uses a fixed delay of (nominal value). The oscillator is used as the clock source, and the operating frequency is 31 kHz. For more information, see [No. 4.5 Section "Internal Clock Mode"](#).

If only PWRT is active, the chip is held in reset. PWRT delay is sufficient time to rise to the required level flat.

Position PWRT can be disabled (if set 1). Although not required, the Power-up Timer should be enabled when Brown-out Reset is enabled.

The delay of the power-on delay timer varies from chip to chip due to the following reasons:

- VDD difference
- Temperature differences
- Manufacturing process differences

See DC parameters for details ([No. 13.0 Section "Electrical Characteristics"](#)).

Note: The pin is below VSS voltage spikes, including those greater than 80 mA current, which can cause latch-up. When a "low" level is applied to the pin, a resistor with a value between 50-100 Ω series resistor instead of pulling the pin directly to VSS.

3.6 Brown-out reset

Configuration Word Register BOREN0 and BOREN1 Bit used to select 4. Two additional Brown-out Reset modes are added to allow software or hardware to control the enable control mode. BOREN<1:0> = 01. When PSTA Register SBOREN Bit Enable/Disable BOR, so that it can be controlled by software. BOREN<1:0>, the brownout reset can be automatically disabled during sleep mode to save power; and re-enabled after wake-up. SBOREN For the definition of the Configuration Word, see [register PSTA](#).

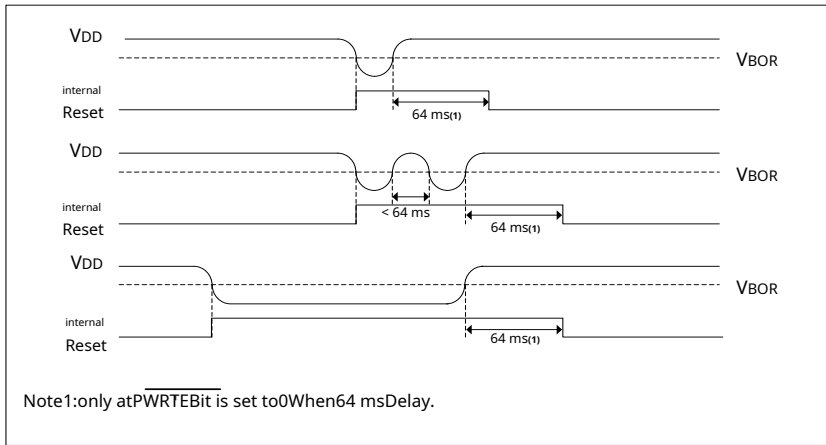
If VDD falls to VBOR below, and the duration exceeds the parameter value (TBOR) (See [No. 13.0 Section "Electrical Characteristics"](#), the undervoltage condition will The device is reset. VDD This will happen regardless of the rate of change of VDD. Lower than VBOR The time is less than the parameter value (TBOR), a reset may not necessarily occur.

Any reset (power-on reset, brown-out reset, or watchdog timer reset, etc.) will keep the chip in reset until VDD up to VBOR. Above (see [picture 3-6](#)) If enabled, the Power-up Timer will start at this time and will keep the device in Reset longer. 64 ms.

Note: The Configuration Word register PWRTEN Bit is used to enable the Power-up Timer.

If during the Power-up Delay Timer running, VDD reduces to VBOR. After that, the chip will return to the Brown-out Reset state and the Power-up Delay Timer will return to its initial state. VDD up to VBOR The power-on delay timer will execute a period of 64 ms reset.

picture3-6: Undervoltage condition



surface3-6: Summary of registers related to undervoltage

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR and BOR	All other
									The value of	Reset value
PSTA	—	—	ULPWUE	SBOREN	—	—	POR	BOR	-- -0 0000	-- -0 0000
STATUS	IRP	PR1	PAGE	TF	PF	Z	HC	C	1111 1111	1111 1111

Figure caption: u = constant, x = Unknown, — = unimplemented bit, read as 0, q = The value depends on the specific situation. BOR Shaded cells are not used. 1: Other

Note (non-power-on) resets include normal operation MCLR reset and watchdog timer reset.

3.7 Delay Timing

The delay sequence during power-on is as follows:

- exist POR After the delay, apply a PWRT Delay.
- PWRT Activate after timeout OST.

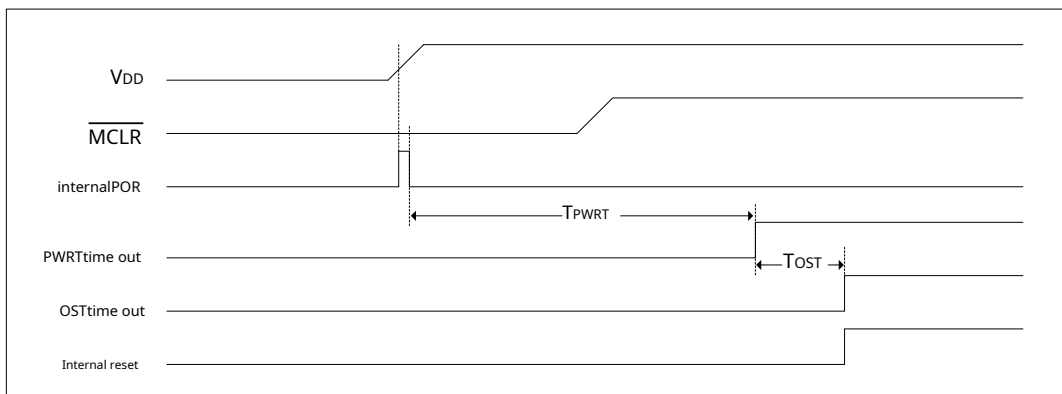
The total delay time depends on the oscillator configuration and PWRT. For example, in EC Mode and PWRT Bits are erased (PWRT Disabled), there will be no delay at all. [picture3-7-1](#), [3-7-2](#) and [3-7-3](#) The delay timing for each case is shown in Figure 1. After the oscillator starts oscillating, the device will start oscillating at INTOSC as a clock source to execute code (see [No.4.7.2 Section "Two-Speed Start-up Sequence"](#) and [No.4.9 Section "Fail-Safe Clock Monitor"](#))

Since the delay is triggered by the power-on reset pulse, if MCLR is pulled high, the low level is maintained long enough, all delays will end.

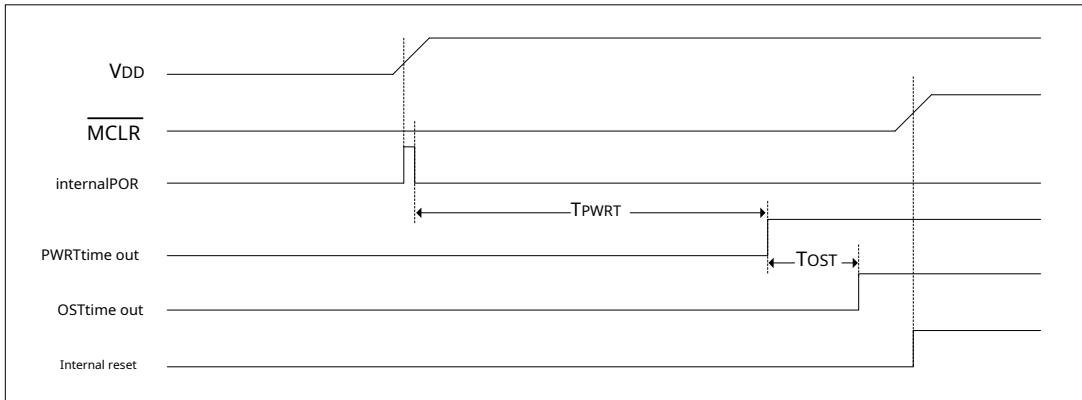
When pulled high, the device will immediately begin executing code (see [picture3-7-2](#)). This is useful for testing or synchronizing multiple parallel jobs. MDT10F684 It is very useful for devices.

[surface3-7-1](#) The reset conditions of some special registers are given, and [surface3-2](#) The reset conditions for all registers are given.

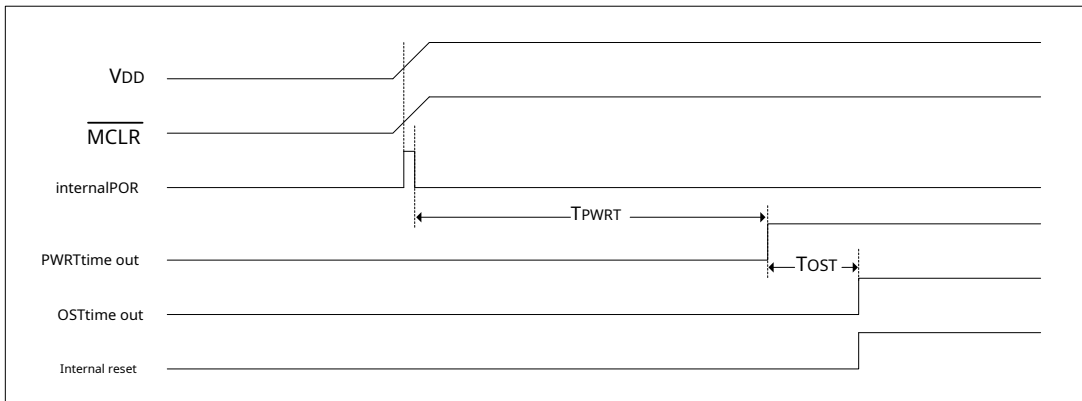
picture3-7-1: Timeout sequence at power-on (MCLR Delay) : situation 1



picture3-7-2: Timeout sequence at power-on (MCLRDelay) :situation2



picture3-7-3: Timeout sequence at power-on (MCLRandVDD)



surface3-7-1: Initial state of special registers

condition	Program Counter	Status Register	PSTRegister
Power-On Reset	0000h	0001 1xxx	-- 01 --0x
During normal working hoursMCLRReset	0000h	000u uuuu	-- 0u --uu
During dormancyMCLRReset	0000h	0001 0uuu	-- 0u --uu
WDTReset	0000h	0000 uuuu	-- 0u --uu
WDTwake	PC + 1	uuu0 0uuu	--uu --uu
Brown-out reset	0000h	0001 1uuu	-- 01 --u0
Wake-up from Sleep via interrupt	PC + 1(1)	uuu1 0uuu	--uu --uu

Legend: u = unchanged, x = unknown, — = unimplemented bit, read as '0'.

Note 1: When the device is awakened by an interrupt and the global interrupt enable bit GIE is set, after executing PC+1, the PC is loaded into the interrupt register (0004h).

surface3-7-2: Timeouts in various situations

Oscillator Configuration	Power-on		Brown-out reset		Waking from Hibernation
	PWRTE = 0	PWRTE = 1	PWRTE = 0	PWRTE = 1	
XT,HS,LP	$TPWRT + 1024 * TOSC$	$1024 * TOSC$	$TPWRT + 1024 * TOSC$	$1024 * TOSC$	$1024 * TOSC$
RC,EC,INTOSC	$TPWRT$	—	$TPWRT$	—	—

3.8 power control (PSTA) register

Power Control Register PSTA ([register 8E](#)) has two status bits that indicate the type of reset that last occurred.

bit 0 yes BOR (Brown-out reset) flag. BOR is unknown at power-on reset. The user must then check on subsequent resets if BOR is 0? If so, it means that a brown-out reset has occurred. BOREN <1:0> = 00) hour, BOR The status bits are "don't cares" and are not necessarily predictable.

bit 1 yes POR (Power-on reset) flag, the value at power-on reset is 0, otherwise it is not affected. After power-on reset, the user must write 1. After a subsequent reset, if POR is 0, it indicates that a power-on reset has occurred (i.e. VDD may have become low).

For more information, see [No. 6.3.4 Section "Ultra-low power wake-up"](#) and [No. 3.6 Section "Brownout Reset \(BOR\)"](#).

4.0 System clock

4.1 Overview

The oscillators have a variety of clock sources and selection features, allowing them to be used in a wide range of applications while maximizing performance and minimizing power consumption. [picture 4-1](#)

A block diagram of the oscillator module is given.

The clock source can be configured to be an external oscillator, a quartz crystal resonator, a ceramic resonator, and a resistor and capacitor (RC) circuit. In addition, the system clock source can be configured to be provided by one of two internal oscillators, with a software selectable speed. Other clock features include:

- External or internal system clock source selectable by software.

- Two-Speed Start-up mode minimizes the delay between external oscillator startup and code execution.

Fail-Safe Clock Monitor (FSCM) is designed to detect failure of the external clock source (LP, XT, HS, EC or RC mode) and automatically switch to the internal oscillator.

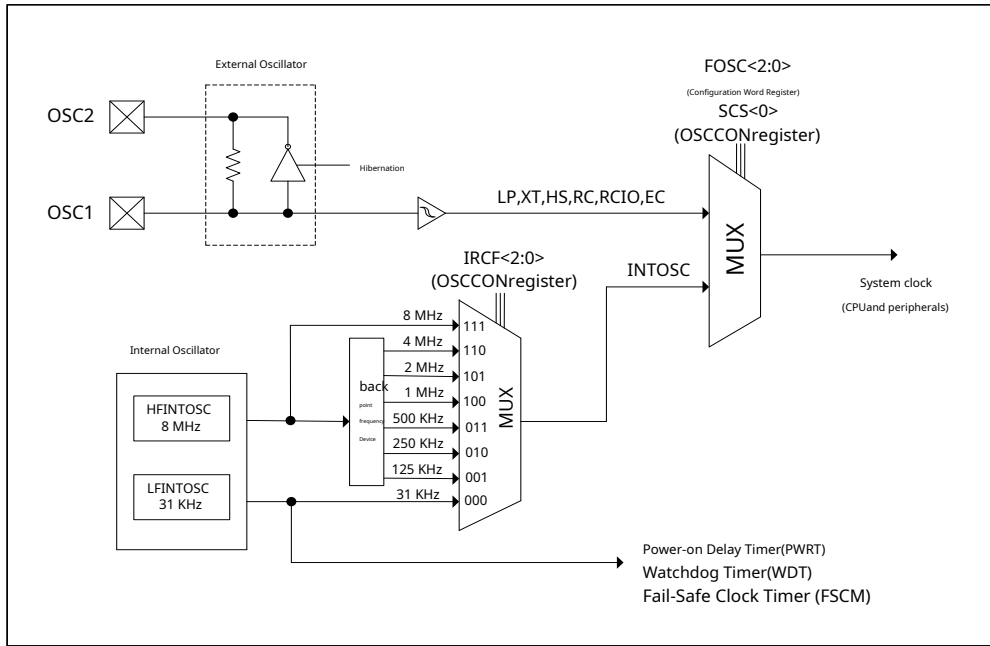
The oscillator module can be configured as follows:

1. EC — External clock, I/O exist OSC2/CLKOUT superior.
2. LP — 32kHz Low power crystal mode.
3. XT — Medium gain crystal or ceramic resonator oscillator mode.
4. HS — High-gain crystal or ceramic resonator mode.
5. RC — External RC (RC), FOSC/4 Output to OSC2/CLKOUT.
6. RCIO — External RC, I/O exist OSC2/CLKOUT superior.
7. INTOSC — Internal oscillator, FOSC/4 Output to OSC2 and I/O exist OSC1/CLKIN superior.
8. INTOSCIO — Internal oscillator, I/O exist OSC2/CLKOUT and OSC1/CLKIN superior.

Through the Configuration Word register (CONFIG) of FOSC <2:0> The internal clock can be generated by two internal oscillators. HFINTOSC is a calibrated high frequency oscillator. LFINTOSC is an uncalibrated low frequency oscillator.

picture 4-1: MCUClock Source Block Diagram





4.2 Oscillator Control

Oscillator Control (OSCCON) register (picture4-1) Controls options such as system clock and frequency selection. OSCCON The register contains the following

Bit:

- Frequency selection bits (IRCF)
- Frequency status bit (HTS and LTS)
- System clock control bit (OSTS and SCS)

register8F: Oscillator Control Register (OSCCON)

U-0	R/W-1	R/W-1	R/W-0	R-1	R-0	R-0	R/W-0
—	IRCF2	IRCF1	IRCF0	OSTS₍₁₎	HTS	LTS	SCS
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7 Unimplemented: Read as0

bit6-4 **IRCF<2:0>**: Internal oscillator frequency selection bit
 000 = 31kHz
 001 = 125kHz
 010 = 250kHz
 011 = 500kHz
 100 = 1MHz
 101 = 2MHz
 110 = 4MHz(Default) 111
 = 8MHz

bit3 **OSTS**: Oscillator start-up timeout status bit₍₁₎
 1 =The device operates in FOSC<2:0> Defined external clock
 0 =The device runs on the internal oscillator (HFINTOSC or LFINTOSC)

bit2 **HTS:**HFINTOSC(high frequency--8MHzarrive125kHz) Status Bit 1
= HFINTOSCStablize
0 = HFINTOSCUstable

bit1 **LTS:**LFINTOSC(Low frequency—31kHz) Status Bit
1 = LFINTOSCStablize
0 = LFINTOSCUstable

bit0 **SCS:** System clock selection bit
1 =Internal oscillator for system clock
0 =Clock sourceFOSC<2:0>Decide

Note 1: Dual-speed start and selectLP,XTorHSWhen the oscillator mode is selected or the fail-safe mode is enabled, this bit will be reset to0.

4.3Clock source mode

The clock source modes can be divided into external and internal modes.

External clock mode relies on external circuitry to provide the clock source. Examples include: Oscillator modules (ECmodel) , quartz crystal resonator or ceramic resonator Oscillator (LP,XTandHSmode) and RC (RC) mode circuit.

The internal clock source is built into the oscillator module. The oscillator module has two internal oscillators, one is8MHzHigh frequency internal oscillator (HFINTOSC) ,the other is31kHzLow frequency internal oscillator (LFINTOSC).

accessibleOSCCONRegisters for system clock selection (SCS) bits to select the system clock between external or internal clock sources. (For more information For more information, see[No.4.6Section "Clock Switching"](#)).

4.4External clock mode

4.4.1Oscillator Start-up Timer (OST)

If the oscillator module is configured asLP,XTorHSmode, Oscillator Start-up Timer (OST)OSC1Oscillation count1024 times. This occurs on power-on reset (POR) and after the Power-up Delay Timer (PWRT) delay has expired (if configured), or after waking up from Sleep. During this time, the program counter does not increment and program execution is suspended.OSTEnsure that the oscillator circuit using a quartz crystal resonator or ceramic resonator has started and is providing a stable system clock signal to the oscillator module. When switching between clock sources, a certain delay is required to allow the new clock to stabilize. [surface4-4-1](#) An example of oscillator delay is given.

To minimize latency between external oscillator start-up and code execution, a Two-Speed Clock Start-up mode can be selected (see[No.4.7Section "Dual-Speed Clock Startup Mode"](#)).

surface4-4-1: Oscillator Delay Example

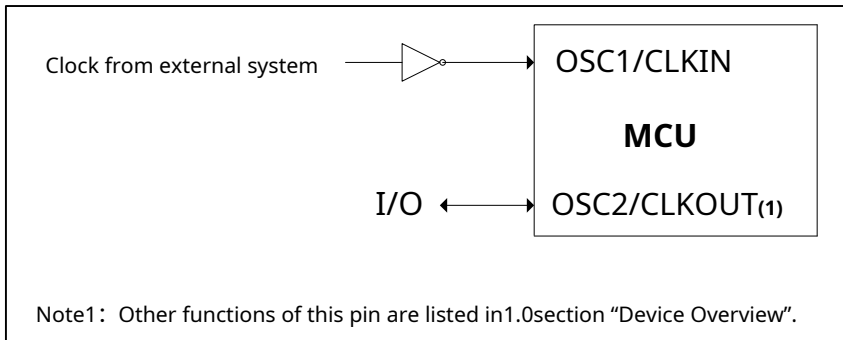
Switch from	Switch to	frequency	Oscillator Delay
Hibernate/POR	LFINTOSC HFINTOSC	31kHz 125kHzarrive8MHz	Oscillator Warm-up Delay (T _{WARM})
Hibernate/POR	EC,RC	DC—20MHz	Double cycle
LFINTOSC (31kHz)	EC,RC	DC—20MHz	One cycle at a time
Hibernate/POR	LP,XT,HS	32kHzarrive20MHz	1024clock cycles (OST)
LFINTOSC (31kHz)	HFINTOSC	125kHzarrive8MHz	1us(approximation)

4.4.2 ECmodel

External clock (EC) mode allows externally generated logic levels to be used as the system clock source. When operating in this mode, the external clock source is connected to OSC1enter, OSC2Pins can be used as general purpose I/O. [picture4-4-2](#) Given EC mode pin connections.

When selecting EC mode, the oscillator start-up timer (OST) is disabled. Therefore, the Power-On Reset (POR) or after waking up from sleep mode. YSPRING MCU The design is completely static, stopping the external clock input will cause the device to stop working and keep all data intact. When the external clock is started again, the device resumes operation as if it had never been stopped.

picture4-4-2: External clock (EC) How the mode works



4.4.3 LP, XT and HS mode

LP, XT and HS mode supports connection to OSC1 and OSC2. The use of quartz crystal resonators or ceramic resonators ([picture4-4-3-1](#)) .mold The internal inverter-amplifier can be selected with a low, medium or high gain setting to support a variety of resonator types and speeds.

LP The oscillator mode selects the lowest gain setting of the internal inverter-amplifier. LP The current consumption of this mode is the smallest among the three modes. This mode is designed only for driving 32.768 kHz tuning fork (Tuning Fork) type crystal oscillator (watch crystal oscillator).

XT The oscillator mode selects a medium gain setting for the internal inverter-amplifier. XT The current consumption of this mode is in the middle of the three modes. This mode is best suited for driving resonators with medium drive level specifications.

HS The oscillator mode selects the highest gain setting for the internal inverter-amplifier. HS The current consumption of this mode is the highest among the three modes. This mode is best suited for driving resonators that require high drive settings.

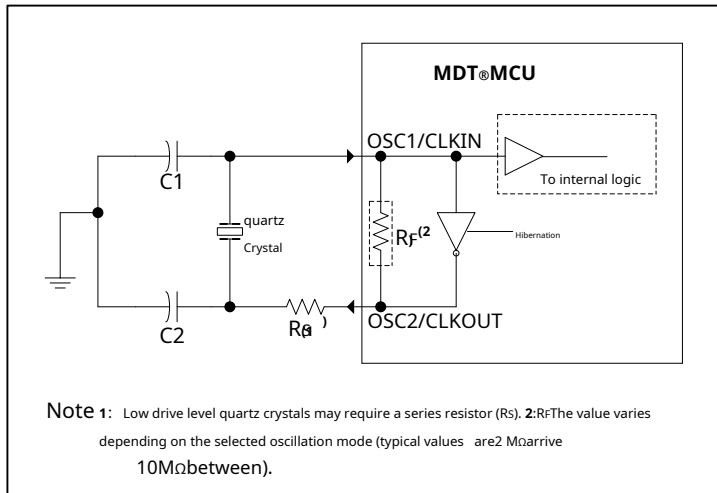
[picture4-4-3-1](#) and [picture4-4-3-2](#) Typical circuits for quartz crystal resonators and ceramic resonators are given respectively.

Note 1: The characteristics of quartz crystals vary with type, package, and manufacturer. For specifications and recommended applications, consult the manufacturer's data sheets.

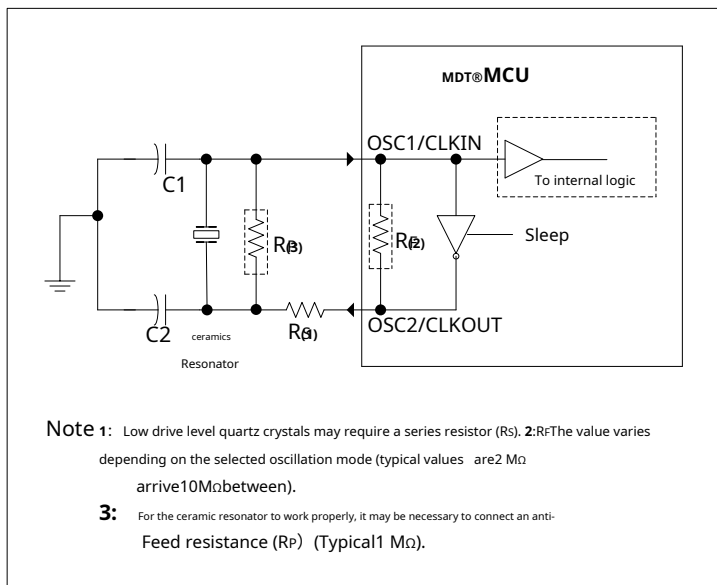
According to the manual.

2: Always verify that the oscillator operates within the intended application VDD and temperature range.

picture4-4-3-1: Working principle of quartz crystal (LP, XT or HS mode)



picture4-4-3-2: Working Principle of Ceramic Resonator (XTorHSmodel)



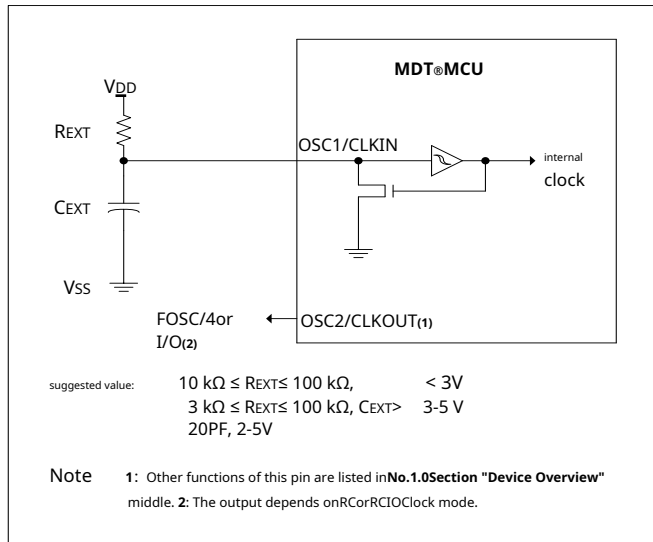
4.4.4 external RC model

External RC (RC) mode supports the use of external RC. When the clock accuracy is not high, this allows designers to have a wide range of frequency choices while keeping costs to a minimum. RC and RC IOT two modes.

exist RC mode, the circuit is connected to OSC1. RC OSC2/CLKOUT Output RC Oscillation frequency. This signal can be used to provide a clock for external circuits, synchronization, calibration, testing or other application requirements. [picture4-4-4](#) Given an external RC connection diagram of the mode.

picture4-4-4: external RC model





existRCIOMode,RCCircuit connected toOSC1.OSC2Become an extra generalI/OPin.I/OThe pin becomesPORTAofbit4 (PA4).
[picture4-5-5](#) GivenRCIOConnection diagram of the mode.

RCThe oscillator frequency is a function of the supply voltage, the resistance (R_{EXT}) and capacitance (C_{EXT}) value and the operating temperature. Other factors that affect the oscillator frequency are:

- Voltage threshold change
- Component tolerance
- Capacitors in different packages

Users should also consider the use of externalRCThe difference is caused by the tolerance of the components.

4.5 Internal clock mode

The oscillator module has two independent internal oscillators that can be configured or selected as system clock sources.

1. HFINTOSC(High Frequency Internal Oscillator) is factory calibrated and operates at a frequency of8MHz, the accuracy range is $\pm 1\%$.
2. LFINTOSC(Low Frequency Internal Oscillator) is not calibrated and operates at a frequency of31kHz.

Through softwareOSCONThe internal oscillator frequency selection bits of the registerIRCF<2:0>Operation allows you to select the system clock speed.

accessibleOSCONRegisters for system clock selection (SCS) bits to select the system clock between an external or internal clock source (see[No.4.6 Section "Clock Switching "](#)).

4.5.1 INTOSCandINTOSCIOmodel

When the Configuration Word register (CONFIG) using the oscillator selection bitsFOSC<2:0>When setting up the device,INTOSCandINTOSCIO In this mode, the internal oscillator is configured as the system clock source.

existINTOSCMODE,OSC1/CLKINCan be used as general purposeI/O.OSC2/CLKOUTOutput of the selected internal oscillator frequency4Frequency division. CLKOUTThe signal can be used to provide a clock for external circuitry, synchronization, calibration, test, or other application requirements.

existINTOSCIO,OSC1/CLKINandOSC2/CLKOUTPins can be used as general purposeI/O.

4.5.2 HFINTOSC

High frequency internal oscillator (HFINTOSC) is factory calibrated 8MHz internal clock source.

HFINTOSC The outputs are connected to the post divider and multiplexer (see [picture4-1](#)). Use OSCCON Register IRCF<2:0> Bit, One of seven frequencies can be selected via software. For more information, see [No.4.5.4Section "Frequency Select Bits \(IRCF\) "](#).

Will OSCCON Register IRCF<2:0> Bit set to #000 choose 8MHz arrive 125kHz Any frequency between HFINTOSC. Then OSCCON Register system clock source (SCS) Location 1, or by setting the configuration register (CONFIG) middle IESO Place 1 Enable Two-Speed Start-up.

OSCCON Register HF Internal Oscillator (HTS) bits are used to display HFINTOSC Whether it is stable.

4.5.3 LFINTOSC

Low frequency internal oscillator (LFINTOSC) is uncalibrated 31kHz internal clock source.

LFINTOSC The outputs are connected to the post divider and multiplexer (see [picture4-1](#)). Through software OSCCON Register IRCF<2:0> Operate on bits, select 31kHz For more information, see [No.4.5.4Section "Frequency Select Bits \(IRCF\) "](#). LFINTOSC or the power-on delay timer (PWRT)、Watchdog Timer (WDT) and a fail-safe clock monitor (FSCM)'s clock source.

Select 31kHz (Will OSCCON Register IRCF<2:0> Bit setting 000) is the system clock source (OSCCON Register bits SCS = 1), or when any of the following is enabled, LFINTOSC Will be enabled:

- Two-Speed Start-up (Configuration Word bit IESO = 1 and OSCCON Register bits IRCF = 000)
- Power-on delay timer (PWRT)
- Watchdog timer (WDT)
- Fail-Safe Clock Monitor (FSCM)

OSCCON Register LF Internal Oscillator (LTS) bit is used to indicate LFINTOSC Whether it is stable.

4.5.4 Frequency Select Bits (IRCF)

8MHz HFINTOSC and 31kHz LFINTOSC The outputs are connected to the post divider and multiplexer (see [picture4-1](#)). OSCCON The internal oscillator frequency selection bits of the register IRCF<2:0> Used to select the frequency output of the internal oscillator. The following can be selected by software: 8 One of the frequencies:

- 8MHz
- 4MHz (Default value after reset)
- 2MHz
- 1MHz
- 500kHz
- 250kHz
- 125kHz
- 31kHz

Note: After any reset, OSCCON Register IRCF<2:0> The bit will be set to 110 and the frequency selection is set to 4MHz. User editable IRCF bits to select other frequencies.

4.5.5 HFINTOSC and LFINTOSC Clock Switching Timing

When LFINTOSC and HFINTOSC are switching between the two oscillators, the new oscillator may be turned off to save power (see picture4-5-5). In this case, OSCCON Register IRCF<2:0> There is a delay after a bit is modified before the frequency selection takes effect. OSCCON Register LTS and HTS The bit will reflect LFINTOSC and HFINTOSC The current active state of the oscillator. The frequency selection sequence is as follows:

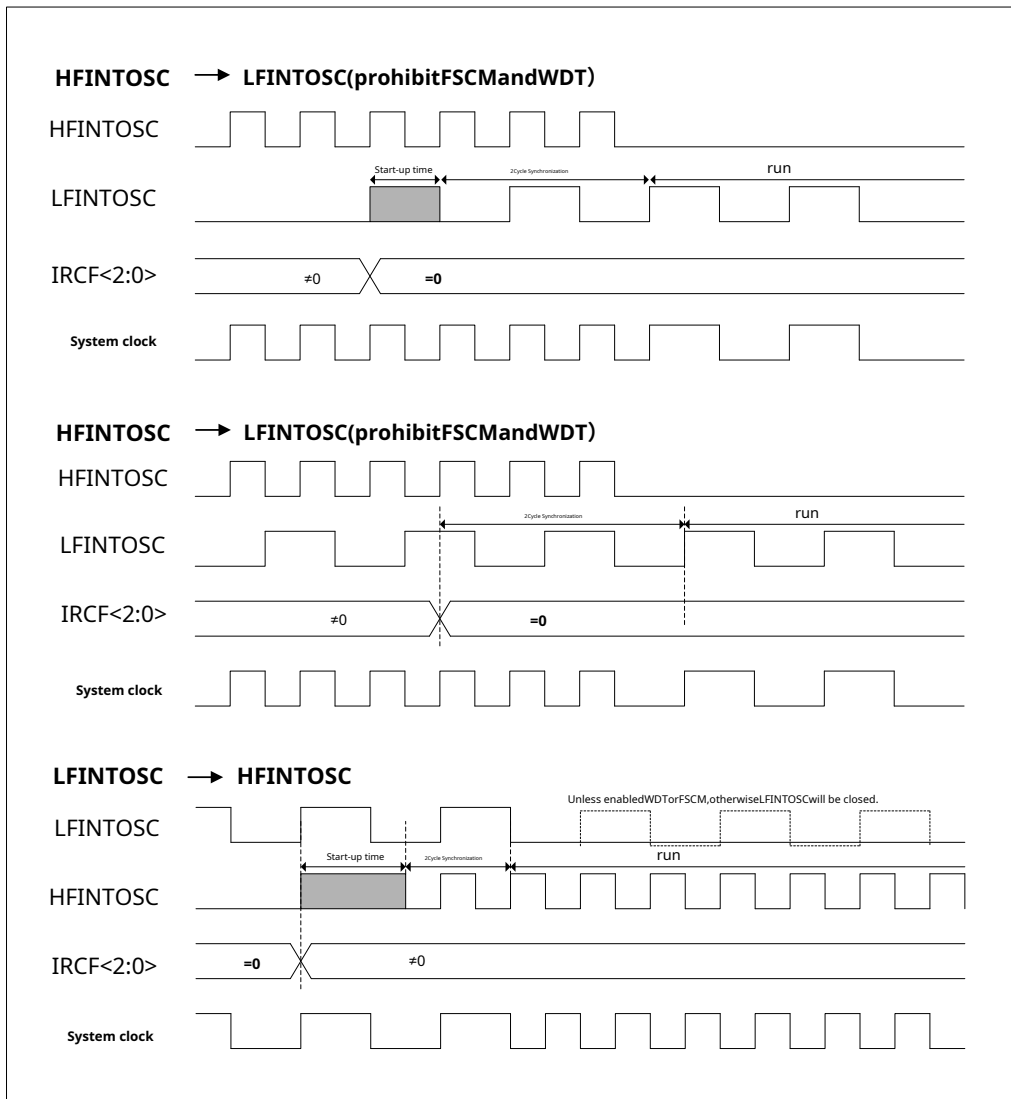
1. OSCCON Register IRCF<2:0> bit is modified.
2. If the new clock is off, a clock start delay begins.
3. The clock switching circuit waits for the falling edge of the current clock to arrive.
4. CLKOUT Keep it low, the clock switching circuit waits for the arrival of a new clock rising edge.
5. Now CLKOUT Connect to the new clock. OSCCON Register HTS and LTS The bits are updated as required.
6. Clock switch complete.

For more information, see picture4-1 .

If the internal oscillator speed is selected within 8MHz arrive 125kHz There is no startup delay when selecting a new frequency. This is because both the old and new frequencies come from the post divider and multiplexer. HFINTOSC.

Start delay specification in No.13.0 Section "Electrical Characteristics" In the table related to oscillators.

picture4-5-5: Internal oscillator switching timing



4.6 Clock Switching

Through software OSCCON Registers for system clock selection (SCS) bits can be operated to switch the system clock source between external and internal clock sources.

4.6.1 System clock selection (SCS) Bit

OSCCON Registers for system clock selection (SCS) bits are used to select CPU and peripherals system clock source.

- OSCCON Register bits SCS = 0 The system clock source is determined by the configuration word register (CONFIG) middle FOSC<2:0> Position Matching Set decision.
- OSCCON Register bits SCS = 1 When, according to OSCCON Register IRCF<2:0> The internal oscillator frequency selected by bit Take the system clock source. After reset, OSCCON Register SCS Always cleared.

Note: Any automatic clock switching (which may occur from Two-Speed Start-up or the Fail-Safe Clock Monitor) does not update OSCCON Register SCS Bit .
User can monitor OSCCON Register OST bits to determine the current system clock source.

4.6.2 Oscillator start-up timeout state (OSTS) Bit

OSCCON Oscillator Start-up Timeout Status Register (OSTS) bit is used to indicate whether the system clock comes from an external clock source or an internal clock source. The external clock source is determined by the configuration word register (CONFIG) of FOSC<2:0> definition. OSTS It also specifically states that LP, XT or HS mode, the oscillator start-up timer (OST) has timed out.

4.7 Dual-Speed Clock Startup Mode

Two-Speed Start-up mode provides additional power savings by minimizing the latency between external oscillator start-up and code execution. For applications that frequently use Sleep mode, Two-Speed Start-up mode reduces the overall power consumption of the device by removing the external oscillator start-up time after the device wakes up.

This mode allows the application to wake up from sleep. INTOSC It can be used as a clock source to execute a few instructions and then return to sleep without waiting for the main oscillator to stabilize.

Note: Execution SLEEP The instruction will stop the oscillator start-up time and make OSCCON Register OSTS Bit remains cleared.

When the oscillator module is configured as LP, XT or HS mode, the oscillator start-up timer (OST) Enable (see [No.4.4.1 Section "Oscillator Start-up Timer \(OST\)"](#)) OST Will pause program execution until it completes. 1024 Oscillation count. Dual-Speed Start-up mode OST Use internal The oscillator is used to operate the oscillator, minimizing the delay in code execution. OST Count to 1024 and OSCCON Register OSTS Location 1, program execution switches to the external oscillator.

4.7.1 Two-Speed Boot Mode Configuration

Configure Two-Speed Boot mode with the following settings:

Configuration Word Register (CONFIG) IESO = 1; Internal/external switch bit (enable two-speed startup mode) .

- OSCCON Register bits SCS = 0.

Configuration Word Register (CONFIG) middle FOSC<2:0> Configured as LP, XT or HS mode.

Two-Speed Startup Mode is entered after the following operations:

Power-on reset (POR) and the Power-On Timer (PWRT) after the delay ends (when enabled), or

Wake up from sleep state.

If the external clock oscillator is configured as LP, XT or HS mode, Two-Speed Start-up will be disabled. This is because POR The external clock oscillator does not require stabilization time after being powered on or when exiting Sleep.

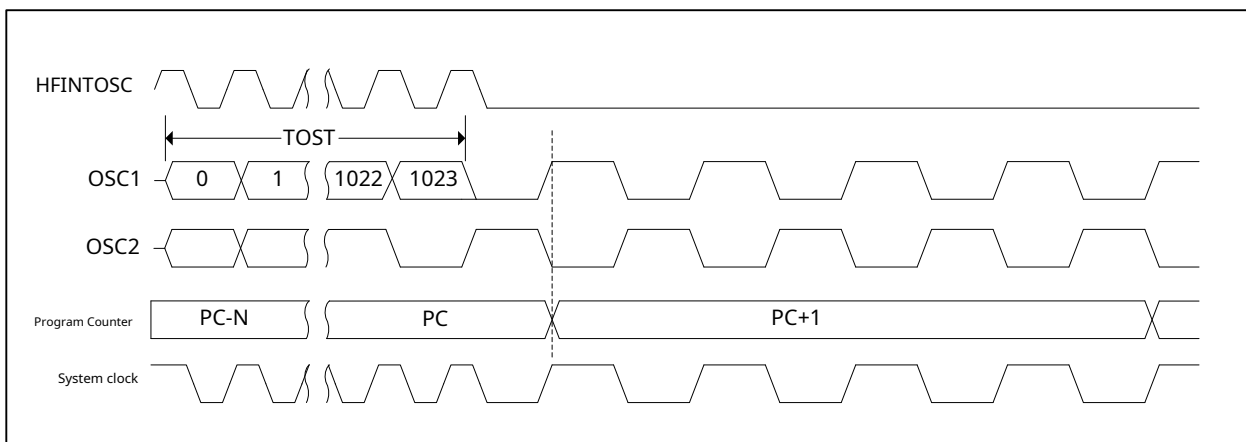
4.7.2 Two-Speed Start Sequence

1. Wake up from power-on reset or sleep mode.
2. Using the internal oscillator to OSCCON Register IRCF<2:0> The frequency set by the bit starts the execution of the instruction.
3. OST Enable, count 1024 clock cycles.
4. OST Timeout, waiting for the falling edge of the internal oscillator.
5. OST SPlace1.
6. The system clock remains low until the next falling edge of the new clock arrives (LP, XT or HS mode).
7. The system clock is switched to the external clock source.

4.7.3 Checking the dual-speed clock status

Passed inspection OSCCON Register OST The state of the bit can determine whether the microcontroller is configured as a word register (CONFIG) middle FOSC<2:0> The MCU can run from an external clock source as defined by the MCU bits, or from the internal oscillator.

picture4-7-3: Dual speed start



4.8 Power-down mode (sleep)

By executing SLEEP instruction to enter power-down mode.

If the watchdog timer is enabled:

- WDT will be cleared and keep running.

Status register PD The bit is cleared.

- TO Position is set 1.
- Turn off the oscillator driver.
- I/O Port Keep Execution SLEEP State before the instruction (driven to high level, low level or high impedance state)

To minimize current consumption in this mode, all I/O pins should be kept as VDD or VSS to ensure that no external circuitry from I/O pin consumes current, the comparator and CVREF to avoid the input pin being left floating and introducing switching current, a high impedance input

of I/O. The pin is pulled high or low. To minimize current consumption, the input should also remain VDD or VSS. It should also be considered PORTA. Effect of on-chip pull-up.

MCLR The pin must be logic high.

Note: Please note that due to WDTA reset caused by a timeout will not MCLR The pin is driven low.

4.8.1 MCLR Pin wakeup from sleep state

The device can wake up from Sleep by any of the following events:

- MCLR External reset input on pin.
- Watchdog timer wakeup (if WDT Enable).
- PA2/INT Pin interrupt, PORTA Interrupt-on-change or peripheral interrupt.

The first event causes a device reset. The second two events are considered to be a continuation of program execution. TO and PD bit is used to determine the SLEEP bit is set at power-on. 1, PD bit is cleared to zero when the command is executed. TO Location is happening WDT Cleared to zero on wakeup.

The following peripheral interrupts can wake the device from Sleep:

1. TMR1 Interrupt. Timer1 Must be used as an asynchronous counter.
2. ECCP Capture mode interrupt.
3. A/D Conversion (when A/D The clock source is F_{sm}hour).
4. EEPROM The write operation is complete.
5. The comparator output state changes.
6. Interrupt on level change.
7. From INT External interrupt pin.

Since no on-chip clocks are active during sleep, other peripherals cannot generate interrupts.

When executing SLEEP instruction, the next instruction (PC+1) If you want to wake up the device through an interrupt event, you must set the corresponding interrupt enable bit (Enabled). Wake-up occurs with GI The state of the bit is irrelevant. If GI bit is cleared (disabled), device will continue to execute SLEEP instruction after the instruction. If GI bit is set (Enabled), the device executes SLEEP instruction after the instruction, then jump to the interrupt location site (0004h) to execute code. If you do not want to execute SLEEP instructions after the instruction, the user should SLEEP place a line after the instruction NOP instruction.

Note: If global interrupts are disabled (GI bit is cleared), but any interrupt source sets its interrupt enable bit and the corresponding interrupt flag position 1, device will wake up immediately from sleep.

When the device wakes up from sleep mode, WDT will be cleared regardless of the wakeup reason.

4.8.2 Using interrupt wakeup

When global interrupts are disabled (GI bit is cleared), and any interrupt source sets its interrupt enable bit and interrupt flag bit 1, one of the following events will occur:

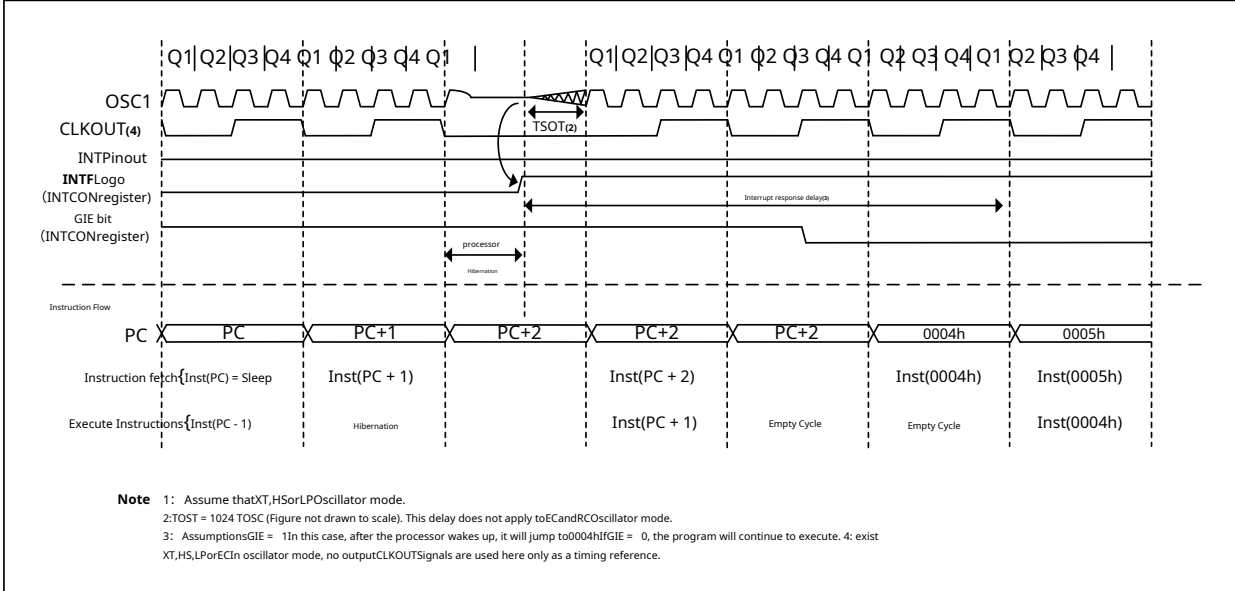
If you are executing SLEEP If an interrupt occurred before, SLEEP instruction will be treated as a NOP instruction is executed. Therefore, WDT and its prescaler and postscaler (if enabled) will not be cleared, and TO The bit will not be set 1, at the same time PD The bit will not be cleared.

If you are executing SLEEP If an interrupt occurs during or after an instruction, the device will immediately wake up from Sleep. SLEEP instruction will be executed before waking up. Therefore, WDT and its prescaler and postscaler (if enabled) will be cleared, and TO The bit will be set 1, at the same time PD The bit will also be cleared.

Even when executing SLEEP Before the instruction, check that the flag is 0, it may also be SLEEP Set before the instruction is executed. To determine whether the SLEEP Instructions, testable PDIF PD Location 1, then it means SLEEP The instruction is treated as a NOP The instruction was executed.

In execution SLEEP Before the instruction, you must first execute a CLR W T Instructions to ensure that W D T For details, see [picture4-8](#).

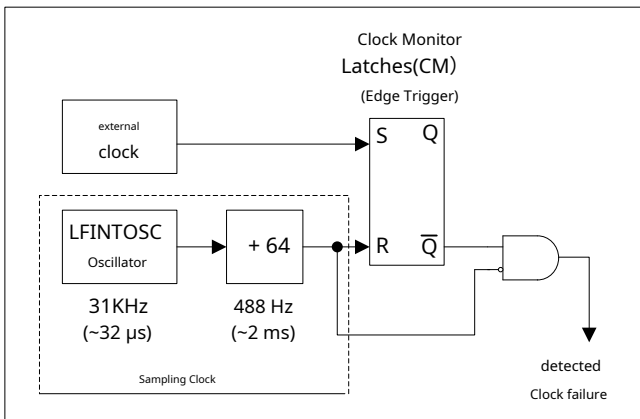
picture4-8: Wake up the device from sleep mode through interrupt



4.9 Fail-Safe Clock Monitor

Fail-Safe Clock Monitor (FSCM) allows the device to continue operating in the event of an external oscillator failure. FSCM can be used during oscillator start-up delay timer (OST) at any time after the expiration of the oscillator. FSCM by setting the Configuration Word register (CONFIG) middle FCMEN Location 1 To enable. FSCM available for all external oscillator modes (LP, XT, HS, EC, RC and RCIO).

picture4-9: FSCM block diagram



4.9.1 Fault protection detection

FSCM The module connects an external oscillator to FSCM Sampling clock comparison to detect oscillator failure. LFINTOSC Divide by 64, the sampling clock is generated. [picture4-9](#) The fault detector has a latch inside. At each falling edge of the external clock, the latch is set. The latch is cleared on each rising edge of the sampling clock. If a full half cycle of the sampling clock elapses without the master clock going low, a fault is detected.

4.9.2 Fail-safe operation

When the external clock fails, FSCM switches the device clock to the internal clock source and PIR2 Register OSFIF Location 1. At the same time, the flag position 1, an interrupt will be generated. The device firmware can then take steps to mitigate the problems that may have been caused by the failed clock. The system clock will continue to be derived from the internal clock source until the device firmware successfully restarts the external oscillator and switches back to external operation.

FSCM The internal clock source is selected by OSCCON Register IRCF<2:0>. This allows the internal oscillator to be configured before a malfunction occurs.

4.9.3 Failsafe condition cleared

Reset, Execute SLEEP Instruction or flip OSCCON Register SCSA. After the Fault Protection condition is cleared, OSCCON Register SCSA. After the bit is modified, OST will restart. OST When running, the device continues to OSCCON Selected INTOSC to perform the operation. OST After the timeout, the Fail-Safe condition is cleared and the device will operate from the external clock source. The Fail-Safe condition must be cleared before the OSFIF Flag bit.

4.9.4 Reset or wake up from sleep

FSCM Designed to be able to start the oscillator during the delay timer (OST) at any time after the expiration of the oscillator. OST The use cases are after waking up from sleep state and after any type of reset. OST Can't be used, so once reset or wake-up is completed, FSCM It is in activated state. FSCM is enabled, when Two-Speed Start-up is also enabled. OST When running, the device is always in the code execution phase.

Note: Due to the wide range of oscillator start-up times, during oscillator start-up (i.e., when exiting from Reset or Sleep), Fault protection circuit After an appropriate period of time, the user should check OSCCON Register OST bits to verify that the oscillator has started successfully and the system clock has switched successfully.

5.0 Interrupt

MDT10F684 There are several interrupt sources:

- External Interrupt PA2/INT
- Timer0 Overflow interrupt
- PORTA Interrupt on Change

Two comparator interrupts

- A/D Interrupt
- Timer1 Overflow interrupt
- Timer2 Match Interrupt
- EEPROM Data write interrupt

Fail-safe clock monitor interrupt

Enhanced CCP Interrupt

Interrupt Control Register (INTS) and peripheral interrupt request registers 1 (PIFB1) Record various interrupt requests in their respective flag bits. INTS The register also includes individual interrupt enable bits and a global interrupt enable bit.

INTS The global interrupt enable bit in the register GIE In place 1 When cleared, all unmasked interrupts are enabled, and when cleared, all interrupts are disabled.

passINTS and PIEB1. The corresponding enable bits in the register can be used to disable each interrupt. GIE is cleared.

In response to an interrupt, the following actions occur automatically:

- GIE cleared to disable any other interrupts.
- The return address is pushed onto the stack.
- PC is loaded with 0004h.

Execute the "return from interrupt" instruction (RTF) to exit the interrupt routine and GIE, thereby re-enabling unmasked interrupts.

INTS: The register contains the following interrupt flag bits:

- INT Pin Interrupt
- PORTA Interrupt on Change
- Timer0 Overflow interrupt

The peripheral interrupt flag is in PIFB1. The corresponding interrupt enable bit is in the PIEB1 register.

PIFB1: The register contains the following interrupt flag bits:

- EEPROM Data write interrupt
- A/D Interrupt
- 2 Comparator interrupt
- Timer1 Overflow interrupt
- Timer2 Match Interrupt
- Fail-Safe Clock Monitor Interrupt
- Enhanced CCP Interrupt

For external interrupt events, such as INT Pin or PORTA Level change interrupt, the interrupt response delay will be 3 μ s. The exact delay time depends on when the interrupt event occurs (see [picture5-1-1](#)). The interrupt response delay is exactly the same for single-cycle or two-cycle instructions.

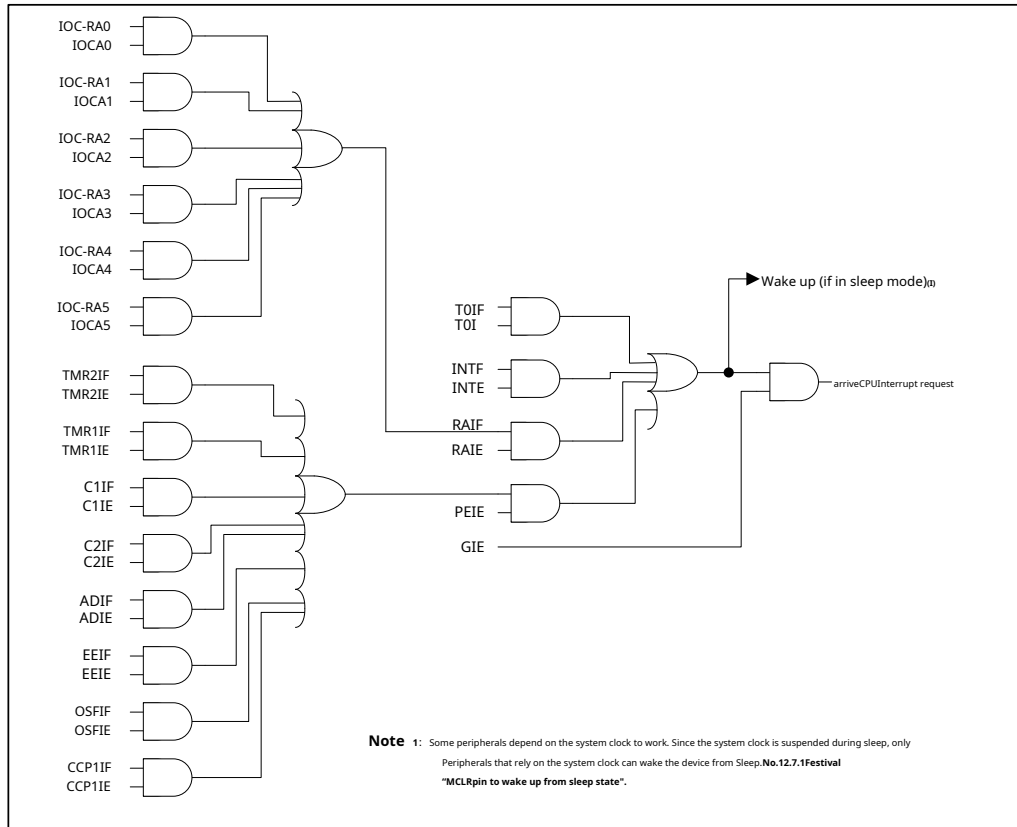
After the interrupt service routine, the interrupt source can be determined by querying the interrupt flag bit. Before re-enabling the interrupt, the interrupt flag bit must be cleared by software to avoid repeated response to the interrupt.

Note 1: Setting of each interrupt flag is not affected by the corresponding interrupt mask bit or GIE. The effect of bit state.

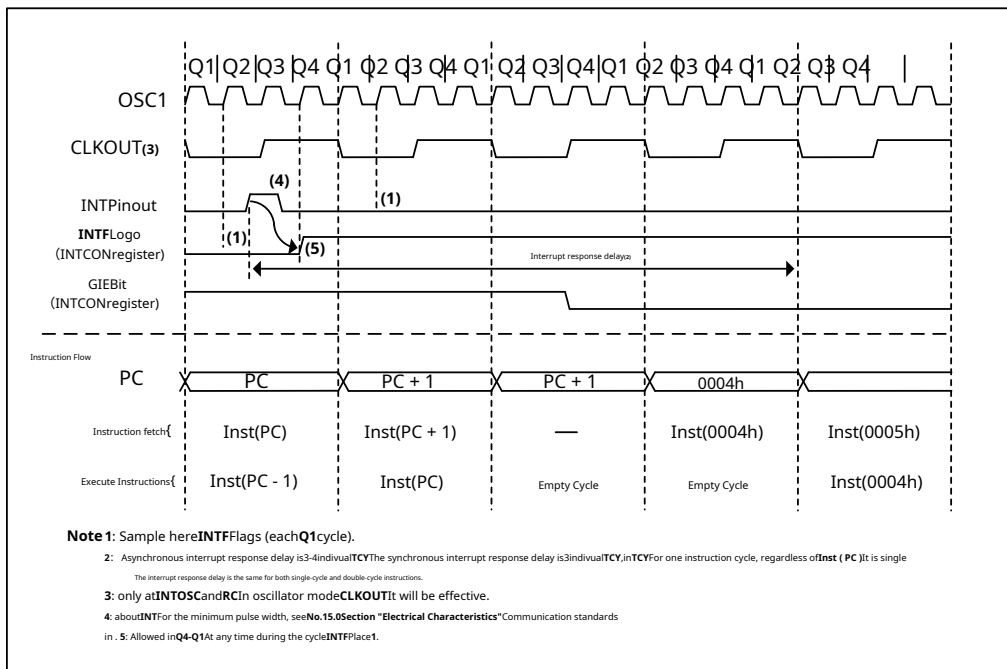
- 2: When executing a clear GIE instruction, any interrupt waiting to be executed in the next cycle will be ignored. The bit is reset. After that, the ignored interrupt will continue to wait for processing.

For more information on a module, see the corresponding peripheral chapter.

picture5-1-1: Interrupt logic



picture5-1-2:INTPin Interrupt Timing



5.1 Interrupt Special Function Register



5.1.1 INTSregister

INTSRegisters are readable and writable registers, containingRTCCRegister overflow,PORTALevel changes and externalPA2/INTVarious enable and flag bits for pin interrupts.

Note: When an interrupt condition occurs, regardless of the corresponding interrupt enable bit orINTSGlobal enable bit of registerGIERegardless of the status, the interrupt flag will be set1
User software should ensure that the corresponding interrupt flag bits are cleared before enabling interrupts.

register0BH/8BH: Interrupt Control Register (INTS)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GIE	PEIE	TIS	INS	PAIE	TIF	INTF	PAIF
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
-n = PORTThe value of 1 =Place1 0 =Clear x =unknown

- bit7** **GIE:** Global interrupt enable bit
1 =Enable all unmasked interrupts
0 =Disable all interrupts
- bit6** **PEIE:** Peripheral interrupt enable bit
1 =Enable all unmasked peripheral interrupts
0 =Disable all peripheral interrupts
- bit5** **TIS:**Timer0Overflow interrupt enable bit
1 =allowTimer0Interrupt 0 =prohibit
Timer0Interrupt
- bit4** **INTS:PA2/INTE**External interrupt enable bit 1
=allowPA2/INTEExternal Interrupt 0 =prohibit
PA2/INTEExternal Interrupt
- bit3** **PAIE:**PORTAInterrupt-on-change enable bit⁽¹⁾
1 =allowPORTAInterrupt on Change 0 =
prohibitPORTAInterrupt on Change
- bit2** **TIF:**Timer0Overflow interrupt flag⁽²⁾
1 = Timer0Register has overflowed (must be cleared by software) 0
= Timer0Registers did not overflow
- bit1** **INTF:**PA2/INTEExternal interrupt flag
1 =happenedPA2/INTEExternal interrupt (must be cleared by software) 0 =
Yet to happenPA2/INTEExternal Interrupt
- bit0** **PAIF:**PORTALevel change interrupt flag⁽¹⁾
1 =at least onePORTA<5:0>The pin level status has changed (must be cleared by
software) 0 =nonePORTA<5:0>The pin level status changes

Note 1: Must be enabled at the same timePAINTRregister.

2:whenRTCCWhen the counter returns to zero,TIFLocation1. When resetRTCCThe status remains unchanged, it should be clearedTIFbits were previously initialized.

5.1.2 PIEB1register

PIEB1Register containing the peripheral interrupt enable bits.

Note: To enable any peripheral interrupt, you mustINTSRegisterPEIELocation1.

register8CH: Peripheral interrupt enable register1 (PIEB1)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE
bit7						bit0	

Figure caption:

R =Readable bit

W =Writable bit

U =Unimplemented bit, read as0

-n = PORThe value of

1 =Place1

0 =Clear

x =unknown

bit7	EEIE :EEPROMWrite completion interrupt enable bit 1 =allowEEWrite completion interrupt 0 =prohibitEEWrite completion interrupt
bit6	ADIE :A/DConverter Interrupt Enable Bit 1 =allowA/DConverter Interrupt 0 = prohibitA/DConverter Interrupt
bit5	CCP1IE :CCP1Interrupt Enable Bit 1 =allowCCP1 Interrupt 0 =prohibitCCP1Interrupt
bit4	C2IE :Comparators2Interrupt Enable Bit 1 =Enable comparator2Interrupt 0 =Disable comparator2Interrupt
bit3	C1IE :Comparators1Interrupt Enable Bit 1 =Enable comparator1Interrupt 0 =Disable comparator1Interrupt
bit2	OSFIE : Oscillator Fault Interrupt Enable Bit 1 =Enable oscillator fault interrupt 0 =Disable oscillator fault interrupt
bit1	TMR2IE :Timer2andPR2Match Interrupt Enable Bit 1 = allowTimer2andPR2Match Interrupt 0 =prohibitTimer2 andPR2Match Interrupt
bit0	TMR1IE :Timer1Overflow interrupt enable bit 1 =allowTimer1Overflow interrupt 0 =prohibitTimer1Overflow interrupt

5.1.3 PIFB1register

PIFB1Register containing the peripheral interrupt flags.

Note: When an interrupt condition occurs, regardless of the corresponding interrupt enable bit orINTSGlobal enable bit of registerGIERegardless of the status, the interrupt flag will be set1
User software should ensure that the corresponding interrupt flag bits are cleared before enabling interrupts.

5.2 PA2/INTInterrupt

PA2/INTThe external interrupt on the pin is edge triggered; whenTMRRegisterIESPosition is set1When the rising edge is triggered,IESTriggered on falling edge when the bit is cleared.PA2/INTWhen a valid edge occurs on the pin,INTSRegisterINTFLocation1. This can be done byINTS RegisterINTEBefore re-enabling the interrupt, the interrupt service routine must first clear the control bit to 0.INTFThe bit is cleared to 0.INTEThe bit is set before entering the sleep state.1,butPA2/INTAn interrupt can wake up the processor from a sleep state.PA2/INTFor the timing of interrupts waking the processor from sleep mode, see[picture4-8](#).

Note: MustANSELandCMSTA0The registers are initialized to configure the analog channels as digital inputs. Pins configured as analog inputs always read as0.

5.3 Timer0Interrupt

RTCCRegister overflow (FWf→00h) willINTSRegisterTOIFLocation1. You can set1/ClearINTSRegisterTOI bit to enable/disable the interrupt.Timer0For module operation, see[No.7.1Festival"Timer0Modules "](#).

5.4 PORTAInterrupt on Change

PORTAChanges in input level will causeINTSRegisterPAIFLocation1. You can set1/ClearINTSRegisterPAIEbit to enable/disable the interrupt.PAINTRThe registers configure the individual pins of the port.

Note: This occurs while a read operation is being performed.I/OIf the pin level changes,PAIFThe interrupt flag may not be set1.

5.5 EEPROMWrite data interrupt

To ensureEEPROMTo generate an interrupt at the end of a write operation, you must set the following location1:

- PIEB1RegisterEEPROMWrite completion interrupt enable bitEEIE
- INTSRegisterPEIEBit
- INTSRegisterGIEBit

In the interrupt service routine,PIFB1RegisterEEIFClearing the bit to 0 will clear the interrupt flag.

5.6 A/DInterrupt

ADCWhen the module completes the analog-to-digital conversion,PIR1RegisterADIFInterrupt flags are set1. You can set1/ClearPIEB1 RegisterADIEbit to enable/disable the interrupt.ADCFor interrupt operation of the module, see[No.9.2.5Festival"ADCModule interrupt "](#).

5.7 Interrupted context protection

During the interruption, only the returned PC often, users may wish to save critical registers during an interrupt (for example, W Registers and Status Registers). This must be done in software. The temporary save register should be W_TEMP and STATUS_TEMP. Place GPR The end 16 Byte (see [picture 2-2](#)). This 16 Each unit is shared by all storage areas and does not require partitioning. This simplifies scene protection and recovery operations. 12-1 The code shown in can be used to:

Save W register

Save status register

· implement ISR Code

· Restore the status register (and bank select bit register)

· recover W register

Note: MDT10F684 Usually no need to save PCHLAT. However, if you want to ISR and the main program using calculations GOTO, it is necessary to ISR Save and restore PCHLAT.

example 5-7: The status register and W Registers are stored in RAM middle

```

STWR    W_TEMP                ;save W Register to W_TEMP
SWAPR   STATUS,W              ;Will STATUS The high and low four bits of the register are swapped and saved to W (Purpose: Do not
STWR    STATUS_TEMP           change the logo) ;save STATUS Register contents to STATUS_TEMP;

;(ISR)                          ;Interrupt Service Routine
;
SWAPR   STATUS_TEMP,W          ;exchange STATUS_TEMP High and low four bytes ;
STWR    STATUS                 Restore Flag Status Register
SWAPR   W_TEMP,R              ;recover W Register value (Purpose of using the exchange instruction: do not change the status register value)
SWAPR   W_TEMP,W

```

6.0 I/O port

This series of devices has 14 Universal I/O. Depending on the peripherals enabled, some (or all) of the pins cannot be used for general purpose I/O. Usually, after a peripheral is enabled, the associated pin cannot be used for general purpose I/O. Pin it out.

6.1 PORTA and CPIOA register

PORTA Yes 6 bit-wide bidirectional port. PORTA The corresponding data direction register is CPIOA ([register 85H](#)). Will CPIOA Location 1 (=1) can make the corresponding PORTA pin as an input pin (i.e. disable the corresponding output driver). CPIOA Bit clear (=0) will make the corresponding PORTA pin as an output pin (i.e., enables the output driver and places the contents of the output latch on the selected pin). PA3 is an exception, it can only be used as an input pin, CPIOA Bit is always read as 1. [example 6-1](#) Given an initialization PORTA Methods.

read PORTA register ([register 05H](#)) will read the pin state and writing to this register will write to the port latch. All write operations are read-modify-write operations. Therefore, writing to a port means reading the pin level of the port, modifying the read value, and then writing the modified value to the port data latch. MCLR = 1 hour, PA3 Read as 0.

Even though PORTA When the pin is used as an analog input, CPIOA Registers still control PORTA When using them as analog inputs, the user must ensure that CPIOA The bits in the register remain set 1 status. I/O The pin always reads 0.

Note: Must ANSEL and CMSTAT The registers are initialized to configure the analog channels as digital input channels. Pins configured as analog inputs read 0.

example6-1:initializationPORTA

```

BCR STATUS,PAGE ;Bank0
CLRR PORTA ; Clear portAOutput Buffer
LDWI 07H ;
STWR CMSTA ; Turn off the comparator
BSR STATUS,PAGE ;Bank1
CLRR ANSEL ;All analog/digital multiplexed ports are set as digital ports
LDWI 0CH ; Set portA<3,2>bits are input, the others are output ;
STWR CPIOA
BCR STATUS,PAGE ;Bank0
    
```

register05H:PORTAregister(PORTA)

U-0	U-0	R/W	R/W-0	Rx	R/W-0	R/W-0	R/W-0
—	—	PA5	PA4	PA3	PA2	PA1	PA0
bit7							bit0

Figure caption:
R =Readable bit W =Writable bit U =Unimplemented bit, read as0
-n = PORTThe value of 1 =Place1 0 =Clear x =unknown

bit7-6 Unimplemented: Read as0

bit5-0 **PA<5:0>**:PORTA I/OPin Position
1 = PORTAPin Level>V_{IH}
0 = PORTAPin level<V_{IL}

register85H:PORTATri-state registers (CPIOA)

U-0	U-0	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
—	—	CPIOA5	CPIOA4	CPIOA3	CPIOA2	CPIOA1	CPIOA0
bit7							bit0

Figure caption:
R =Readable bit W =Writable bit U =Unimplemented bit, read as0
-n = PORTThe value of 1 =Place1 0 =Clear x =unknown

bit7-6 Unimplemented: Read as0

bit5-0 **CPIOA<5:0>**:PORTATri-state control bit 1 =
PORTAPin configured as input (tri-state) 0 =
PORTAPins configured as output

Note 1:CPIOA<3>Always read as1.
2:existXT,HSandLPIn oscillation mode,CPIOA<5:4>Always read as1.



6.2 PORTC and CPIOC Register

PORTC by 8A universal bidirectional pin. The pin can be configured as I/O or as A/D converter (ADC) or the analog input of a comparator. For specific information on each function (such as CCP or ADC), see the corresponding section of this data sheet.

Note: Must ANSEL and CMSTAT. The registers are initialized to configure the analog channels as digital input channels. Pins configured as analog inputs read 0.

example 6-4: initialization PORTC

```
BCR    STATUS,PAGE    ;Bank 0
CLRR   PORTC          ; Clear output buffer
LDWI   07h
STWR   CMSTA          ; Turn off the comparator
BSR    STATUS,PAGE    ;Bank 1
CLRR   ANSEL          ; Set the port to a digital port
LDWI   0Ch            ; set up PC1, PC2 for input, other PC output
STWR   CPIOC
```

register 07H: PORTC register (PORTC)

R/W	R/W	R/W	R/W	R/W-0	R/W-0	R/W-0	R/W-0
PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
bit7							bit0

Figure caption:

R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORTC value of 1 = Place 1 0 = Clear x = unknown

bit7-0 **PC<7:0>**: PORTC I/O pin
 Position 1 = PORTC pin level >
 VIH 0 = PORTC pin level < VIL

register 87H: PORTC tri-state registers (CPIOC)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
CPIOC7	CPIOC6	CPIOC5	CPIOC4	CPIOC3	CPIOC2	CPIOC1	CPIOC0
bit7							bit0

Figure caption:

R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORTC value of 1 = Place 1 0 = Clear x = unknown

bit7-0 **CPIOC<7:0>**: PORTC tri-state control bit 1 =
 PORTC pin configured as input (tri-state) 0 =
 PORTC pins configured as output

6.3 Other pin functions

MDT10F684 Every one of PORTA The pins have level change interrupt and weak pull-up function (PA3 When used as a reset pin, enable pull-up. IO port, there is no pull-up resistor). PA0 It has ultra-low power wake-up functions. The following three subsections will introduce these functions.

6.3.1 ANSEL register

ANSEL Registers are used to I/O The input mode of the pin is configured as analog. ANSEL Setting the pin high will cause all read operations on this pin to result in 0 and enable the analog function of the pin to operate normally.

ANSEL The state of the bit has no effect on the digital output function. CPIO Clear and ANSEL Place 1 The pins will still function as digital outputs, but their output mode will be analog. This can result in unexpected operation when executing read-modify-write instructions on the affected ports.

register 91H: Analog Select Register (ADINS)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0
bit7							bit0

Figure caption:

R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORTThe value of 1 = Place 1 0 = Clear x = unknown

bit7-0 **ANS<7:0>**: Analog selection bit
 existAN<7:0> Analog selection of analog or digital functions on the pins respectively. 1 = Analog Input. Pins are assigned as analog input_n. 0 = number I/O Pins are assigned to ports or special functions.

Note 1: Setting a pin as an analog input automatically disables the digital input circuitry, weak pull-up, and interrupt-on-change (if any).
 corresponding CPIO The OUTPUT_MODE bit must be set to input mode to allow external control of the voltage on this pin.

6.3.2 Weak pull-up

Every PORTA Pins (except PA3) have their own configurable internal weak pull-ups. Control bits WPUA Enables or disables each weak pull-up. See [register 95H](#) When a port pin is configured as an output, its weak pull-up circuit is automatically cut off. PAPU Bit (TMR<7>) disable the pull-up function. PA3 for I/O And configured as MCLR When it is disabled, it starts automatically PA3 Weak pull-up function. MCLR The pull-ups are not under software control.

register 95H: Weak pull-up PORTA register (PAPHR)

U-1	U-1	R/W-1	R/W-1	U-1	R/W-1	R/W-1	R/W-1
—	—	PHA5	PHA4	—	PHA2	PHA1	PHA0
bit7							bit0

Figure caption:

R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORTThe value of 1 = Place 1 0 = Clear x = unknown

bit7-6 Unimplemented: Read as 0

bit5-4 **PHA<5:4>**: Weak pull-up control bit
 1 =Enable pull-up
 0 =Disable pull-up

bit3 Unimplemented: Read as0

bit2-0 **PHA<2:0>**: Weak pull-up control bit
 1 =Enable pull-up
 0 =Disable pull-up

Note 1: Must be enabledTMRGlobal registersPAPHbits to enable individual pull-ups. 2: If the pin is in output mode (CPIOC = 0) , the weak pull-up device is automatically disabled. 3: Configured in the configuration word register asMCLRhourPA3The pull-up is enabled and configured asI/OThe pull-up is disabled. 4:existXT,HSandLPIn oscillation modePHA<5:4>Always read as1.

6.3.3Interrupt on Change

EveryPORTAEach pin can be configured as an interrupt-on-change pin.PAINTR_xEnable or disable the interrupt function for each pin.[register85H](#) . Disable interrupt-on-change on power-on reset.

For pins that have enabled interrupt-on-change, the value on the pin is the same as the last read.PORTAThe last “mismatch” is compared with the latched value. The outputs are logically ORed togetherIn order toINTSregister([register0BH](#))middlePORTALevel change interrupt flag (PAIF) Place1.

This interrupt can wake up the device in sleep mode. The user can clear the interrupt in the interrupt service routine by:

- a)rightPORTAPerform a read or write operation. This ends the pin level mismatch condition.
- b)Set the flagPAIFCleared.

The level mismatch condition will continue toPAIFLogo location1. And readPORTAwill end the mismatch condition and allowPAIFThe flag bit is cleared.

The latch will hold the last value read regardless ofMCLRand brown-out resets. After these resets, if a level mismatch occurs,PAIF The flag will continue to be set1.

Note: Before executing anyPORTAIf the operationI/OIf the pin level changes,PAIFThe interrupt flag may not be set1.

register96H: Interrupt on level changePORTAregister(PAINTR)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	PAINTR5	PAINTR4	PAINTR3	PAINTR2	PAINTR1	PAINTR0
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORTThe value of 1 =Place1 0 =Clear x =unknown

bit7-6 Unimplemented: Read as0.

bit5-0 **PAINTR<5:0>**: Interrupt on level changePORT AControl bit 1
 =Enable interrupt-on-change
 0 =Disable interrupt on change

Note 1: Global interrupt enable must be enabled (GIE) so that each interrupt can be recognized. 2:existXT,HSandLPIn oscillation modePAINTR<5:4>Always read as1.

6.3.4 Ultra-low power wake-up

The ultra-low power wake-up on the MCU allows the voltage to fall slowly, thus generating a PA0 Interrupt on Change. ULPWUE Bit (PSTA<5>1) This will generate a small current sink that can be used to PA0 The capacitor on the .

To use this feature, PA0 The pin should first be configured as a high output to charge the capacitor, allowing PA0 The level change interrupt and PA0 To configure the pin as an input. ULPWUE Place 1 Start discharging and execute SLEEP Instructions. PA0 The voltage on VIL After that, the device will wake up and execute the next instruction. INTS Register GIE Location 1, the device will call the interrupt service routine (0004h) For more information, please See also [No.6.3.3 Section "Interrupt on Change"](#) and [No.5.4 Section "PORT A Interrupt on Change"](#) .

This feature provides a low-power technique that can periodically wake up a device from sleep mode. The delay depends on PA0 superior RCTo learn how to initialize the ultra-low power wake-up module, see [example6-3-4](#) .

The series resistor provides PA0 The overcurrent protection function of the pin allows the delay to be calibrated in software (see [picture6-3-1](#)). A timer can be used to measure the charge and discharge time of the capacitor. The charge time is then adjusted to provide the desired interrupt delay. This technique compensates for the effects of temperature, voltage, and component accuracy. The ultra-low power wake-up peripheral can also be configured as a simple programmable low voltage detection device or temperature sensor.

example6-3-4: Initialization of ultra-low power wake-up

```

BCR    STATUS,PAGE    ;Bank 0
BSR    PORTA,0        ;clear 0 port A of PA0
LDWI   07H
STWR   CMSTA         ; Turn off the comparator
BSF    STATUS,PAGE    ;Bank 1
BCR    ANSEL,0        ;port A of PA0 Set to digital port;
BCR    CPIOA,0        port A of PA0 Set as output ;
LCALL  CapDelay       Delay
BSR    PSTA,ULPWUE    ; Enable ultra low power wake-up function
BSR    PAINTR,0       ;set up PA0 Port level change wake-up function
BSR    CPIOA,0        ;set up PA0 Input
LDWI   B'10001000'    ; Enable level change interrupt, global interrupt
STWR   INTS           ;
SLEEP                                     ; Sleep, wait to wake up
NOP
NOP

```

6.4 PORTA Pin Description and Pin Diagram

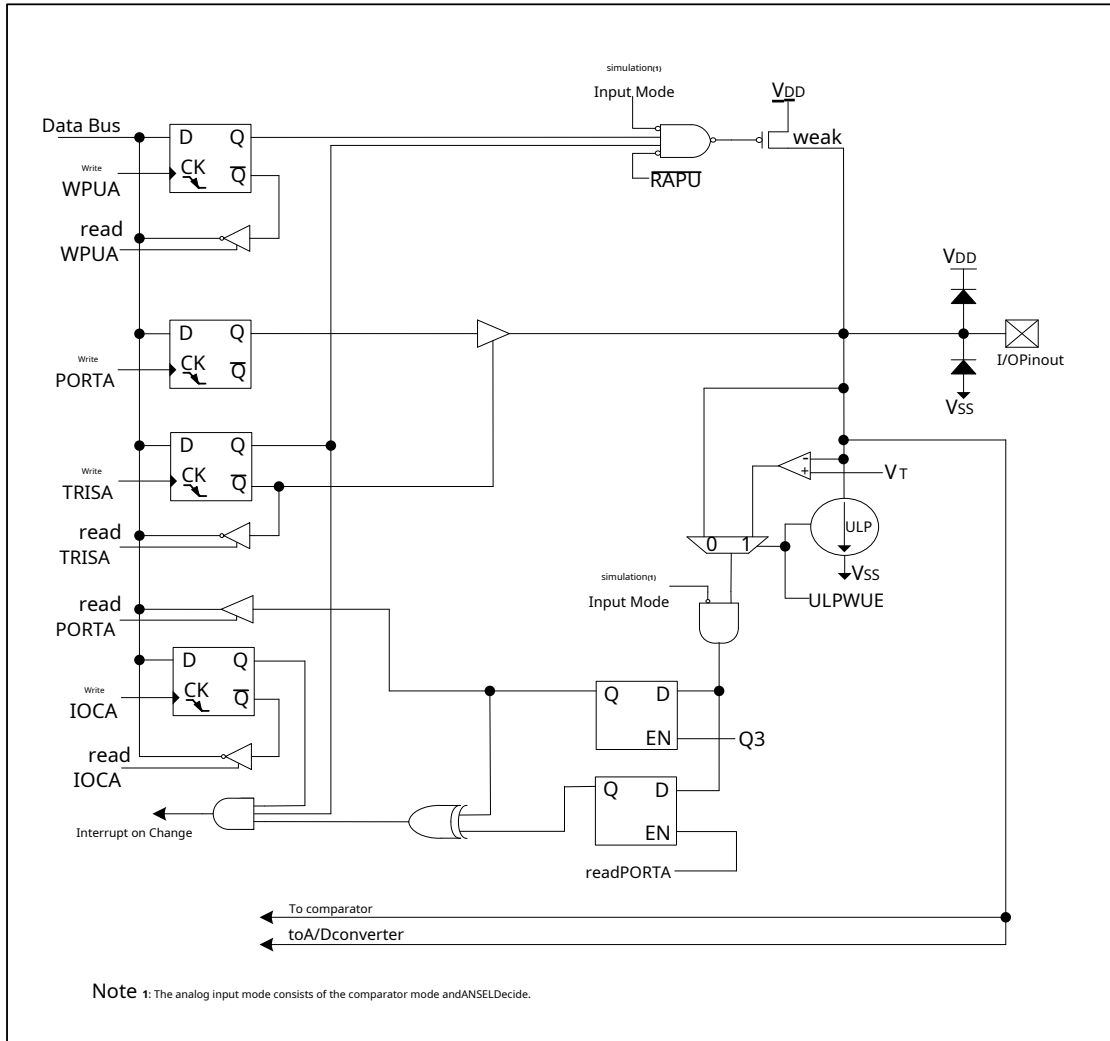
Each PORTA pins are multiplexed with other functions. Here we briefly describe the pins and their multiplexed functions. A/D converter (A/D Converter, ADC), please refer to the relevant chapters in this data sheet for specific information.

6.4.1 PA0/AN0/C1IN+/SCK/ULPWU

picture6-4-1 The pinout diagram for this pin is given. PA0/AN0/C1IN+/SCK/ULPWU The pin can be configured to function as one of the following:

- General I/O
- Connect to ADC Analog input
- Connect to the analog input of the comparator
- In-Circuit Serial Programming (In-CircuitSerialProgramming™) programming and debugging clock SCK
- Analog input for ultra-low power wake-up

picture6-4-1: PA0 block diagram

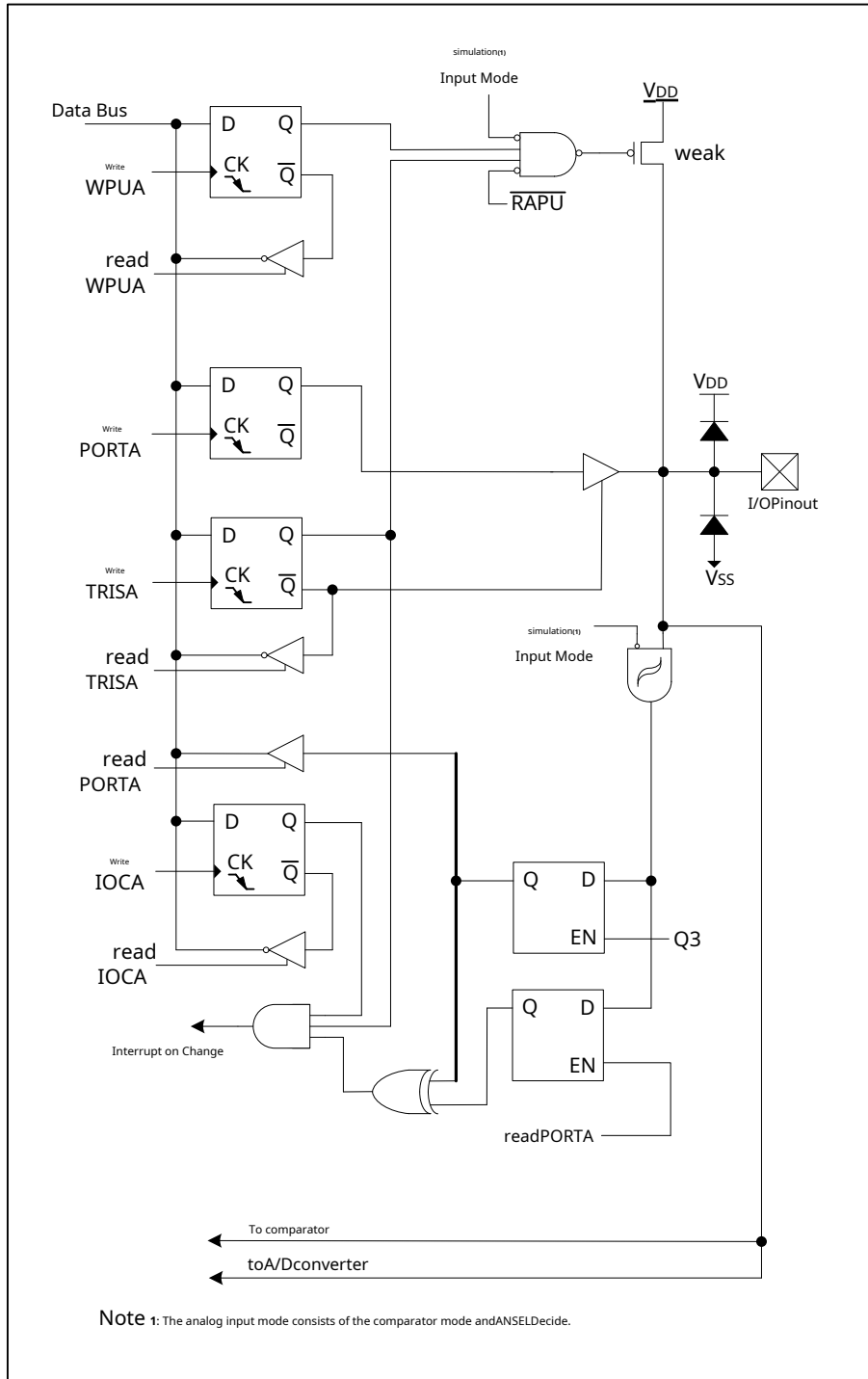


6.4.2 PA1/AN1/C1IN-/VREF/SDA

picture6-4-2 The pin diagram for this pin is given.PA1/AN1/C1IN-/VREF/SDAThe pin can be configured to function as one of the following:

- General I/O
- Connect to ADC Analog input
- Connect to the analog input of the comparator
- ADC Reference voltage input
- Data port for in-circuit serial programming and debugging SDA

picture6-4-2:PA1 block diagram

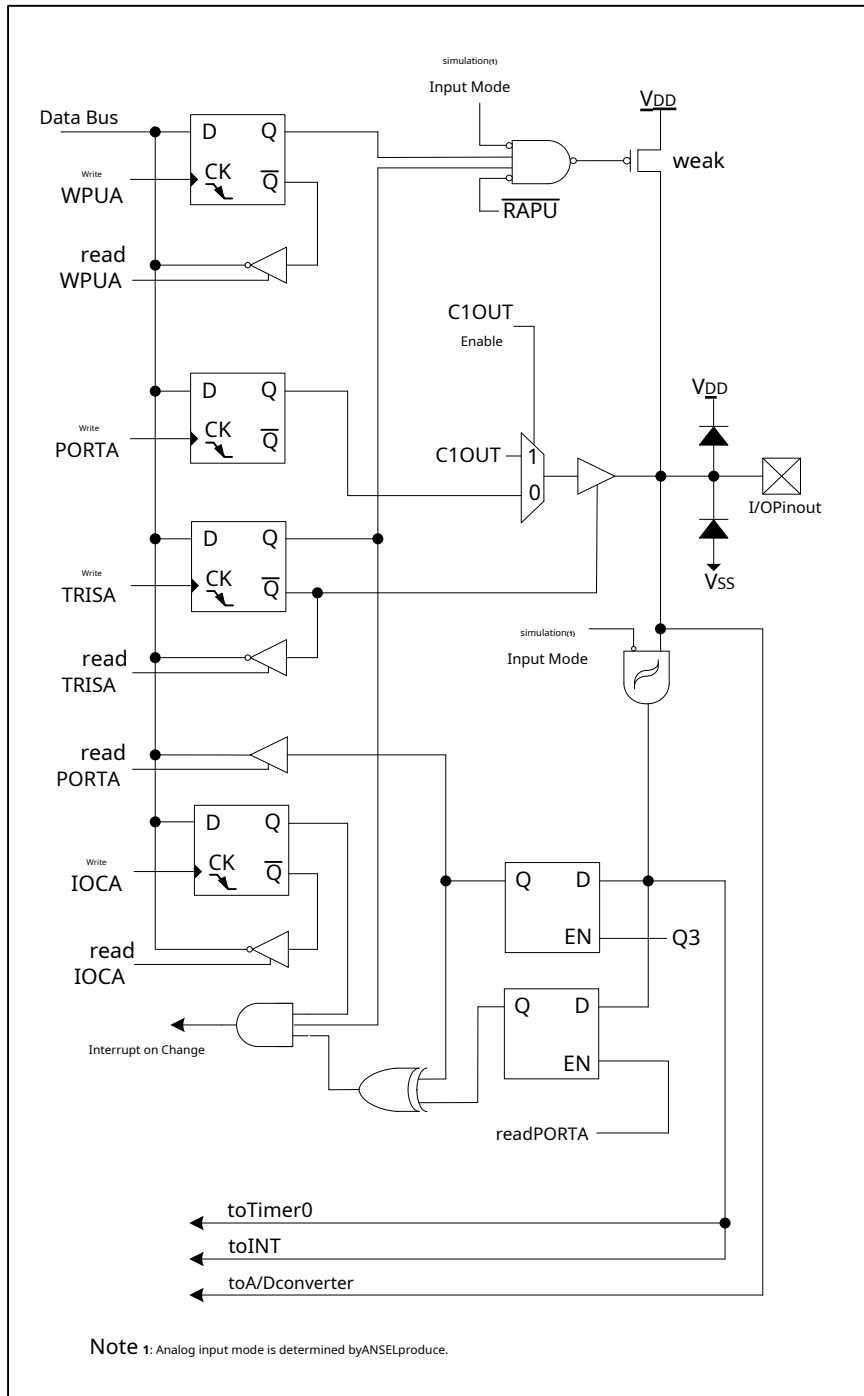


6.4.3 PA2/AN2/T0CKI/INT/C1OUT/EE_EW_CLK

picture6-4-3 The pinout diagram for this pin is given.PA2/AN2/T0CKI/INT/C1OUT/EE_EW_CLKThe pin can be configured to function as one of the following:

- General I/O
- Connect to ADC Analog input
- RTCClock input
- External edge-triggered interrupt
- From the comparator1 Digital output
- DATA EEPROM Erase and Write Self_timer Clock

picture6-4-3:PA2block diagram

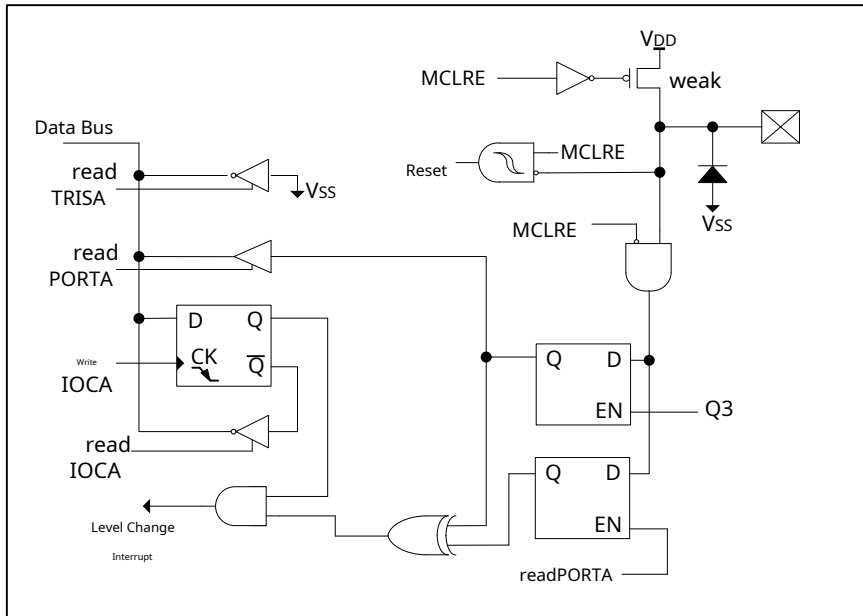


6.4.4 PA3/MCLR/VPP

[picture6-4-4](#) The pinout diagram for this pin is given. PA3/MCLR/VPP The pin can be configured to function as one of the following:

- Universal input
- Master Clear Reset with weak pull-up

picture6-4-4: PA3 block diagram

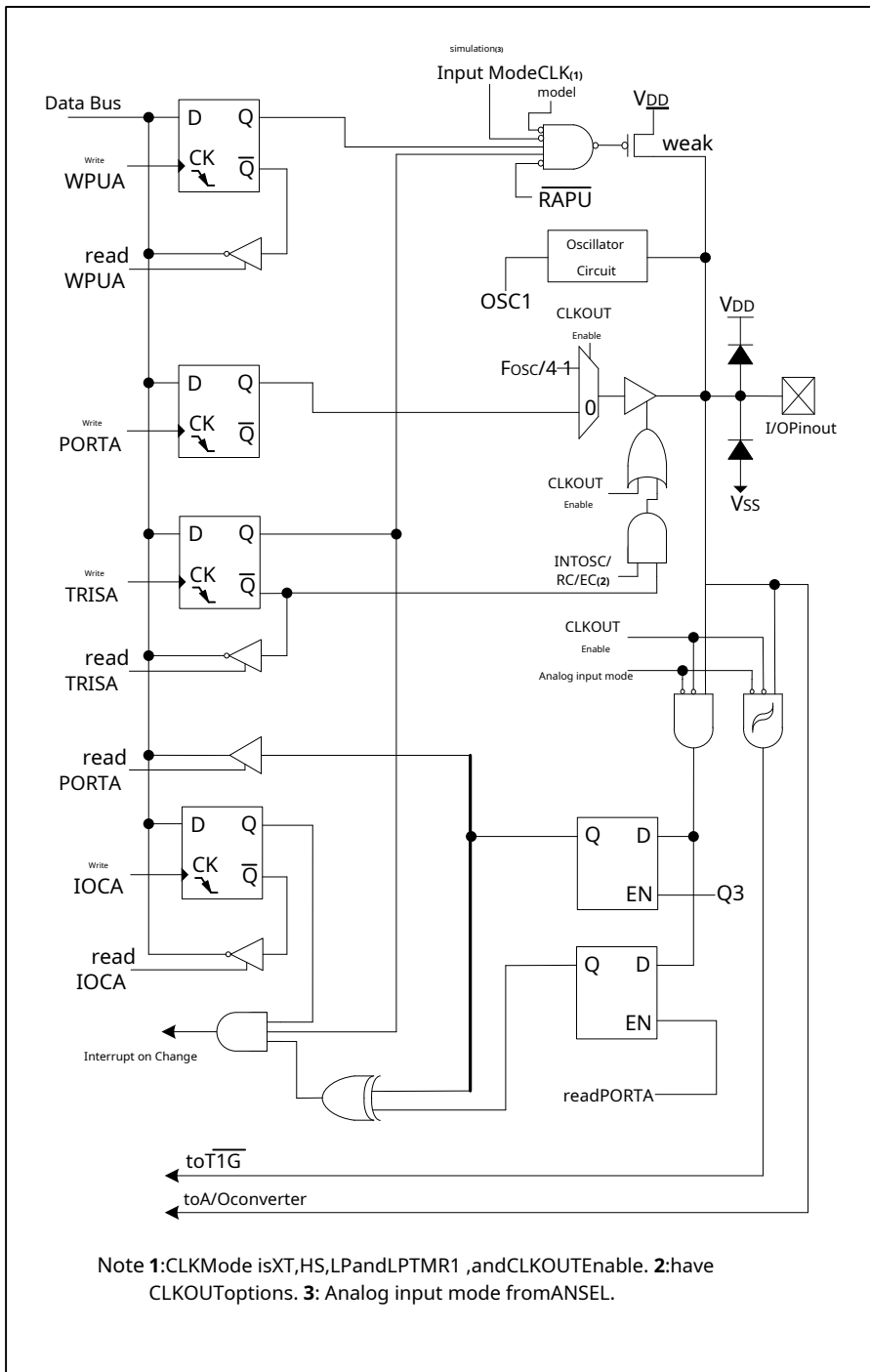


6.4.5 PA4/AN3/T1G/OSC2/CLKOUT

The pinout diagram for this pin is given. PA4/AN3/ T1G /OSC2/CLKOUT The pin can be configured to function as one of the following:

- General I/O
- Connect to ADC Analog input
- Timer1 Gate control (count enable)
- Crystal/Resonator Connections
- Clock output

picture6-4-5:PA4block diagram

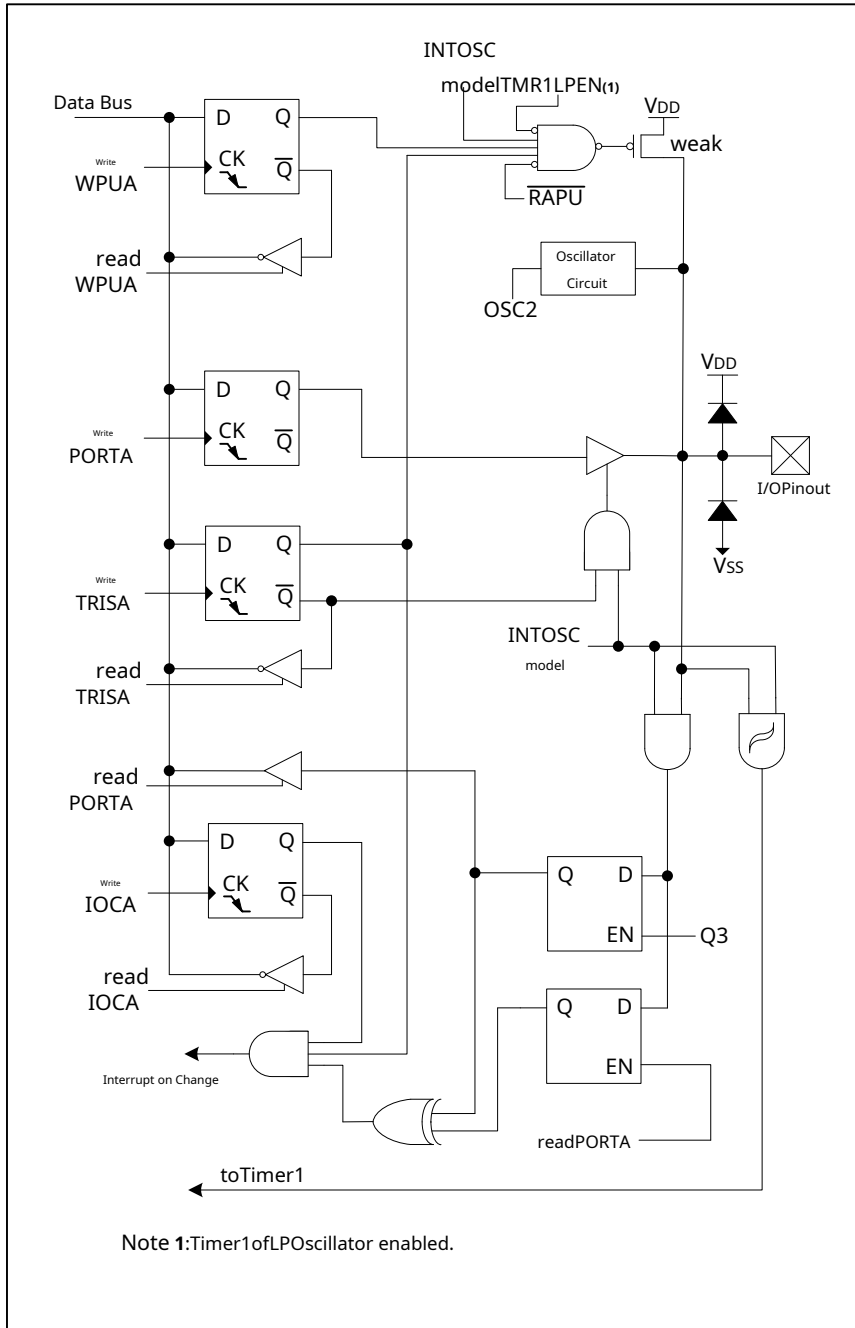


6.4.6 PA5/T1CKI/OSC1/CLKIN

picture6-4-6 The pin diagram for this pin is given.PA5/T1CKI/OSC1/CLKINThe pin can be configured to function as one of the following:

- General I/O
- TMR1 Clock Input
- Crystal/Resonator Connections
- Clock input

picture6-4-6:PA5block diagram



surface6-4:andPORT ASummary of related registers

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	PORandBOR The value of	All other Reset value
ADINS	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CMSTA	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
PSTA	—	—	ULPWUE	SBOREN	—	—	POR	BOR	--01 --qq	--0u --uu
INTS	GIE	PE1E	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
PAINTR	—	—	PAINTR5	PAINTR4	PAINTR3	PAINTR2	PAINTR1	PAINTR0	--00 0000	--00 0000
TMR	PA PH	IES	TCS	TCE	PSC	PS2	PS1	PS0	1111 1111	1111 1111
PORTA	—	—	PA5	PA4	PA3	PA2	PA1	PA0	--x0x000	--uu uu00
CPIOA	—	—	CPIOA5	CPIOA4	CPIOA3	CPIOA2	CPIOA1	CPIOA0	--11 1111	--11 1111
PAPHR	—	—	PHA5	PHA4	—	PHA2	PHA1	PHA0	--11 -111	--11 -111

Figure caption: x =unknown,u =unchanged, — = unimplemented (read as0) .PORT AShaded cells are not used.

6.5 PORTC Pin Description and Pin Diagram

6.5.1 PC0/AN4/C2IN+

PC0The pin can be configured to function as one of the following:

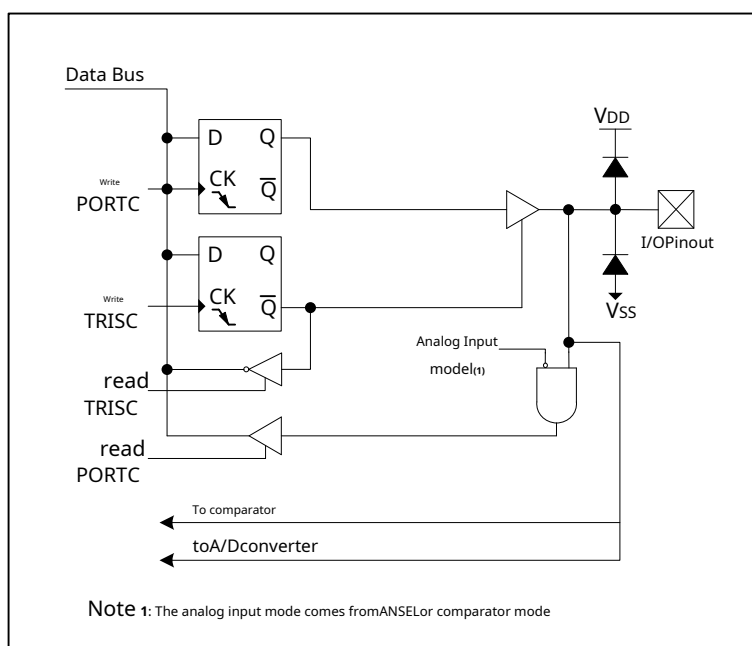
- General I/O
- ADC Analog input
- Comparator analog input

6.5.2 PC1/AN5/C2IN-

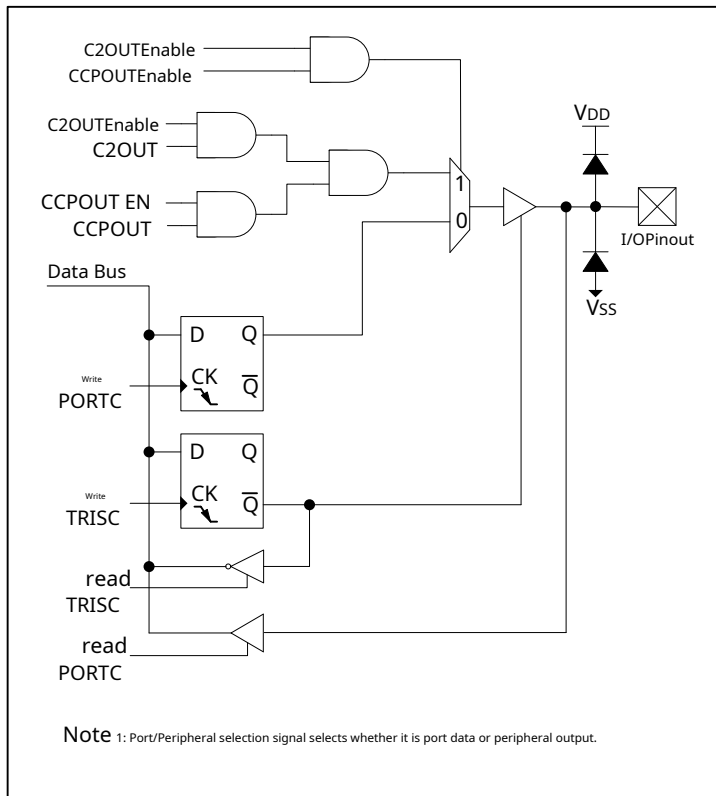
PC1The pin can be configured to function as one of the following:

- General I/O
- ADC Analog input
- Comparator analog input

picture6-5-2:PC0andPC1block diagram



picture6-5-5:PC4block diagram

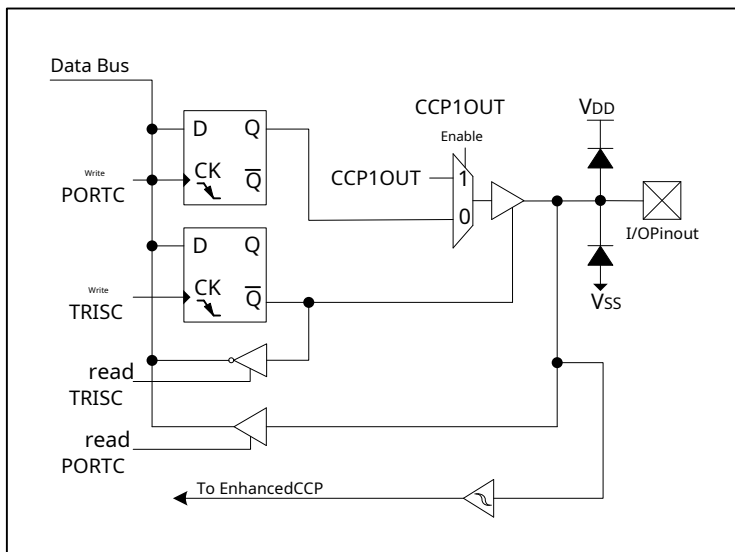


6.5.6 PC5/CCP1/P1A

PC5The pin can be configured to function as one of the following:

- General I/O
- From Enhanced CCP Digital Input/Output

picture6-5-6:PC5block diagram



6.5.7 PC6

PC6The pin can be configured to function as one of the following:

- GeneralI/O

6.5.8 PC7

PC7The pin can be configured to function as one of the following:

- GeneralI/O

surface6-5:andPORTCSummary of related registers

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POrandBOR	All other
									The value of	Reset value
ADINS	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CMSTA	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
PORTC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	xxxx 0000	uuuuuu00
CPIOC	CPIOC7	CPIOC6	CPIOC5	CPIOC4	CPIOC3	CPIOC2	CPIOC1	CPIOC0	1111 1111	1111 1111

Figure caption: x =unknown,u =unchanged, — = unimplemented (read as0) .PORT CShaded cells are not used.

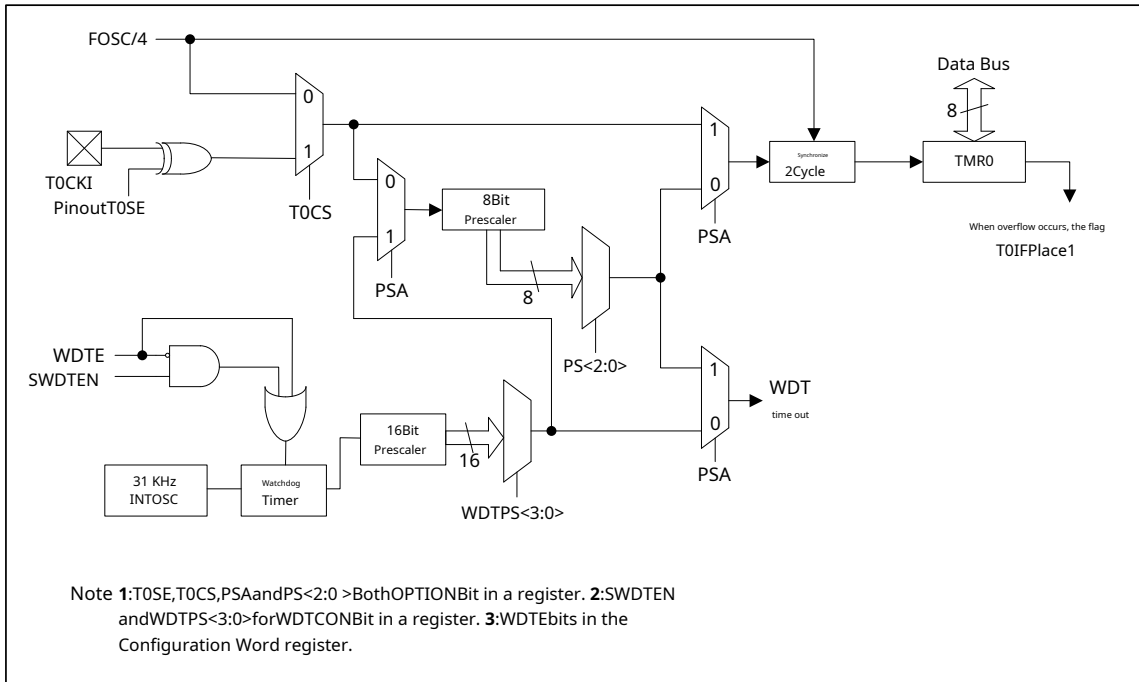
7.0 Timer

7.1 Timer0 Modules

Timer0The module is 88-bit timer/counter with the following features:

- 8-bit Timer/Counter Register (RTCC)
- 81-bit prescaler (shared with watchdog timer)
- Programmable internal or external clock source
- Programmable external clock edge selection
- Overflow interrupt

picture7-1:Timer0Block diagram of the module



7.1.1 Timer0How it works

When used as a timer,Timer0Modules can be used as8bit timer or8Bit counter.

register01H:Timer0Module Registers (RTCC)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
RTCC7	RTCC6	RTCC5	RTCC4	RTCC3	RTCC2	RTCC1	RTCC0
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7-0 RTCC <7:0>:81-bit timer/counter register

7.1.2 8Bit Timer Mode

When used as a timer, Timer0The module will increment on every instruction cycle (no prescaler) .WillTMRregister([register81H](#)) ofTCSClearing this bit to 0 selects timer mode.

whenRTCCWhen written, it will be disabled immediately after writing2Cycle increments.

Note:RTCCWhen written, in order to take into account the delay of two instruction cycles, the writeRTCCThe value of the register.

7.1.3 8Bit Counter Mode

When used as a counter, Timer0The module will beTCKIEEach rising edge on the pin is incremented. The incrementing edge is determined byTMRregister([register 81H](#)) ofTCEThe position is determined. TMRRegisterTCSLocation1Selects counter mode.

7.1.4 Software programmable prescaler

Timer0or watchdog timer (WDT) can use a software programmable prescaler, but not both. The prescaler assignment is determined byTMRRegisterPSCbit control. To assign the prescaler toTimer0, you mustPSCBit cleared.

Timer0The module's prescaler ratio is8Options, from1:2to1:256The prescaler ratio can be set byTMRregister([register81H](#)) of PS<2:0>bit to select.Timer0Module Get1:1The prescaler must be assigned toWDTmodule.

The prescaler is not readable or writable.Timer0module, all writesRTCCRegister instructions will clear the prescaler.WDTWhen, aCLRWTThe instruction will clear both the prescaler andWDT.

7.1.4.1 existTimer0andWDTSwitching prescalers between modules

Since the prescaler can be assignedTimer0orWDT, so an unwanted device reset may occur when switching prescaler ratios.Timer0Switch toWDTmodule, you must execute[example7-1-4-1-1](#) The instruction sequence shown.

example7-1-4-1-1: Change the prescaler (TIMER0→WDT)

```
BCR    STATUS,PAGE    ;Bank 0
CLRWT                      ; Clear watchdog timer
CLRR   TMR0           ;clearTMR0
BSR    STATUS,PAGE    ;Bank 1
LDWI   b'10001111'    ; Disable pull-up,RTCCFrequency division for watchdog, frequency division ratio1:128 ;
TMODE
CLRWT                      ; Clear watchdog
BCR    STATUS,PAGE    ;Bank 0
```

When the prescaler is switched fromWDTSwitch toTimer0module, the following instruction sequence must be executed (see[example7-1-4-1-2](#))

example7-1-4-1-2: Change the prescaler (WDT→TIMER0)

```

CLRWT                ; Clear watchdog
BSR  STATUS,PAGE    ;Bank 1
LDWI  b'xxxx0xxx'   ; Assign the divider toTMR0
TMODE                ;
BCR STATUS, PAGE    ;Bank 0

```

7.1.5 Timer0Interrupt

Timer0An interrupt will be generated when register overflows, regardless of whether it is allowed. You must use software to clear the interrupt enable bit. The interrupt flag will be set every timer overflow, regardless of whether it is allowed. The interrupt enable bit is INTS Register Bit.

Note: Since the timer is frozen during sleep, Timer0An interrupt cannot wake the processor from sleep.

7.1.6 Timer0Use with an external clock

When processing counter mode, the registers are synchronized by sampling the internal phase clock. Therefore, the high and low periods of the external clock source must meet [No.13.0Section "Electrical Characteristics"](#) The timing requirements listed in .

Summary of related registers

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POr and BOR The value of	All other Reset value
RTCC	Timer0Module Registers								xxxx xxxx	uuuu uuuu
INTS	GIE	PE1E	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
TMR	PA PH	IES	TCS	TCE	PSC	PS2	PS1	PS0	1111 1111	1111 1111
CPIOA	—	—	CPIOA5	CPIOA4	CPIOA3	CPIOA2	CPIOA1	CPIOA0	--11 1111	--11 1111

Figure caption: x = unknown, u = unchanged, — = unimplemented (read as 0). Timer0Shaded cells are not used.

7.2 With gate Timer1 Modules

The module is 168-bit timer/counter with the following features:

- 16-bit Timer/Counter Register (TMR1H:TMR1L)
 - Programmable internal or external clock source
 - 3Bit Prescaler
 - Optional LPOscillator
 - Synchronous or asynchronous operation
 - Through comparators or T1GPinTimer1Gate control (count enable)
 - Overflow interrupt
 - Wake-up on overflow (external clock and asynchronous mode only)
 - Time base for capture/compare functions
 - Special event trigger (with ECCP)
 - Comparator output and Timer1Clock synchronization

Figure caption:

R =Readable bit

W =Writable bit

U =Unimplemented bit, read as0

-n = PORThe value of

1 =Place1

0 =Clear

x =unknown

bit7-0 **TMR1 <15:8>**:16BitTimer1Timer/Counter High Byte Register

7.2.2Clock source selection

T1STARegisterTMR1CSbits are used to select the clock source.TMR1CS = 0When the clock source isFOSC/4.TMR1CS = 1 When the clock source is provided externally.

Clock Source	TMR1CS
FOSC/4	0
T1CKIPinout	1

7.2.2.1Internal clock source

When the internal clock source is selected,TMR1H:TMR1LThis pair of registers will beFOSCA multiple ofTimer1Prescaler
Certainly.

7.2.2.2External clock source

When selecting an external clock source,Timer1The module can work as a timer or a counter.

When counting,Timer1On external clock inputT1CKIIn addition, the Counter Mode clock can be synchronized to the microcontroller system clock or run asynchronously.

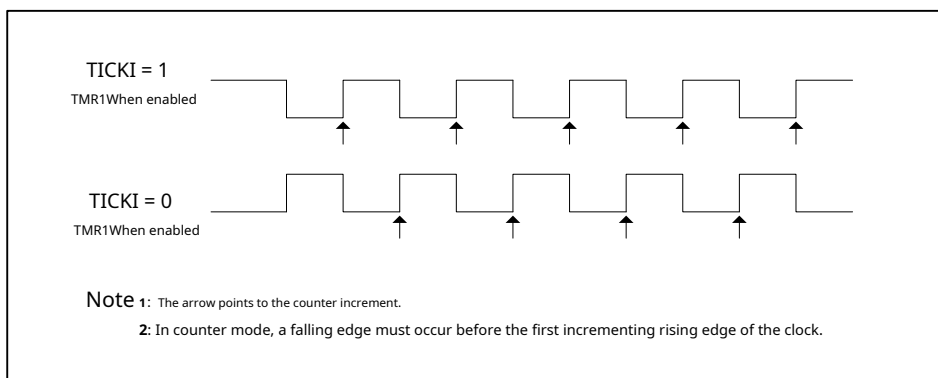
If an external clock oscillator is required (and the microcontroller is used withoutCLKOUTofINTOSC),butTimer1be usableLPOscillator as clock source.

Note: In counter mode, after any one or more of the following conditions occur, the counter must first experience a falling edge before the first rising edge increments :

- POREnable after resetTimer1
- WriteTMR1HorTMR1L
- Timer1Prohibited
- T1CKIWhen high levelTimer1Prohibited
- (TMR1ON = 0) , thenT1CKIWhen low levelTimer1Enabled (TMR1ON = 1).

See[picture7-2-2](#)

picture7-2-2:TIMER1Incrementing edge



7.2.3 Timer1 Prescaler

There are four prescaler options to adjust the clock input frequency division. The prescaler counter cannot be read or written directly; however, the prescaler counter is cleared when a write operation is performed.

7.2.4 Timer1 Oscillator

The register will control location 1. Enable the oscillator. The oscillator continues to run during sleep mode.

Oscillator sharing. In this way, this can only be done when the main system clock comes from the internal oscillator or the oscillator is in LP mode. The user must provide a software delay to ensure proper oscillator start-up.

When the oscillator is enabled, location 1.

The bits are read as 0 and 1.

Note: The oscillator requires a period of start-up and stabilization time before use. There should be an appropriate delay.

7.2.5 Timer1 Working in asynchronous counter mode

Control bits of registers, the external clock input is not synchronized. The timer increments asynchronously with the internal phase clock. If an external clock source is selected, the timer will continue to run during sleep and can generate an interrupt on overflow to wake the processor. However, extreme caution should be exercised when reading and writing the timer (see [Section "Reading and Writing in Asynchronous Counter Mode" Timer1](#)).

Note: When switching from synchronous to asynchronous operation, an increment may be missed. When switching from asynchronous to synchronous operation, an extra increment may be generated.

7.2.5.1 Reading and Writing in Asynchronous Counter Mode Timer1

When the timer runs on an external asynchronous clock, reading will ensure that the read operation is valid (this is the responsibility of the hardware). However, it should be noted that using two 8-bit values to read 16-bit timers themselves present certain problems because the timer may overflow between read operations.

For write operations, it is recommended that the user stop the counter directly and then write the desired value. If the register is counting up, writing to the timer register may cause write contention, which may produce unpredictable values in the registers.

7.2.6 Timer1Gating

The gate source can be software configured as T1GPin or comparatorC2. This allows the device to be used directly for timing external events, or using comparators for timing of simulation events. For selection of gating source, see CMCON1 register ([register99H](#)). This feature can simplify Δ - Σ /D converter and many other applications.

Note: T1STAR register TMR1GE Position must be set to T1GorC2OUT Use as Timer1 Gating source. More about selection of gating sources, see CMCON1 register ([register99H](#)).

use T1STAR register T1GINV Bit flippable for Timer1 Gating, regardless of its source T1GPin or comparatorC2. This will configure Timer1 to ensure that there is an active low or active high time between events.

7.2.7 Timer1Interrupt

A pair of registers (TMR1H:TMR1L) increases to 0FFFh. After Return 0000h. When the counter is full, PIEB1 Register Timer1 The interrupt flag is set. To ensure that an interrupt occurs on a rollover, you must set the following locations:

- T1STAR register TMR1ON Bit
- PIEB1 Register TMR1IE Bit
- INTS Register PEIE Bit
- INTS Register GIE Bit

In the interrupt service routine, TMR1IF Clearing the bit to 0 will clear the interrupt flag.

Note: TMR1H:TMR1L This register and TMR1IF Bit should be cleared before enabling interrupts.

7.2.8 Timer1How it works in sleep mode

Only when the asynchronous counter mode is set, Timer1 To operate in Sleep mode, an external crystal or clock source signal can be used to increment the counter in this mode. To set up the timer to wake up the device:

- You must T1STAR register TMR1ON Location 1
- You must PIEB1 Register TMR1IE Location 1
- You must INTS Register PEIE Location 1
- You must T1STAR register T1SYNCLocation 1
- You must T1STAR register TMR1CS Location 1
- You can T1STAR register T1OSCEN Location 1

When an overflow occurs, the device wakes up and executes the next instruction. INTS Register GIE Location 1, the device will call the interrupt service routine (0004h).

7.2.9 ECCP Capture/Compare Time Base

When operating in capture or compare mode, ECCP TMR1H:TMR1L Register as time base.

In capture mode, TMR1H:TMR1L The value of this register pair is copied to CCPR1H:CCPR1L This pair of registers.

In compare mode, when $CCPR1H:CCPR1L$ The value of this register is $TMR1H:TMR1L$ When the value of matches, an event will be triggered. This event can be a special event trigger.

For more information, see [No.8.0Section "Enhanced Capture/Compare/PWM+\(With automatic shutoff and deadband\) module"](#).

7.2.10 ECCP Special event triggers

when ECCP When configured to trigger a special event, the trigger will $TMR1H:TMR1L$ This clears the register to 0. This special event does not generate Timer1 Interrupt. ECCP The module can still be configured to produce ECCP Interrupt.

In this working mode, $CCPR1H:CCPR1L$ This pair of registers becomes Timer1 The period register.

Timer1 Should be synchronized FOSC to take full advantage of special event triggers. Timer1 Working asynchronously can result in missing special event triggers.

When $TMR1H$ or $TMR1L$ The write operation is similar to a ECCP When special event triggers occur simultaneously, the write operation has priority.

For more information, see [No.8.4.5Section "Special Event Triggers"](#).

7.2.11 Comparator Synchronization

Used to make Timer1 The incrementing clock can also be used to synchronize the comparator outputs. This feature can be enabled in the comparator module.

Use the comparator for Timer1 When gated, the comparator output should be synchronized to Timer1 This will ensure that when the comparator changes Timer1 Do not miss an increment. For more information, see [No.10.9Section "Comparator C2 Output and Timer1 Synchronize"](#).

7.2.12 and Timer1 Summary of related registers

surface7-2-12:andTIMER1Summary of related registers

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR and BOR The value of	All other Reset value
CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	----- 10	----- 10
INTS	GIE	PEIE	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
PIEB1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIFB1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMR1IF	0000 0000	0000 0000
TMR1H	16BitTMR1High Byte Holding Register								xxxx xxxx	uuuu uuuu
TMR1L	16BitTMR1Low Byte Holding Register								xxxx xxxx	uuuu uuuu
T1STA	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCN	T1SYNC	TMR1CS	TMR1ON	0000 0000	1111 1111

Figure caption: x = unknown, u = unchanged, — = unimplemented (read as 0). Timer1 Shaded cells are not used by the module.

7.3 Timer2 Modules

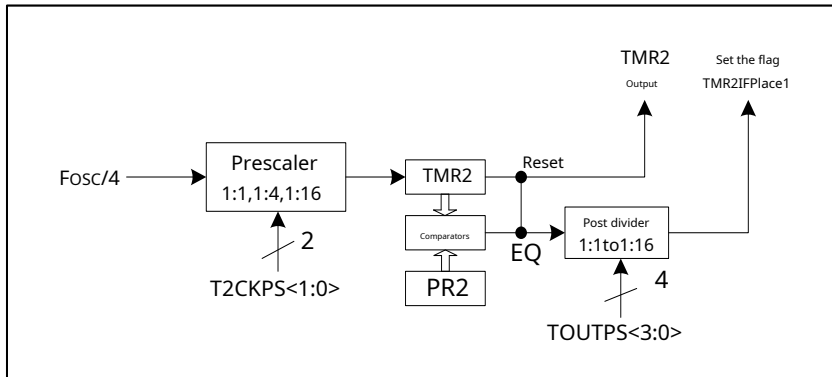
Timer2A module is a 8Bit timer with the following functions:

- 8Bit Timer Register (TMR2)
- 8Bit Period Register (PR2)
- TMR2 and PR2 Break on Match
- Software programmable prescaler ratio (1:1, 1:4 or 1:16)

- Software programmable postscaler (1:1to1:16)

Timer2For a block diagram, see [picture7-3](#) .

picture7-3:TIMER2block diagram



7.3.1 Timer2How it works

Timer2The clock input of the module is the system instruction clock (FOSC/4) The clock is fed intoTimer2The prescaler has a prescaler ratio of1:1, 1:4or1:16The output of the prescaler is then used to incrementTMR2register.

TMR2andPR2The values of are constantly compared to determine when there is a match.TMR2Will be from00hStart increasing until it reachesPR2The following two things happen when they match:

- TMR2Resets to00h.
- Timer2Post-divider ratio increment

Timer2/PR2The match output of the comparator is fed intoTimer2Post divider. The options for the post divider range from1:1to1:16.Timer2The output of the post divider is used toPIR1RegisterTMR2IFInterrupt flag set1.

TMR2andPR2The registers are fully readable and writable.TMR2Register set to00handPR2Register set toFWf.

WillT2CONRegisterTMR2ONLocation1OpenableTimer2.WillTMR2ONClear bit to 0 to turn offTimer2.

Timer2The prescaler is composed ofT2CONRegisterT2PBit control.Timer2The post divider is composed ofT2CONRegisterTOUTPS The prescaler and postscaler counters are cleared when:

- TMR2of write operations.
- T2CONof write operations.
- Any device reset occurs (power-on reset,MCLRreset, watchdog timer reset or brownout reset) .

Note:T2CONWhen writtenTMR2Not cleared.

register12H:TIMER2Control Register (T2CON)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit7							bit0

Figure caption:
 R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7

Unimplemented: Read as0

bit6-3

TOUTPS<3:0>:Timer2Output post-division ratio selection

bits 0000 = 1:1

0001 = 1:2

0010 = 1:3

0011 = 1:4

0100 = 1:5

0101 = 1:6

0110 = 1:7

0111 = 1:8

1000 = 1:9

1001 = 1:10

1010 = 1:11

1011 = 1:12

1100 = 1:13

1101 = 1:14

1110 = 1:15

1111 = 1:16

bit2

TMR2ON:Timer2Open Bit

1 = Timer2Open

0 = Timer2closure

bit1-0

T2CKPS<1:0>:Timer2Clock prescaler ratio selection bit

00 = 1:1

01 = 1:4

1x = 1:16

register11H:Timer2Module Registers (TMR2)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
TMR2<7>	TMR2<6>	TMR2<5>	TMR2<4>	TMR2<3>	TMR2<2>	TMR2<1>	TMR2<0>
bit7							bit0

Figure caption:

R =Readable bit

W =Writable bit

U =Unimplemented bit, read as0

-n = PORThe value of

1 =Place1

0 =Clear

x =unknown

bit7-0

TMR2<7:0>:Timer2Timer Registers**register92H:Timer2Module Period Register (PR2)**

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
PR2<7>	PR2<6>	PR2<5>	PR2<4>	PR2<3>	PR2<2>	PR2<1>	PR2<0>
bit7							bit0

Figure caption:

R =Readable bit

W =Writable bit

U =Unimplemented bit, read as0

-n = PORThe value of

1 =Place1

0 =Clear

x =unknown

bit7-0

PR2<7:0>:Timer2Module Period Register

surface7-3-4:andTimer2Summary of related registers

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	PORandBOR The value of	All other Reset value
INTS	GIE	PEIE	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
PIEB1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIFB1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMRIF	0000 0000	0000 0000
PR2	Timer2Module Period Register								1111 1111	1111 1111
TMR2	Timer2Module Registers								0000 0000	0000 0000
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	- 000 0000	- 000 0000

Figure caption: x =unknown,u =unchanged, — = unimplemented (read as0) .Timer2Shaded cells are not used by the module.

8.0Enhanced Capture/Compare/PWM+(With automatic shutoff and deadband) module

Enhanced Capture/Compare/PWMThe module is a peripheral that the user can use to time and control different events. In capture mode, the peripheral can time the duration of an event. Compare mode allows the user to trigger an external event after a predetermined period of time.PWMmode can generate pulse width modulated signals with variable frequency and duty cycle.

8.1surface:ECCPMode - Timer resources required

ECCPmodel	Timer Resources
Capture	Timer1
Compare	Timer1
PWM	Timer2

8.2register:CCP1CON: EnhancedCCP1Control Register

register15H: EnhancedCCP1Control Register (CCP1CON)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
bit7							bit0

Figure caption:

R =Readable bit

W =Writable bit

U =Unimplemented bit, read as0

-n = PORThe value of

1 =Place1

0 =Clear

x =unknown

bit7-6

P1M<1:0>:PWMOutput Configuration

Bits IfCCP1M<3:2> = 00,01or10:

xx = P1AConfigured as capture/compare input;P1B,P1CandP1DConfigured as a port pin ifCCP1M<3:2> = 11:

00 =Single output;P1Amodulation;P1B,P1CandP1DConfigured as port pins 01 =

Full bridge forward output;P1Dmodulation;P1Aefficient;P1BbandP1Cinvalid

10 =Half-bridge output;P1AandP1Bmodulation, with dead-band control;P1CandP1DConfigured as port pins 11

=Full bridge reverse output;P1Bmodulation;P1Cefficient;P1AandP1Dinvalid

Note: To use PWM for 8-pin 684, only 01 can be set, other modes are invalid.

bit5-4 **DC1B<1:0>**:PWMDuty cycle least significant bit
 capture mode:
 Unused.
 Comparison Mode:
 Unused.
 PWMmodel:
 These bits arePWMLow duty cycle2High8atCCPR1Lmiddle.

bit3-0 **CCP1M<3:0>**:ECCPMode select bits
 0000 =Capture/Compare/PWMClose (ResetECCPmodule)
 0001 =Not used (reserved)
 0010 =Compare mode, flip output on match (CCP1IFLocation1)
 0011 =Not used (reserved)
 0100 =Capture mode, each falling edge
 0101 =Capture mode, each rising edge
 0110 =Capture mode, each4Rising edge
 0111 =Capture mode, each16Rising edge
 1000 =Compare mode, output is set when match occurs1 (CCP1IFLocation1)
) 1001 =Compare mode, output cleared on match (CCP1IFLocation1)
 1010 =Compare mode, software interrupt is generated when a match occurs (CCP1IFLocation1,CCP1pins are not affected)
 1011 =Compare mode, trigger special events (CCP1IFLocation1,CCP1ResetTMR1, and ifA/DThe module is enabled.
 Start onceA/DConversion
 1100 = PWMmodel;P1AandP1CHigh level is valid;P1BandP1DHigh level is effective
 1101 = PWMmodel;P1AandP1CHigh level is valid;P1BandP1DLow level is effective
 1110 = PWMmodel;P1AandP1CLow level is valid;P1BandP1DHigh level is effective
 1111 = PWMmodel;P1AandP1CLow level is valid;P1BandP1DLow level is effective

8.3Snap Mode

In capture mode, whenCCP1When an event occurs on the pin,CCPR1H:CCPR1LCaptureTMR1Register16An event is defined as one of the following and is represented byCCP1CONRegisterCCP1M<3:0>Bits to configure:

- Each falling edge
- Each rising edge
- Every4Rising edge
- Every16Rising edge

After capturing,PIFB1Interrupt request flag in registerCCP1IFPlaced1This bit must be cleared to 0 by software.CCPR1H andCCPR1LIf another capture occurs before the value in this register pair is read, the old captured value will be overwritten by the new captured value (see[picture 8-3-1](#)).

register13H:capture/compare/PWMregister1The low byte ofCCPR1L)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
CCPR1<7	CCPR1<6	CCPR1<5	CCPR1<4	CCPR1<3	CCPR1<2	CCPR1<1	CCPR1<0
>	>	>	>	>	>	>	>
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7-0 **CCPR1<7:0>**:capture/compare/PWMregister1The low byte



register14H:capture/compare/PWMregister1The high byte ofCCPR1H)

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
CCPR1<15	CCPR1<14	CCPR1<13	CCPR1<12	CCPR1<11	CCPR1<10	CCPR1<9	CCPR1<8
>	>	>	>	>	>	>	>
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

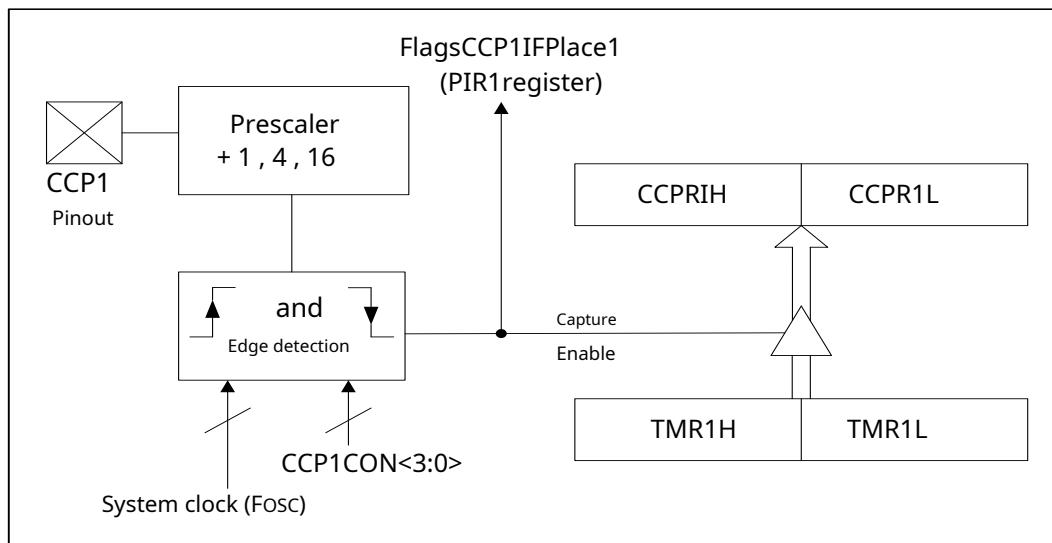
bit7-0 CCPR1<15:8>:capture/compare/PWMregister1High Byte

8.3.1 CCP1Pin Configuration

In capture mode, the relevantCPIOControl location1WillCCP1Pin configured as input.

Note: IfCCP1If the pin is configured as an output, writing to the port will generate a capture condition.

picture8-3-1: Capture mode working principle block diagram



8.3.2 Timer1Mode Selection

To makeCCPThe module uses the capture feature,Timer1Must be running in Timer mode or Synchronous Counter mode. In Asynchronous Counter mode, capture operation may not work.

8.3.3Software interrupts

When the capture mode is changed, a false capture interrupt may be generated. The user should keepPIEB1RegisterCCP1IEIn addition, the user should also clear this bit to zero after any such operating mode change.PIFB1Interrupt flag registerCCP1IF.



8.3.4 CCP Prescaler

CCP1CON Register CCP1M<3:0> Bit specifies 4 different prescaler ratios. CCP module, or CCP when the module is not in capture mode, the prescaler counter will be cleared. Any reset will clear the prescaler counter.

Switching from one capture prescaler to another does not clear the prescaler and may generate an interrupt. To avoid unexpected operation, set the CCP1CON Clear the register to turn off CCP module (see [example8-3-4](#)).

example8-3-4: Switch between capture prescaler ratios

```
CLRR    CCP1CON           ;Reset CCP Modules
LDWI    NEW_CAPT_PS      ;
STWR    CCP1CON           ; Set new CCP parameter
```

8.4 Comparison Mode

In compare mode, 16-bit CCP1 Register values are continuously compared with TMR1. When a match occurs, CCP Modules may:

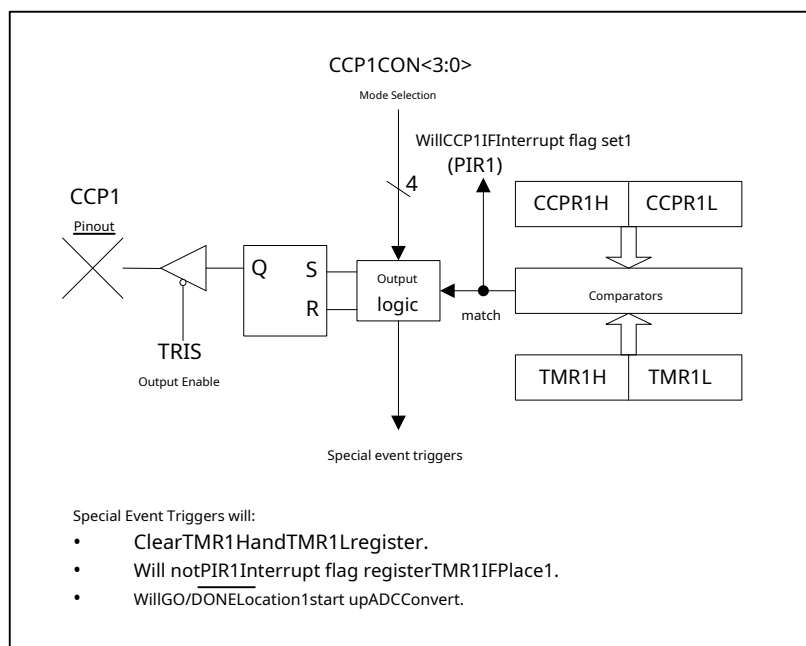
- Flip CCP1 Output
- Will CCP1 Output 1
- Will CCP1 Output clear
- Trigger special events
- Generate software interrupt

The action on the pin depends on CCP1CON Register CCP1M<3:0> The value of the control bit.

All compare modes can generate an interrupt.

8.4.1 Compare mode working diagram

picture8-4-1: Compare mode working diagram



8.4.2 CCP1Pin Configuration

The user must clear the relevant CPIO Position CCP1 Pins are configured as outputs.

Note: Clear CCP1 CON The register will force CCP1 The compare output is latched at the default low level. PORTC of I/O Data latch.

8.4.3 Timer1 Mode Selection

In compare mode, Timer1 Must be run in Timer mode or Synchronous Counter mode. Compare operations may not work properly in Asynchronous Counter mode.

8.4.4 Software interrupt mode

When software interrupt mode is selected (CCP1M<3:0> = 1010) hour, CCP The module does not get CCP1 pin control (see CCP1 CON register).

8.4.5 Special event triggers

When the special event trigger mode is selected (CCP1M<3:0> = 1011) hour, CCP The model will:

- Reset Timer1
- like ADC Enable, then start once ADC Conversion

In this mode, CCP Module does not get CCP1 pin control (see CCP1 CON register) .

once TMR1H:TMR1L This register and CCPR1H:CCPR1L When a match occurs between this pair of registers, CCP Special event trigger output. TMR1H:TMR1L Register in Timer1 The reset will not occur until the next rising edge of the clock. CCPR1H:CCPR1L Registers can be used as Timer1 of 16 Bit-programmable period register.

Note: 1: CCP The module's special event trigger will not PIFB1 Interrupt flag register TMR1IF Place 1.

- 2: On the clock edge that generates the Special Event Trigger and the reset Timer1 By changing the clock edge between CCPR1H and CCPR1L This pair
Removing the match condition by changing the contents of the register will prevent a reset from occurring.

8.5 PWM model

PWM The mode will be CCP1 The pin generates a pulse width modulated signal. Its duty cycle, period and resolution are determined by the following registers:

- PR2
- T2CON
- CCPR1L
- CCP1CON

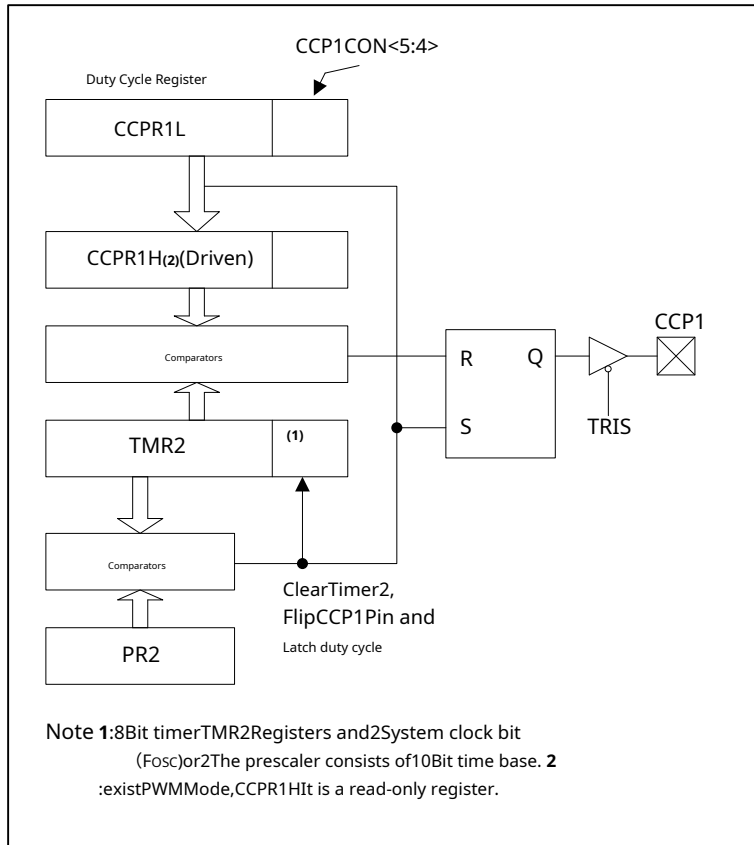
In pulse width modulation (Pulse-Width Modulation, PWM) mode, CCP Module in CCP1 The pin generates up to 10 bit resolution PWM Output. CCP1 Pins and PORT To multiplex the data latch, the pin must be CPIO Clear to enable CCP1 Pin output driver.

Note: CCP1CON Clearing the register to zero will discard the CCP1 pin control.

8.5.1 Figure: Simplified PWM block diagram

To learn more about CCP The module is set to PWM For detailed steps, see [No.8.5.9 Festival "PWM Working Settings"](#).

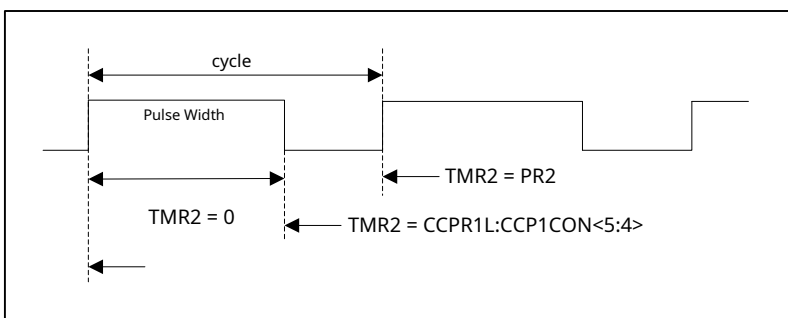
picture8-5-1: simplified PWM block diagram



8.5.2 CCP PWM Output

PWM Output ([picture8-5-2](#)) consists of a time base (period) and a period of time that the output remains high (duty cycle).

picture8-5-2: CCP PWM Output



8.5.3 PWMcycle

The cycle is Timer2 of PR2 Register specified. [formula8-5-3](#) Computable PWM cycle.

formula8-5-3: PWM cycle

$$PWM\text{Period} = [(PR2) + 1] \cdot 4 \cdot T_{osc} \cdot$$

(TMR2 Prescaler value)

when TMR2 equal PR2, the following three events will occur in the next increment cycle: .

- TMR2 is cleared.
- CCP1 Pin is set (Exception: If PWM Duty Cycle = 0%, the pin is not set).
- PWM Duty cycle from CCPR1L Latch to CCPR1H.

Note: OK PWM Frequency not used Timer2 Register (see [No.7.3.1 Festival "Timer2 How it works "](#)).

8.5.4 PWM Duty Cycle

By writing the following registers 10Bit value can be specified PWM Duty Cycle: CCPR1L Register and CCP1CON Register DC1B<1:0> Bit. CCPR1L Contains eight MSb, CCP1CON Register DC1B<1:0> The bit contains two LSB. CCP1CON Register CCPR1L and DC1B<1:0> can be written at any time. The duty cycle is not reset until the cycle is complete (i.e. PR2 and TMR2 registers) is latched into CCPR1H. In. Use PWM hour, CCPR1H The registers are read-only.

[formula8-5-4-1](#) Used for calculation PWM Pulse width.

[formula8-5-4-2](#) Used for calculation PWM Duty cycle.

formula8-5-4-1: Pulse Width

$$Pulse\ Width = (CCPR1L:CCP1CON<5:4>) \cdot$$

T_{osc} \cdot (TMR2 Prescaler value)

formula8-5-4-2: Duty cycle

$$Duty\ Cycle = \frac{(CCPR1L:CCP1CON<5:4>)}{4(PR2 + 1)}$$

CCPR1H Register and 2 The internal latch of the bit is used to PWM The duty cycle is double buffered. PWM Burr-free work plays a very important role.

8Bit Timer TMR2 Register and 2bit of the internal system clock (FOSC) or 2 The prescaler of the bit is connected to form 10bit time base. If Timer 2 The prescaler is set to 1:1 The system clock is used.

when 10Bit time base and CCPR1H and 2 When the latch of the bit matches, CCP1 pin is cleared (see [picture8-5-1](#)).

8.5.5 PWMResolution

The resolution determines the effective duty cycle for a period. For example, 10-bit resolution is 1024 discrete duty cycle, and 8-bit resolution is 256 discrete duty cycle.

Resolution is a function of register values, such as [formula 8-5-5](#) shown.

formula 8-5-5: PWMResolution

$$\text{Resolution} = \frac{\log[4(PR2 + 1)]}{\log(2)} \text{ Bit}$$

Note: (1) If the pulse width is greater than the period, then the allocated PWM pin will remain unchanged.

(2) 10-bit PWM resolution can only be generated when bits <1,0> of T2CON and <5,4> of CCP1CON are set to 00 at the same time.

surface 8-5-5-1: PWM Frequency and resolution examples (Fosc = 20MHz)

PWM frequency	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
	=5000/(256*16)	=5000/(256*4)	=5000/256	=5000/64	=5000/32	=5000/24
Timer prescale ratio (1,4 or 16)	16 (The lower two bits are invalid)	4 (The lower two bits are invalid)	1	1	1	1
PR2 Value	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
	=2 ⁸	=2 ⁸	=2 ⁸	=2 ⁶	=2 ⁵	=2 ^{4.6}
Maximum resolution (bits)	8	8	10	8	7	6.6
	=8	=8	=8+2	=6+2	=5+2	4.6+2

surface 8-5-5-2: PWM Frequency and resolution examples (Fosc = 8MHz)

PWM frequency	1.22 kHz	4.9 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
	=2000/(102*16)	=2000/(102*4)	=2000/102	=2000/26	=2000/13	=2000/10
Timer prescale ratio (1,4 or 16)	16 (The lower two bits are invalid)	4 (The lower two bits are invalid)	1	1	1	1
PR2 Value	0x65	0x65	0x65	0x19	0x0C	0x09
	=2 ^{6.67}	=2 ^{6.67}	=2 ^{6.67}	=2 ^{4.7}	=2 ^{3.7}	=2 ^{3.33}
Maximum resolution (bits)	6.67	6.67	8.67	6.7	5.7	5.33
	=6.67	=6.67	=6.67+2	=4.7+2	=3.7+2	=3.33+2

8.5.6 Operation in sleep mode

In sleep mode, TMR2 register is not incremented and the module state is unchanged. CCP1 pin is driving a value and it will continue to drive that value. TMR2 previous state will continue.

8.5.7 Changes in system clock frequency

PWM frequency is derived from the system clock frequency. Any change in the system clock frequency will result in PWM. For details, see [No. 1.1 Section "System Structure](#)

[Diagram](#)



8.5.8 Effect of reset

Any reset will force all ports to input mode and force CCPThe register is in its reset state.

8.5.9 PWM Working Settings

Follow the steps below to CCPThe module is configured as PWMWork:

1. The relevant CPIO Location 1 prohibit PWM Pin (CCP1) output driver.
2. Fitting PR2 Registers to set PWM cycle.
3. Load with appropriate values CCP1CON The register will CCPThe module is configured as PWM model.
4. Load CCPR1L Registers and CCP1CON Register DC1B<1:0> set up PWM Duty cycle.
5. Configure and start Timer2:
 - Will PIFB1 Register TMR2IF The interrupt flag is cleared.
 - load T2CON Register T2P Bit Setting Timer2 Prescaler ratio.
 - Will T2CON Register TMR2ON Location 1 Enable Timer2.
6. Start a new one PWM After the cycle, enable PWM Output:
 - wait Timer2 overflow (PIR1 Register TMR2IF Location 1).
 - The relevant CPIO Bit clear enable CCP1 pin's output driver.

8.6 PWM (Enhanced Mode)

Enhanced PWM mode can generate up to 10-bit resolution PWM. It uses four PWM. The output mode does:

- one PWM
- Half Bridge PWM
- Full Bridge PWM, forward mode
- Full Bridge PWM, reverse mode

To select Enhanced PWM model, CCP1CON Register P1M The bit must be set correctly 1.

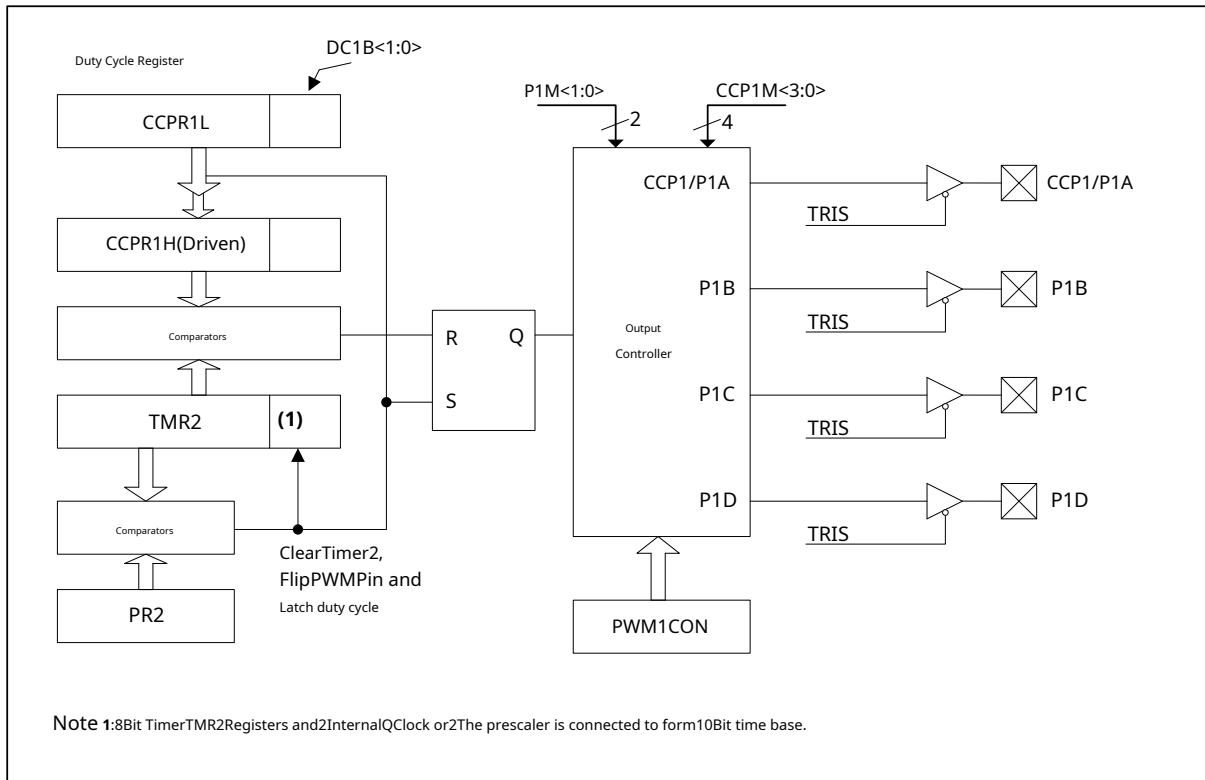
PWM Output and I/O pins are multiplexed and are designated as P1A, P1B, P1C and P1D. PWM The polarity of the pin is configurable by setting CCP1CON Register CCP1M Position appropriately 1. Select the polarity.

[surface8-6-1](#) Shown is each enhanced PWM mode pinout.

[picture8-6-1](#) Enhanced PWM Simplified block diagram of the module.

Note: Enable at the beginning PWM to prevent incomplete waveforms, ECCP The module is generating PWM. Wait for a new signal before PWM. The cycle begins.

picture8-6-1: EnhancedPWMSimplified block diagram example of a pattern

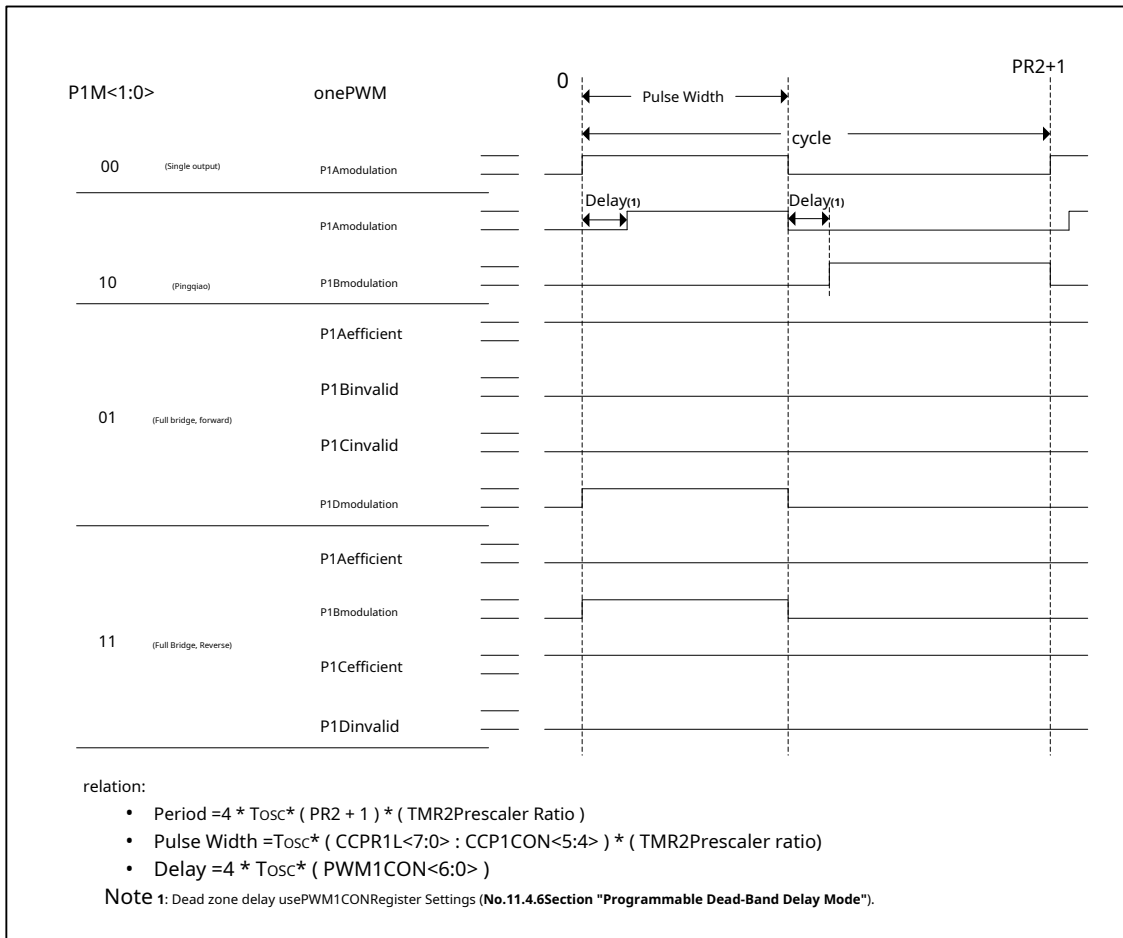


Note: 1: Each must be configured correctlyPWMOutputCPIORegister value.

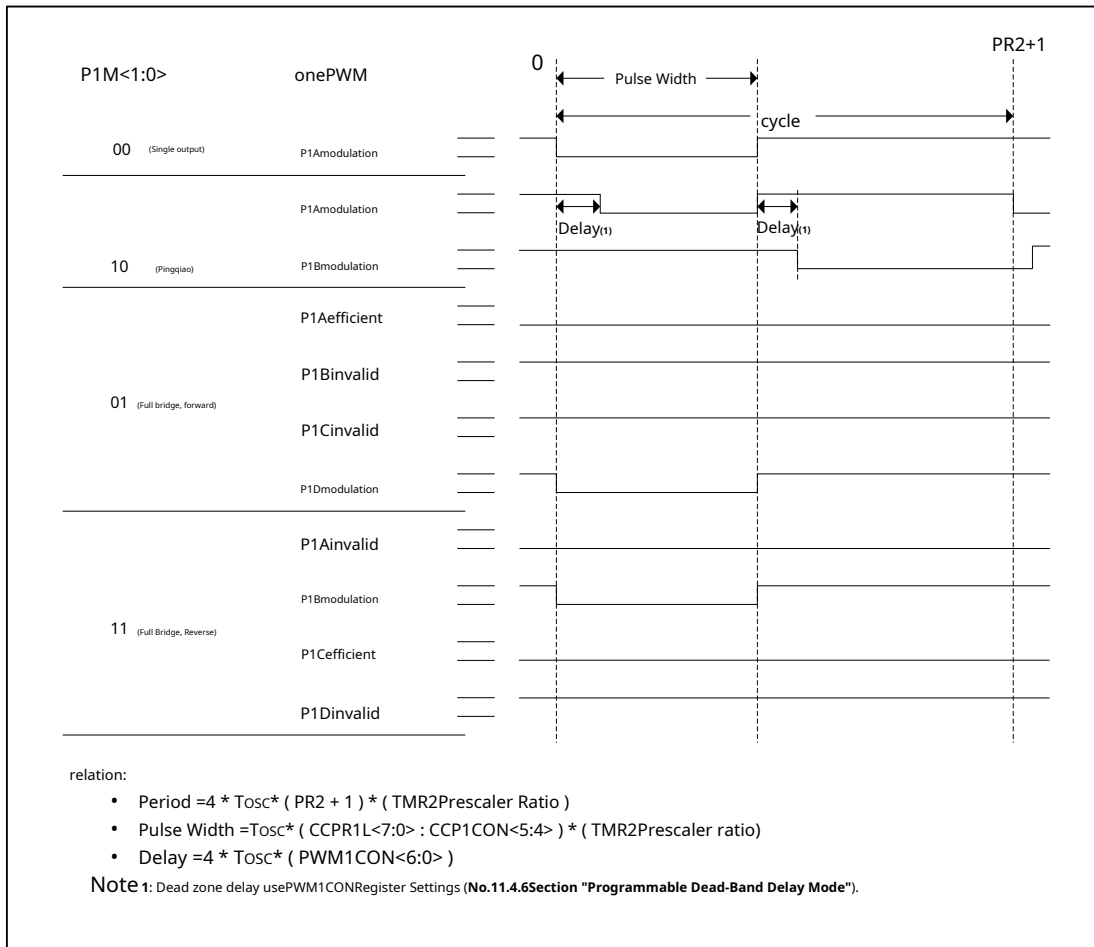
2: ClearCCP1CONThe register will discard allPWMOutput pinsECCPControl.

3: EnhancedPWMAny pins not used by a mode can be used for other pin functions.

picture8-6-2:PWM(Enhancement Mode) Output Relationship Example (Active High State)



picture8-6-3:PWM(Enhancement Mode) Output Relationship Example (Active Low State)



surface8-6-1:differentPWMExample pinout for enhanced mode

ECCPmodel	P1M<1:0>	CCP1/P1A	P1B	P1C	P1D
onePWM	00	yes	yes	yes	yes
Half BridgePWM	10	yes	yes	no	no
Full bridge, forward	01	yes	yes	yes	yes
Full Bridge, Reverse	11	yes	yes	yes	yes

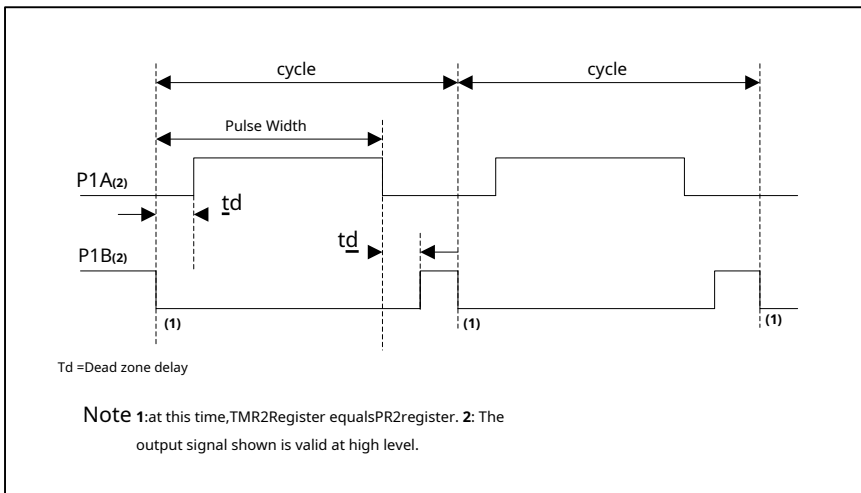
8.6.1 Half-bridge mode

In half-bridge mode, two pins are used as outputs to drive push-pull loads. PWM The output signal is sent to CCP1/P1A pins, while complementary PWM The output signal is sent to P1B pins (see [picture8-6-2](#)) . This mode can be used in half-bridge applications such as [picture8-6-1-2](#) as shown, or for full bridge applications . Use, at this time use two PWM The signal modulates four power switches.

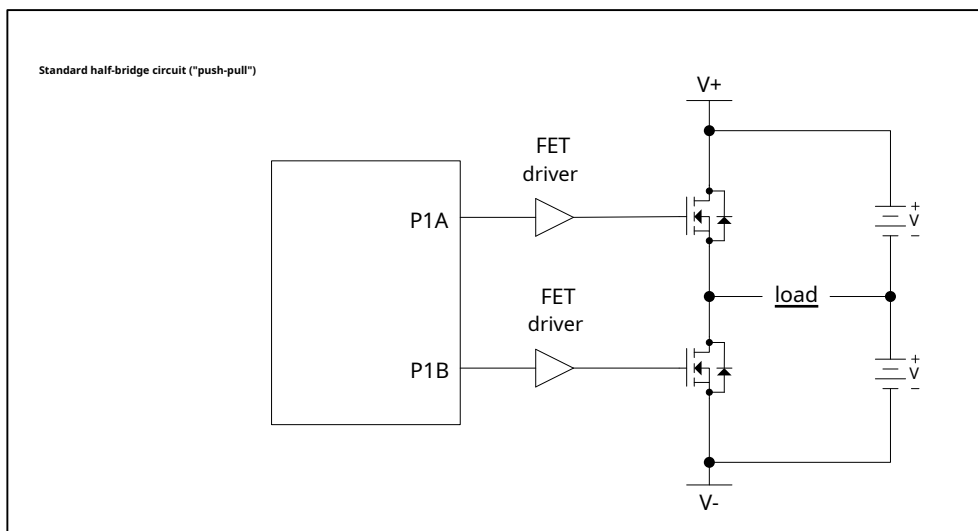
In half-bridge mode, programmed dead-band delays can be used to prevent shoot-through current in the half-bridge power devices. PWM1CON Register PDC<6:0> The bits are used to set the number of instructions before the output is driven active. If this value is greater than the duty cycle, the corresponding output will remain inactive for the entire cycle. See the [No.8.6.6 Section "Programmable Dead-Band Delay Mode"](#) .

because P1A and P1B Output and PORT Data latch multiplexing must clear the associated CPIO Position P1A and P1B Configured as output.

picture8-6-1-1: Half Bridge PWM Output example



picture8-6-1-2: Half-bridge application example



8.6.2 Full-bridge mode

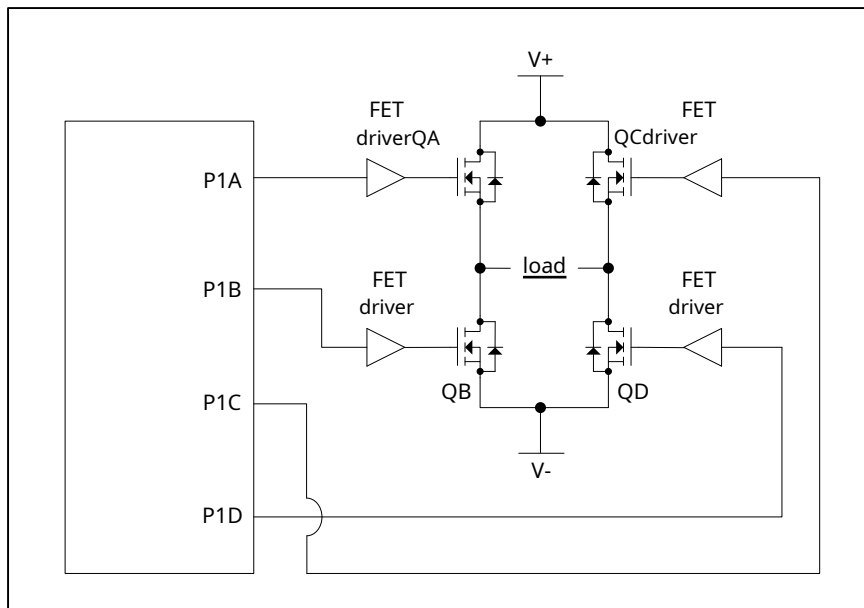
In full-bridge mode, all four pins function as outputs. [picture8-6-2-1](#) An example full-bridge application is shown.

In forward mode, CCP1/P1A The pin is driven to the active state. P1D pin is the modulation output, and P1B and P1C is driven to an invalid state, such as [picture8-6-2-2](#) shown.

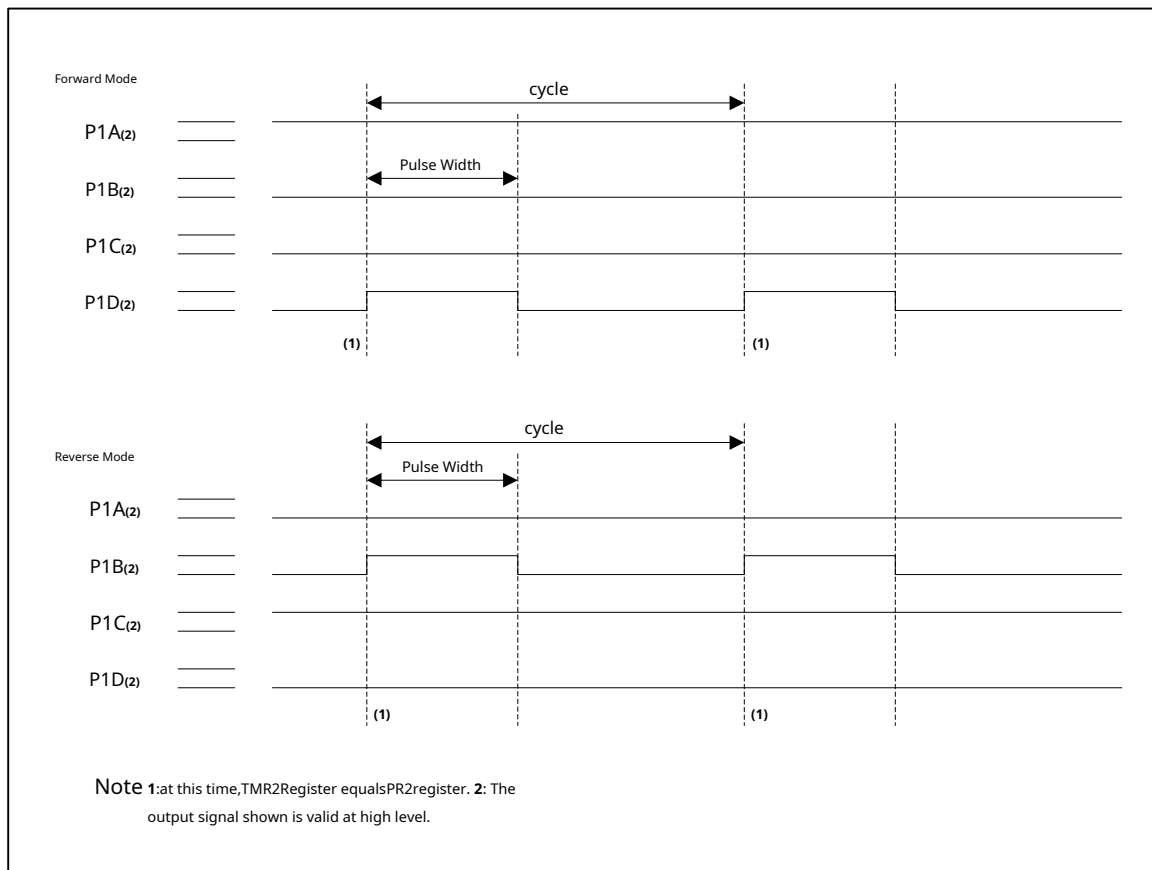
In reverse mode, P1C The drive is in a valid state. P1B pin is the modulation output, and P1A and P1D is driven to an invalid state, such as [picture 8-6-2-2](#) shown.

P1A, P1B, P1C and P1D Output and PORTData latch multiplexing. The relevant CPIO Position P1A, P1B, P1C and P1D Pins are configured as outputs.

picture8-6-2-1: Full-bridge application example



picture8-6-2-2: Full BridgePWMOutput example



8.6.3 Startup Considerations

Use any PWM mode, the application hardware must use appropriate external pull-up/pull-down resistors on output pins.

Note: When the microcontroller exits from reset, all I/O pins are all in high impedance state. I/O pin or Activate PWM Before output, the external circuit must put the power switch in the off state.

CCP1CON Register CCP1M<1:0> bits are user selectable per pair PWM output pins (P1A/P1C and P1B/P1D) is high level active or low level active. PWM The output polarity must be enabled PWM We do not recommend PWM Do not change the polarity configuration while the pin's output driver is enabled as this may damage the application circuit.

exist PWM When the module is initialized, P1A, P1B, P1C and P1D The output latch may not be in the correct state. PWM The pin output drivers are enhanced PWM Modes enabled simultaneously may result in damage to the application circuit. PWM mode must be enabled in the correct output mode and PWM The pin output driver completes a full PWM cycle. A whole PWM Whether the cycle is completed can be checked by PIR1 Register TMR2IF In the second PWM Is it set at the beginning of the cycle? 1 Check.

8.6.4 Enhanced PWM Auto-off mode

The module supports an auto-shutdown mode, which disables the PWM output. The auto-off mode will place the output pin in a predetermined state. This module is used to prevent PWM damage application.

The auto-shutdown source can be selected by the bit. A shutdown event can be generated by:

- INT pin logic appears
- Comparators C1
- Comparators C2
- In the firmware, ECCPASE Location 1

The closed state is ECCPASE (Auto-shutdown event status) bit indication. If this bit is 0, PWM pin operates normally. If this bit is 1, PWM output is in the OFF state.

When a shutdown event occurs, two things happen:

ECCPASE Location 1. ECCPASE Keep 1 status until it is cleared by the firmware or an automatic restart occurs (see [No. 8.6.5 Section "Auto Restart Mode"](#)).

The pin is asynchronously placed in its off state. The output pins are divided into two pairs [P1A/P1C] and [P1B/P1D]. The status of the two pairs of pins is determined by PSSAC and PSSBD. Each pair of pins can be placed in one of three states:

- Driven by logic 1
- Driven by logic 0
- Tri-state (high impedance)

register 17H: Enhanced Capture/Compare/PWM Auto Shutdown Control Register (ECCPAS)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
bit 7							bit 0

Figure caption:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as 0

-n = PORT value of

1 = Place 1

0 = Clear

x = unknown

bit 7 **ECCPASE:** ECCP Auto-shutdown event status bit
 1 = An automatic shutdown event occurred; ECCP output is in OFF state
 0 = ECCP output works normally

bit 6-4 **ECCPAS<2:0>:** ECCP Auto-shutdown source select bit
 000 = Disable automatic shutdown
 001 = Comparators 1 Output changes
 010 = Comparators 2 Output changes
 011 = Comparators 1 or 2 One of the changes
 100 = INT The pin voltage is VIL
 101 = INT The pin voltage is VIL or comparator 1 Variety
 110 = INT The pin voltage is VIL or comparator 2 Variety
 111 = INT The pin voltage is VIL or comparator 1/2 One of the changes

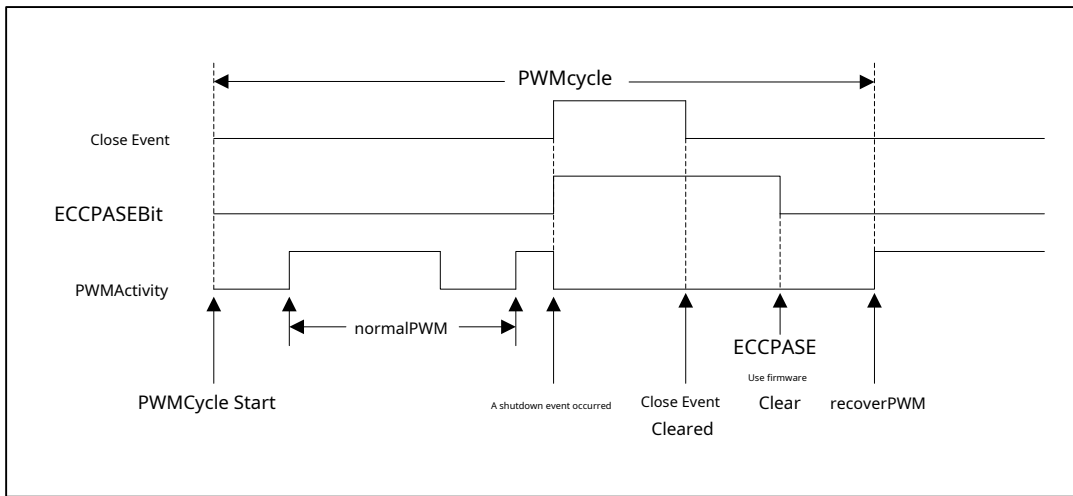
bit 3-2 **PSSAC:** P1A and P1C pin shutdown status control bit
 00 = Drive pin P1A and P1C for 0
 01 = Drive pin P1A and P1C for 1
 1x = P1A and P1C pins are tri-stated

bit1-0 **PSSBD**:P1BandP1DPin shutdown status control bit
 00 =Drive pinP1BandP1Dfor0 01 =Drive pinP1B
 andP1Dfor1 1x = P1BandP1DPins are tri-stated

Note:

- 1: The auto-off condition is based on the level signal, not the edge signal. As long as the level does not change, the auto-off does not change.
- 2: Write disabled under auto-close conditionECCPASEBit.
- 3: Once the auto-shutdown condition is cleared andPWMReboot (either by firmware or automatically),PWMThe signal will always be on the nextPWMweek
Restart at the beginning of the period.

picture8-6-4: Firmware restartPWMAutomatic shutdown (PRSEN=0)

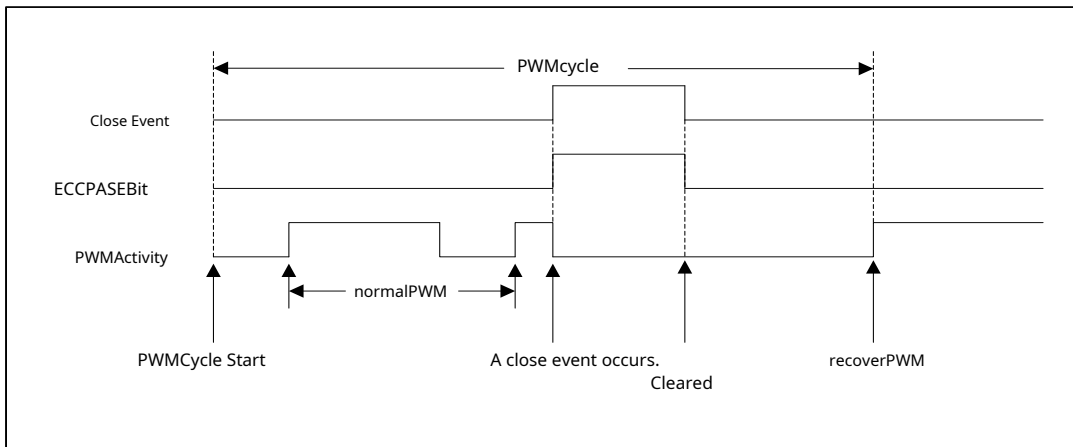


8.6.5 Automatic restart mode

EnhancedPWMConfigurable to automatically restart when auto-shutdown condition is clearedPWMsignal.PWM1CONRegister PRSEN Location1Automatic restart can be enabled.

When automatic restart is enabled, as long as the automatic shutdown condition is valid,ECCPASEKeep the position1. When the auto-close condition is cleared,ECCPASE The bit will be cleared by hardware, resuming normal operation.

picture8-6-5: When automatic restart is enabledPWMAutomatic shutdown (PRSEN=1)



8.6.6 Programmable dead-band delay mode

All power switches are modulated PWM in a half-bridge application with a low frequency, it usually takes a long time for the power switch to switch on from off. If the upper and lower power switches are activated at the same time (one is on and the other is off), before one switch is completely turned off, the two switches may be in a short period of time. During this short period of time, a very high current (punch-through) will flow through the two power switches. To avoid this destructive shoot-through current during switching, the on-time of either power switch is usually delayed to allow the other switch time to completely turn off.

In half-bridge mode, digitally programmable dead-band delays are used to prevent shoot-through current from damaging the bridge power switches. A delay occurs when a signal changes from an inactive state to an active state. See [picture 8-6-1-2](#) Related PWM1CON register ([register 16H](#)) is used as the instruction cycle of the microcontroller (TCY, Right now 4TOSC) to set the delay period for the unit.

register 16H: Enhanced PWM Control Register (PWM1CON)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0
bit7							bit0

Figure caption:

R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORThe value of 1 = Place 1 0 = Clear x = unknown

bit7 PRSEN: PWM Restart enable bit
 1 = When automatically closed, ECCPASE bit is automatically cleared on exit shutdown event) PWM Automatic restart
 0 = When the software is turned off, ECCPASE clear to restart PWM

bit6-0 PDC<6:0>: PWM Delay count bit
 PDCn = Reserve PWM The signal should turn valid and PWM The signal actually becomes valid FOSC/4(4*TOSC) Number of cycles

Note 1: Quick start and selected LP, XT or HS When the fault protection mode is enabled, this bit is reset to 0.

surface 8-6-6: capture/compare/PWM Summary of related registers

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR and BOR The value of	All other Reset value
CCPR1L	Capture/Compare/PWM register 1 The low byte								xxxx xxxx	uuuu uuuu
CCPR1H	Capture/Compare/PWM register 1 High Byte								xxxx xxxx	uuuu uuuu
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
CMSTA	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	----- 10	----- 10
ECCPAS	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000 0000	0000 0000
INTS	GIE	PEIE	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
PIEB1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIFB1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMR1IF	0000 0000	0000 0000
PR2	Timer 2 Module Period Register								1111 1111	1111 1111
PWM1CON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000	0000 0000
T1STA	T1GINV	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	0000 0000	uuuu uuuu
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	- 000 0000	- 000 0000
TMR1L	16Bit TMR1 Low Byte Holding Register								xxxx xxxx	uuuu uuuu
TMR1H	16Bit TMR1 High Byte Holding Register								xxxx xxxx	uuuu uuuu
TMR2	Timer 2 Module Registers								0000 0000	0000 0000
CPIOA	—	—	CPIOA5	CPIOA4	CPIOA3	CPIOA2	CPIOA1	CPIOA0	- -11 1111	- -11 1111
CPIOC	CPIOC7	CPIOC6	CPIOC5	CPIOC4	CPIOC3	CPIOC2	CPIOC1	CPIOC0	1111 1111	1111 1111

Figure caption: x = unknown, u = unchanged, — = unimplemented (read as 0) . Capture/Compare/PWM Shaded cells are not used.

9.0 Analog-to-digital converter (ADC) module

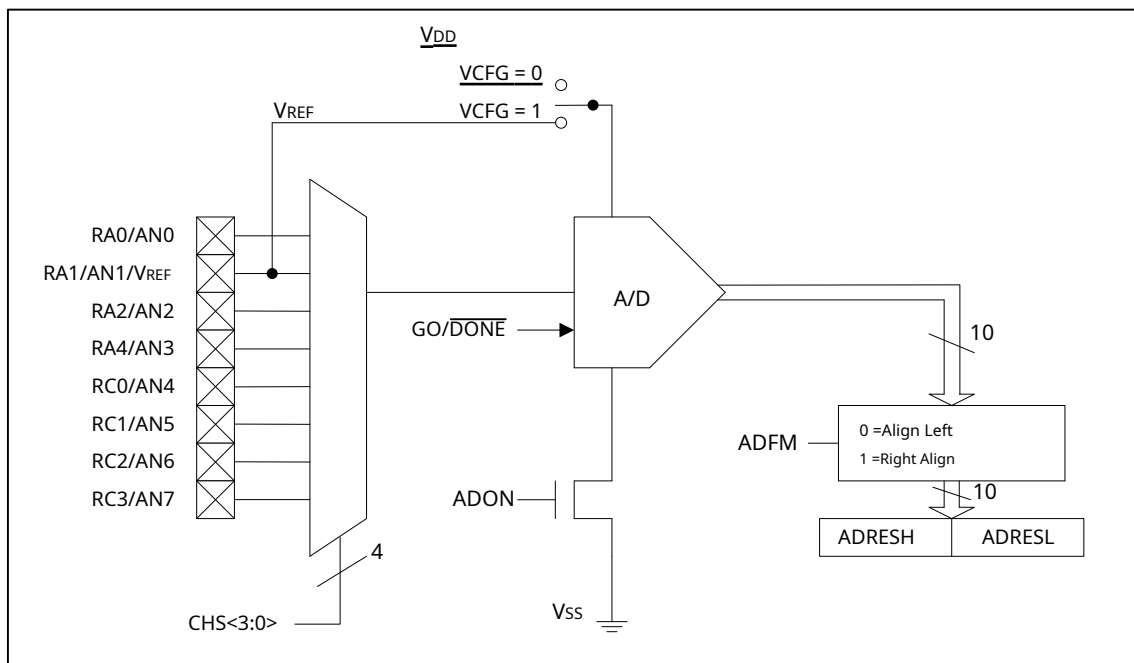
Analog-to-digital converter (Analog-to-digital Converter, ADC) can convert the analog input signal into the corresponding 10-bit digital value. This series of devices uses multiple analog inputs to multiplex into a sample-and-hold circuit. The output of the sample-and-hold circuit is connected to the input of the converter. The converter generates the 10-bit digital value. The converted binary value is stored in ADC Result Register (ADRESL:ADRESH) middle.

The reference voltage can be selected by software as V_{DD} or the voltage applied to the external reference pin.

An interrupt can be generated when a conversion is complete. This interrupt can be used to wake the device from Sleep.

9.1 picture: ADC block diagram

picture9-1: ADC block diagram



9.2 ADC Configuration

The following features must be considered:

- Port Configuration
- Channel selection
 - ADC Reference voltage selection
 - ADC Convert clock source
- Interrupt control
- The format of the conversion result

9.2.1 Port Configuration

ADC can be used to convert analog and digital signals. When converting analog signals, the relevant pin should be configured as an analog function. See the corresponding port chapter for more information.

Note: The presence of analog voltages on pins defined as digital inputs can cause the input buffers to conduct excessive current.

9.2.2 Channel Selection

Register CHSThe 1/2 bit determines which channel is connected to the sample-and-hold circuit.

When changing channels, a delay is required before starting the next conversion. For more information, see [No.9.3 "ADCHow it works"](#).

9.2.3 ADC Reference voltage

Register ADVRS<1:0>The 0x00 bit provides control of the positive reference voltage.

The positive reference voltage can be:

- Select power supply voltage VDD as ADC's reference.
- Select External (from PA1) as ADC's reference.
- Internal reference voltage selected via configuration register 2V, 3V, 4V (calibrated to 1%).

The negative reference voltage is always connected to the reference ground.

9.2.4 Conversion clock

The conversion clock source can be selected by software, by setting Register ADCSThere are seven clock options:

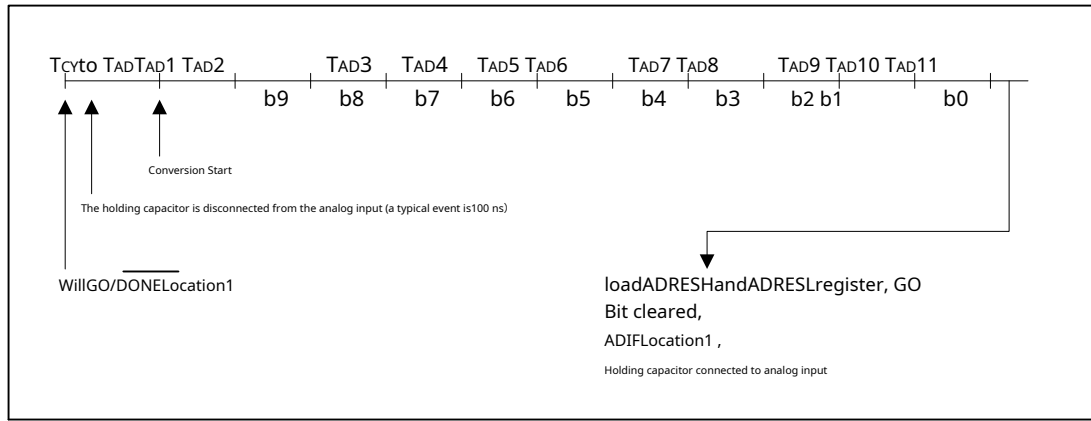
- FOSC/2
- FOSC/4
- FOSC/8
- FOSC/16
- FOSC/32
- FOSC/64
- F_{SM} (Dedicated internal oscillator)

Complete one (bit) is defined as T_{AD}. 10Bit conversion requires 11 individual T_{AD} Cycle, such as [picture9-2-4](#) shown.

The correct conversion must satisfy the corresponding T_{AD} For more information, see [No.13.0 Section "Electrical Characteristics"](#) middle/AD Conversion requirements. [surface 9-2-4](#) Correct selection shown ADCExample of a clock.

Note: Unless you are using F_{SM} Otherwise, any change in the system clock frequency will change ADCThe clock frequency will ADCThe results had a negative impact.

picture9-2-4: Analog-to-digital conversion T_{AD} cycle



surface9-2-4:ADCClock cycle (TAD)—Device operating frequency (Vdd≥3.0V)

ADCClock cycle (TAD)		Device frequency (Fosc)			
ADCClock Source	ADCS<2:0>	20 MHz	8 MHz	4 MHz	1 MHz
Fosc/2	000	100 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	2.0 μs
Fosc/4	100	200 ns ⁽²⁾	500 ns ⁽²⁾	1.0 μs ⁽²⁾	4.0 μs
Fosc/8	001	400 ns ⁽²⁾	1.0 μs ⁽²⁾	2.0 μs	8.0 μs ⁽³⁾
Fosc/16	101	800 ns ⁽²⁾	2.0 μs	4.0 μs	16.0 μs ⁽³⁾
Fosc/32	010	1.6 μs	4.0 μs	8.0 μs ⁽³⁾	32.0 μs ⁽³⁾
Fosc/64	110	3.2 μs	8.0 μs ⁽³⁾	16.0 μs ⁽³⁾	64.0 μs ⁽³⁾
F _{SM}	x11	2-6 μs ^(1,4)	2-6 μs ^(1,4)	2-6 μs ^(1,4)	2-6 μs ^(1,4)

Figure caption: Shaded cells are outside the recommended range.

Note

- 1: exist VDD > 3.0V in the case of F_{SM} typical clock sources TAD Time is 4 μs.
- 2: These values violate the minimum TAD time requirement.
- 3: To speed up the conversion, it is recommended to select another clock source.
- 4: When the device frequency is greater than 1 MHz it is recommended to use it only when the conversion is done during sleep. F_{SM} Clock source.

9.2.5 ADC Module interrupt

ADC When the conversion is complete, if ADC Interrupt flag ADIE Enable and global interrupt flag GIE If it is also enabled, it will generate ADC Interrupt, enter the interrupt service routine; in the interrupt service routine, the interrupt flag must be cleared by software ADIF (cleared to 0).

Note: regardless ADC whether interrupts are allowed, ADIF The bit is set at the completion of each conversion. 1.

ADC none F_{SM} Mode, need to use ADC When waking up, the system clock must be one of the internal fast/slow clocks.

The interrupt can be generated while the device is operating or in sleep mode. If the device is in sleep mode, the interrupt will wake the device. When waking from sleep mode, the interrupt will always execute SLEEP If the user attempts to wake up the device and resume sequential code execution, global interrupts must be disabled. If global interrupts are enabled, code execution will be transferred to the Interrupt Service Routine.

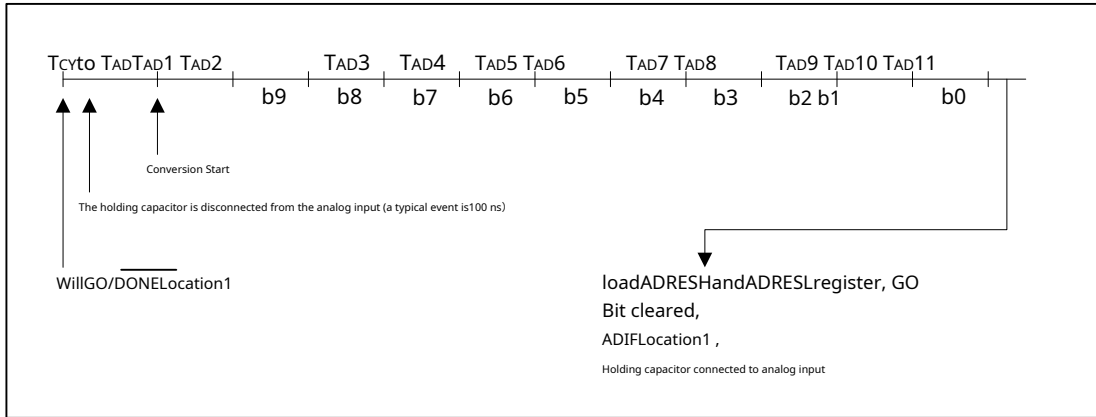
9.2.6 A/D Conversion result format

10 Bit A/D The conversion result has two formats, left-aligned and right-aligned. ADS0 Register ADFM bits control the output format.

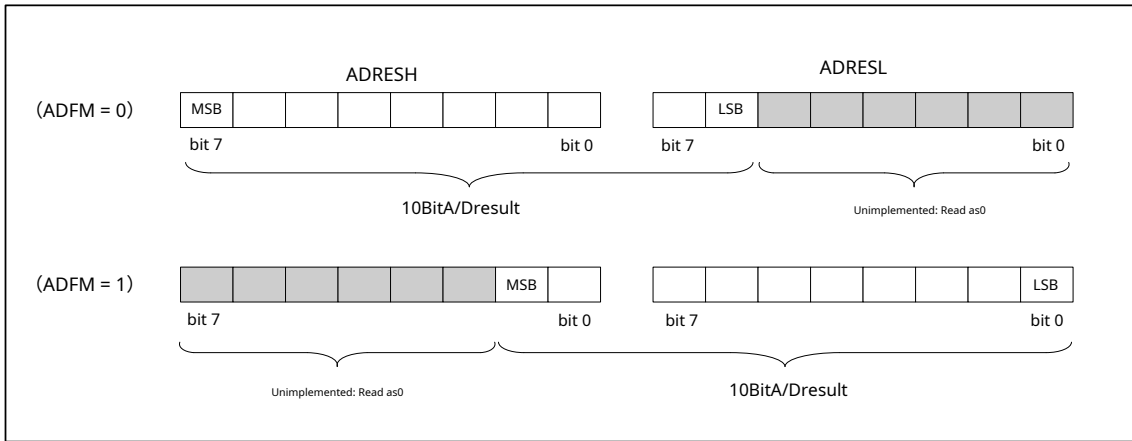


picture9-2-6-1 Two output formats are shown.

picture9-2-6-1: Analog-to-digital conversionTadcycle



picture9-2-6-2: Analog-to-digital conversionTadcycle



9.3 ADCHow it works

9.3.1Start the conversion

To enableADCmodule, you mustADS0RegisterADONLocation1.WillADS0RegisterGO/DONELocation1The analog-to-digital conversion will be started.

Note: See [No.9.3.5Festival"A/DConversion steps "](#).

9.3.2Conversion Complete

When the conversion is complete,ADCThe module will:

- WillGO/DONEBit clear
- WillADIFLogo location1
- Update with new transformation resultsADRESH:ADRESLregister

9.3.3 Termination Conversion

If the conversion must be terminated before completion, the available software will GO/DONE cleared. ADRESH:ADRESL The registers are not updated by pending analog-to-digital conversion samples. ADRESH:ADRESL This register pair will hold the values from the previous conversion. In addition, wait until 2 T_{AD} After the delay, input acquisition of the selected channel will start automatically.

Note: A device reset forces all registers back to their reset states. ADCThe module is closed and any pending conversions are terminated.

9.3.4 Special event triggers

ECCP Special event triggers can be performed periodically without software intervention ADC When a trigger event occurs, GO/DONE Set by hardware 1, Timer 1 The counter is reset to zero.

The use of special event triggers does not ensure normal ADCTiming. Ensure satisfaction ADCTiming requirements are the responsibility of the user.

For more information, see [No.8.0 Section "Enhanced Capture/Compare/PWM+\(With automatic shutoff and deadband\) module"](#).

9.3.5 A/D Conversion steps

The following is the use ADC Example of steps to perform an analog-to-digital conversion:

1. Configure ports:
 - Disable pin output drivers (see CPIO register)
 - Configure the pin as analog
2. Configuration ADC Modules:
 - choose ADC Conversion clock
 - Configure reference voltage
 - choose ADC Input Channels
 - Select the format of the conversion result
 - Open ADC Modules
3. Configuration ADC Interrupt (optional) :
 - Will ADC Interrupt flag cleared
 - allow ADC Interrupt
 - Enable peripheral interrupts
 - Enable global interrupts⁽¹⁾
4. Wait for the required acquisition time⁽²⁾
5. GO/ DONE Place 1 Start the conversion
6. Wait through one of the following situations ADC Conversion completed:
 - Inquire GO/ DONE Bit
 - wait ADC Interrupt (when interrupts are enabled)
7. Read ADC result
8. ADC Clear the interrupt flag (interrupts must be enabled) .

Note: 1: If the user attempts to wake the device from Sleep and execute code sequentially, global interrupts can be disabled.

2: See [No.9.5 Section "A/D Collection time requirements"](#).

example9-3-5:A/DConversion

```

;ADCConversion program snippet:
;
BSR    STATUS,PAGE    ;chooseBANK1
LDWI   B'01110000'   ;Select InternalRCAsADCClock Source
STWR   ADS1
BSR    CPIOA,0       ;set upPA0enter
BSR    ANSEL,0       ;set upPA0For analog port
BCR    STATUS,PAGE   ;chooseBANK0
LDWI   B'10000001'   ; Set the result to right-align, enableADC,aisle0,VDDFor reference
STWR   ADS0
LCALL  SampleTime    ; Delay for a short time
BSR    ADS0,GO       ; Start conversion
BTSC   ADS0,GO       ;waitADCConversion Complete
LJUMP  $-1           ;
LDR    ADRESH,W      ; Read the result high2Bit
STWR   RESULTHI
BSR    STATUS,PAGE   ;Bank 1
LDR    ADRESL,W      ;Read result low8Bit
STWR   RESULTLO
BCR    STATUS,PAGE
BCR    ADS0,ADON     ;closureADC

```

9.4 ADCRegister Definition

The following registers are used to controlADCwork.

9.4.1 ADS0:A/DControl Register0

register1FH:A/DControl Register0 (ADS0)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	ADVRS1	ADVRS0	CHS2	CHS1	CHS0	GO/DONE	ADON
bit7						bit0	

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
-n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7 **ADFM**:A/DConversion result format selection bit 1
=Right Align
0 =Align Left

bit6-5 **ADVRS<6:5>**:chooseADCReference voltage 00 =Select
power supply voltageVDDAsADCReferences 10 =Select
External (fromPA1) asADCReferences
X1 =Internal reference voltage, this voltage is selected by the configuration register2V,3V,4V.

bit4-2 **CHS<2:0>**: Analog channel selection bit
 000 = AN0
 001 = AN1
 010 = AN2
 011 = AN3
 100 = AN4
 101 = AN5
 110 = AN6
 111 = AN7

bit1 **GO/DONE**: A/D Conversion status bit
 1 = A/D Conversion in progress. 1 Start once A/D Convert.

A/D This bit is automatically cleared by hardware when the conversion is complete.

0 = A/D Conversion completed/not in progress

bit0 **ADON**: ADC Enable bit
 1 = Enable ADC
 0 = prohibit ADC, does not consume working current

9.4.2 ADS1: A/D Control Register1

register9F: A/D Control Register1 (ADS1)

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
ADRVO	ADCS2	ADCS1	ADCS0	—	—	—	—
bit7							bit0

Figure caption:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as 0

-n = PORThe value of

1 = Place 1

0 = Clear

x = unknown

bit7 **ADRVO**: Control whether PA1 Output internal reference voltage (Note: this voltage has no driving capability) 1 = sent to PA1 port
 0 = Not sent to PA1 port.

bit6-4 **ADCS<2:0>**: A/D Conversion clock select
 bits 000 = FOSC/2
 001 = FOSC/8
 010 = FOSC/32

x11 = Fsm Intelligent frequency division, ADCThe conversion clock will select the division ratio according to the oscillation mode:

When the system clock selects LPorRCorRCIOMode, ADCThe conversion clock is the system clock 2 Frequency division.

When the system clock selects XTMode, ADCThe conversion clock is the system clock 8 When the

system clock selects HFMode, ADCThe conversion clock is the system clock 32 Frequency division.

100 = FOSC/4

101 = FOSC/16

110 = FOSC/64

bit3-0 Unimplemented: Read as 0

How to output the internal reference voltage to PA1 On the port (this voltage has no driving capability and can only be used as a test internal reference voltage):

- (1) set up PA1 For analog, input port;
- (2) set up ADS1 First 7 Bit ADRVO for 1;
- (3) set up ADS0 First 5 Bit ADVRS0 for 1;
- (4) Enable ADC;

9.4.3 ADRESH:ADCResult High Register

register1EH:ADCResult High Register (ADRESH),ADFM = 0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADRES9	ADRES8	ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7-0 ADRES<9:2>:ADCResult register bits,10The high bit of the conversion result8Bit

register1EH:ADCResult High Register (ADRESH),ADFM = 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
—	—	—	—	—	—	ADRES9	ADRES8
bit7						bit0	

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7-2 Reserved: Do not use

bit1-0 ADRES<9:8>:ADCResult register bits,10The high bit of the conversion result2Bit

9.4.4 ADRESL:ADCResult low register

register9E:ADCResult low register (ADRESL),ADFM = 0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADRES1	ADRES0	—	—	—	—	—	—
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7-6 ADRES<1:0>:ADCResult register bits,10The low bit of the conversion result2Bit

bit5-0 reserved: Do not use

register9E:ADCResult low register (ADRESL),ADFM = 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
bit7							bit0

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7-0 ADRES<7:0>:ADCResult register bits,10The low bit of the conversion result8Bit

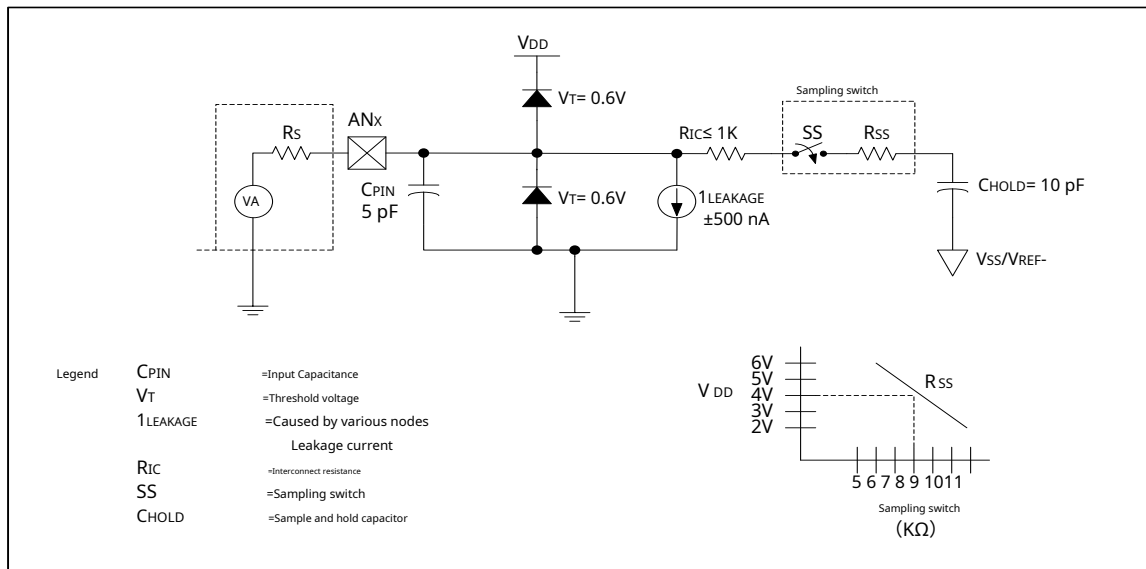
9.5 A/D Collection time requirements

because ADCTo achieve the specified accuracy, the charge holding capacitor (CHOLD) to the input channel level. For analog input models, see [picture9-5-1](#) . Source Impedance (RS) and the internal sampling switch (RSS) Impedance directly affects capacitanceCHOLDThe charging time of the sampling switch (RSS) Impedance varies with device voltage (VDD), see [picture9-5-1](#) The maximum impedance of the analog signal source is recommended to be 10kΩThe acquisition time decreases as the source impedance decreases. After selecting (or changing) the analog input channel, the acquisition must be completed before starting the conversion. [formula9-5](#) To calculate the minimum acquisition time. This formula assumes that the error is 1/2 LSB (ADConversion required 1024step). 1/2LSbThe error is ADCThe maximum error allowed to achieve the specified accuracy.

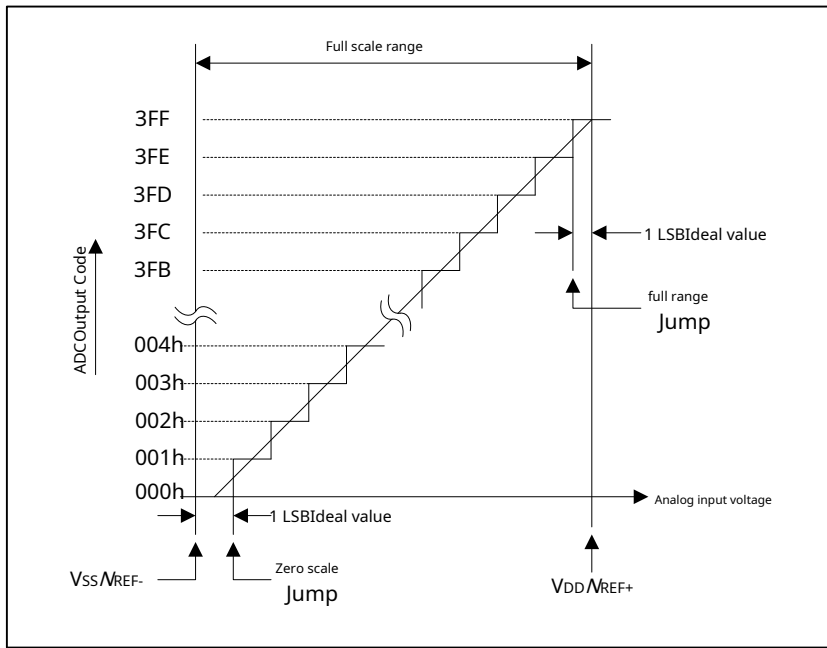
Note:

- 1: Since the reference voltage (VREF) eliminates itself and therefore has no effect on the formula.
- 2: Charge holding capacitor (CHOLD) is not discharged after each conversion.
- 3: The maximum recommended source impedance for the analog input is 10kΩ. This is to meet the pin leakage specification.

picture9-5-1: Simulate input model



picture9-5-2:ADCTransfer Function



formula9-5: Sample time example

假设 温度 = 50°C, 外部电阻 10 kΩ, 5.0V VDD

$$\begin{aligned}
 T_{ACQ} &= \text{放大器稳定时间} + \text{保持电容充电时间} + \text{温度系数} \\
 &= T_{AMP} + T_C + T_{COFF} \\
 &= 5 \mu\text{s} + T_C + [(\text{温度} - 25^\circ\text{C}) (0.05 \mu\text{s}/^\circ\text{C})]
 \end{aligned}$$

Tc 值可以用以下公式近似计算:

$$V_{APPLIED} \left(1 - \frac{1}{2047} \right) = V_{CHOLD} \quad ; [1] \text{ 在 } 1/2 \text{ lsb 误差范围内对 } V_{CHOLD} \text{ 充电}$$

$$V_{APPLIED} \left(1 - e^{-\frac{T_C}{RC}} \right) = V_{CHOLD} \quad ; [2] \text{ 依照 } V_{APPLIED} \text{ 对 } V_{CHOLD} \text{ 充电}$$

$$V_{APPLIED} \left(1 - e^{-\frac{T_C}{RC}} \right) = V_{APPLIED} \left(1 - \frac{1}{2047} \right) \quad ; \text{结合 } [1] \text{ 和 } [2]$$

求解 Tc:

$$\begin{aligned}
 T_C &= -CHOLD / (R_{IC} + R_{SS} + R_S) \ln(1/2047) \\
 &= -10 \text{ pF} / (1 \text{ k}\Omega + 7 \text{ k}\Omega + 10 \text{ k}\Omega) \ln 0.004885 \\
 &= 1.37 \mu\text{s}
 \end{aligned}$$

因此

$$\begin{aligned}
 T_{ACQ} &= 5 \mu\text{s} + 1.37 \mu\text{s} + [(50^\circ\text{C} - 25^\circ\text{C}) (0.05 \mu\text{s}/^\circ\text{C})] \\
 &= 7.67 \mu\text{s}
 \end{aligned}$$

surface9-5:andADCSummary of related registers

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	PORandBOR	All other
									The value of	Reset value
ADS0	ADFM	ADVRS1	ADVRS0	CHS2	CHS1	CHS0	GO/DONE	ADON	0000 0000	0000 0000
ADS1	ADRVO	ADCS2	ADCS1	ADCS0	—	—	—	—	0000 ----	0000 ----
ANINS	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
ADRESH	A/DResult register high byte								xxxx xxxx	uuuu uuuu
ADRESL	A/DResult register low byte								xxxx xxxx	uuuu uuuu
INTS	GIE	PEIE	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
PIEB1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIFB1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMR1IF	0000 0000	0000 0000
PORTA	—	—	PA5	PA4	PA3	PA2	PA1	PA0	- -x0x000	- -uu uuuu
PORTC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	- -xx 0000	- -uu uuuu
CPIOA	—	—	CPIOA5	CPIOA4	CPIOA3	CPIOA2	CPIOA1	CPIOA0	- -11 1111	- -11 1111
CPIOC	CPIOC7	CPIOC6	CPIOC5	CPIOC4	CPIOC3	CPIOC2	CPIOC1	CPIOC0	1111 1111	1111 1111

Figure caption: x =unknown,u =unchanged, — = unimplemented (read as0) .ADCShaded cells are not used by the module.

10.0 Comparator Module

Comparators are used to interface analog circuits with digital circuits by comparing two analog voltages and providing a digital value that represents their relative magnitude.

Comparators are very useful for mixed-signal building blocks because they provide analog functionality independent of program execution. The analog comparator module includes the following features:

- Independent comparator control
- Programmable input selection
- Internal/external comparator output
- Programmable output polarity
- Interrupt on Change
- Wake up from sleep
- PWMclosure
- Timer1Gate control (count enable)
- Output andTimer1Clock input synchronization
- SRLatches
- Programmable and fixed reference voltages

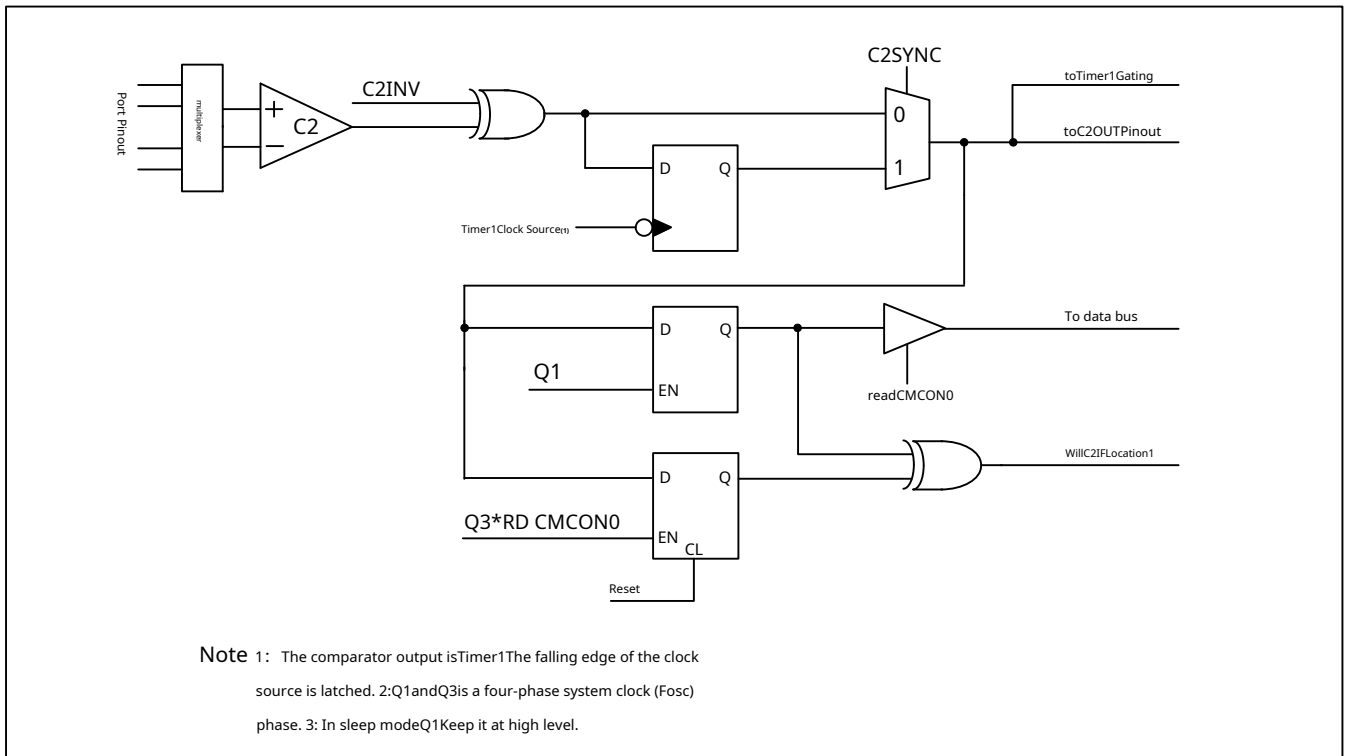
Note: Only the comparatorC2Connect toTimer1.

10.1 Comparator Overview

[picture10-1-1](#) Shown is a single comparator and the relationship between the analog input levels and the digital output.VIN+The analog voltage is less thanVIN-When the analog voltage is higher than , the comparator outputs a digital low level.VIN+The analog voltage is greater thanVIN-When the analog voltage is higher than , the comparator outputs a digital high level.

picture10-1-1: Single comparator

picture10-1-3:ComparatorsC2Output Block Diagram



10.1.1 Analog Input Connection Considerations

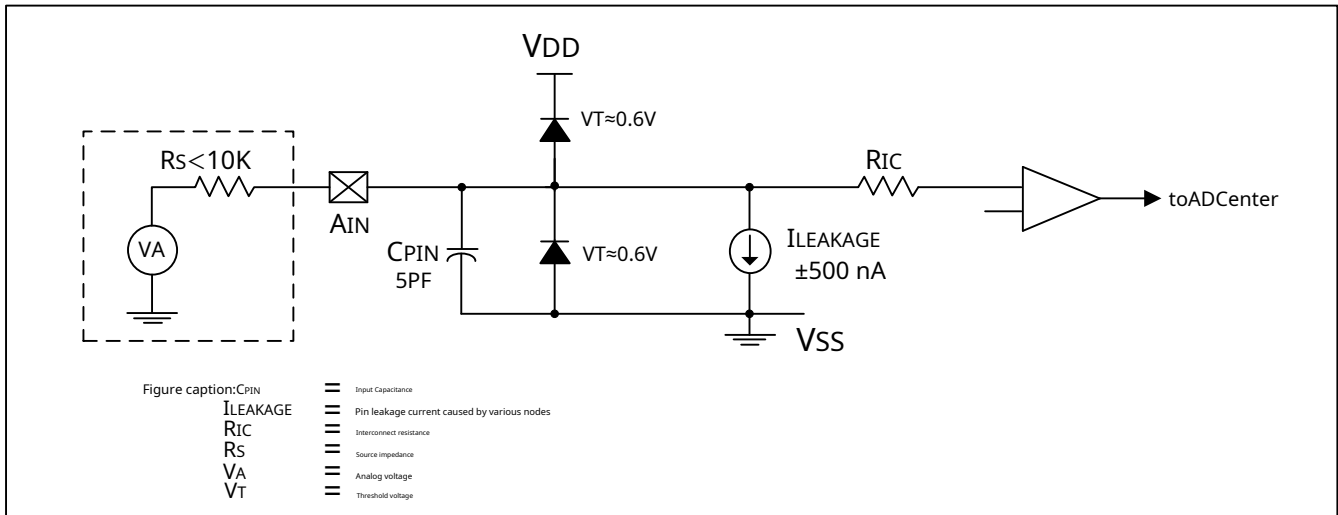
[picture10-1-1-1](#) A simplified circuit for the analog inputs is shown. Since the analog input pins share connections with the digital inputs, they have VDDandVSSReverse biasESDprotection diode. Therefore, the analog input voltage must be betweenVSSandVDDIf the input voltage deviates from this range by more than0.6V, one of the diodes will become forward biased and latch-up may occur.

It is recommended to connect a maximum of10kΩIn addition, external components connected to the analog input pins, such as capacitors or Zener diodes, should have very little leakage current to minimize the introduction of errors.

Note:1: ReadPORTregisters, all pins configured as analog inputs will read as0According to the input specification, the input pin will be converted into an analog input.

2: The presence of analog levels on any pin defined as a digital input will cause the input buffer to consume current beyond the specification.

picture10-1-1-1: Simulate input model



10.2 Comparator Configuration (Figure)

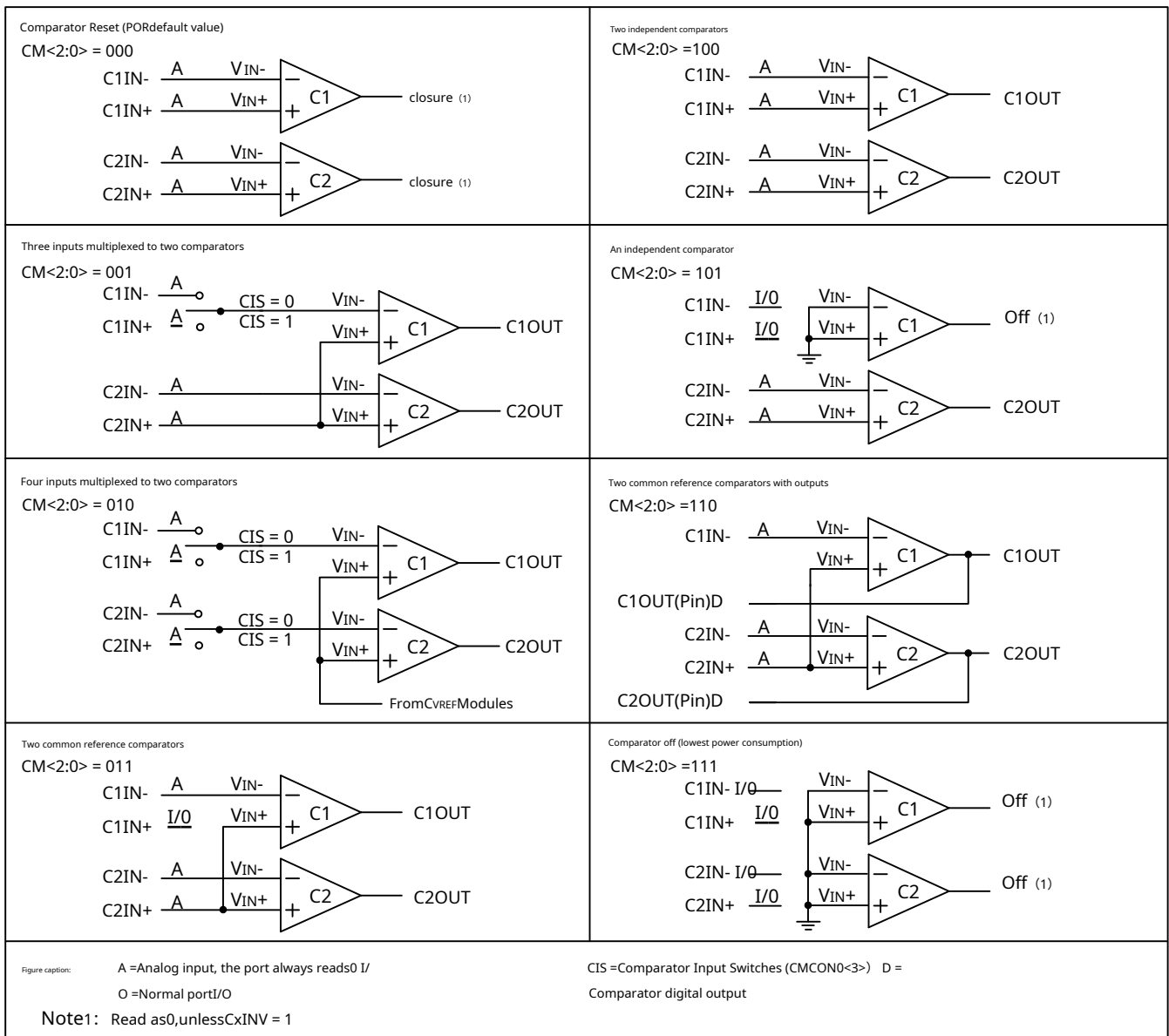
The comparator has eight operating modes. CMSTARRegisterCM<2:0>Bits are used to select these modes, such as [picture10-2](#) shown. I/OThe lines change as the mode changes and are assigned the following functions:

- Simulation function (A): Digital input buffer is disabled
- Digital functions (D): Comparator digital output, covering port function
- Normal port functionality (I/O): Not related to the comparator

Expressed as "A" The port pins will read 0, regardless I/OPin or I/OcontrolCPIOThe pins used as analog inputs should also have their corresponding CPIOLocation1To disable the digital output driver. D"The pins should be connected to their corresponding CPIOClear this bit to enable the digital output drivers.

Note: The comparator interrupts should be disabled during comparator mode changes to prevent unexpected interrupts.

picture10-2:ComparatorsI/OWorking Mode



10.3Comparator Control

CMSTAreger([register19H](#)) provides control over the following comparator functions:

- Mode selection
- Output status
- Output polarity
- Input switch

10.3.1 Comparator output status

The status of each comparator can be read from the relevant register $CMxOUT$. The bit is read internally. When the comparator outputs after selecting this mode, the related $CMxOUT$ pin must be cleared to enable the output drivers.

10.3.2 Comparator output polarity (table)

Inverting the comparator output has the same effect as swapping the comparator inputs. The comparator output polarity can be changed by $CMxCON0$ register $CxPOL$. Flipping $CxPOL$ clearing the bit to 0 will produce a non-inverting output. [surface10-3-2](#) Shown is the output state compared to the input conditions including polarity control.

surface10-3-2: Comparator output state - input condition

Input conditions	CxI	CxOUT
$V_{IN-} > V_{IN+}$	0	0
$V_{IN-} < V_{IN+}$	0	1
$V_{IN-} > V_{IN+}$	1	1
$V_{IN-} < V_{IN+}$	1	0

Note: $CxOUT$ refers to register bits and output pins.

10.3.3 Comparator Input Switch

The comparator's toggling input can be switched between analog pins in the following modes:

- $CM<2:0> = 001$ (Comparator only C1)
- $CM<2:0> = 010$ (Comparators C1 and C2)

In the above modes, both pins are in analog mode regardless of which pin is selected as input. $CMSTA$ register $CIST$ bit controls the comparator input switch.

10.4 Comparator response time

After a comparator input source changes or a different reference voltage is selected, the comparator output is uncertain for a period of time. This period of time is called the response time. The comparator response time varies with the reference voltage settling time. Therefore, both times must be taken into account when determining the total response time to a comparator input change. For more information, see [No.13.0 Section "Electrical Characteristics"](#) Medium comparator and reference voltage specifications.

10.5 How Comparator Interrupts Work

Whenever the comparator output value changes, the comparator interrupt flag can be set. The change is detected by a mismatch circuit consisting of two latches and an XOR gate (see [picture10-1-2](#) and [picture10-1-3](#)) is implemented. When reading $CMxSTA0$ when the comparator output level is read, a latch is updated. The latch holds the value until the next read. $CMxSTA0$ The other latch of the mismatch circuit is reset by each system clock. Q1 Update. Q1 Clock cycle, when the change of comparator output is sent to the second latch, a mismatch condition will be generated. At this time, the two mismatched latches have opposite output levels, which are detected by the XOR gate and sent to the interrupt circuit. The mismatch condition will last until $CMxSTA0$ register is read or the comparator output returns to its previous state.

Note: rightCMSTAA write operation to the register will also clear the mismatch condition because all write operations include a read operation at the beginning of the write cycle.

The software needs to save information about the state of the comparator output to determine the actual change that occurred.

PIR1RegisterCxFbit is the comparator interrupt flag. This bit must be cleared to zero by software to reset. Since this register can be written, it is possible that a software interrupt will be generated.

MustPIEB1RegisterCxIbit andINTSRegisterPEIEandGIEAll positions are set1To enable the comparator interrupt. If any of the above bits are cleared, the interrupt is not enabled, but when the interrupt condition occurs,PIR1RegisterCxFTThe bit will still be set1.

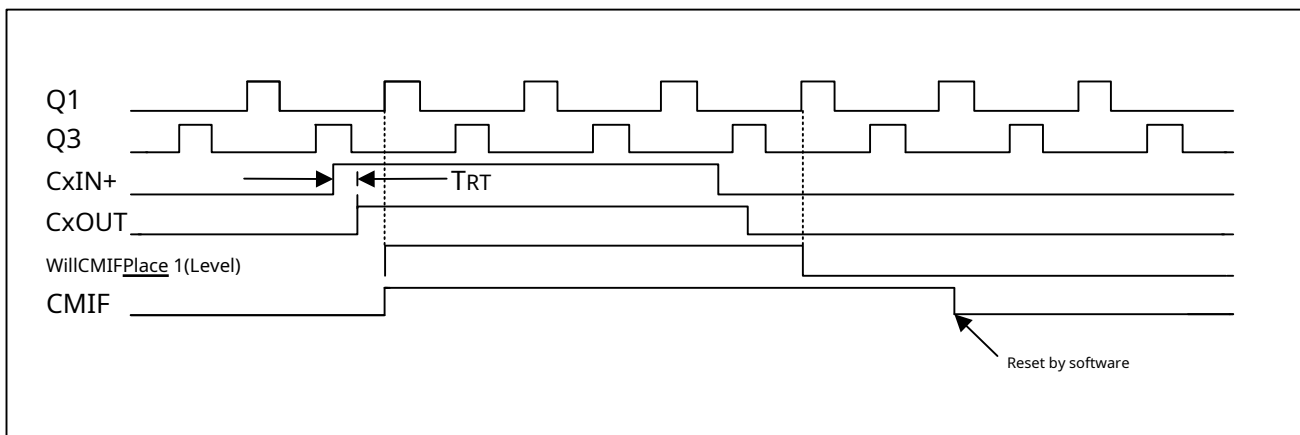
The user can clear the interrupt in the interrupt service routine by the following methods:

a)rightCMSTAThis will end the mismatch condition.

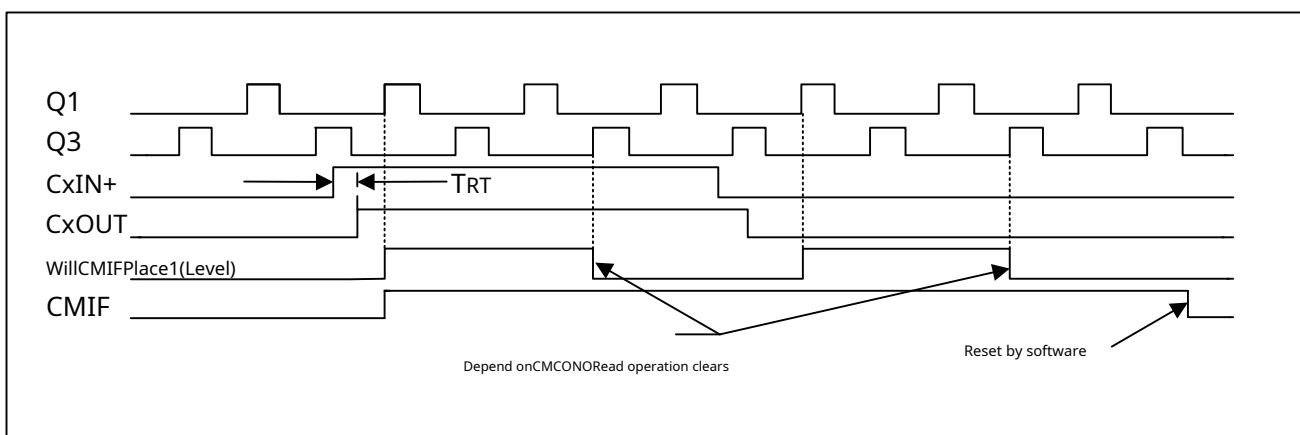
b)The interrupt flagCxFCleared.

If the mismatch condition persists,CxFTThe interrupt flag will not be cleared.CMSTAWill end the unmatched condition and allowCxFTThe bit is cleared zero.

picture10-5-1: Not readCMSTAComparator interrupt timing at



picture10-5-2: ReadCMSTAComparator interrupt timing at



Note:1: If you have to change while a read operation is in progressCMxSTA0register(CxOUT)(Q2The start of the cycle), thenPIR1register ofCxFTThe interrupt flag may not be set.1.

2: When any comparator is enabled first, the bias circuit in the comparator module can produce invalid comparator output until the bias circuit stabilizes to There should be1usThe bias circuit stabilizes in time and then clears the mismatch condition before enabling the comparator interrupt and clearing the interrupt flag.

10.6 How sleep works

If the comparator is enabled before entering sleep mode, it will remain active during sleep. The additional current consumed by the comparator is [No.13.0 Section "Electrical Characteristics"](#) If the comparator is not used to wake up the device, the comparator can be turned off to minimize power consumption during sleep. CMSTARRegisterCM<2:0> = 000 or CM<2:0> = 111 mode, the comparator can be turned off.

A change in the comparator output can wake the device from sleep mode. To enable the comparator to wake the device from sleep mode, you must PIEB1Register CxIbit and INTSRegister PEIELocation1. When the device wakes from Sleep, the instruction immediately following the Sleep instruction will always be executed. INTSRegister GIEThe position is also set 1, the device will execute the interrupt service routine.

10.7 Effect of reset

A device reset will force CMSTA and CMCON1 registers to their reset state. This forces the comparator module to the comparator reset mode (CM<2:0> = 000). This way, all comparator inputs are analog inputs and the comparator is disabled, consuming minimal current.

register19H: Comparator Configuration Register (CMSTA)

R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0
Bit7							bit0

Figure caption:

R = Readable bit

W = Writable bit

U = Unimplemented bit, read as 0

-n = PORTThe value of

1 = Place 1

0 = Clear

x = unknown

bit7

C2OUT: Comparators2 Output

bits C2INV = 0 hour:

1 = C2VIN+>C2VIN-

0 = C2VIN+<C2VIN-

C2INV = 1 hour:

1 = C2VIN+<C2VIN-

0 = C2VIN+>C2VIN-

bit6

C1OUT: Comparators1 Output

bits C1INV = 0 hour:

1 = C1VIN+>C1VIN-

0 = C1VIN+<C1VIN-

C1INV = 1 hour:

1 = C1VIN+<C1VIN-

0 = C1VIN+>C1VIN-

bit5

C2INV: Comparators2 Output toggle bit

1 = C2Output inversion

0 = C2Output does not flip

bit4

C1INV: Comparators1 Output toggle bit

1 = C1Output inversion

0 = C1Output does not flip

bit3

CIS: Comparator input switch position

CM<2:0> = 010 hour: 1 =

C1IN+Connect to C1VIN-

C2IN+Connect toC2VIN-
 0 = C1IN-Connect toC1VIN-
 C2IN-Connect toC2VIN-
 CM<2:0> = 001hour: 1 =
 C1IN+Connect toC1VIN-0 =
 C1IN-Connect toC1VIN-

bit2-0 **CM<2:0>**: Comparator mode selection bits (see Figure8-5) 000 =
 The comparator is turned off.CxDpins configured as analog 001
 =Three inputs multiplexed to two comparators
 010 =Four inputs multiplexed to two comparators
 011 =Two common reference comparators
 100 =Two independent comparators
 101 =An independent comparator
 110 =Two common reference comparators with outputs
 111 =The comparator is turned off.CxDpin configuration as digitalI/O

10.8ComparatorsC2GatingTimer1

You can use this feature to time the duration or interval of simulated events.CMCON1RegisterT1GSSClearing this bit will enableTimer1According to the comparatorC2The output of is incremented. This requiresTimer1Open and gate enabled. See[No.7.2Section "GatedTimer1Modules "](#)

When the comparator is used asTimer1When the gate source is It is recommended toC2SYNCLocation1The comparatorC2andTimer1This will ensureTimer1
 If the comparator changes during incrementing,Timer1Will not miss incrementing.

10.9ComparatorsC2Output andTimer1Synchronize

ByCMCON1RegisterC2SYNCLocation1The comparator canC2The output ofTimer1When enabled, the comparator output isTimer1The falling edge of the clock source is latched.Timer1When a prescaler is used, the comparator output is latched after prescaling. To prevent race conditions, the comparator output is latched afterTimer1The falling edge of the clock source is latched, andTimer1Increments on the rising edge of the clock source. For more information, see the Comparator Block Diagram ([picture10-1-2](#) and[picture10-1-3](#))as well asTimer1block diagram([picture7-2](#)).

register1AH: Comparator Configuration Register (CMCON1)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0
—	—	—	—	—	—	T1GSS	C2SYNC
bit7						bit0	

Figure caption:

R =Readable bit W =Writable bit U =Unimplemented bit, read as0
 -n = PORThe value of 1 =Place1 0 =Clear x =unknown

bit7-2 Unimplemented: Read as0

bit1 **T1GSS**:Timer1Gate source select bits (1)
 1 = Timer1The gate source isT1GPin (pin should be configured as digital input) 0
 = Timer1The gate source is a comparatorC2Output



bit0 **C2SYNC**:ComparatorsC2Output Sync Bit(2) 1 =Output andTimer1Falling edge of the clock is synchronized 0 = Asynchronous Output

Note 1:See[No.7.2.6Festival"Timer1Gating"](#) .
2:See[picture10-1-3](#) .

10.10 Comparator reference voltage

The comparator voltage reference module provides an internally generated reference voltage for the comparator. It has the following features:

- Independent of comparator operation
- two16Voltage range
- Output clamped toVSS
- andVDDProportional
- Fixed reference voltage (0.6V)

VRSTARegister([register99H](#)) controls the reference voltage module, such as [picture10-10-4](#) shown.

10.10.1 Work independently

The comparator reference voltage is independent of the comparator configuration.VRSTARegisterVRENbit will enable the voltage reference.

10.10.2 Output voltage selection

CVREFThere are two ranges of reference voltage, each of which is16The range is selected byVRSTARegisterVRRBit control.16The level isVRSTARegisterVR<3:0>Bit setting.

CVREFThe output voltage is determined by the following formula:

formula10-10-2:CvrefThe output voltage

$$\begin{aligned}
 &VRR = 1 \text{ (低电压范围):} \\
 &\quad CVREF = (VR<3:0>/24) \times VDD \\
 &VRR = 0 \text{ (高电压范围):} \\
 &\quad CVREF = (VDD/4) + (VR<3:0> \times VDD/32)
 \end{aligned}$$

Due to the limitations of the module structure, it is not possible toVSStoVDDSee[picture10-10-4](#) .

10.10.3 Output clamped toVss

ByVRSTThe following configuration can beCVREFThe output voltage is set toVSS, thus consuming no power:

- VREN = 0
- VRR = 1
- VR<3:0> = 0000

This allows the comparator to perform zero-crossing detection without consuming additional CVREF module current.

10.10.4 Output and VDD Proportional

The comparator reference voltage is given by $V_{DD} \times \text{CVREF}$ Output with V_{DD} . The absolute accuracy of the comparator reference voltage is [No.13.0 Section "Electrical Characteristics"](#) Listed in.

register99H: Reference Voltage Control Register (VRSTA)

R/W-0	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
VREN	—	VRR	—	VR3	VR2	VR1	VR0
bit7							bit0

Figure caption:

R = Readable bit W = Writable bit U = Unimplemented bit, read as 0
 -n = PORThe value of 1 = Place 1 0 = Clear x = unknown

bit7 **VREN**: CVREF Enable bit 1 = CVREF Circuit energized
 0 = CVREF Circuit is powered off, no leakage current I_{DD} and $CVREF = VSS$.

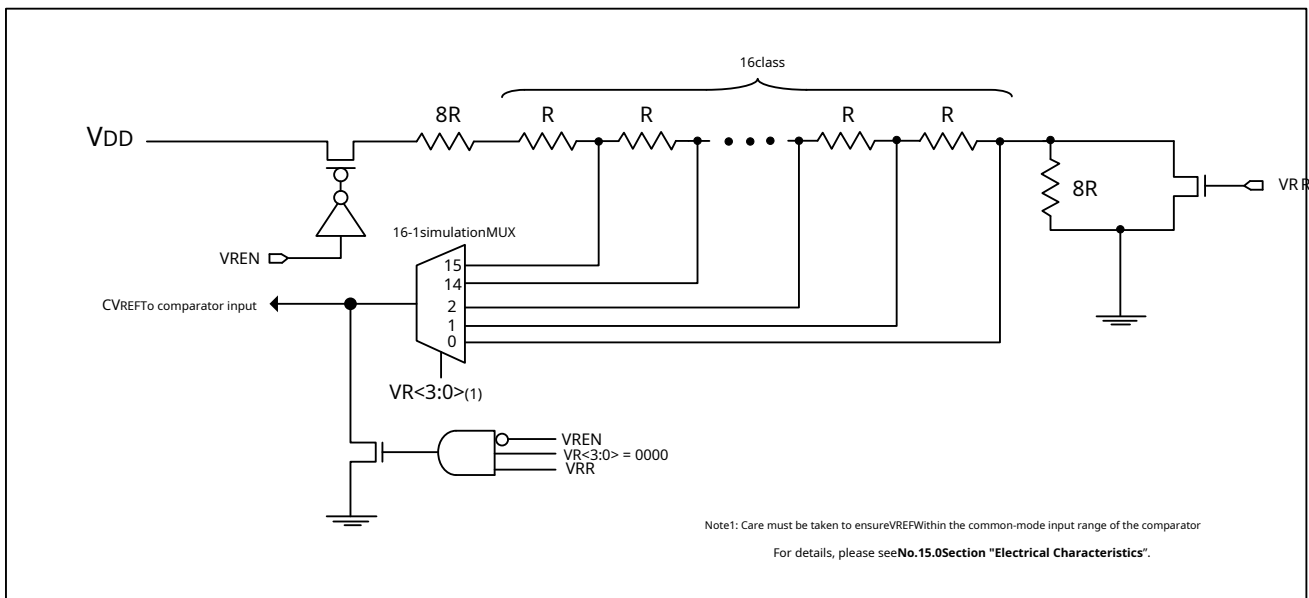
bit6 Unimplemented: Read as 0

bit5 **VRR**: CVREF Range selection bits
 1 = Low level range
 0 = High level range

bit4 Unimplemented: Read as 0

bit3-0 **VR<3:0>**: CVREF Value Select Bits ($0 \leq VR<3:0> \leq 15$)
 $VRR = 1 \text{ hour: } CVREF = (VR<3:0>/24) * VDD$
 $VRR = 0 \text{ hour: } CVREF = VDD/4 + (VR<3:0>/32) * VDD$

picture10-10-4: Comparison reference voltage block diagram



surface10-10-4: Summary of registers associated with the comparator voltage reference module

name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	PORandBOR The value of	All other Reset value
ANINS	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	1111 1111
CMSTA	C2OUT	C1OUT	C2INV	C1INV	CIS	CM2	CM1	CM0	0000 0000	0000 0000
CMCON1	—	—	—	—	—	—	T1GSS	C2SYNC	----- 10	----- 10
INTS	GIE	PEIE	TIS	INS	PAIE	TIF	INTF	PAIF	0000 0000	0000 0000
PIEB1	EEIE	ADIE	CCP1IE	C2IE	C1IE	OSFIE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIFB1	EEIF	ADIF	CCP1IF	C2IF	C1IF	OSFIF	TMR2IF	TMRIF	0000 0000	0000 0000
PORTA	—	—	PA5	PA4	PA3	PA2	PA1	PA0	- -x0x000	- -uu uu00
PORTC	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0	- -xx 0000	- -uu uu00
CPIOA	—	—	CPIOA5	CPIOA4	CPIOA3	CPIOA2	CPIOA1	CPIOA0	--11 1111	--11 1111
CPIOC	CPIOC7	CPIOC6	CPIOC5	CPIOC4	CPIOC3	CPIOC2	CPIOC1	CPIOC0	1111 1111	1111 1111
VRSTA	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0-0000	0-0-0000

Figure caption: x =unknown,u =unchanged, — = unimplemented (read as0) The shaded cells are not used by the comparator.

11.0 Configuration Registers

The configuration register is used to configureMCUThe original value of the setting isFlashIn some locations,MCUWhen starting fromFlash Bootto the configuration register, the configuration register bit width is8bit, due toFlashThe bit width is14Bit,BootWhenBootLow8bits to the configuration register.

11.1 Config Option

ConfigOptionUsed to provide configuration options to customers.

register1201H: Configuration word register high8Bit(Config1 Option)

—	RD_CTRL	LVDS1	LVDS0	FCMEN	IESO	LVREN1	LVREN0
bit15							bit8

register1200H: Configuration word register low8Bit(Config0 Option)

CPDB	CPB	MCLRE	PWRTE	WDTE	FOSC2	FOSC1	FOSC0
bit7							bit0

bit15 Unimplemented: Reserved

bit14 **RD_CTRL**: Read port control in output mode
1: Read data port returnsPINThe value on 0: Reading the data port returns the value on the latch

bit13-12 **LVDS <1:0>**: Low voltage reset voltage setting
11:1.7V
10:1.7V
01:2.1V
00:3.8V

bit11	<p>FCMEN: System clock failure detection enable bit</p> <p>1: Enable system clock failure detection, which only works when the system clock is in external crystal oscillator mode 0: Disable system clock failure detection</p>
bit10	<p>IESO: Internal and external clock switching enable bit, dual-speed startup mode enable,</p> <p>1: Enable the dual-speed startup mode, which only works when the system clock is in external crystal oscillator mode 0: Disable two-speed start mode</p>
bit9-8	<p>LV DEN <1:0>: Low voltage reset selection⁽¹⁾</p> <p>11: Low voltage reset enable</p> <p>10:MCUWhen working, low voltage reset is turned on;MCUDuring sleep, low voltage reset is disabled</p> <p>01: Low voltage reset byPCONOFBBORENPosition decision 00: Disable low voltage reset</p>
bit7	<p>-----</p> <p>CPDB: Data code protection bit⁽²⁾</p> <p>1:Data EEPROMContent is not protected</p> <p>0:start upData EEPROMContent protection,MCUCan read, serial port cannot read</p>
bit6	<p>CPB: Code protection bit⁽³⁾</p> <p>1:FlashContent is not protected</p> <p>0:start upFlashContent protection,MCUCan read, serial port cannot read</p>
bit5	<p>-----</p> <p>MCLRE:PA3/MCLRPin function selection bits⁽⁴⁾</p> <p>1:PA3/MCLRFoot executionMCLRFunction, is the reset pin 0:PA3/ MCLRFoot executionPA3Function, is a digital input pin</p>
bit4	<p>-----</p> <p>PWRTE: Power-on delay timer enable bit</p> <p>1:PWRTProhibit</p> <p>0:PWRTEEnable</p>
bit3	<p>WDTE: Watchdog timer enable bit</p> <p>1:WDTEnable, the program cannot disable</p> <p>0:WDTDisabled, but the program can be setWDTCONofSWDTENPositionWDTEnable</p>
bit2-0	<p>FO SC<2:0>: Oscillator selection bits</p> <p>111:INTOSCIO Oscillatormodel,PA4forIOPins,PA5forIOPinout 110:RCIO Oscillatormodel, PA4forIOPins,PA5Clock input pinCLKIN 101:INTOSC Oscillatormodel,PA4Output Clock CLKOUT,PA5forIOPinout 100:RC Oscillatormodel,PA4Output ClockCLKOUT,PA5It is the clock input pinCLKIN 011:ECmodel,PA4forIOPins,PA5Clock input pinCLKIN 010:HS Oscillatormodel,PA4,PA5Connect to high frequency crystal 001:XT Oscillatormodel,PA4, PA5Connect to crystal 000:LP Oscillatormodel,PA4,PA5Connect to low frequency crystal</p>

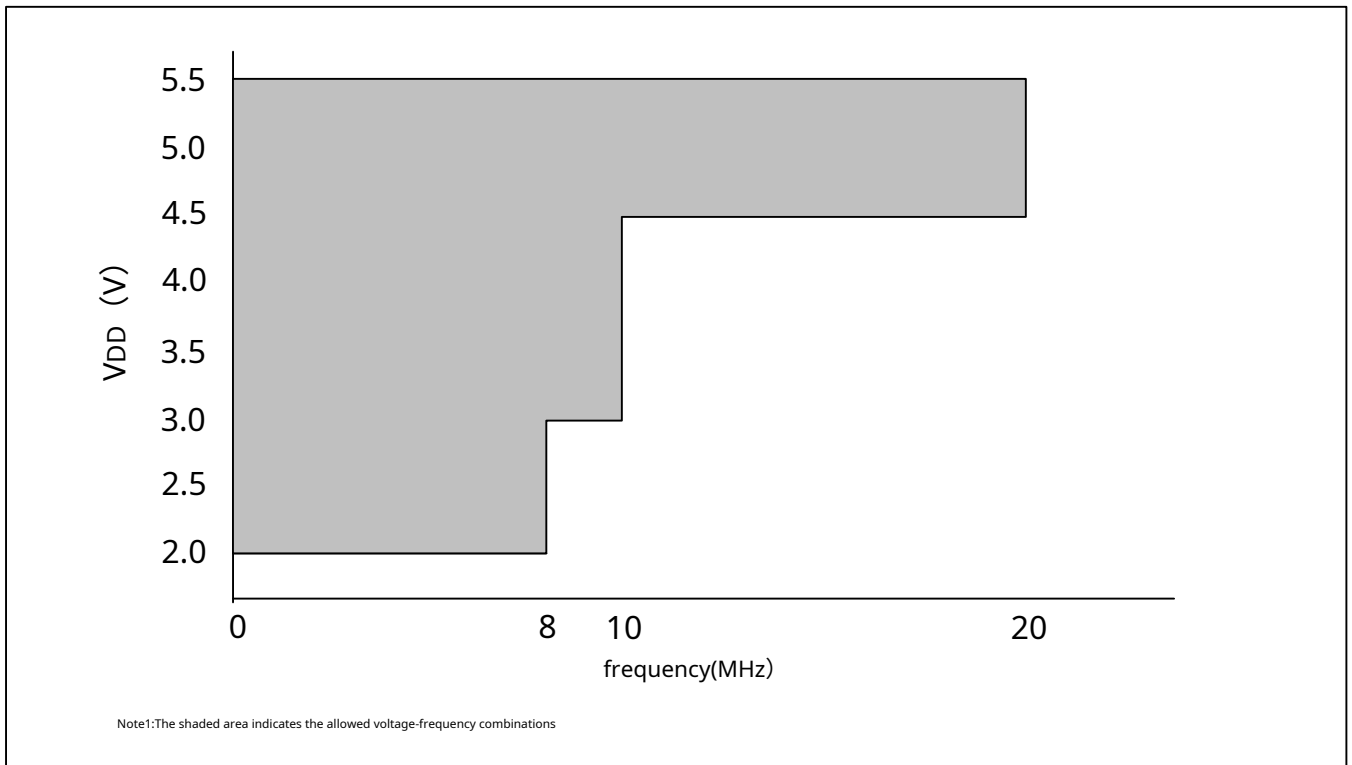
Note

- 1: Enabling Brown-out Reset does not automatically enable the Power-up Timer. 2:
When code protection is turned off, the entire dataEEPROMwill be erased. 3: When code
protection is turned off, the entire programEEPROMwill be erased.
-
- 4:existINTOSCorRCWhen modeMCLRWhen active, the internal clock oscillator is disabled.

12.0 Instruction List

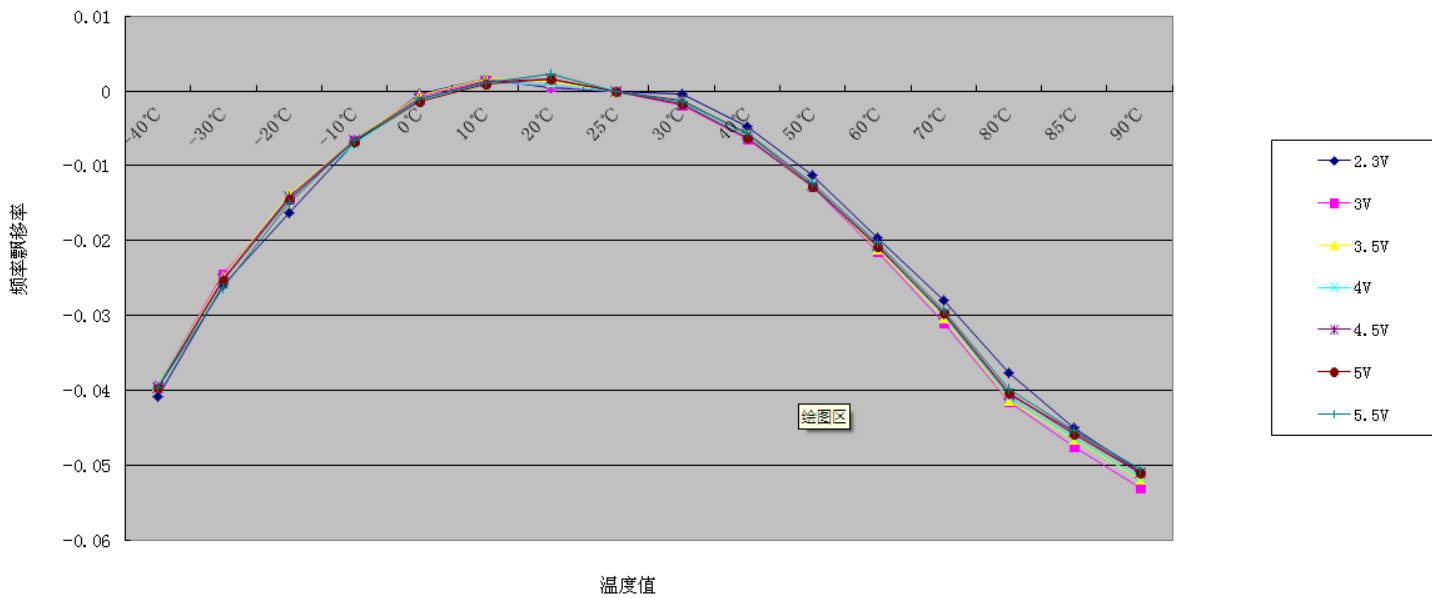
Instruction code	instruction	Function	operate	Impact Position
010000 00000000	NOP	No operation	none	none
010000 00000001	CLRWT	Clear Watchdog	0→WT	/TF, /PF
010000 00000010	SLEEP	Entering sleep mode	0→WT, stop OSC	/TF, /PF
010000 00000011	TMODE	Load the W register value into the option register (81H)	W→81H	none
010000 00000rrr	CPIO R	Set port direction register (1 input, 0 output)	W→CPIO r	none
010001 1rrrrrrr	STWR	Send the contents of the W register to the R register	W→R	none
011000 trrrrrrr	LDR R, t	Read register R, the result is stored in R (t=1) or W (t=0)	R→t	Z
111010 iiiiii	LDWI	Send immediate data to W register	I→W	none
010111 trrrrrrr	SWAPR R, t	Exchange the high and low four bits of register R, and the result is stored in R (t=1) or W (t=0)	[R(0~3) R(4~7)]→t	none
011001 trrrrrrr	INCR R, t	Increment register R, the result is stored in R (t=1) or W (t=0)	R + 1 → t	Z
011010 trrrrrrr	INCRSZ R, t	Increment register R, save the result in R (t=1) or W (t=0); if the result is equal to 0, skip The following instructions	R + 1 → t	none
011011 trrrrrrr	ADDWR R, t	The W register is added to the R register, and the result is stored in R (t=1) or W (t=0)	W + R → t	C, HC, Z
011100 trrrrrrr	SUBWR R, t	The R register is subtracted from the W register, and the result is stored in R (t=1) or W (t=0)	R AW→t (R+/W+1→t)	C, HC, Z
011101 trrrrrrr	DECR R, t	Decrement register R, the result is stored in R (t=1) or W (t=0)	R A1→t	Z
011110 trrrrrrr	DECRSZ R, t	Decrement register R, save result in R (t=1) or W (t=0); jump if result is 0 The following command	R A1→t	none
010010 trrrrrrr	ANDWR R, t	The R register and the W register perform an "AND" operation, and the result is stored in R (t=1) or W (t=0)	R ∩ W→t	Z
110100 iiiiii	ANDWI I	The W register and the immediate value I perform an "AND" operation, and the result is saved in the W register.	I ∩ W→W	Z
010011 trrrrrrr	IORWR R, t	The R register is ORed with the W register, and the result is stored in R (t=1) or W (t=0).	R ∪ W→t	Z
110101 iiiiii	IORWI I	The W register is ORed with the immediate value I, and the result is stored in R (t=1) or W (t=0).	I ∪ W→W	Z
010100 trrrrrrr	XORWR R, t	The R register and the W register perform an "XOR" operation, and the result is stored in R (t=1) or W (t=0)	R ⊕ W→t	Z
110110 iiiiii	XORWI I	The W register performs an "XOR" operation on the immediate value I, and the result is stored in R (t=1) or W (t=0)	I ⊕ W→W	Z
011111 trrrrrrr	COMR R, t	R register "invert" operation, the result is stored in R (t=1) or W (t=0)	/R→t	Z
010110 trrrrrrr	RRR R, t	The R register is cyclically "shifted right" and the result is stored in R (t=1) or W (t=0)	R(n)→R(n-1), C→R(7), R(0)→C	C
010101 trrrrrrr	RLR R, t	The R register is cyclically "shifted left" and the result is stored in R (t=1) or W (t=0)	R(n)→r(n+1), C→R(0), R(7)→C	C
010000 1xxxxxxx	CLRW	W register cleared to 0	0→W	Z
010001 0rrrrrrr	CLRR	R register clear 0	0→R	Z
0000bb brrrrrrr	BCR R, b	Clear bit b of R register to 0	0→R(b)	none
0010bb brrrrrrr	BSR R, b	Bit b of R register is set to 1	1→R(b)	none
0001bb brrrrrrr	BTSC R, b	If bit b of the R register is 0, the following instruction is skipped.	Skip if R(b)=0	none
0011bb brrrrrrr	BTSS R, b	If bit b of the R register is 1, the following instruction is skipped.	Skip if R(b)=1	none
100nnn nnnnnnnn	LCALL N	Call instruction in the entire 2K region	N→PC, PC+1→Stack	none
101nnn nnnnnnnn	LJUMP N	Jump instructions within the entire 2K area	N→PC	none
110001 iiiiii	RTIW I	Return from a subroutine with an immediate value	Stack→PC, I→W	none
110111 iiiiii	ADDWI I	Add the W register to the immediate value I and save the result in W	PC+1→PC, W+I→W	C, HC, Z
111000 iiiiii	SUBWI	The immediate value I minus the W register, and the result is stored in the W register.	IW→W	C, HC, Z
010000 00001001	RTFI	Interrupt return	Stack→PC, 1→GIS	none
010000 00001000	RET	Return from a subroutine	Stack→PC	none

picture13-1-1:MDT10F684Voltage-frequency diagram, $-40^{\circ}\text{C} \leq \text{TA} \leq +85^{\circ}\text{C}$



picture13-1-2: DeviceVDDand temperature rangeHFINTOSCFrequency accuracy

IRC温度曲线



13.2DC Electrical Characteristics

DC Characteristics		Standard working conditions Operating temperature -40°C≤Ta≤+85°C						
symbol	characteristic	Minimum	Typical Value (1)	Maximum	unit	condition		
VDD	voltage	2		5.5	V			
VDR	RAMData retention voltage (2)	—	0.5*	—	V	Device in Sleep Mode		
VPOR	VddThe starting voltage ensures that the Electrical reset signal	—	Vss	—	V			
SVDD	VddThe rate of rise ensures that the Electrical reset signal	0.05*	—	—	V/ms			
IDD	Working current (3)	—	230	—	uA	3.3V	1M	WDT PED Dis IRC
			370				4M	
			570			5V	8M	
			320				1M	
			580				4M	
			880				8M	
IPD	Current drop (4)	—	1	—	uA	WDT Disable VDD=2.5V		
ΔIWDT	WDTCurrent (4)	—	0.5	—	uA	VDD=5V		
VIL	Input low voltage	VSS	— —	1.1	V	3V TLL		
		VSS	— —	1.1		3V SCHMITT		
		VSS	— —	1.6		5V TLL		
		VSS	— —	1.6		5V SCHMITT		
VIH	Input high voltage	1.1	— —	VDD	V	3V TLL		
		1.9	— —	VDD		3V SCHMITT		
		1.6	— —	VDD		5V TLL		
		3.5	— —	VDD		5V SCHMITT		
IOL	Output Sink Current	—	twenty four	—	mA	VOL=0.7V	3V	
		—	28	—		VOL=0.9V		
		—	35	—		VOL=0.7V	5V	
		—	42	—		VOL=0.9V		
IOH	Output source current	—	13	—	mA	VOH=1.7V	3V	
		—	10	—		VOH=2.1V		
		—	twenty two	—		VOH=3.6V	5V	
		—	12	—		VOH=4.2V		
VPED	Low voltage reset voltage	1.7 -20%	1.7	1.7 +20%	V	LOW		
		2.1 -20%	2.1	2.1 +20%		MIDDLE		
		3.8 -20%	3.8	3.8 +20%		HIGH		
R	Pull-up resistor	—	32	—	K	3V		
		—	18	—		5V		

Note: "—" means none or pending.

(1) The data in the typical column are 25°C, this part of the data is for reference only. (2)

This voltage is guaranteed not to be lost RAM Minimum data VDD.

(3) Operating current varies primarily with operating voltage and frequency. Other factors such as bus loading, bus speed, internal code execution pattern, and temperature also affect the current.

Flow consumption.

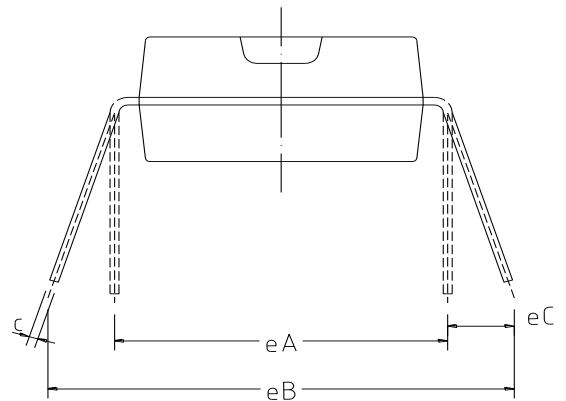
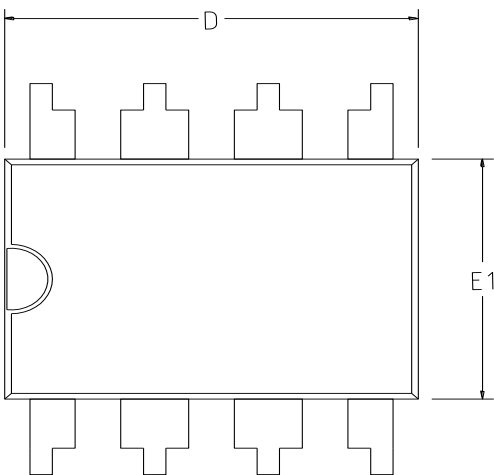
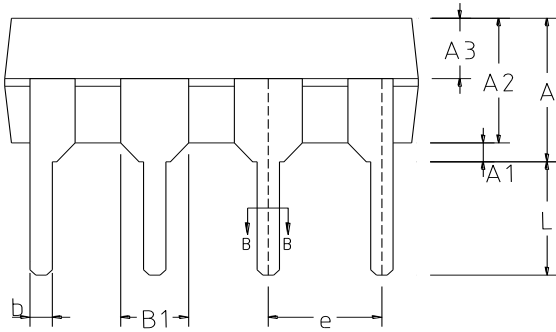
(4) Power-down current is when the device is in sleep mode, all I/O pins are in high impedance state and connected to VDD or VSS Measured when.

13.3AC electrical characteristics

AC Characteristics		Standard working conditions Operating temperature $-40^{\circ}\text{C} \leq T_{\text{as}} \leq +85^{\circ}\text{C}$				
symbol	characteristic	Minimum	Typical Value ⁽¹⁾	Maximum	unit	condition
FOSC	System clock	0	8M	20M	Hz	5V
FHIRC	Internal high speed clock	— —	4M	8M	Hz	5V
FLIRC	Internal low speed clock	— —	31K	— —	Hz	5V
T _{EERD}	EEPROM Reading time	— —	1	— —	Tins	Instruction cycle
T _{EEWR}	EEPROM Write time	— —	400us	1ms	s	5V
T _{INT}	Interrupt pulse		2	3	Tins	Instruction cycle
T _{SST}	System startup time (Power-On Reset)		13	20	ms	5V
	System startup time (Depend on SLEEP Mode wake-up, Fsyexist SLEEP Mode Off close)		1024		Tsys	5V
	System startup time (Depend on SLEEP Mode wake-up, mainly at SLEEP mode)		1024		Tsys	5V
T _{RSTD}	System reset delay time (Power-On Reset)		13	20	ms	5V
	System reset time (WDT Normal reset)		77	84	ms	5V

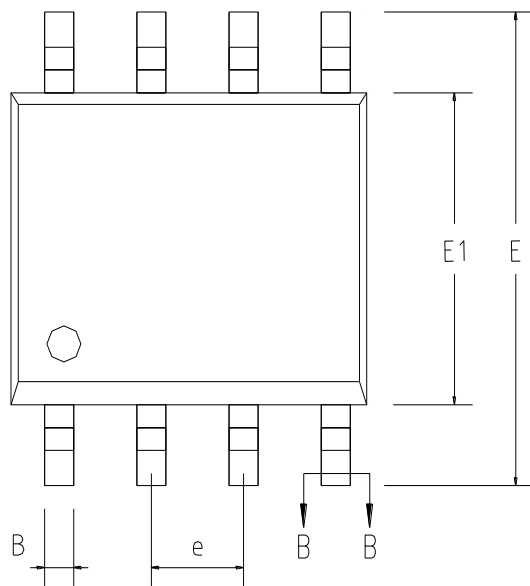
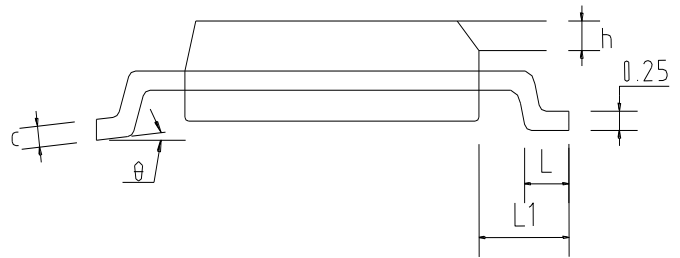
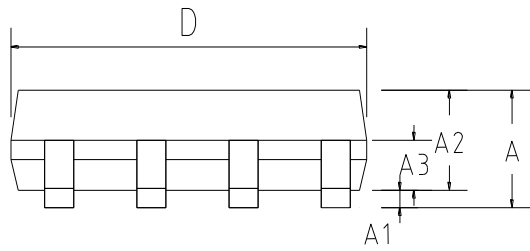
15.0 Packaging information

15.1 P-DIP 8 PIN



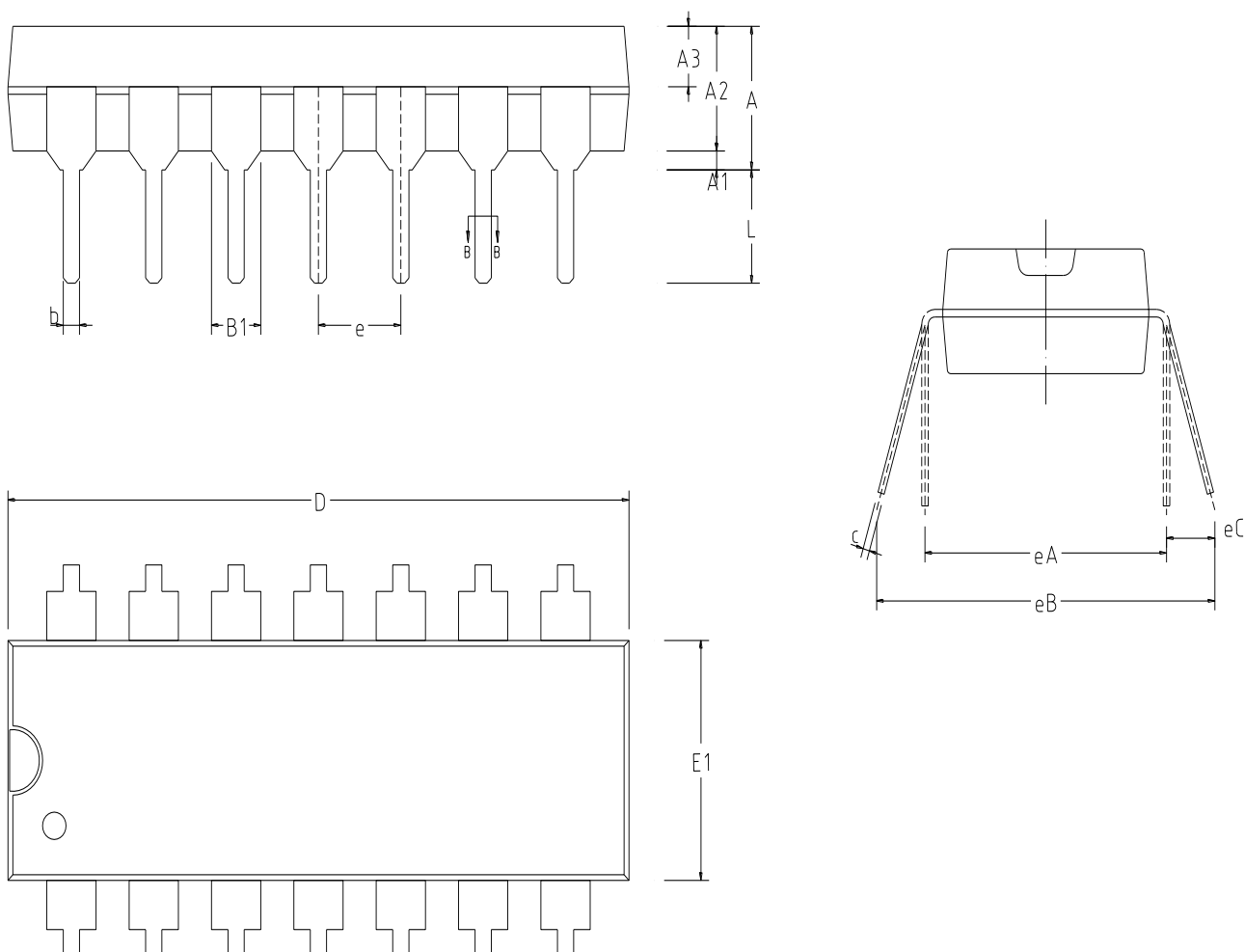
SYMBOLS	MIN	NOR	MAX	SYMBOLS	MIN	NOR	MAX
	(mm)				(mm)		
A	3.60	3.80	4.00	E1	6.15	6.35	6.55
A1	0.51	—	—	e	2.54BSC		
A2	3.00	3.30	3.40	eA	7.62BSC		
A3	1.55	1.60	1.65	eB	7.62	—	9.30
B1	1.52BSC			eC	0	—	0.84
D	9.05	9.25	9.45	L	3.00	—	—
L/FCarrier size(mil)	80*80						

15.2 SOP 8 PIN



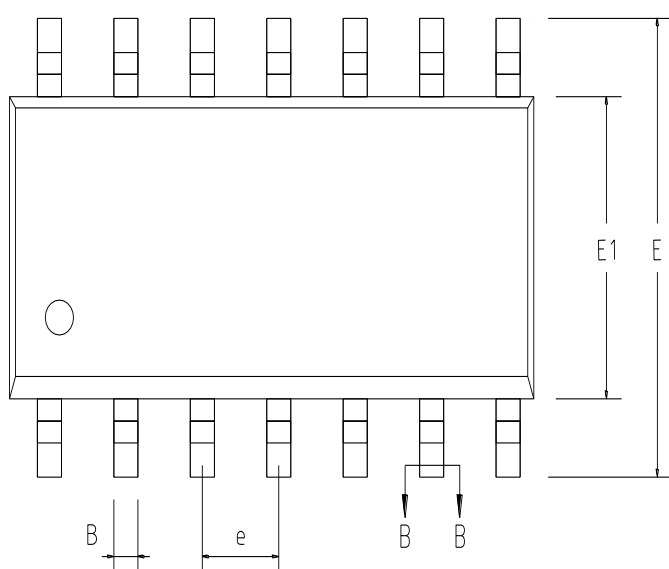
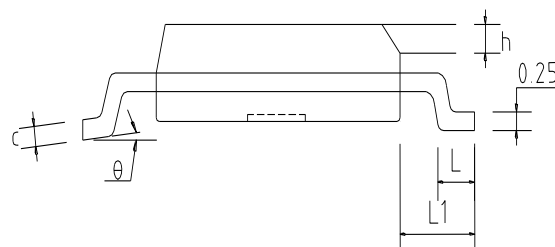
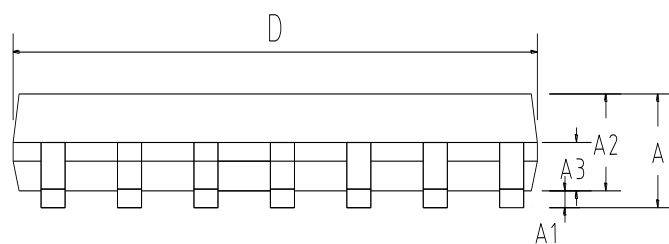
SYMBOLS	MIN	NOR	MAX	SYMBOLS	MIN	NOR	MAX
	(mm)				(mm)		
A	—	—	1.75	E1	3.70	3.90	4.10
A1	0.10	—	0.225	e	1.27BSC		
A2	1.30	1.40	1.50	h	0.25	—	0.50
A3	0.60	0.65	0.70	L	0.50	—	0.80
D	4.70	4.90	5.10	L1	1.05 BSC		
E	5.80	6.00	6.20	-	0	—	8-
L/FCarrier size(mil)	80*80	90*90	95*130	— —			

15.3 P-DIP 14 PIN



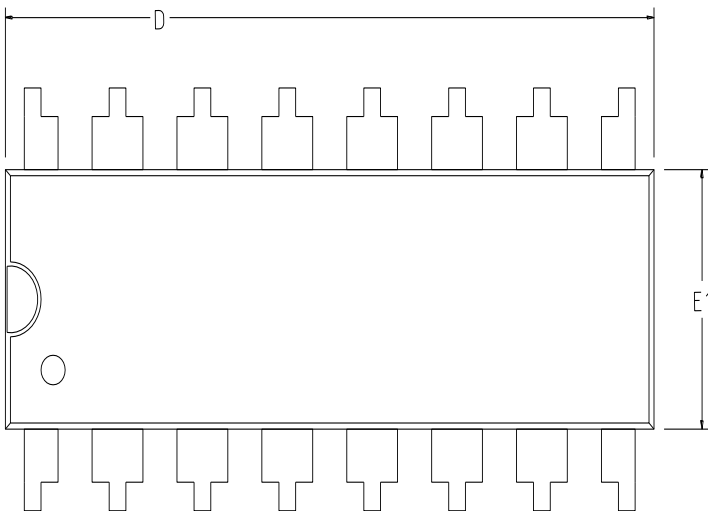
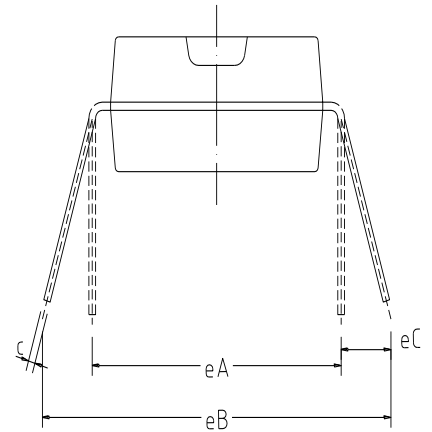
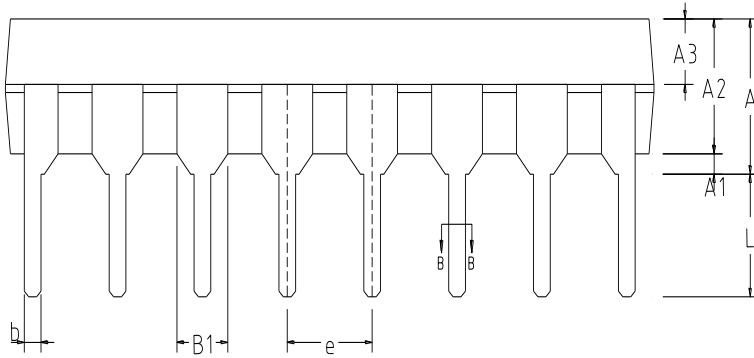
SYMBOLS	MIN	NOR	MAX	SYMBOLS	MIN	NOR	MAX
	(mm)				(mm)		
A	3.60	3.80	4.00	E1	6.15	6.35	6.55
A1	0.51	—	—	e	2.54BSC		
A2	3.20	3.30	3.40	eA	7.62BSC		
A3	1.47	1.52	1.57	eB	7.62	—	9.30
B1	1.52BSC			eC	0	—	0.84
D	18.90	19.10	19.30	L	3.00	—	—

15.4 SOP 14 PIN



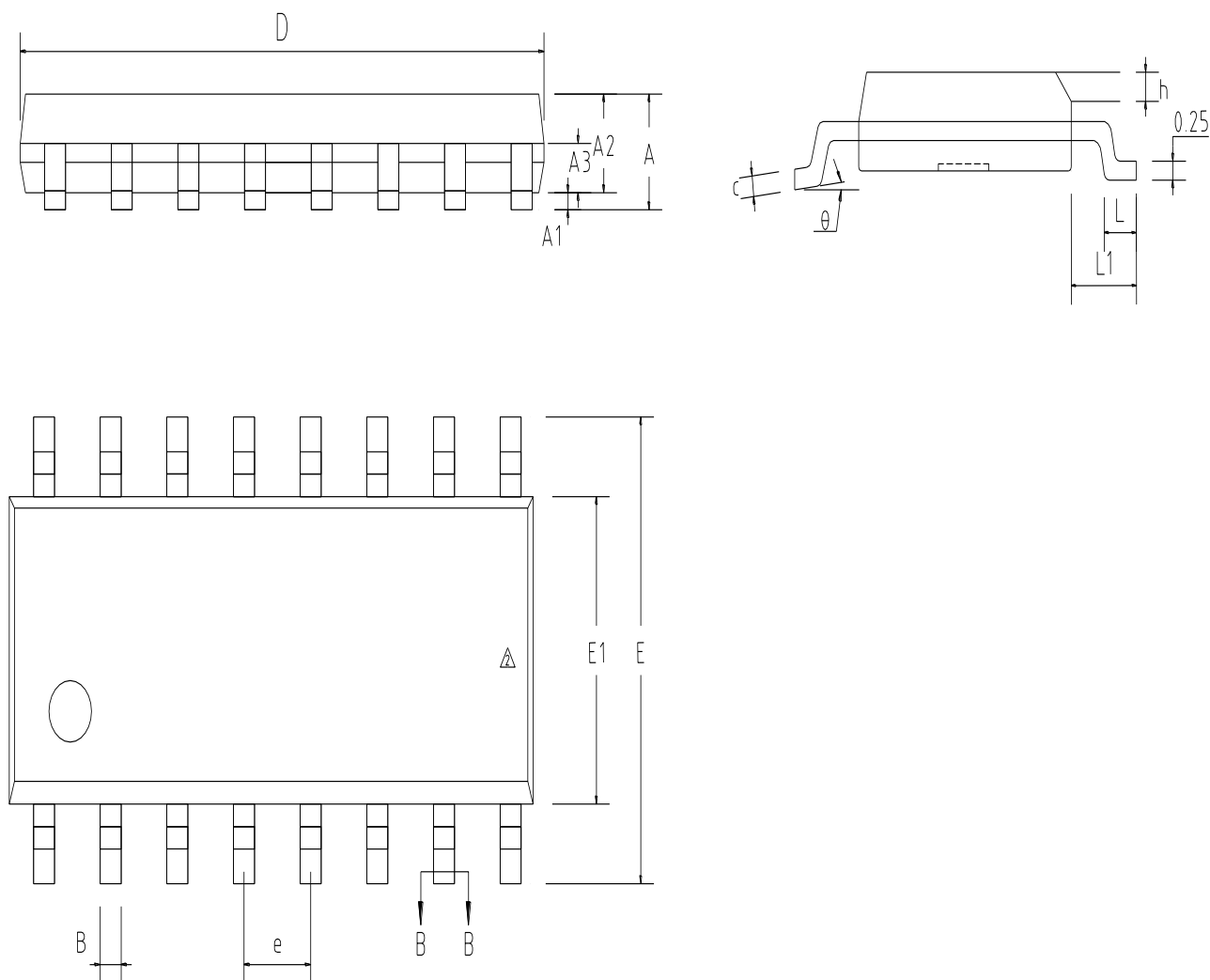
SYMBOLS	MIN	NOR	MAX	SYMBOLS	MIN	NOR	MAX
	(mm)				(mm)		
A	—	—	1.75	E1	3.70	3.90	4.10
A1	0.10	—	0.225	e	1.27BSC		
A2	1.30	1.40	1.50	h	0.25	—	0.50
A3	0.60	0.65	0.70	L	0.50	—	0.80
D	8.45	8.65	8.85	L1	1.05 BSC		
E	5.80	6.00	6.20	-	0	—	8-
L/Carrier size(mil)	70*70 90*110	98*150	100.4*210	— —			

15.5 P-DIP 16 PIN



SYMBOLS	MIN	NOR	MAX	SYMBOLS	MIN	NOR	MAX
	(mm)				(mm)		
A	3.60	3.80	4.00	E1	6.15	6.35	6.55
A1	0.51	—	—	e	2.54BSC		
A2	3.20	3.30	3.40	eA	7.62BSC		
A3	1.47	1.52	1.57	eB	7.62	—	9.30
B1	1.52BSC			eC	0	—	0.84
D	18.90	19.10	19.30	L	3.00	—	—

15.6 SOP 16 PIN



SYMBOLS	MIN	NOR	MAX	SYMBOLS	MIN	NOR	MAX
	(mm)				(mm)		
A	—	—	1.75	E1	3.70	3.90	4.10
A1	0.05	—	0.225	e	1.27BSC		
A2	1.30	1.40	1.50	h	0.25	—	0.50
A3	0.60	0.65	0.70	L	0.50	—	0.80
D	9.70	9.90	10.10	L1	1.05 BSC		
E	5.80	6.00	6.20	-	0	—	8-