EM85F765N

8-BIT Microcontroller

Product Specification

DOC. VERSION 0.9

ELAN MICROELECTRONICS CORP.

February 2017

Trademark Acknowledgments:
IBM is a registered trademark and PS/2 is a trademark of IBM.
Windows is a trademark of Microsoft Corporation.

ELAN and ELAN logo are trademarks of ELAN Microelectronics Corporation.

The contents of in this specification are subject to change without notice. ELAN Microelectronics assumes no responsibility concerning the accuracy, adequacy, or completeness of this specification. ELAN Microelectronics makes no commitment to update, or to keep current the information and material contained in this specification. Such information and material may change to conform to each confirmed order.

In no event shall ELAN Microelectronics be made responsible to any claims attributed to errors, omissions, or other inaccuracies in the information or material contained in this specification. ELAN Microelectronics shall not be liable for direct, indirect, special incidental, or consequential damages arising out of the use of such information or material.

The software (if any) described in this specification is furnished under a license or nondisclosure agreement, and may be used or copied only in accordance with the terms of such agreement.

ELAN Microelectronics products are not intended for use in life support appliances, devices, or systems. Use of ELAN Microelectronics product in such applications is not supported and is prohibited.
NO PART OF THIS SPECIFICATION MAY BE REPRODUCED OR TRANSMITTED IN ANY FORM OR BY ANY MEANS WITHOUT THE EXPRESS WRITTEN PERMISSION OF ELAN MICROELECTRONICS.

ELAN MICROELECTRONICS CORPORATION

Headquarters:
No. 12, Innovation Road 1
Hsinchu Science Park
Hsinchu, TAIWAN 308
Tel: +886 3 563-9977
Fax:+886 3 563-9966
webmaster@emc.com.tw
http://www.emc.com.tw

Hong Kong:
Elan (HK) Microelectronics Corporation, Ltd.
Flat A, 19F., World Tech Centre
95 How Ming Street, Kwun Tong
Kowloon, HONG KONG
Tel: +852 2723-3376
Fax:+852 2723-7780

USA:
Elan Information Technology Group (U.S.A.)
PO Box 601
Cupertino, CA 95015
U.S.A.
Tel: +1 408 366-8225
Fax:+1 408 366-8225

Shenzhen:
Elan Microelectronics Shenzhen, Ltd.
8A Floor, Microprofit Building
Gaoxin South Road 6
Shenzhen Hi-tech Industrial Park
South Area, Shenzhen
CHINA 518057
Tel: +86 755 2601-0565
Fax:+86 755 2601-0500
elan-sz@elanic.com.cn

Shanghai:
Elan Microelectronics Shanghai, Ltd.
6F, Ke Yuan Building
No. 5 Bibo Road
Zhangjiang Hi-Tech Park
Shanghai, CHINA 201203
Tel: +86 21 5080-3866
Fax:+86 21 5080-0273
elan-sh@elanic.com.cn

Contents

APPENDIX

Specification Revision History

Version	Revision Description	Date
0.9	Preliminary version	2017/02/07

1 System Overview

- System operating condition: 16KHz~20 MHz @ 2.7~5.5V, -40°C~85°C
- High-speed pipelined 8051-compatible microcontroller

Memory

- 18KB of on-chip Flash memory
- Flash Memory 10000 writing cycle endurance
- Built-in In-Application-Programmable (IAP)
- (256+768)B of on-chip RAM

Clock Sources

- High frequency Internal Resistor & Capacitor oscillator (HIRC) with ± 2% total of deviation: 4MHz

Internal RC Frequency	Drift Rate			
	Process	Temperature (-40℃~ +85℃)	Voltage (2.7V~ 5.5V)	Total
128KHz	±2%	±3%	±2%	±7%
4MHz	±0.25%	±1.75%		±2%

- External crystal oscillator
- External clock input
- PLL 48MHz
- LXT 32.768KHz
- Low frequency (128K/32K/16KHz) Internal Resistor & Capacitor oscillator (LIRC)
- Operation Frequency: 16KHz~20MHz

Analog Peripherals

- 12-bit 100ksps single-ended ADC with (21+2) analog input channel
- One Operational Amplifier/Comparator with 2mV input offset voltage and 200ns response time
- Internal voltage reference: 4.096V/3.072V/2.560V/2.048V (1LSB: 1mV/0.75mV/0.625mV/0.5mV)
- On-chip regulator: 1.5V ± 3% only for kernel use
- Power on Reset (POR): 2.3/2.4V ±0.2V @ -40°C~85°C
- On-chip Low-voltage detect (HLVD): 2.5V, 2.6V,2.8V,2.9V, 3.1V, 3.3V,3 .5V, 3.7V, 3.9V, 4.1V, 4.3V, 4.5V, 4.7V ±0.15V @ -40°C~85°C
- Brown Out Reset (BOR): 2.5V, 2.6V,2.8V,2.9V, 3.1V, 3.3V,3.5V, 3.7V, 3.9V, 4.1V, 4.3V, 4.5V, 4.7V ±0.15V @ -40°C~85°C

Digital Peripherals

- 1-set I2C(16 Bytes Buffer)
- 1-set UART: 8051's own UART and the other one is Elan's UART
- 1-set UART (16 Bytes Buffer)
- 1-set SPI (16 Bytes Buffer)
- 26 I/O pins, each supports pull-high, pull-low, normal drive/sink, and high drive/sink control functions
- Two standard 8051 16-bit timer/counters (TC0, TC1) with four operation modes (Mode 0~3)
- Two general purpose 16-bit timer/counters (TC3, TC4) with six modes: Timer/Counter/Capture/Window/Buzzer/PWM/PDO(programmable divider output) modes. It's clock source comes from PLL 48 MHz
- Three 16-bit Dual PWMs with either normal or trigger mode (need to be used together with CMP1), their clock sources come from PLL 48 MHz
- LED Driver

Packages

- 28-pin SSOP
- 24-pin SSOP
- 20-pin SSOP
- 32-pin QFN(4X4X0.6mm)
- On-chip Debug
- On-chip Debug system (OCD)
- Provide 6 ROM breakpoints, 1 IDATA/XDATA/SFR register breakpoint.
- Provide emulation instructions (such as free run, step into, system halt, wake-up, reset, etc.), inspection memory and registers information.

1.1 Block Diagram

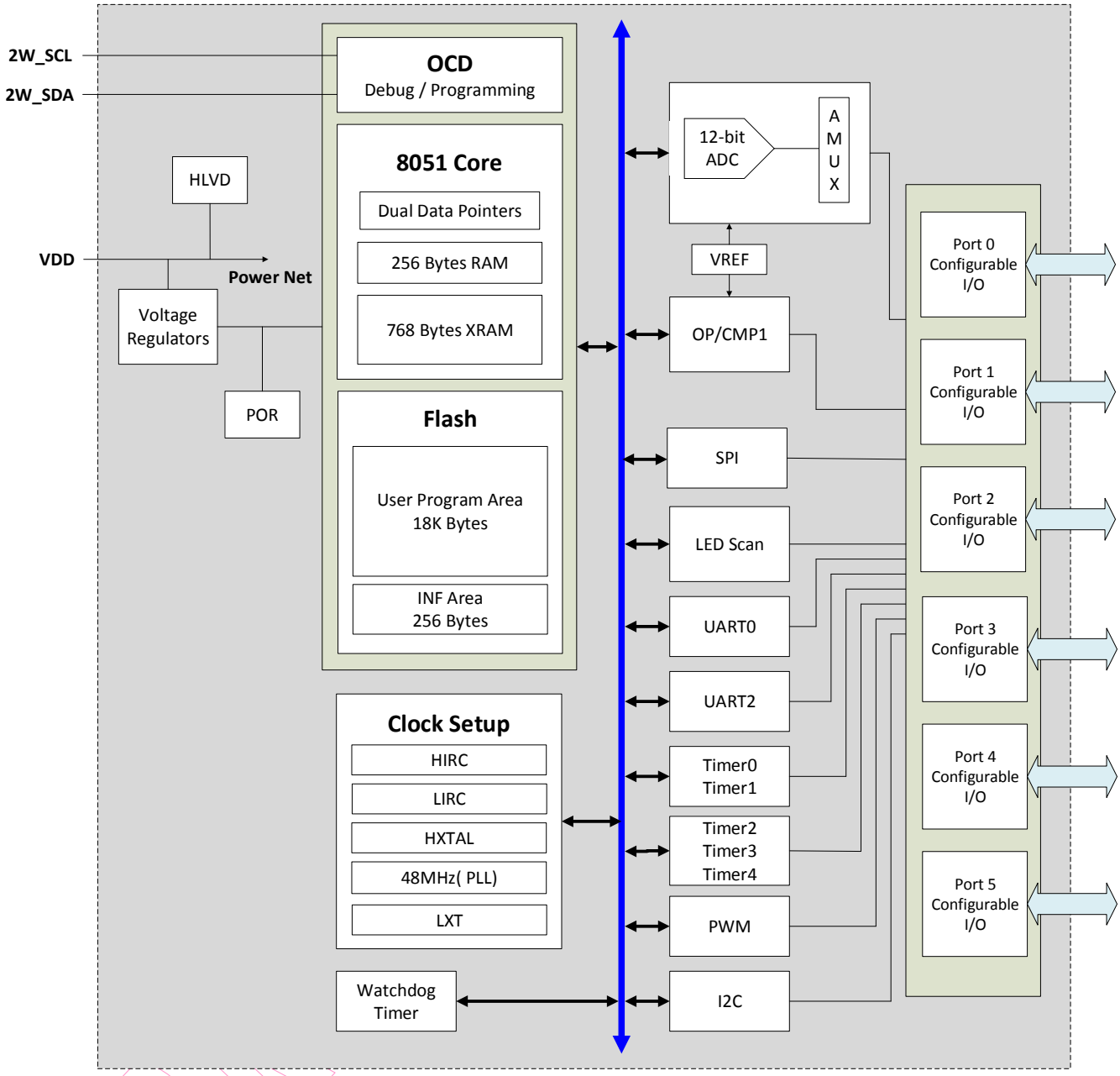


Figure 1 System Block Diagram

2 Pin Configurations

When I/O is multiplexing with analog and digital functions, its sequential priority is analog > digital > I/O. If the multiplexing pin is used with more than two analog functions or more than two digital functions, the hardware will not handle any anti-wrong action. Users should take note and take precaution of the condition.

2.1 Pin Configuration

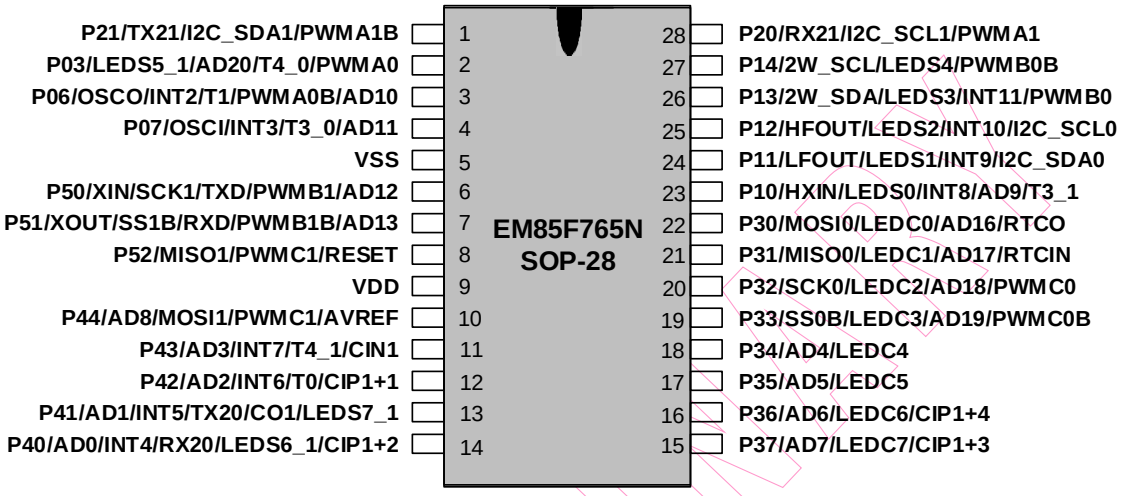


Figure 2- 1 SOP28 Pinout

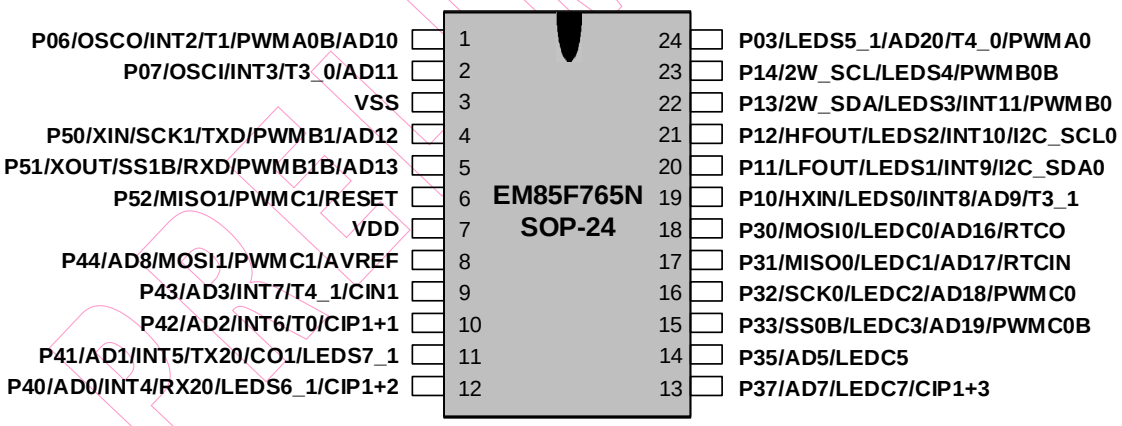


Figure 2- 2 SOP24 Pinout

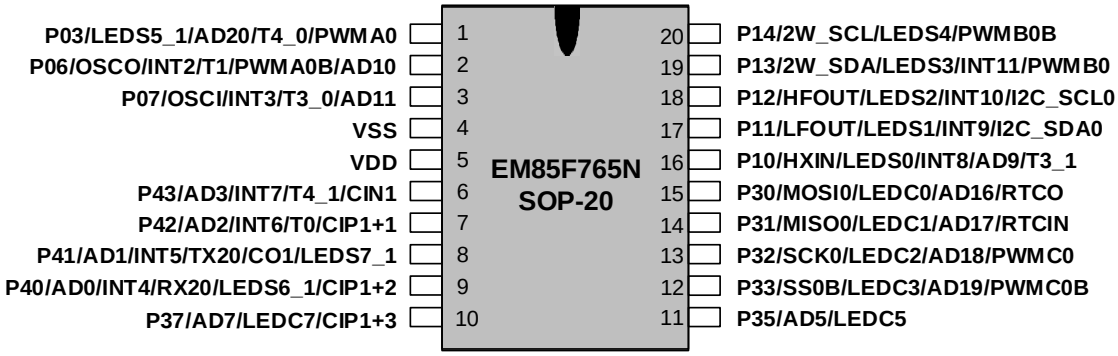


Figure 2- 3 SOP20 Pinout

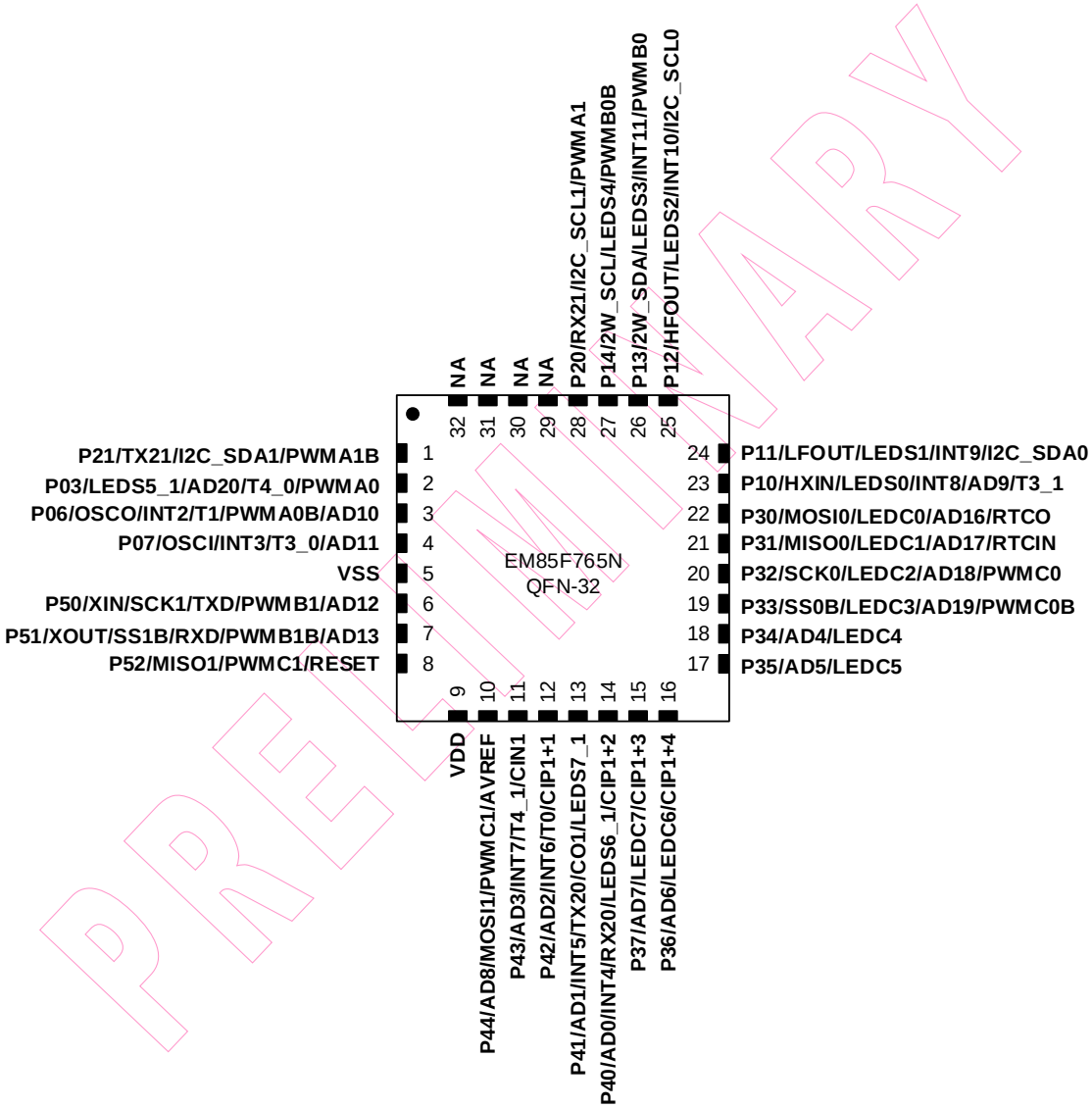


Figure 2- 4 QFN32 Pinout

2.2 Pin Description

Name	Function	Input Type	Output Type	Description
–	VDD	Power	–	Power
–	VSS	Power	–	Ground
P03/LEDS5_1/AD20/T4_0/PWMA0	P03	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	LEDS5_1	ST	CMOS	LED driver Segment Output 5_1
	AD20	AN	–	ADC input Channel 20
	T4_0	ST	CMOS	Timer 4 Channel 0
	PWMA0	ST	CMOS	PWMA Chanel 0 output
P06/OSCO/INT2/T1/PWMA0B/AD10	P06	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	OSCO	AN	–	High Frequency Crystal output
	INT2	ST	–	External Interrupt Pin 2
	T1	ST		8051's Timer 1
	PWMA0B			PWMA Chanel 0 Inverse output
	AD10	AN		ADC Input Channel 10
P07/OSCI/INT3/T3_0/AD11	P07	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	OSCI	AN	–	High Frequency Crystal input
	INT3			External Interrupt Pin 3
	T3_0			Timer 3 Channel 0
	ADC11	AN	–	ADC input Channel 11
P10/HXIN/LEDS0/INT8/AD9/T3_1	P10	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	HXIN	–	CMOS	Output pin of Comparator 2
	LEDS0	AN	–	LED driver Segment Output 0
	INT8	ST	CMOS	External Interrupt Pin 8
	AD9			ADC Input Channel 9
	T3_1			Timer 3 Channel 1
P11/LFOUT/LEDS1/INT9/I2C_SDA0	P11	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wakeup
	LFOUT	AN	–	Low Frequency Clock output
	LEDS1	AN	–	LED driver Segment Output 1
	INT9	AN	–	External Interrupt Pin 9
	I2C_SDA0	ST		I^2C channel 0 serial data line. It is open-drain

Name	Function	Input Type	Output Type	Description
P12/HFOUT/LEDS2/INT10/I2C_SCL0	P12	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	HFOUT	AN	–	Hi Frequency Clock Output
	LEDS2	AN	–	LED driver Segment Output 2
	INT10			External Interrupt Pin 10
	I2C_SCL0			I^2C Channel 0 Serial Clock line. It is open-drain
P13/2W_SDA/LEDS3/INT11/PWMB0	P13	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	2W_SDA	ST	CMOS	On-chip Debug System Data Pin
	LEDS3	ST	–	LED driver Segment Output 3
	INT11	ST	CMOS	External Interrupt Pin 11
	PWMB0			PWMB Chanel 0 Output
P14/2W_SCL/LEDS4/PWMB0B	P14	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	2W_SCL	AN	CMOS	On-chip Debug System clock pin
	LEDS4	ST	–	LED driver Segment Output 4
	PWMB0B	ST	CMOS	PWMB Chanel 0 Inverse Output
P20/RX21/I2C_SCL1/PWMA1	P20	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	RX21	AN	–	UART2 RX Channel 1
	I2C_SCL1	ST	–	I^2C Channel 1 Serial Clock line. It is open-drain
	PWMA1	ST	CMOS	PWMA Chanel 1 Output
P21/TX21/I2C_SDA1/PWMA1B	P11	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	TX21	AN	–	UART2 TX Channel 1
	I2C_SDA1	ST	CMOS	I^2C Channel 1 Serial Clock line. It is open-drain
	PWMA1B	–	AN	PWMA Chanel 1 Inverse Output
P30/MOSI0/LEDC0/AD16/RTCO	P30	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	MOSI0			SPI Channel0 Master Output Slave Input
	LEDC0	ST	–	LED driver Common Output 0
	AD16	ST	–	ADC Input Channel 16
	RTCO			Real Time Clock output

Name	Function	Input Type	Output Type	Description
P31/MISO0/LEDC1/AD17/RTCIN	P31	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	MISO0	–	CMOS	SPI Channel0 Master Input Slave Output
	LEDC1	–	CMOS	LED driver Common Output 1
	AD17			ADC Input Channel 17
	RTCIN			Real Time Clock Input
P32/SCK0/LEDC2/AD18/PWMC0	P32	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	SCK0	AN	–	SPI Channel0 Clock line
	LEDC2			LED driver Common Output 2
	AD18	AN	–	ADC Input Channel 18
	PWMC0			PWMC Chanel 0 Output
P33/SS0/LEDC3/AD19/PWMC0B	P33	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	SS0	ST	–	SPI Channel0 Slave select
	LEDC3	AN	–	LED driver Common Output 3
	AD19	AN		ADC Input Channel 19
	PWMC0B			PWMC Chanel 0 Inverse Output
P34/AD4/LEDC4	P34	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	AD4	ST	–	ADC input Channel 4
	LEDC4	ST	CMOS	LED driver Common Output 4
P35/AD5/LEDC5	P35	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	AD5	ST	–	ADC input Channel 5
	LEDC5	ST	CMOS	LED driver Common Output 5
P36/AD6/LEDC6/CIP1+4	P36	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	AD6	AN	–	ADC input Channel 6
	LEDC6	ST	–	LED driver Common Output 6
	CIP1+4	ST	CMOS	Positive input end Number4 of Comparator
P37/AD7/LEDC7/CIP1+3	P37	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	AD7	AN	–	ADC Input Channel 7
	LEDC7	ST	CMOS	LED driver Common Output 7
	CIP1+3	–	AN	Positive input end Number3 of Comparator

Name	Function	Input Type	Output Type	Description
P40/AD0/INT4/RX20/LEDS6_1/CIP1+2	P40	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	AD0	ST	–	ADC Input Channel 0
	INT4	ST	–	External Interrupt Pin 4
	RX20			UART2 RX Channel 0
	LEDS6_1			LED driver Segment Output 6_1
	CIP1+2			Positive input end Number2 of Comparator
P41/AD1/INT5/TX20/CO1/LEDS7_1	P41	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	AD1	–	CMOS	ADC Input Channel 1
	INT5	–	CMOS	External Interrupt Pin 5
	TX20			UART2 TX Channel 0
	CO1			Output of Comparator
	LEDS7_1			LED driver Segment Output 7_1
P42/AD2/INT6/T0/CIP1+1	P42	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	AD2	AN	–	ADC Input Channel 2
	INT6	AN	–	External Interrupt Pin 6
	T0			8051's Timer 0
	CIP1+1			Positive input end Number1 of Comparator
P43/AD3/INT7/T4_1/CIN1	P43	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	AD3	AN	–	ADC Input Channel 3
	INT7	AN	–	External Interrupt Pin 7
	T4_1			Timer 4 channel 1
	CIN1			Negative input end of Comparator
P44/AD8/MOSI1/PWMC1/AVREF	P44	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	AD8	ST	–	ADC Input Channel 8
	MOSI1	ST	CMOS	SPI Channel1 Master Output Slave Input
	PWMC1			PWMC Chanel 1 Output
	AVREF			External voltage reference pin for ADC

Name	Function	Input Type	Output Type	Description
P50/XIN/SCK1/TXD/PWMB1/AD12	P50	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	XIN	ST	–	Low Frequency Crystal input
	SCK1	ST	CMOS	SPI Channel1 serial clock line.
	TXD			UART0 TX pin
	PWMB1			PWMB Chanel 1 Output
	AD12			ADC input Channel 12
P51/XOUT/SS1/RXD/PWMB1B/AD13	P51	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	XOUT	AN	–	Low Frequency Crystal output
	SS1	ST	–	SPI Channel1 Slave select
	RXD	ST	CMOS	UART0 RX pin
	PWMB1B			PWMB Chanel 1 Inverse Output
	AD13			ADC input Channel 13
P52/MISO1/PWMC1/RESET	P52	ST	CMOS	Bidirectional I/O pin with programmable pull-down, pull-high, open-drain
	MISO1	AN	–	SPI Channel1 Master Input Slave Output
	PWMC1	ST	CMOS	PWMC Chanel 1 Output
	RESET	–	AN	External reset pin

3 Memory Organization

The MCU On-Chip Memory Map as shown in Figure 3.1, has its Flash Memory divided into Flash Program Memory (18K bytes) and Flash INF Area (256 bytes). The INF Area is divided into four Pages, each Page has different uses: the Code Option provides hardware options allowing users to set their own setting; Info Pages 1 ~ 2 information store info in the MCU and retain the info after power down; The Data Memory is divided into Internal Data Memory (IDATA), Special Function Registers (SFR) and External Memory (XDATA). Each of the above-mentioned memory blocks is described in detail in the subsequent sections.

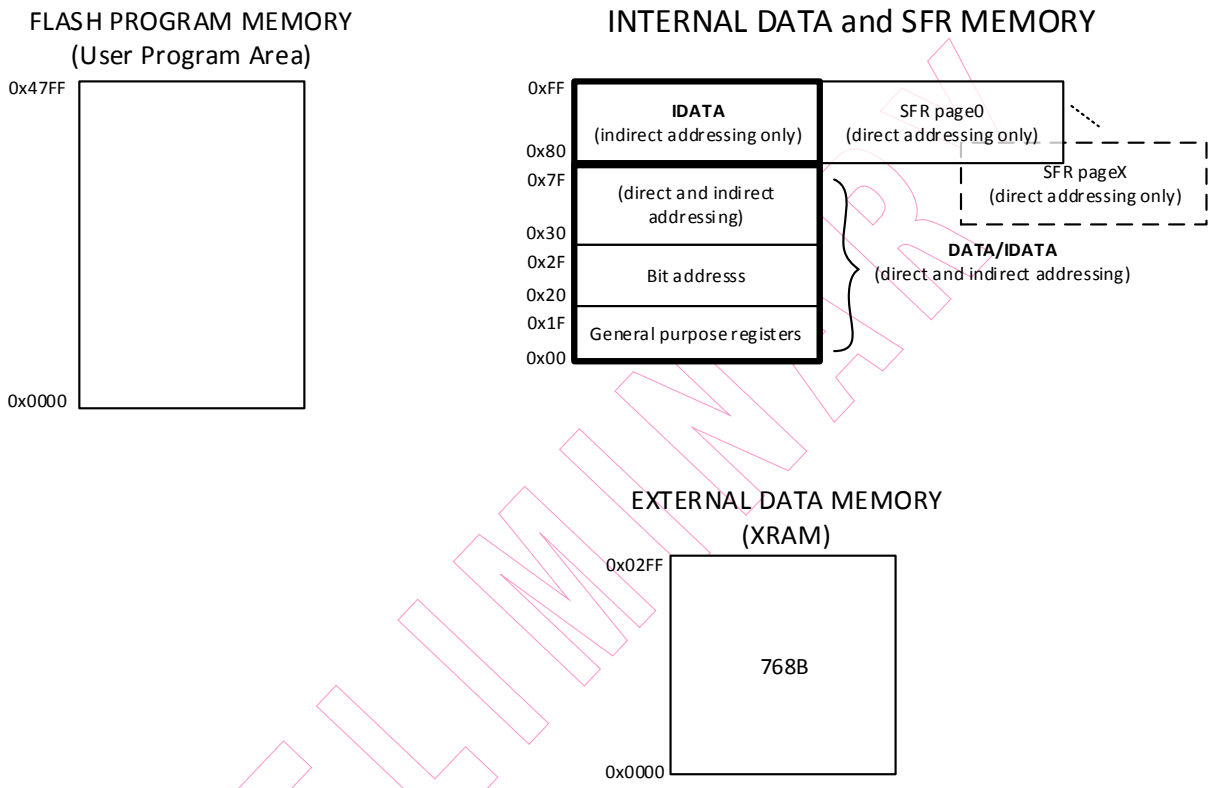


Figure 3-1On-Chip Memory Map

3.1 Flash Program Memory

The MCU incorporates 18K bytes of Flash Program Memory space which can be accessed through the use of IAP. A memory space size can be identified by setting SFR. However, if that space is under enhanced protection, it cannot be accessed by IAP Code as it is in the space without enhanced protection. For more details, refer to Section 7.3 Flash Security.

3.2 Internal Data Memory

Included in the MCU are 256 bytes of internal data memory, divided into lower 128 bytes and upper 128 bytes. This internal RAM is mapped into the data memory space from 0x00 through 0xFF.

The lower 128 bytes of data memory are accessible either by direct or indirect addressing mode, and used for General Purpose Registers and scratch pad memory. Locations 0x00~0x1F are addressed as four banks of General Purpose Registers, each bank has 8-bytes width registers. The subsequent 16 bytes, from locations 0x20~0x2F, can be addressable either as bytes or as 128-bit locations, which can be accessed through direct addressing mode.

The upper 128 bytes of data memory are accessible by indirect addressing only. This area occupies the same address space as the Special Function Registers (SFR) but is physically apart from the SFR space. The addressing mode utilized by an instruction when accessing locations above 0x7F determines whether the CPU is accessing the SFRs or the upper 128 bytes of data memory space. Instructions that employ direct addressing will access the SFR space. Instructions that use indirect addressing above 0x7F access the upper 128 bytes of data memory. For detailed description on SFRs configuration, refer to Section 3.3.

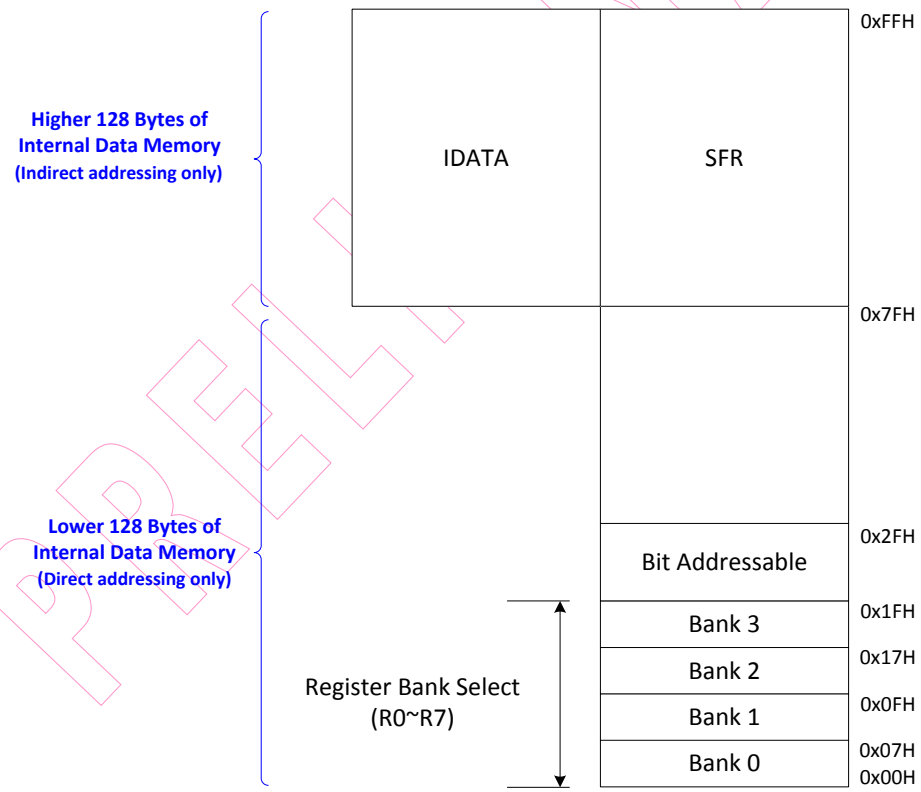


Figure 3- 2Internal Data Memory

3.3 External Data Memory

This External Data Memory (XDATA) of MCU has 768 bytes of space, and MOVX instruction can be used to access. There are two methods to access XRAM by MOVX instruction. The first method uses DPTR, this 16-bit register containing XRAM address that can be read or written to. The second method uses R0/R1 and XPAGE register as XRAM address. These two methods are illustrated below:

16-Bit MOVX Example:

MOV DPTR, #0x0123H ; load DPTR with 16-bit address to read (0x0123)

MOVX A, @DPTR ; load contents of 0x0123 into accumulator A

8-Bit MOVX Example:

MOV XPAGE, #0x01H ; load high byte of address into XPAGE

MOV R0, #0x23H ; load low byte of address into R0 (or R1)

MOVX A, @R0 ; load contents of 0x0123 into accumulator A

XPAGE: XDATA Page

Bit	7	6	5	4	3	2	1	0
Name	XPAGE.7	XPAGE.6	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA9; SFR Page = 0

(Select which 256-byte page of XDATA is currently accessible by MOVX @ Ri instructions. However, this value will not output on P2.)

3.4 Special Function Registers (SFR)

The Special Function Registers (SFRs) is composed of data memory locations from 0x80~0xFF which can be directly accessed anytime using direct addressing mode. SFRs with addresses ending of 0x0 or 0x8 are bit-addressable or as byte-addressable.

This MCU expands the SFR space from its original 128 bytes by paging mechanism. There are three SFR pages which are selected by PAGESW register.

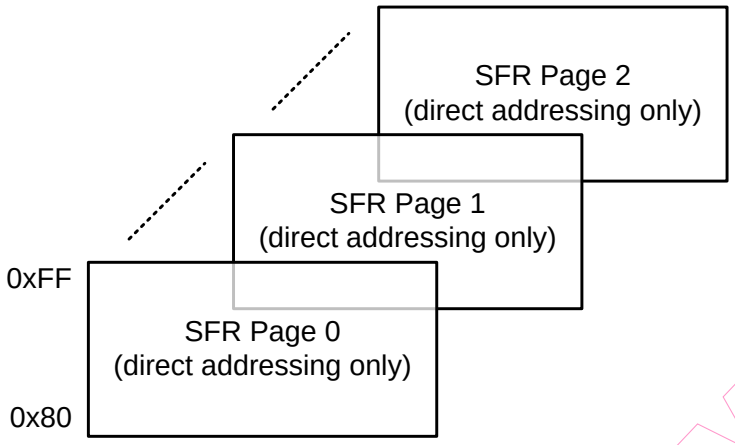


Figure 3- 3SFR Memory Map

PAGESW: Page Switch

Bit	7	6	5	4	3	2	1	0
Name	PAGESW .7	PAGESW .6	PAGESW .5	PAGESW .4	PAGESW .3	PAGESW .2	PAGESW .1	PAGESW .0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x86; SFR Page = All Pages

(When PAGESW is set above 0x02, except the Common area, reading any address will result in “0” value and writing is Don’t Care)

3.4.1 SFR Map

PAGE0	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
F8	P5	P2M0	P2M1	P3M0	P3M1	PHCON1	PLCON1	PHDSC1
F0	B	P0M0	P0M1	P1M0	P1M1	PHCON0	PLCON0	PHDSC0
E8	P4	ICEN	QBODICEN			OPCMP1CR2	OPCMP1CR1	COSF1
E0	ACC						SDPWMCR1	CMPNRCR1
D8	PSW1	TC4CR1	TC4CR2	TC4DAL	TC4DAH	TC4DBL	TC4DBH	TC4CR3
D0	PSW	TC3CR1	TC3CR2	TC3DAL	TC3DAH	TC3DBL	TC3DBH	TC3CR3
C8		FLKEY	FLCR	EPKEY	EPPOINTL	EPPOINTH	WDTKEY	WDTCR
C0	RSTSC	SPISR2	SPITDBR	SPIRDBR	SPITDBC	SPIRDBC	SPIBUFPTR1	SPIBUFPTR2
B8	IP	EIP1	EIP2	EIP3		SPICON1	SPICON2	SPISR1
B0	P3	EIE1	EIE2	EIE3			EIESC1	
A8	IE	XPAGE			SPIRDAFBC		PRST	
A0	P2							
98	SCON0	SBUF0	ADCVL	ADCVH	ADER3		HLVDCR	HLVDCR1
90	P1	ADCR1	ADCR2	ADISR	ADER1	ADER2	ADDL	ADDH
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON0	CKCON1
80	P0	SP	DPL	DPH	DPL1	DPH1	PAGESW	PCON

Figure 3- 4SFR Memory Map in Page 0

PAGE1	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
F8	P5							
F0	B	P4M0	P4M1	P5M0	P5M1	PHCON2	PLCON2	PHDSC2
E8	P4							
E0	ACC	PWMCCR1	PRDCL	PRDCH	DTCL	DTCH	TMRCL	TMRCH
D8	PSW1	PWMBCR1	PRDBL	PRDBH	DTBL	DTBH	TMRBL	TMRBH
D0	PSW	PWMACR1	PRDAL	PRDAH	DTAL	DTAH	TMRAL	TMRAH
C8		DeadTR	PWMENCR	PWMACR2	PWMBCR2	PWMCCR2		
C0	RSTSC	I2CTXLEN	I2CRXLEN	I2CSTASU	I2CSTAHD	I2CSCLFIR	I2CSDAFIR	I2CCR4
B8	IP	I2CRXAF	I2CTO1	I2CTO1R	I2CTO2	I2CTO2R	I2CTO3	I2CTO3R
B0	P3	I2CCR1	I2CCR2	I2CSA	I2CDB	I2CDAL	I2CDAH	I2CSF
A8	IE	LCDCR1						
A0	P2	LCDADDR	LCDDATA		LEDFR			
98	SCON0	SBUF0	CO1TRL	CO1TRH				
90	P1	RTCCON	RTCDATAH	RTCDATAL	ALARM	RTCCAL1	RTCCAL2	RTCKEY
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON0	CKCON1
80	P0	SP	DPL	DPH	DPL1	DPH1	PAGESW	PCON

Figure 3- 5 SFR Memory Map in Page 1

PAGE2	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
F8	P5			LIRCTR				
F0	B	UR2CR	UR2S	UR2TD	UR2RDL	UR2RDH	UR2S2	UR2BRS
E8	P4	UR2BUFTXSF	UR2BUFRXSF	UR2BUFTXIM	UR2BUFRXIM	UR2TXLEN	UR2RXLEN	
E0	ACC							
D8	PSW1							
D0	PSW							
C8		EIOPWMA	EIOPWMB	EIOPWMC		EIOT3	EIOT4	
C0	RSTSC	EIOCOM1	EIOSEG1				EIOSEG11	
B8	IP	EXSF2	EXSF3	EXSF4		EIES2	EIES3	EIES4
B0	P3	EIOUART2	EIOI2C0	EIOSPI0		EXEN0	EXEN1	
A8	IE	ARITH	MA0	MA1	MA2	MA3	MB0	MB1
A0	P2	PLL1CR						
98	SCON0	SBUF0						
90	P1							
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON0	CKCON1
80	P0	SP	DPL	DPH	DPL1	DPH1	PAGESW	PCON

Figure 3-6 SFR Memory Map in Page 2

3.5 Stack

The stack area can be located within the 256-byte data memory and it is designated using the Stack Pointer (SP, 0x81) SFR. The Stack Pointer SP will point to the last location used. The next value pushed onto the stack is located at SP+1 and SP is then incremented (The Stack Pointer is incremented prior to storing data during PUSH and CALL execution, and decremented after data is popped during POP, RET and RETI execution). A reset initializes the Stack Pointer SP to Location 0x07, as shown in Figure 3-7. Hence, the first value that is pushed onto the stack is placed at Location 0x08, which is also R0, the first register of Register Bank 1. Therefore, if it is necessary to use more than one register banks, SP should be initialized to a data memory location that is not used for data storage. The depth of the stack can be extended up to 256 bytes.

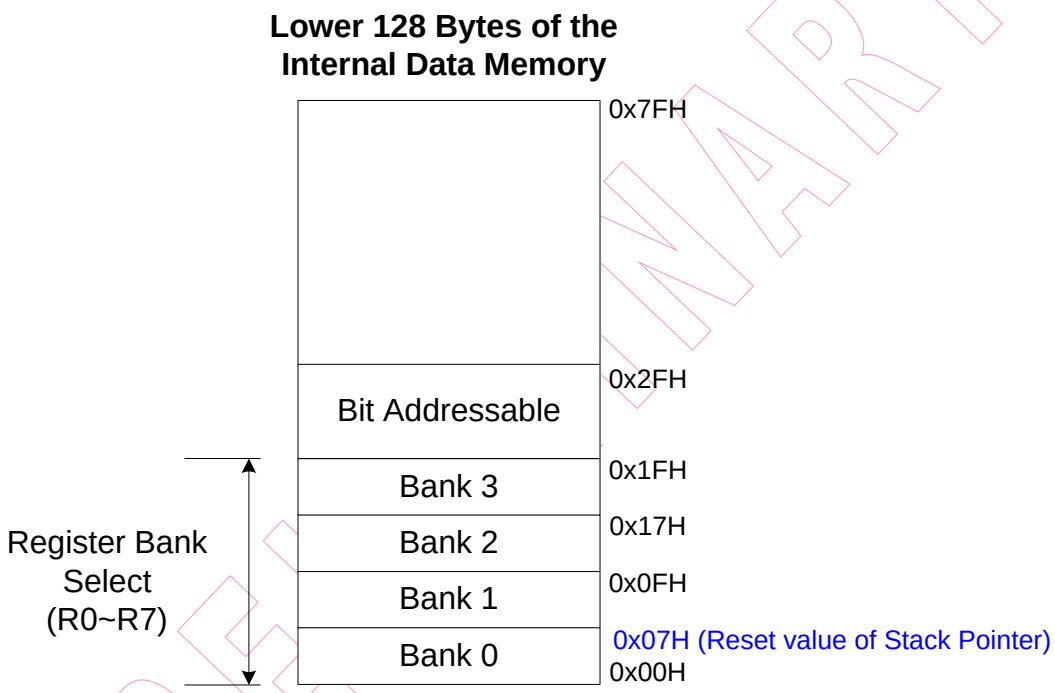


Figure 3-7 Stack Pointer Reset Value

SP: Stack Pointer

Bit	7	6	5	4	3	2	1	0
Name	SP.7	SP.6	SP.5	SP.4	SP.3	SP.2	SP.1	SP.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	1	1	1

SFR Address = 0x81; SFR Page = All Pages

4 System Clock

The EM85F765N System clock uses the control register of SCLKS (CKCON1.0) or HLFS (Code Option 2 Bit 3) to enable choosing and switching between high-speed oscillation source (FHS) and low-speed oscillator source (FLS). In addition, frequency through FOSCPS (CKCON1 Bits 1 ~ 3) can be selected as System Clock in order to reduce the Power consumption by having CPU running at low speeds.

4.1 Clock Structure

This MCU's System Clock has a variety of oscillator sources (F_{HS} by the HSOSC (Code Option 2 Bits 0 ~ 1) that can choose either High IRC, High XTAL or HXIN pin as an oscillation source) and a variety of selections. Its application range is very wide. While providing high performance, it also has power consumption reduction characteristics. The detailed Clock Tree is shown in Figure 4-1 below. Note that only when FHS selects IRC as the oscillator source under PLL, will the output frequency (FPLL) be enabled.

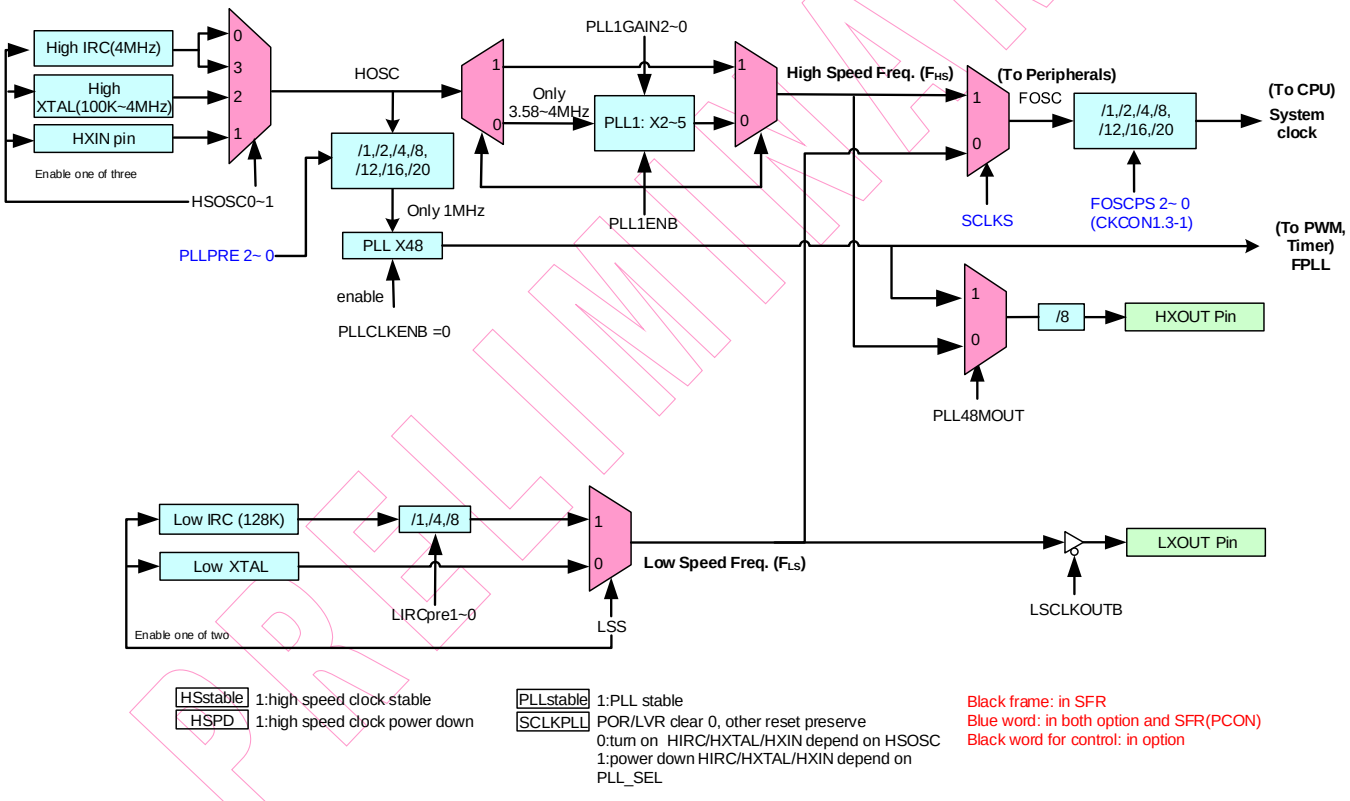


Figure 4-1 System Clock Diagram

4.2 Crystal Oscillator/Ceramic Resonators (XTAL)

In most applications, Pin OSCI and Pin OSCO can be connected with a crystal or ceramic resonator to generate oscillation and such circuitry are depicted in the following Figures. The same thing applies whether it is in HXT mode or in LXT mode. The following Table provides the recommended values of C1 and C2. Since each resonator has its own attribute, user should refer to its specification for appropriate values of C1 and C2. The serial resistor, RS, may be necessary for AT strip cut crystal or low frequency

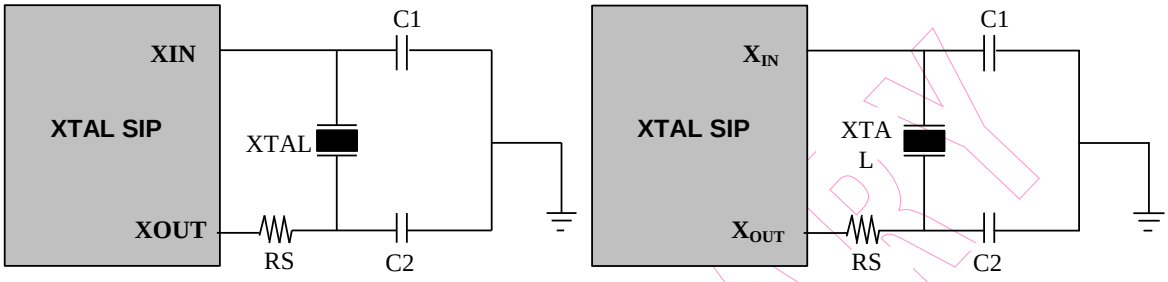


Figure 4-2 Crystal/Resonator Circuits

Oscillator Type	Frequency Mode	Frequency	C1(pF)	C2(pF)
Ceramic Resonators	HXT1 (100K~1MHz)	100KHz	60pF	60pF
		200KHz	60pF	60pF
		455KHz	100pF	100pF
		1MHz	30pF	30pF
	HXT2 (1M~4MHz)	1.0 MHz	30pF	30pF
		2.0 MHz	30pF	30pF
		4.0 MHz	20pF	20pF
Crystal Oscillator	LXT1 (32.768KHz)	32.768KHz	40pF	40pF
	HXT1 (100K~1MHz)	100KHz	60pF	60pF
		200KHz	60pF	60pF
		455KHz	40pF	40pF
		1MHz	30pF	30pF
	HXT2 (1~4MHz)	1.0 MHz	30pF	30pF
		2.0 MHz	30pF	30pF
		4.0 MHz	20pF	20pF

4.3 Internal RC Oscillator Mode

The MCU only offer a internal RC mode with default frequency value of 4 MHz.

CKCON1: Clock Control 1

Bit	7	6	5	4	3	2	1	0
Name	-	RCXT2	RCXT1	RCXT0	FOSCPS2	FOSCPS1	FOSCPS0	SCLKS
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	Code Option	Code Option	Code Option	Code Option	Code Option	Code Option	Code Option

SFR Address = 0x8F; SFR Page = All Pages

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bits 6~4: HIRC Clock Select, set to “000”

RCXT [2:0]	IRC (MHz)
000	4
001	Reserved
010	Reserved
011	Reserved
100	Reserved
101	Reserved
110	Reserved
111	4

Bits 3~1: FOSC pre-scaler

FOSCPS2	FOSCPS1	FOSCPS0	Function Description
0	0	0	FOSC/20
0	0	1	FOSC/16
0	1	0	FOSC/12
0	1	1	FOSC/10
1	0	0	FOSC/8
1	0	1	FOSC/4
1	1	0	FOSC/2
1	1	1	FOSC/1

Bit 0: System clock selection.

Setting this bit activates CPU to enter low-speed operation.

0: CPU frequency using Low Speed Frequency (F_{LS}).

1: CPU frequency using High Speed Frequency (F_{HS}).

PLL1CR: PLL1 Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	PLL1GAIN2	PLL1GAIN1	PLL1GAIN0	PLL1ENB
Type	R	R	R	R	R/W	R/W	R/W	R/W
Default	1	1	1	1	Code Option	Code Option	Code Option	Code Option

SFR Address = 0XA1; SFR Page = 0x2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bits 3~1 (PLL1GAIN2~0): PLL Gain.

PLL1GAIN2	PLL1GAIN1	PLL1GAIN0	GAIN
0	0	0	Reserved
0	0	1	Reserved
0	1	0	Reserved
0	1	1	5
1	0	0	4
1	0	1	3
1	1	0	2.5
1	1	1	2

Bit 0 (PLL1ENB): PLL Enable.

0: Enable.

1: Disable.

5 System Reset

The MCU has six Reset Sources (including OCD Reset) including. Power-On Reset, External Reset, Watchdog Timer Reset, Software Reset and Low Voltage Reset, respectively. When the MCU enters Reset state, all SFR are initialized to their preset values and the program counter (PC) is reset. The program will start execution at Reset Vector (0000H) during Reset, the internal data memory contents will not change (the data stored prior to Reset will remain unchanged).

After POR and BOR events occurred, a period of TWARM-UP time during which the Latch Code Option operation is carried out is needed before the CPU can execute the program. The TWARM-UP period is typically 64ms (with Latch Code Option starts operating at 60ms). Figure 5.1 below shows the timing diagram of POR (1.1/1.2V) events. It also illustrates that after the first Latch Code Option of BOR1, BOR position is shifted to BOR2.

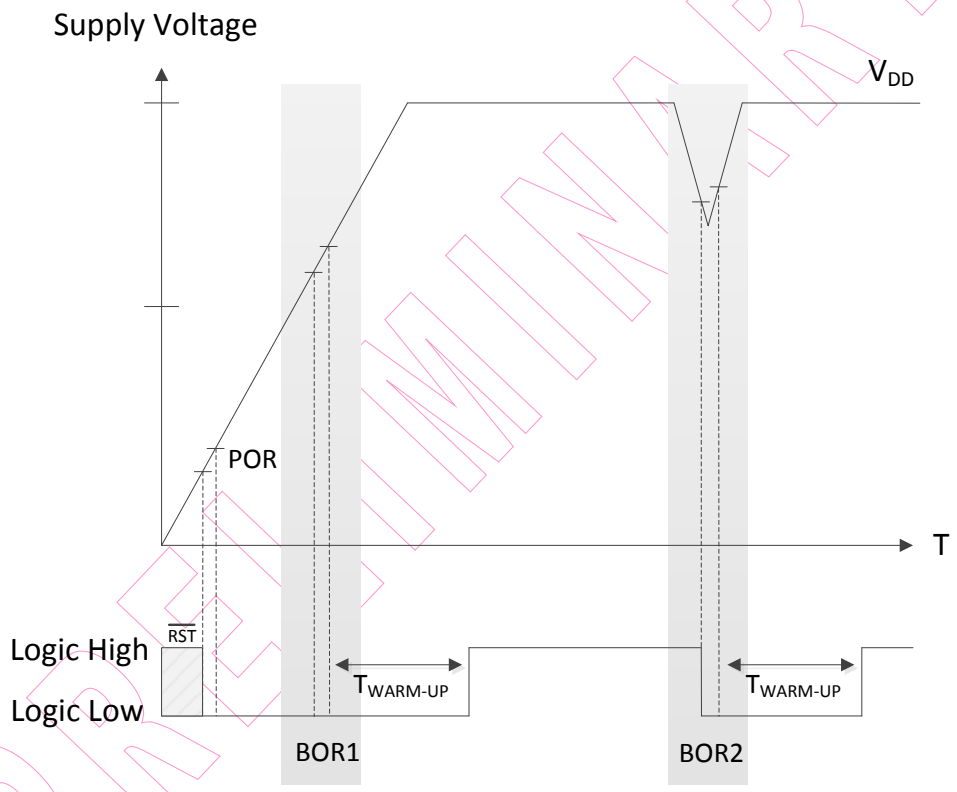


Figure 5-1 POR and BOR Reset Timing

5.1 Power on Reset

Power-On Reset (POR) will generate System Reset during power-up. At the same time, the CPU will stop operating and maintain at Reset state until VDD rises above POR voltage. Then, when VDD drops below the POR voltage, the System Reset returns to a Reset state again. When POR occurs, PORSF (RSTSC.0) will be set to 1.

5.2 External Reset

External RST Pin (low electric potential) provides a means for external circuitry to force the MCU into reset state, in order to avoid noise interference caused by Reset. The RST Pin must be connected to Pull-High or with decoupling capacitors. When Reset is activated by External Reset, ERSF (RSTSC.1) is set to 1.

5.3 Watchdog Timer Reset

With the WDT activated (ENWDT of Code Option and WDTCR.7 of SFR are set to 1) and counting started, System Reset is generated when WDT overflows and then the WTSF (RSTSC.2) is set to 1. For more details on the WDT control, refer to Section 10 Watchdog Timer.

5.4 Software Reset

Setting SWRSF (RSTSC.7) to 1 will force generate System Reset. When Reset is activated by the Software Reset, SWRSF (RSTSC.7) will be set to 1.

5.5 Brown out Reset (BOR)

This MCU provides a Brown out Reset function. When the system voltage drops below the set BOR voltage level, the system will reset. At the same time, BORSF will be set to 1. This could help user to determine whether the system reset is caused by BOR or not BOR level can be set by BORS3~0 (Code Option 0 Bit 3~0).

5.6 Peripheral Communication interface Reset

This MCU provides reset of peripheral communication interface reset through PRST registers setting. It enables user to return to the initial communication interface state status when an unknown or uncontrollable confusion condition occurs.

RSTSC: Reset Source

Bit	7	6	5	4	3	2	1	0
Name	SWRSF	-	-	-	BORSF	WTSF	ERSF	PORSF
Type	R/W	R	R	R	R/W	R/W	R/W	R/W
Reset	Varies	0	0	0	Varies	Varies	Varies	Varies

SFR Address = 0xC0; SFR Page = All Pages

Bit 7: Software Reset Force and Flag.

Write:

0: No effect

1: Set to “1” to cause a system reset if SWRSF was cleared by software.

Read:

0: No software reset occurs.

1: Software reset caused the system reset.

Bits 6~4: Reserved. Read = 0, Write = Don't Care.

Bit 3: BOR Flag.

0: No effect

1: Set to 1 if BOR caused the system reset. It must be cleared by software.

Bit 2: Watchdog Reset Flag.

0: No effect

1: Set to 1 if a watchdog reset occurs. It must be cleared by software.

Bit 1: External Reset Flag.

0: No effect

1: Set to 1 if RST pin caused the reset. It must be cleared by software.

Bit 0: Power-On Flag.

0: No effect

1: Set to 1 anytime a power-on occurs. It must be cleared by software.

P: Previous status before reset

Reset Event	Reset Flag				
	SWRSF	BOR	WTSF	ERSF	PORSF
Power-On Reset	0	1	0	0	1
BOR Voltage Reset	0	1	0	0	P
Watchdog Reset	P	P	1	P	P
External Reset	P	P	P	1	P
Software Reset	1	P	P	P	P
OCD Reset	0	0	0	0	1

Table 5- 1 Reset Flag Status after Reset Event Occurred

PRST: Peripheral Reset

Bit	7	6	5	4	3	2	1	0
Name	-	-	UART2RST		UART0RST	SPIRST	-	I2CARST
Type	R	R	R/W	R	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAE; SFR Page = 0

Bits 7~6: Reserved. Read = 0, Write = Don't Care.

Bit 5: UART2 reset bit.

0: No effect

1: Reset UART2 registers and pin state as default with initialize state machine.

Bit 4: Reserved. Read = 0, Write = Don't Care.

Bit 3: UART0 reset bit.

0: No effect

1: Reset UART0 registers and pin state as default with initialize state machine.

Bit 2: SPI reset bit.

0: No effect

1:Reset SPI registers and pin state as default with initialize state machine.

Bit 1: Reserved. Read = 0, Write = Don't Care.

Bit 0: I2CA reset bit.

0: No effect

1: Reset I2CA registers and pin state as default with initialize state machine.

6 Power Management

This MCU contains five power management modes which meet a variety of application needs. These five modes are Normal Mode, Idle Mode, Power-Down Mode, Slow Mode and Green Mode. User can set HLFS (Code Option 2 Bit 3) to decide which MCU initial power management mode is to be used after POR or Reset event.

6.1 Operation Mode Structure

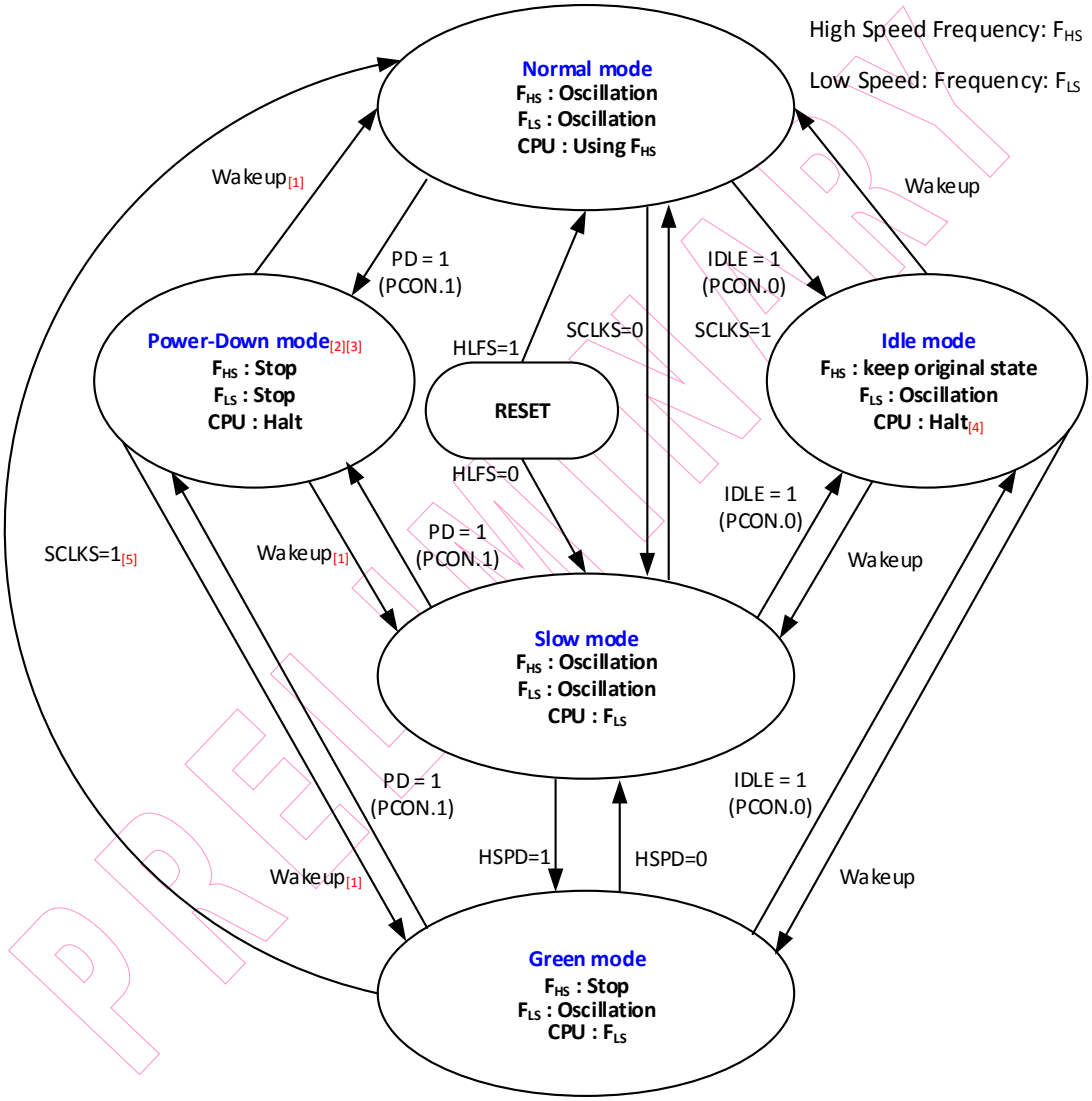


Figure 6-1Operation Mode Structure

Notes:

[1]. Valid Wakeup Source in Power-Down mode, for detail refer to Section 6.1.3

[2]. F_{LS} will keep running when WDT is working in Power-Down Mode.

[3]. F_{HS} will keep running when ADC is working in Power-Down Mode.

[4]. CPU is halt, but system clock is non-stop.

[5]. After setting SCLKS during Green mode, the CPU will return to Normal mode and the hardware will automatically clear HSPD.

CPU Mode	HSPD	SCLKS	PD	IDLE
Normal Mode	0	1	0	0
Idle Mode	X	X	0	1
Power-Down Mode	X	X	1	X
Slow Mode	0	0	0	0
Green Mode	1	0	0	0

Table 6- 1 CPU Mode Control Bit Summary

Notes:

1. If PD and Idle is 1, CPU enter Power-Down Mode.
2. HSPD can be set to 1 when CPU is in low speed mode.

6.1.1 Normal Mode

During Normal Mode, the high-speed oscillation source (FHS) and low-speed oscillation source (FLS) a real active. Then the CPU uses the high-frequency oscillation source for high-speed operation and all analog and digital peripheral remain active. When user needs to reduce the power consumption user can set SCLKS (CKCON1.0) to allow the CPU to use the low frequency oscillation source for low-speed operation (enter into Slow Mode). However, FHS oscillation source does not stop automatically. User must set the software HSPD (CKCON0.0) to close FHS in order to reduce power consumption. For detailed information on Slow Mode characteristics, refer to Section 6.1.4.

6.1.2 Idle Mode

To enter Idle Mode, select bits (PCON.0). The CPU will stop running while all analog and digital peripheral remain active. Under Idle Mode, any active Interrupt or Reset events can awake CPU. Note that after entering idle mode, FHS will maintain its original oscillation state before Idle Mode. That is, if of the state of FHS was activated before going into Idle Mode, it will remain in sustained oscillation after entering Idle Mode. Likewise, if FHS has stopped oscillating before going into Idle Mode, after entering Idle Mode, it will remain in none-oscillating state.

6.1.3 Power-Down Mode

Setting the Power-Down mode select bits (PCON.1). Let the CPU into the power-down mode. When the CPU enters into Power-Down mode, certain peripheral will continue to run the function it was operating before entering into this mode. After each of the peripheral corresponding Wake-Up Source event is triggered, it will then wake-up to exit from this mode. Note that after user set the instruction for the CPU to enter into Power-Down/Idle mode, at least add one "NOP" instruction to ensure interrupt execute immediately after Wake-up.

Notes:

1. Before entering this mode, ADC is in enabled status. After entering this mode, FHS will not stop.
2. Before entering this mode, WDT is in enabled status. After entering this mode, FLS will not stop.
3. Before entering this mode, I2C is in enabled status. CPU will wake up when a valid I2C Address is received.
4. Before entering this mode, Pin-Change is in enabled status. CPU will wake up when the interrupt event occurs.
5. Before entering this mode, CMP1 are in enabled status. CPU will wake up when the interrupt event occurs.

6.1.4 Slow Mode

In Normal Mode, set SCLKS from 1 to 0 to enter into Slow Mode. Under this mode, high-speed oscillation source (FHS) and low-speed oscillation source (FLS) are both active, but the CPU is running at low speed with low frequency oscillation source. Hence, under this mode, low power consumption is achieved. Furthermore, after setting HSPD (CKCON0.0) to close FHS, it will further reduce power consumption (enter into Green Mode). For detailed features of Green Mode, refer to Section 6.1.5.

6.1.5 Green Mode

While in Slow Mode, set HSPD (CKCON0.0) to enter into Green Mode. This mode offers lower power consumption than under Slow Mode. Note that setting SCLKS from 0 to 1 while in Green Mode will switch Green Mode back to Normal Mode, whereas when HSPD is set to 0, it will switch Green Mode back to Slow Mode.

Only after FHS oscillator source stabilizes will the CPU oscillator source switch to FHS. User can check the HS stable (CKCON0.1) flag to confirm whether FHS oscillation source is in stable state.

6.1.6 Warm-Up Time

When the CPU is in mode switching or in system resetting, the oscillation source needs come time to stabilize. The needed time will vary depending on the CPU operating mode or on the type of oscillation source being used. Details are shown in the following Table 6.2 and Table 6.3. Note that FHS are listed in both tables already in steady state or before Reset or CPU mode switching event.

WSTO: Waiting Time from Start-to-Oscillation, N: Normal mode, G: Green mode, S: Sleep mode, I: Idle mode.

POR: Power-On Reset, BOR: Low voltage Reset, EXR: External Reset, SWR: Software Reset, WDT: Watchdog Reset.

Mode	F_{HS}	F_{LS}	POR / BOR	EXR / WDT / SWR			
				N/S	G	P/I(HSPD=1)	I(HSPD=0)
HLFS = 1 Normal [1]	EXclk	RC/XT	64ms+12/F_{HS}	12/F_{HS}+2/128 KHz	12/F_{HS}+10/128KHz	12/F_{HS}+10/128KHz	300/F_{HS}
	RC		64ms+WSTO+12/F_{HS}	12/F_{HS}+2/128 KHz	WSTO + 12/F_{HS}+10/128KHz	WSTO + 12/F_{HS}+10/128KHz	
	XT		64ms+WSTO+512/F_{HS}	12/F_{HS}	WSTO+512/F_{HS}	WSTO+512/F_{HS}	
	RC/EXclk+PLL		64ms+WSTO+28/ 128KHz	12/F_{HS}	WSTO + 28/ 128KHz	WSTO + 28/ 128KHz	
	XT+PLL		64ms+WSTO+512/FHS+28/ 128KHz	12/F_{HS}+2/128KHz	WSTO+512/FHS+28/128KHz	WSTO+512/FHS+28/128KHz	
HLFS = 0 Slow [2]	ALL	RC	64ms+WSTO+8/128KHz	0/128KHz	0/128KHz	WSTO+8/128KHz	WSTO+8/128 KHz
HLFS = 0 Slow [2]	ALL	XT	64ms+WSTO+512/128KHz	0/128KHz	0/128KHz	WSTO+512/128KHz	WSTO+8/128 KHz

Table 6-2 Warm up Time from Reset Condition

Notes:

[1]. LS stable is used to check if the low-speed clock is stable. On the other hand. If the total warm-up time of low speed clock is shorter, then the low-speed clock LS stable is activated before the CPU works.

[2]. If HSPD = 0, HS stable is used to check if the high-speed clock is stable at the start of the program. On the other hand, if the total warm-up time of low speed clock is longer, then the high-speed clock HS stable is activated before the CPU works.

WSTO: Waiting Time from Start-to-Oscillation, N: Normal mode, G: Green mode, S: Sleep mode, I: Idle mode.

F_{HS}	F_{LS}	G → N[1]	I → N	P → N[2]
EXclk	RC/XT	$12/F_{HS}$	$300/F_{HS}$	$12/F_{HS}$+8/FLS
RC		WSTO +$12/F_{HS}$+8/128KHz		WSTO +$12/F_{HS}$+8/128KHz
XT		WSTO + $512/F_{HS}$		WSTO + $512/F_{HS}$
EXclk+PLL		28/ 128KHz		28/ 128KHz
RC+PLL		WSTO +28/ $F_{LS\ (LIRC)}$		WSTO +26/ $F_{LS\ (LIRC)}$
XT+PLL		WSTO + $512/F_{HS}$+28/ $F_{LS\ (LIRC)}$		WSTO + $512/F_{HS}$+26/ $F_{LS\ (LIRC)}$
F_{HS}	**F_{LS}**	**I → G / S**	**P → G (HSPD = 1)**	**P → S[3] (HSPD = 0)**
ALL	RC	$8/F_{LS}$	(WSTO) + $8/F_{LS}$	(WSTO) + $8/F_{LS}$
ALL	XT	$8/F_{LS}$	(WSTO)+$512/F_{LS}$	(WSTO)+$512/F_{LS}$

Table 6- 3 Warm up Time by Mode Change

Notes:

[1]. CPU wouldn't be on hold. Low speed clock is used as CPU clock until HS stable = 1, Then the high speed clock is used as CPU clock.

[2]. LS stable is used to check if the low speed clock is stable. On the other hand, if the total warm-up time of low speed clock is shorter, then the high speed clock LS stable is activated before the CPU works.

[3]. HSPD is set to 0 by hardware as waked-up. HS stable is used to check if the high speed clock is stable. On the other hand, if the total warm-up time of low speed clock is longer, then the high speed clock HS stable is activated before the CPU works.

PCON: Power Control

Bit	7	6	5	4	3	2	1	0
Name	SMOD0	-	-	-	DPS	-	PD	IDLE
Type	R/W	R	R	R	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x87; SFR Page = All Pages

Bit 7: UART0 double baud rate bit.

0: Disable double Baud rate of the UART0.

1: Enable double Baud rate of the UART0 in mode 1, 2, or 3.

Bits 6~4: Reserved. Read = 0, Write = Don't Care.

Bit 3: Data Pointer Select.

This bit is used to switch between DPTR and DPTR1.

0: Select DPTR.

1: Select DPTR1.

Bit 2: Reserved. Read = 0, Write = Don't Care.

Bit 1: Power-down control bit.

Setting this bit activates the Power-down operation. The PD bit is cleared automatically by hardware when waking up from power-down.

Bit 0: Idle mode control bit.

Setting this bit activates the Idle mode operation. The IDL bit is cleared automatically by hardware when waking up from Idle (If PD and IDLE is 1, CPU enters into Power-Down Mode).

CKCON0: Clock Control 0

Bit	7	6	5	4	3	2	1	0
Name	-	T1SC	T0SC	T1M	T0M	LSstable	HSstable	HSPD
Type	R	R/W	R/W	R/W	R/W	R	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8E; SFR Page = All Pages

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bits 6~3: refer to Chapter 14 Timer about Timer0 and 1.

Bit 2: Low Speed Frequency stable.

0: Low speed oscillator source is unused or not yet stable.

1: Low speed oscillator source is running and stable.

Bit 1: High Speed Frequency stable.

0: High speed oscillator source is unused or not yet stable.

1: High speed oscillator source is running and stable.

Bit 0: High Speed Frequency power down (Only affective under low speed mode).

0: High speed oscillator source is running.

1: High speed oscillator source is power down.

CKCON1: Clock Control 1

Bit	7	6	5	4	3	2	1	0
Name	-	RCXT2	RCXT1	RCXT0	FOSCPS2	FOSCPS1	FOSCPS0	SCLKS
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	code option	code option	code option	code option	code option	code option	code option

SFR Address = 0x8F; SFR Page = All Pages

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bits 6~4: HIRC Clock Select.

RCXT [2:0]	IRC (MHz)
000	4
001	4
010	4
011	4
100	4
101	4
110	4
111	4

Bits 3~1: FOSC pre-scaler.

FOSCPS2	FOSCPS1	FOSCPS0	Function Description
0	0	0	FOSC/20
0	0	1	FOSC/16
0	1	0	FOSC/12
0	1	1	FOSC/10
1	0	0	FOSC/8
1	0	1	FOSC/4
1	1	0	FOSC/2
1	1	1	FOSC/1

Bit 0: System clock selection.

Setting this bit activates CPU to enter low-speed operation.

0: CPU frequency using Low Speed Frequency (F_{LS}).

1: CPU frequency using High Speed Frequency (F_{HS}).

6.2 Peripheral Power Down

The EM85F765N supply peripheral power down function, user can power down peripherals to reduce power consumption. Actually, the control just do peripheral clock gating, it is not really to power down peripheral.

DEVPD1: Device1 Power Down

Bit	7	6	5	4	3	2	1	0
Name	-	PWMAPD	Timer3PD	ADCPD	SPIPD	OPCMP1PD	LVDPD	RTCPD-
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAD; SFR Page = 1

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Setting this bit to power down PWMA.

0: PWMA power down is disabled.

1: PWMA power down is enabled.

Bit 5: Setting this bit to power down Timer/Counter 3.

0: Timer/Counter 3 power down is disabled.

1: Timer/Counter 3 power down is enabled.

Bit 4: Setting this bit to power down ADC Conversion.

0: ADC Conversion power down is disabled.

1: ADC Conversion power down is enabled.

Bit 3: Setting this bit to power down SPI.

0: SPI power down is disabled.

1: SPI power down is enabled.

Bit 2: Setting this bit to power down OPCMP1.

0: OPCMP1 power down is disabled.

1: OPCMP1 power down is enabled.

Bit 1: Setting this bit to power down LVD.

0: LVD power down is disabled.

1: LVD power down is enabled.

Bit 0: Setting this bit to power down RTC.

0: RTC power down is disabled.

1: RTC power down is enabled.

DEVPD2: Device2 Power Down

Bit	7	6	5	4	3	2	1	0
Name	-	PWMCPD	-	Timer4PD	PWMBPD	-	-	I2CPD
Type	R	R/W	R	R/W	R/W	R	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAE; SFR Page = 1

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Setting this bit to power down PWMC.

0: PWMC power down is disabled.

1: PWMC power down is enabled.

Bit 5: Reserved. Read = 0, Write = Don't Care.

Bit 4: Setting this bit to power down Timer/Counter 4.

0: Timer/Counter 4 power down is disabled.

1: Timer/Counter 4 power down is enabled.

Bit 3: Setting this bit to power down PWMA.

0: PWMA power down is disabled.

1: PWMA power down is enabled.

Bit 2~1: Reserved. Read = 0, Write = Don't Care.

Bit 0: Setting this bit to power down I2C.

0: I2C power down is disabled.

1: I2C power down is enabled.

DEVPD3: Device3 Power Down

Bit	7	6	5	4	3	2	1	0
Name	-	CMPPD	LCDLEDPD	UART2PD	-	-	-	-
Type	R	R/W	R/W	R/W	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAF; SFR Page = 1

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Setting this bit to power down Comparator1 (CMP1).

0: CMP1 power down is disabled.

1: CMP1 power down is enabled.

Bit 5: Setting this bit to power down LCD/LED.

0: LCD/LED power down is disabled.

1: LCD/LED power down is enabled.

Bit 4: Setting this bit to power down UART2.

0:UART2 power down is disabled.

1: UART2 power down is enabled.

Bit 3~0: Reserved. Read = 0, Write = Don't Care.

7 Flash Program

The Flash Memory is the location where the user's code or program is stored. It can be programmed in-system through OCD interface (using ICP) or by software using the MOVX instruction (using IAP). **Once cleared to logic 0, a Flash bit must be erased to set it back to Logic 1.** Flash bytes would typically be erased (set to 0xFF) before being reprogrammed.

7.1 In-Circuit Programming (ICP)

Programming and reprogramming of the microcontroller is carried out through In-Circuit Programming (ICP). It is done by writing two lines for clock and data and another two lines for power and ground. These pins are the same as OCDS. For more information on Pins Definition, refer to Section 20 On-chip Debug Support.

7.2 In-Application Programming (IAP)

The MCU supports In-Application Programming (IAP), allowing the Flash Memory to be modified during execution. The Flash modify procedure is the preferred way to program the Flash Memory from the application code (During implementation of the IAP program, the System Clock needs to operate at below 8 MHz).

7.2.1 Flash Key Function

Flash Write and Erase through user software are protected with a lock and key function. If flash is protected, using instructions to erase and write will be invalid..

The Flash Lock and Key Register (FLKEY) must be written with the correct key codes in sequence, before Flash operations may be performed. The Key-Code are 0xA9,and 0x7F. If the key codes are written out of order, or the wrong codes are written, Flash writes and erases will be disabled. After completing Flash Program sequence, Flash will return to protected Lock (the System Reset will also cause Flash to Lock). When running the next Flash Program operation, user must write the Key-Code again. Note that after writing the Flash Key, one system clock must be delayed to let programming flash successfully as shown in Figure 7.1.

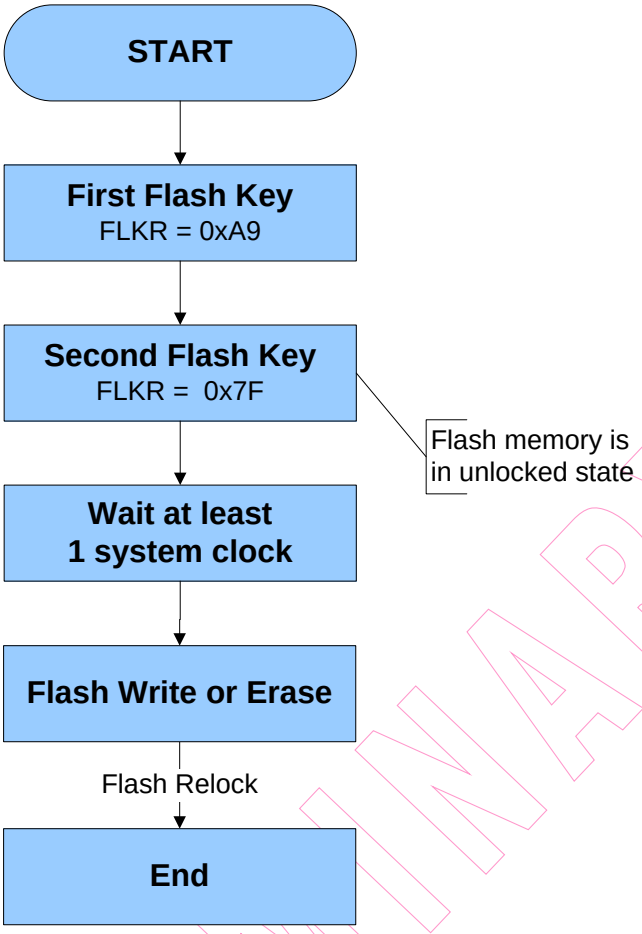


Figure 7- 1 Flash Unlock Procedure

7.2.2 Flash Erase Procedure

After Erasing Flash, the Flash data will be cleared as 1. Flash Erase must be performed using the MOVX instruction (MOVX @ DPTR, A), to provide the address user wants to Erase (data can be of any values). Before writing to Flash memory using MOVX, Flash write operations must be enabled by successful Flash unlock (the specified address of MOVX write instruction will point to the Flash Address). Regardless of which address of page is being pointed, the relevant contents of page are cleared as 1. Finally, set FLEE (FLCR.1) to 1 to enable the Flash Page Erase program. Each time Flash Page Erase is completed, Flash will return to locked state. Before proceeding to the next Flash Page Erase task, flash key must be written. Refer to detailed operational flow chart shown in Figure 7.2.

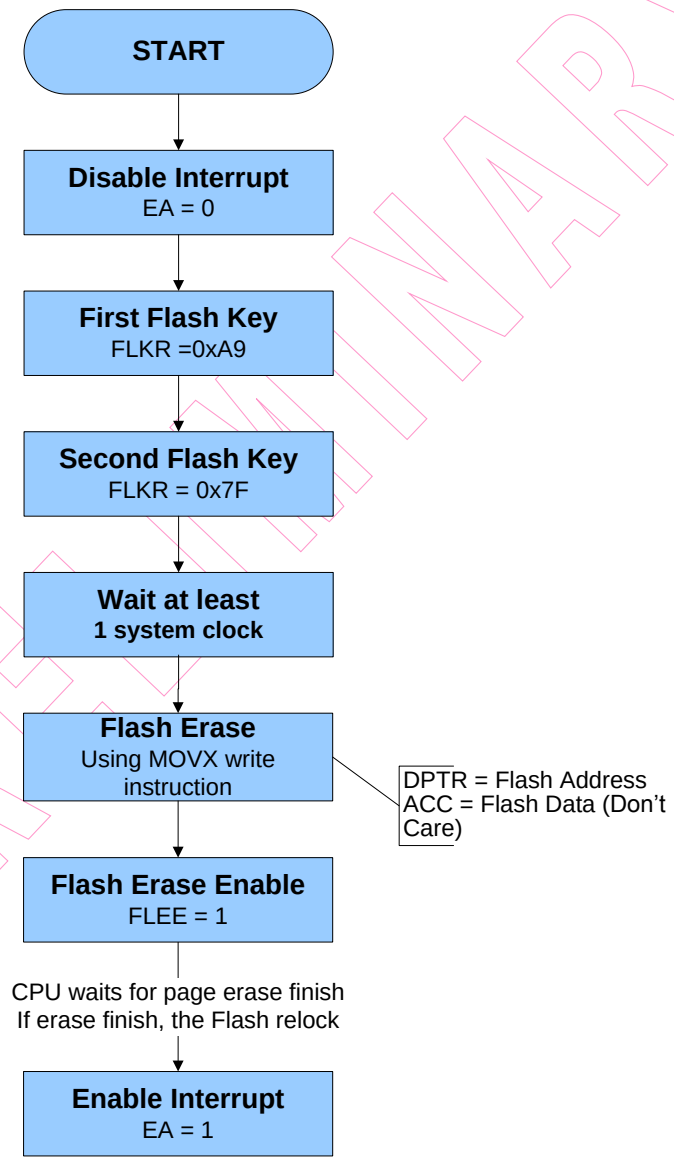


Figure 7- 2 Flash Erase Procedure

7.2.3 Flash Read Procedure

Flash Read operation is accomplished with MOVC instruction. Note that this operation is subject to Flash Enhanced Protection restriction. For details, refer to Section 7.3.2 Flash Enhanced Protection.

7.2.4 Flash Write Procedure

In Flash Write operation, data can be changed to 0, but cannot be changed to 1because only an erase operation can set bits to Logic 1 in Flash. Flash Write is carried out in page write manner by unlocking the Flash first. The moment unlock is successful, the specified address of MOVX write instruction will point to the Flash Address. Then use MOVX to provide the address and data user wants to write into Flash. Flash write operations must start from the page starting address. The last data address will determine which page to write. Lastly, set FLWE (FLCR.0) to 1 to enable the Flash Page Write program. When using the MOVX instruction to write data, the data are temporarily stored in the Data FIFO. The written data cannot be less than 64 strokes. Otherwise, the written data in the page will be incorrect. If the strokes exceed 64, only the last 64 strokes will be written into the Data FIFO. After each Flash Page Write is completed, Flash will revert to locked state. Before writing to the next page, repeat the unlock operation again. Refer to detailed operational flow chart shown in Figure 7.3.

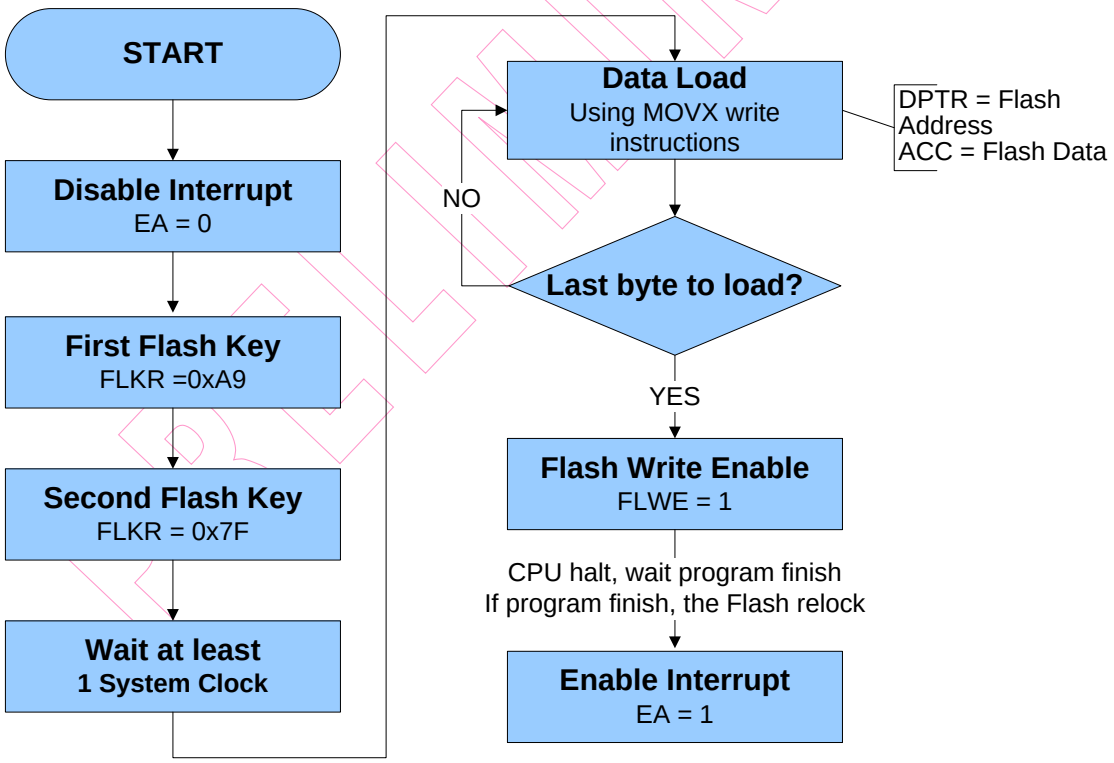


Figure 7- 3Flash Page Write Procedure

7.2.5 Flash Write with Erase Procedure

Users can simultaneously carry out Flash Erase while performing Flash Page Write as long as user is setting FLEE while setting FLWE in Flash Page Write procedure. For detailed operating procedure, see Figure 7.4.

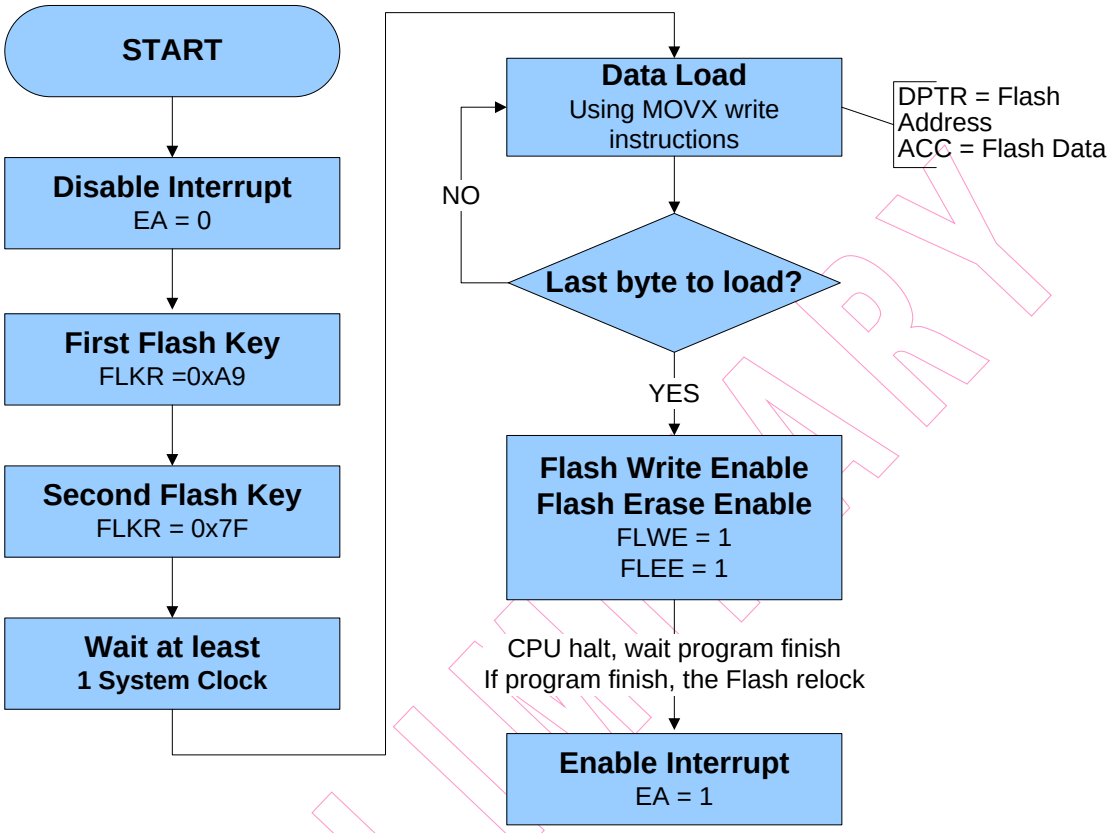


Figure 7-4 Flash Page Write with Erase Procedure

MOVX Write Instruction	
16-Bit MOVX Example1	MOV DPTR, #0x0800H; load DPTR with 16-bit address to read MOVX @DPTR, A ; write contents of accumulator A into flash
16-Bit MOVX Example2	MOV DPTR1, #0x0800H; load DPTR1 with 16-bit address to read MOVX @DPTR1, A ; write contents of accumulator A into flash

8-Bit MOVX Example	MOV XPAGE, #0x08H; load high byte of address into XPAGE
	MOV R0, #0x00H ; load low byte of address into R0 (or R1)
	MOVX @R0, A ; write contents of accumulator A into flash

FLKEY: Flash Key

Bit	7	6	5	4	3	2	1	0
Name	FLKR.7	FLKR.6	FLKR.5	FLKR.4	FLKR.3	FLKR.2	FLKR.1	FLKR.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC9; SFR Page = 0

This register provides a lock and key function for Flash erasures and writes. Flash writes and erases are enabled by writing 0xA9 followed by 0x7F to the FLKEY register. Flash writes and erases are automatically disabled after the next write or erase is completed.

FLCR: Flash Control Register

Bit	7	6	5	4	3	2	1	0
Name	EPEN	-	-	-	-	MEMSP	FLEE	FLWE
Type	R/W	R	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCA; SFR Page = 0

Bit7: Enhanced Protection enable

Setting this bit activates Flash Enhanced Protection.

0: Enhanced Protection disabled.

1: Enhanced Protection enabled.

Bits 6~3: Reserved. Read = 0, Write = Don't Care.

Bit 2: Set memory space.

0: Flash Program Memory.

1: Flash INF Area (only Info. Pages).

Bit 1: Flash erase enabled.

Setting this bit activates Flash page erase. The bit is cleared automatically by hardware when page erase finishesed.

0: Flash Memory page erase disabled.

1: Flash Memory page erase enabled.

Bit 0: Flash write enabled.

Setting this bit activates Flash page write. The bit is cleared automatically by hardware when page write finishes.

0: Flash Memory page write disabled.

1: Flash Memory page write enabled.

7.3 Flash Security

The MCU provides two types of security protection mechanism to Flash memory, namely, Flash POR Protection and Flash Enhanced Protection. The Flash POR Protection restricts ICP access to User Program Area operation when the MCU is powered off and then on again. See Section 7.3.1 for more details on Flash POR Protection. Likewise, the Flash Enhanced Protection restricts IAP access to User Program Area operation after the protection mechanism started. See Section 7.3.2 Flash Enhanced Protection for more details. Table 7.1 Flash Protection Summary below shows the restrictions on Flash memory access operation when the two protection mechanisms are activated.

(- : Always invalid.)

Flash POR Protection Enable		Read	Write	Erase
ICP	Flash Program Memory (Use Program Area)	$X_{[2]}$	X	O
Flash Enhanced Protection Enable		**Read**	**Write**	**Erase**
IAP	Flash Program Memory (Use Program Area)	$X_{[1][2]}$	$X_{[1]}$	$X_{[1]}$

Figure 7-52 Flash Protection Summary

7.3.1 Flash Enhanced Protection

An enhanced protected area cannot be erased or written by IAP in non-protected area is decided by EPPOINT (The value of 0x0000 means there is no enhanced protected area).

To start, set EPEN (FLCR.7) to 1 and in EPKEY, enter 0xC5 to enable and permit EPPOINT write. Next, fill in the Page number user wants to protect in EPPOINT (EPPOINTH and EPPOINTL). After EPPOINTL is written, EPKEY is cleared. The scope of protection begins and is calculated from Page 0 (Address 0x0000). For example, when filling 0x00 into EPPOINTL and 0x20 into EPPOINTH, it means that the range to be protected is from Address 0x0000 to 0x07FF, totaling of 32 Pages of codes illustrated in Figure 7.5 Enhanced Protection.

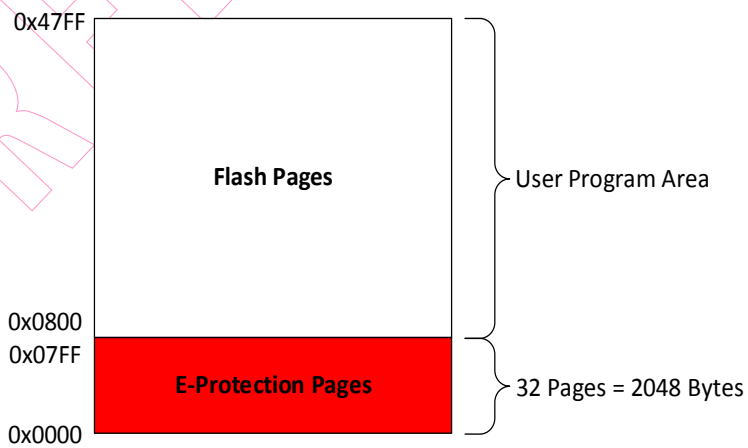


Figure 7-5 Enhanced Protection Pages

Note that after the start of protection mechanism, the EPPOINT contents can no longer be changed. Only after going through POR can the EPPOINT be cleared. After the protective mechanism is

activated and in operation under IAP of the protected region, the entire contents of User Program Area program can now be accessed. However, such program contents cannot be accessed from non-protected area. This feature also provides protection to user developed Library Code to prevent others from reading user's program. On application, the Library Code is configured within the protected area. After system reset, program execution from this area take precedence with protection mechanism started simultaneously. It is followed by the program and then it jumps to the non-protected area to execute the main program. As result, the programs in the main program area cannot access the contents of the Library Code program. This accomplishes the protection of the Library Code program from being read or modified by others. Detailed steps on starting Flash Enhanced Protection are described below:

Enable Flash Enhanced Protection

Step 1: Launch the Flash Enhanced Protection by setting the enable bit (EPEN = 1).

Step 2: Write the 0xC5 to EPKEY register.

Step 3: Fill page number (High byte) in EPPOINTH register.

Step 4: Fill page number (Low byte) in EPPOINTL register.

EPKEY: Enhanced Protection Key

Bit	7	6	5	4	3	2	1	0
Name	EPKEY.7	EPKEY.6	EPKEY.5	EPKEY.4	EPKEY.3	EPKEY.2	EPKEY.1	EPKEY.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCB; SFR Page = 0

EPPOINTL: Enhanced Protection Point Low Byte

Bit	7	6	5	4	3	2	1	0
Name	EPPOINTL.7	EPPOINTL.6	EPPOINTL.5	EPPOINTL.4	EPPOINTL.3	EPPOINTL.2	EPPOINTL.1	EPPOINTL.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCC; SFR Page = 0

EPPOINTH: Enhanced Protection Point High Byte

Bit	7	6	5	4	3	2	1	0
Name	EPPOINTH.7	EPPOINTH.6	EPPOINTH.5	EPPOINTH.4	EPPOINTH.3	EPPOINTH.2	EPPOINTH.1	EPPOINTH.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCD; SFR Page = 0

8 Port Configuration

All port pins on the MCU may be configured in one of four modes: Quasi-Bidirectional (standard 8051 port output), Push-Pull Output, Open-Drain Output, or Input-Only. Port modes may be assigned in software on a pin-by-pin basis as shown in Table 8.1 (Total mode registers summary in Table 8.2). The data registers are listed in Table 8.3, they can set the Port data logic value and read the Port Pin logic state in Port cells configured for digital I/O.

I/O pins can be multiplexed with other functions. When I/O is working on other function, it is not allowed to change the PxMy register value to change the I/O mode. The corresponding register is allowed to change the I/O mode only when the other function is disabled. Note that under Push-Pull Output, Input Only and Open-Drain Output modes, P0 and P1 can individually start internal pull-high function through PHCON0 and PHCON1, and start internal pull-low function through PLCON0 and PLCON1. At the same time, it is able to start High Drive and High Sink Current functions through PHDSC0 and PHDSC1. Furthermore, P2 can also start the above-mentioned functions through PHCON2, PLCON2 and PHDSC2.

Depending on which instruction is used, a read from a port will either reads the state of the pins or the state of the port register. Simple read instructions will always directly access the port pins. The port register is always accessed by some instructions that read a value and probably modify it and then write it back. Such instructions include bit write instruction such as CLR or SETB as they actually read the entire port, modify a single bit, then write the data back to the entire port. For a complete list of these instructions, refer to Table 8-1.

Mnemonic	Instruction	Example
ANL	Logical AND	ANL P0, A
ORL	Logical OR	ORL P0, A
XRL	Logical EX-OR	XRL P0, A
JBC	Jump if bit set and clear bit	JBC P0.1, LABEL
CPL	Complement bit	CPL P0.1
INC	Increment	INC P0
DEC	Decrement	DEC P0
DJNZ	Decrement and jump if not zero	DJNZ P0, LABEL
MOV Px.y, C	Move carry to bit y of Portx	MOV P0.1, C
CLR Px.y	Clear bit y of Portx	CLR P0.1
SETB Px.y	Set bit y of Portx	SETB P0.1

Table 8-1Instructions is always access the port register

Table 8- 2 Port Configuration Modes

PxM0.y	PxM1.y	Port Mode
0	0	Quasi-Bidirectional
0	1	Push-Pull Output
1	0	Input Only (High Impedance)
1	1	Open-Drain Output

(x = 0, 1, 2; y = 0, 1, 2)

Table 8- 3 Port Configuration Mode 0 Registers

Register	SFR	7	6	5	4	3	2	1	0
P0M0 (0xF1)	PAGE0	P0M0.7	P0M0.6	-	-	P0M0.3	-	-	-
P1M0 (0xF3)	PAGE0	-	-	-	P1M0.4	P1M0.3	P1M0.2	P1M0.1	P1M0.0
P2M0 (0xF9)	PAGE0	-	-	-	-	-	-	P2M0.1	P2M0.0
P3M0 (0xFB)	PAGE0	P3M0.7	P3M0.6	P3M0.5	P3M0.4	P3M0.3	P3M0.2	P3M0.1	P3M0.0
P4M0 (0XF1)	PAGE1	-	-	-	P4M0.4	P4M0.3	P4M0.2	P4M0.1	P4M0.0
P5M0 (0xF3)	PAGE1	-	-	-	-	-	P5M0.2	P5M0.1	P5M0.0
Type		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		1	1	1	1	1	1	1	1

Table 8- 4 Port Configuration Mode 1 Registers

Register	SFR	7	6	5	4	3	2	1	0
P0M1 (0xF2)	PAGE0	P0M1.7	P0M1.6	-	-	P0M1.3	-	-	-
P1M1 (0xF4)	PAGE0	-	-	-	P1M1.4	P1M1.3	P1M1.2	P1M1.1	P1M1.0
P2M1 (0xFA)	PAGE0	-	-	-	-	-	-	P2M1.1	P2M1.0
P3M1 (0xFC)	PAGE0	P3M1.7	P3M1.6	P3M1.5	P3M1.4	P3M1.3	P3M1.2	P3M1.1	P3M1.0
P4M1 (0xF2)	PAGE1	-	-	-	P4M1.4	P4M1.3	P4M1.2	P4M1.1	P4M1.0
P5M1 (0xF4)	PAGE1	-	-	-	-	-	P5M1.2	P5M1.1	P2M1.0
Type		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset		0	0	0	0	0	0	0	0

Table 8- 5 Port Data Registers

Register	7	6	5	4	3	2	1	0
P0 (0x80)	P0.7	P0.6	-	-	P0.3	-	-	-
P1 (0x90)	-	-	-	P1.4	P1.3	P1.2	P1.1	P1.0
P2 (0xA0)	-	-	-	-	-	-	P2.1	P2.0
P3 (0xB0)	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
P4 (0xE8)	-	-	-	P4.4	P4.3	P4.2	P4.1	P4.0
P5 (0xF8)	-	-	-	-	-	P5.2	P5.1	P5.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

Notes:

1. When writing the data register, 0: Set output latch to logic LOW, 1: Set output latch to logic HIGH.
2. When reading the data register, 0: Port pin is logic LOW, 1: Port pin is logic HIGH.

PHCON0: Pull-High Control 0

Bit	7	6	5	4	3	2	1	0
Name	P0PH.7	P0PH.6	-	-	P0PH.3	-	-	-
Type	R/W	R/W	R	R	R/W	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF5; SFR Page = 0

Bits 7~6: Setting these bits enables Port 0 Pull-High.

Bits 5~4: Reserved. Read = 0, Write = Don't Care.

Bit 3: Setting this bit enables Port 0 Pull-High.

Bits 2~0: Reserved. Read = 0, Write = Don't Care.

PHCON1: Pull-High Control 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	P1PH.4	P1PH.3	P1PH.2	P1PH.1	P1PH.0
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xFD; SFR Page = 0

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bits 4~0: Setting these bits to enable Port 1 Pull-High.

PHCON2: Pull-High Control 2

Bit	7	6	5	4	3	2	1	0
Name	-	P5PH0	P4PH1	P4PH0	P3PH1	P3PH0	-	P2PH0
Type	R	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF5; SFR Page = 1

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Port 5 pull-high enables control on low nibble.

0: Disable the P5.3 ~ P5.0 pull-high.

1: Enable the P5.3 ~ P5.0 pull-high.

Bit 5: Port 4 pull-high enables control on high nibble.

0: Disable the P4.7~ P4.4 pull-high.

1: Enable the P4.7 ~ P4.4 pull-high.

Bit 4: Port 4 pull-high enables control on low nibble.

0: Disable the P4.3 ~ P4.0 pull-high.

1: Enable the P4.3 ~ P4.0 pull-high.

Bit 3: Port 3 pull-high enables control on high nibble.

0: Disable the P3.7 ~ P3.4 pull-high.

1: Enable the P3.7 ~ P3.4 pull-high.

Bit 2: Port 3 pull-high enables control on low nibble.

0: Disable the P3.3 ~ P3.0 pull-high.

1: Enable the P3.3 ~ P3.0 pull-high.

Bit 1: Reserved. Read = 0, Write = Don't Care.

Bit 0: Port 2 pull-high enables control on low nibble.

0: Disable the P2.3 ~ P2.0 pull-high.

1: Enable the P2.3 ~ P2.0 pull-high.

PLCON0: Pull-Low Control 0

Bit	7	6	5	4	3	2	1	0
Name	P0PL.7	P0PL.6	-	-	P0PL.3	-	-	-
Type	R/W	R/W	R	R	R/W	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF6; SFR Page = 0

Bits 7~6: Setting these bits to enable Port 0 Pull-Low.

Bits 5~4: Reserved. Read = 0, Write = Don't Care.

Bits 3: Setting this bit to enable Port 0 Pull-Low.

Bits 2~0: Reserved. Read = 0, Write = Don't Care.

PLCON1: Pull-Low Control 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	P1PL.4	P1PL.3	P1PL.2	P1PL.1	P1PL.0
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xFE; SFR Page = 0

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bits 4~0: Setting these bits to enable Port 1 Pull-Low.

PLCON2: Pull-Low Control 2

Bit	7	6	5	4	3	2	1	0
Name	-	P5PL0	P4PL1	P4PL0	P3PL1	P3PL0	-	P2PL0
Type	R	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF6; SFR Page = 1

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Port 5 pull-low enables control on low nibble.

0: Disable the P5.3 ~ P5.0 pull-low.

1: Enable the P5.3 ~ P5.0 pull-low.

Bit 5: Port 4 pull-low enables control on high nibble.

0: Disable the P4.7 ~ P4.4 pull-low.

1: Enable the P4.7 ~ P4.4 pull-low.

Bit 4: Port 4 pull-low enables control on low nibble.

0: Disable the P4.3 ~ P4.0 pull-low.

1: Enable the P4.3 ~ P4.0 pull-low.

Bit 3: Port 3 pull-low enables control on high nibble.

0: Disable the P3.7 ~ P3.4 pull-low.

1: Enable the P3.7 ~ P3.4 pull-low.

Bit 2: Port 3 pull-low enables control on low nibble.

0: Disable the P3.3 ~ P3.0 pull-low.

1: Enable the P3.3 ~ P3.0 pull-low.

Bit 1: Reserved. Read = 0, Write = Don't Care.

Bit 0: Port 2 pull-low enables control on low nibble.

0: Disable the P2.3 ~ P2.0 pull-low.

1: Enable the P2.3 ~ P2.0 pull-low.

PHDSC0: Port High Drive/Sink Control 0

Bit	7	6	5	4	3	2	1	0
Name	P0PHS.7	P0PHS.6	-	-	P0PHS.3	-	-	-
Type	R/W	R/W	R	R	R/W	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF7; SFR Page = 0

Bits 7~6: Setting these bits to enable Port 0 High Drive/Sink.

Bit 5~4: Reserved. Read = 0, Write = Don't Care.

Bit 3: Setting these bits to enable Port 0 High Drive/Sink.

Bits 2~0: Reserved. Read = 0, Write = Don't Care.

PHDSC1: Port High Drive/Sink Control 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	P1PHS.4	P1PHS.3	P1PHS.2	P1PHS.1	P1PHS.0
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xFF; SFR Page = 0

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bit 4~0: Setting these bits to enable Port 1 High Drive/Sink.

PHDSC2: Port High Drive/Sink Control 2

Bit	7	6	5	4	3	2	1	0
Name	-	P5PHS0	P4PHS1	P4PHS0	P3PHS1	P3PHS0	-	P2PHS0
Type	R	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF7; SFR Page = 1

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Port 5 high drive/sink enables control on low nibble.

0: Disable the P5.3 ~ P5.0 high drive/sink.

1: Enable the P5.3 ~ P5.0 high drive/sink.

Bit 5: Port 4 high drive/sink enables control on high nibble.

0: Disable the P4.7 ~ P4.4 high drive/sink.

1: Enable the P4.7 ~ P4.4 high drive/sink.

Bit 4: Port 4 high drive/sink enables control on low nibble.

0: Disable the P4.3 ~ P4.0 high drive/sink.

1: Enable the P4.3 ~ P4.0 high drive/sink.

Bit 3: Port 3 high drive/sink enables control on high nibble.

0: Disable the P3.7 ~ P3.4 high drive/sink.

1: Enable the P3.7 ~ P3.4 high drive/sink.

Bit 2: Port 3 high drive/sink enables control on low nibble.

0: Disable the P3.3 ~ P3.0 high drive/sink.

1: Enable the P3.3 ~ P3.0 high drive/sink.

Bit 1: Reserved. Read = 0, Write = Don't Care.

Bit 0: Port 2 high drive/sink enables control on low nibble.

0: Disable the P2.3 ~ P2.0 high drive/sink.

1: Enable the P2.3 ~ P2.0 high drive/sink.

8.1 Quasi-Bidirectional Mode

Quasi-Bidirectional Mode has three pull-high resistances. Their respective roles are described below:

Weak pull-up MOS: When the data is 1 and the Pin itself is also 1, the MOS is switched on. If the Pin output is high and is pulled down by external device to Low, the Weak pull-up is closed and the very weak pull-up maintains its open state. Note that the Weak pull up is enabled through internal pull high control register.

Very weak pull-up MOS: When the data is 1, it opens. When the Pin base is floating, this Very weak pull-up will generate a very weak pull-up current, which will pull the Pin base to High level.

Strong pull-up MOS: When the port data changes from 0 to1, the pull-up is used to speed up the quasi-bidirectional port through logic conversion from Logic 0 to Logic 1.

(Under this mode, the High drive and High Sink Current features can be enabled through Port Control Register)

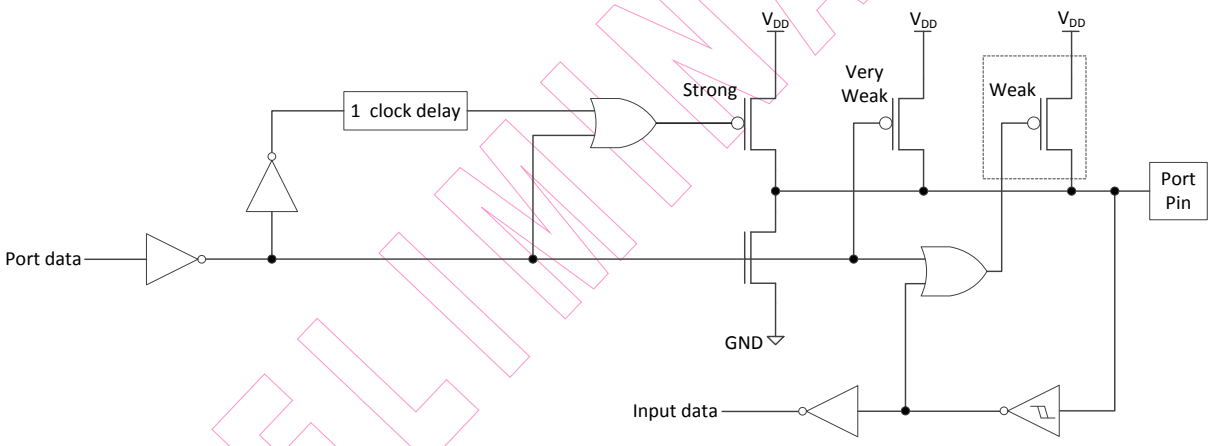


Figure 8-1 Quasi-Bidirectional Mode

8.2 Push-Pull Output Mode

The pull-low structure in Push-Pull Output mode is same as open-drain and Quasi-Bidirectional mode, but the port provides a continuous strong pull-up when the port data is 1. Push-Pull Output mode port structure diagram is shown in Figure 8.2 Push-Pull Output Mode.

(Under this mode, user can provide internal pull-high or pull-low through the Port Control Register. At the same time, it also enables High drive and High Sink Current features).

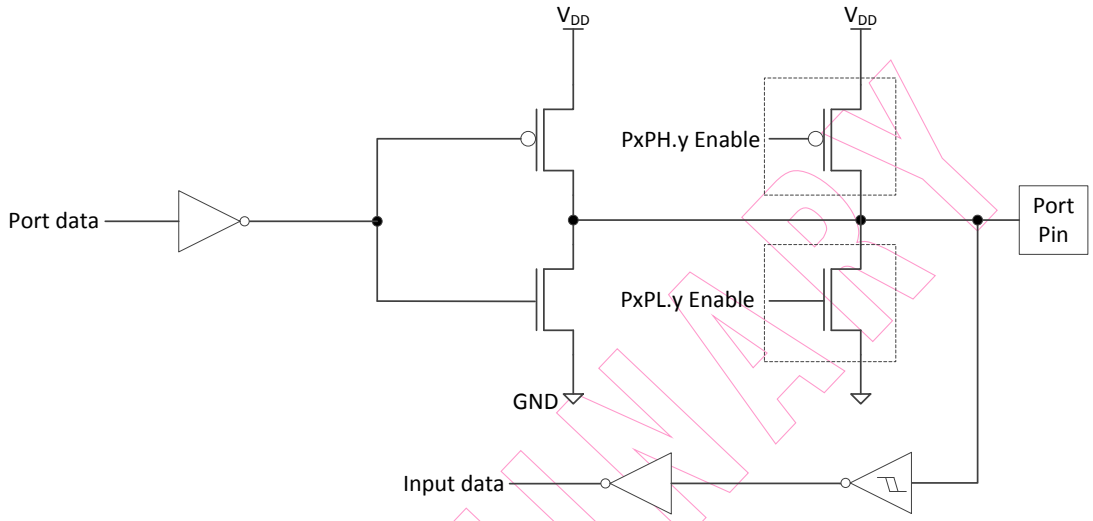


Figure 8-2 Push-Pull Output Mode

8.3 Input-Only Mode

In Input-Only mode, port is input only and has no output capability. Input-Only mode port structure diagram is shown in Figure 8.3 Input-Only Mode.

(Under this mode, user can provide internal pull-high or pull-low through the Port Control Register. At the same time, it also enables the High drive and High Sink Current features).

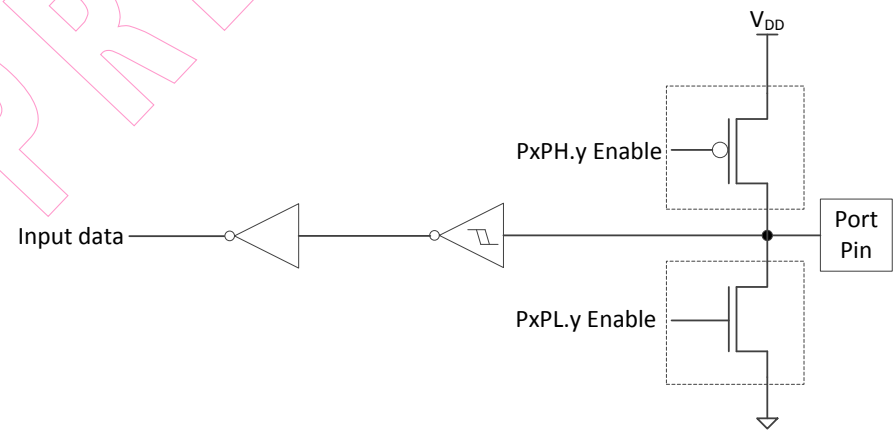


Figure 8-3 Input-Only Mode

8.4 Open-Drain Output Mode

In Open-Drain Output mode, the ports have no output high capability. Open-Drain Output mode port structure diagram is shown in Figure 8.4 Open-Drain Output Mode.

(Under this mode, user can provide internal pull-high or pull-low through the Port Control Register. At the same time, it also to enable the High drive and High Sink Current features).

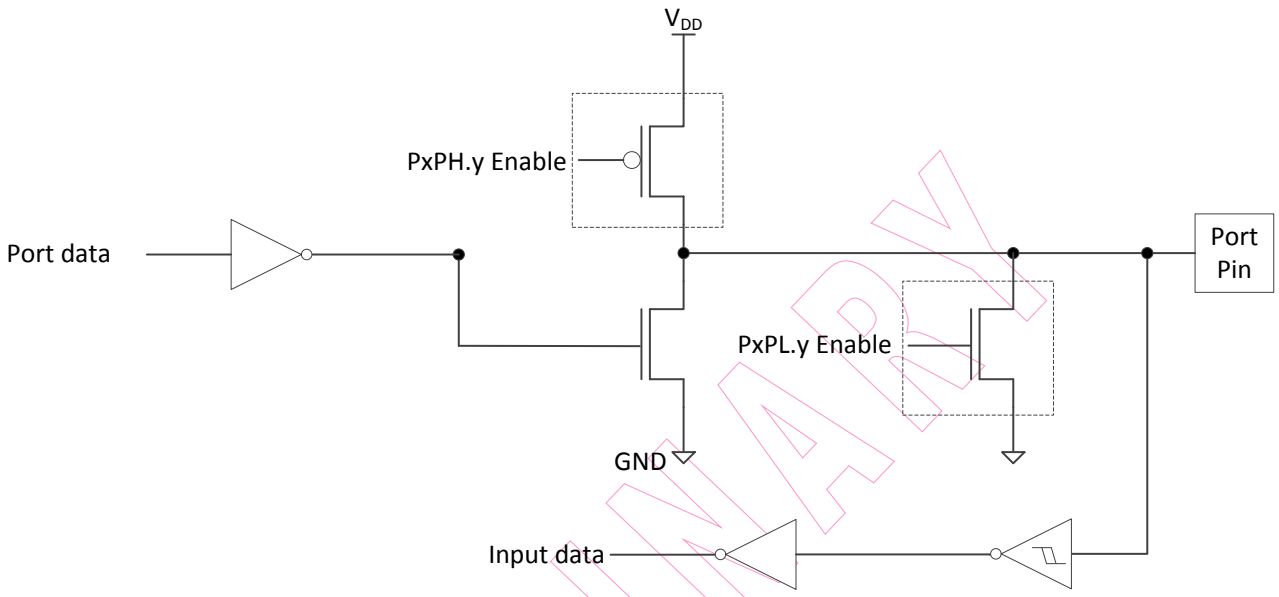


Figure 8-4 Open-Drain Output Mode

8.5 External IO Pins Functional Select

EIOCOM1: External IO Pins Function Select for LCD/LED COM Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0XC1; SFR Page = 0x2

Bits 7~0 (SEG7~0): LCD/LED Pin Switch for COM7~COM0

0: Function as normal I/O or other functions (default)

1: Function as LCD/LED common pins

EIOSEG1: External IO Pins Function Select for LCD/LED SEG Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	SEG4	SEG3	SEG2	SEG1	SEG0
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0XC2; SFR Page = 0x2

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bits 4~0 (SEG4~0): LCD/LED Pin Switch for SEG4~SEG0

0: Function as normal I/O or other functions (default)

1: Function as LCD/LED segment pins

EIOSEG11: External IO Pins Function Select for LCD/LED SEG1 Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	SEG7_1	SEG6_1	SEG5_1	-	-	-	-	-
Type	R/W	R/W	R/W	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address =0xC6; SFR Page = 0x2

Bits 7~5 (SEG7_1~SEG5_1): LCD/LED Pin Switch for LEDS7_1_LEDS5_1(SEG7_1~SEG5_1)

0: Function as normal I/O or other functions (default)

1: Function as LCD/LED segment pins

Bits 4~0: Reserved. Read = 0, Write = Don't Care.

EIOPWMA: External IO Pins Function Select for PWMA Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	/PWMA1	PWMA1	/PWMA0	PWMA0
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0xC9; SFR Page = 0x2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bits 3 (/PWMA1): PWM Pin Switch for /PWMA1

0: Function as normal I/O or other functions (default)

1: Function as /PWMA1 pins

Bits 2 (PWMA1): PWM Pin Switch for PWMA1

0: Function as normal I/O or other functions (default)

1: Function as PWMA1 pins

Bits 1 (/PWMA0): PWM Pin Switch for /PWMA0

0: Function as normal I/O or other functions (default)

1: Function as /PWMA0 pins

Bits 0 (PWMA0): PWM Pin Switch for PWMA0

0: Function as normal I/O or other functions (default)

1: Function as PWMA0 pins

EIOPWMB: External IO Pins Function Select for PWMB Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	/PWMB1	PWMB1	/PWMB0	PWMB0
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0xCA; SFR Page = 0x2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bits 3 (/PWMB1): PWM Pin Switch for /PWMB1

0: Function as normal I/O or other functions (default)

1: Function as /PWMB1 pins

Bits 2 (PWMB1): PWM Pin Switch for PWMB1

0: Function as normal I/O or other functions (default)

1: Function as PWMB1 pins

Bits 1 (/PWMB0): PWM Pin Switch for /PWMB0

0: Function as normal I/O or other functions (default)

1: Function as /PWMB0 pins

Bits 0 (PWMB0): PWM Pin Switch for PWMB0

0: Function as normal I/O or other functions (default)

1: Function as PWMB0 pins

EIOPWMC: External IO Pins Function Select for PWMC Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	/PWMC1	PWMC1	/PWMC0	PWMC0
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0xCB; SFR Page = 0x2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bits 3 (/PWMC1): PWM Pin Switch for /PWMC1

0: Function as normal I/O or other functions (default)

1: Function as /PWMC1 pins

Bits 2 (PWMC1): PWM Pin Switch for PWMC1

0: Function as normal I/O or other functions (default)

1: Function as PWMC1 pins

Bits 1 (/PWMC0): PWM Pin Switch for /PWMC0

0: Function as normal I/O or other functions (default)

1: Function as /PWMC0 pins

Bits 0 (PWMC0): PWM Pin Switch for PWMC0

0: Function as normal I/O or other functions (default)

1: Function as PWMB0 pins

EIOT3: External IO Pins Function Select for Timer 3 Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	T3_1	T3_0
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0xCD; SFR Page = 0x2

Bits 7~2: Reserved. Read = 0, Write = Don't Care.

Bits 1 (T3_1): Timer 3 Pin Switch for T3_1

0: Function as normal I/O or other functions (default)

1: Function as T3_1 pins

Bits 0 (T3_0): Timer 3 Pin Switch for T3_0

0: Function as normal I/O or other functions (default)

1: Function as T3_0 pins

EIOT4: External IO Pins Function Select for Timer 4 Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	T4_2	T4_1	T4_0
Type	R	R	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0xCE; SFR Page = 0x2

Bits 7~3: Reserved. Read = 0, Write = Don't Care.

Bits 2 (T4_2): Timer 3 Pin Switch for T4_2

0: Function as normal I/O or other functions (default)

1: Function as T4_2 pins

Bits 1 (T4_1): Timer 4 Pin Switch for T4_1

0: Function as normal I/O or other functions (default)

1: Function as T4_1 pins

Bits 0 (T4_0): Timer 4 Pin Switch for T4_0

0: Function as normal I/O or other functions (default)

1: Function as T4_0 pins

EIOUART2: External IO Pins Function Select for UART2 Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	TX21	RX21	TX20	RX20
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0XB1; SFR Page = 0x2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bits 3 (TX21): UART Pin Switch for TX21

0: Function as normal I/O or other functions (default)

1: Function as TX21 pins

Bits 2 (RX21): UART Pin Switch for RX21

0: Function as normal I/O or other functions (default)

1: Function as RX21 pins

Bits 1 (TX20): UART Pin Switch for TX20

0: Function as normal I/O or other functions (default)

1: Function as TX20 pins

Bits 0 (RX20): UART Pin Switch for RX20

0: Function as normal I/O or other functions (default)

1: Function as RX20 pins

EIOI2C0: External IO Pins Function Select for I2C0 Control Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	I2C_SCL01	I2C_SDA01	I2C_SCL00	I2C_SDA00
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0xB2; SFR Page = 0x2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bits 3 (I2C_SCL01): I2C Pin Switch for I2C_SCL01

0: Function as normal I/O or other functions (default)

1: Function as I2C_SCL01 pins

Bits 2 (I2C_SDA01): I2C Pin Switch for I2C_SDA01

0: Function as normal I/O or other functions (default)

1: Function as I2C_SDA01 pins

Bits 1 (I2C_SCL00): I2C Pin Switch for I2C_SCL00

0: Function as normal I/O or other functions (default)

1: Function as I2C_SCL00 pins

Bits 0 (I2C_SDA00): I2C Pin Switch for I2C_SDA00

0: Function as normal I/O or other functions (default)

1: Function as I2C_SDA00 pins

EIOSPI0: External IO Pins Function Select for SPI0 Control Register 0

Bit	7	6	5	4	3	2	1	0
Name	SCK01	MOSI01	MISO01	/SS01	SCK00	MOSI00	MISO00	/SS00
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0xB3; SFR Page = 0x2

Bits 7 (SCK01): SPI Pin Switch for SCK01

0: Function as normal I/O or other functions (default)

1: Function as SCK01 pins

Bits 6 (MOSI01): SPI Pin Switch for MOSI01

0: Function as normal I/O or other functions (default)

1: Function as MOSI01 pins

Bits 5 (MISO01): SPI Pin Switch for MISO01

0: Function as normal I/O or other functions (default)

1: Function as MISO01 pins

Bits 4 (/SS01): SPI Pin Switch for /SS01

0: Function as normal I/O or other functions (default)

1: Function as /SS01 pins

Bits 3 (SCK00): SPI Pin Switch for SCK00

0: Function as normal I/O or other functions (default)

1: Function as SCK00 pins

Bits 2 (MOSI00): SPI Pin Switch for MOSI00

0: Function as normal I/O or other functions (default)

1: Function as MOSI00 pins

Bits 1 (MISO00): SPI Pin Switch for MISO00

0: Function as normal I/O or other functions (default)

1: Function as MISO00 pins

Bits 0 (/SS00): SPI Pin Switch for /SS00

0: Function as normal I/O or other functions (default)

1: Function as /SS00 pins

9 Interrupt

The Watchdog Timer starts counting upward when the WDTE bit is set to “1” and stops (or WDT is not used) by first clear WDTE bit. Next, we write 0xB1 to WDTKEY in order. The WDT is enabled in default. The clock source of WDT is 16 KHz.

The MCU has an extended interrupt system. Such interrupt system supports many interrupt sources with two priority levels, and each source has one or more related interrupt flags in the Special Function Register. When an external source or a peripheral meets a valid interrupt condition, the related interrupt flag is set to Logic 1.

When the status flag is set and if an interrupt is enabled for the source, an interrupt request is generated. Upon completion of the execution of the current instruction, the CPU generates an LCALL to a predetermined address to start execution of an Interrupt Service Routine or ISR. Each ISR must end with RETI instruction. Such instruction returns the program execution to the next instruction, which would have been executed had there been no interrupt request. When a function is enabled, the corresponding status flag will be activated regardless of the enabled/disabled status of the interrupt mask.

An associated interrupt enable bit in SFR (IE, EIE1, EIE2 or EIE3) can be used to individually enable each interrupt source. But prior to recognizing such individual interrupt enabling, they must first be globally enabled by setting the EA bit (IE.7) to Logic 1. Setting the EA bit to Logic 0 disables all interrupt sources, regardless of the individual interrupt-enable settings.

When the CPU vectors to the ISR, some interrupt flags are automatically cleared by hardware, those not cleared by hardware must be cleared by software before returning from the ISR. If the CPU has completed the return-from-interrupt (RETI) instruction and an interrupt flag still remains set, a new interrupt request will immediately be generated.

Interrupt response time depends on the status of the CPU when the interrupt occurs. For each system clock cycle, interrupts are sampled and priorities are decoded. If the CPU is executing an ISR for an interrupt with equal or higher priority, the new interrupt will not be serviced until the current ISR is completed, including the RETI and the instruction following it.

9.1 Interrupt Sources

The EM85F765N support several interrupt sources. Software can simulate an interrupt by setting any interrupt flag to logic 1. If interrupts are enabled for the flag, an interrupt request will be generated and the CPU will vector to the ISR address associated with the interrupt flag. Table 9.1 summarized MCU interrupt sources, associated vector addresses, fixed priority order and control bits are.

Fixed Priority Order	Interrupt Vector	Interrupt Source	Status Flag	Clear by HW	Interrupt Mask	Normal Wake-up Valid Time[1]	
						IDLE	PD
Top	0x0000	Reset	None	N/A	Always Enabled	-	-
0	0x0003						
1	0x000B	Timer 0 Overflow	TF0 (TCON.5)	Y	ET0 (IE.1)	4T	x
2	0x0013						
3	0x001B	Timer 1 Overflow	TF1 (TCON.7)	Y	ET1 (IE.3)	4T	x
4	0x0023	UART0	RI0 (SCON0.0) TI0 (SCON0.1)	N	ES0 (IE.4)	x	x
5	0x002B	RTC	ALARMF (ALARM.6)	N	ERTCIE (IE.5)	5T	x
6	0x0033	Pin-Change Interrupt	IC0SF (ICEN.0) IC1SF (ICEN.1) IC2SF (ICEN.2) IC3SF (ICEN.3)	N	ICIE (IE.6)	6T	6T
7	0x003B	HLVD	HLVDSF (LVDCR1.7)	N	HLVDIE (EIE1.0)	x	x
8	0x0043	System-Hold	SHSF (PSW1.0)	N	SHIE (EIE1.1)	x	x
9	0x004B	Comparator1	CMP1SF (COSF1.0)	N	CMP1IE (EIE1.2)	6T	6T
10	0x0053						
11	0x005B	SPI	SPISF (SPIS.1)	N	SPIE (EIE1.4)	By Flag[2]	By Flag[2]
12	0x0063						
13	0x006B	ADC Conversion Complete	ADSF (ADCR2.7)	N	ADIE (EIE1.6)	8T	8T
14	0x0073	Timer 3	TC3DASF (TC3CR1.4) TC3DBSF (TC3CR1.5)	N	TC3IE (EIE1.7)	7T	x
15	0x007B	PWMA	PWMADSF (PWMACR2.6) PWMAPSF (PWMACR2.7)	N	PWMAIE (EIE2.0)	7T	x
16	0x0083	PWMB	PWMBDSF (PWMBCR2.6) PWMBPSF (PWMBCR2.7)	N	PWMBIE (EIE2.1)	7T	x
17	0x008B						
18	0x0093	Timer 4	TC4DASF (TC4CR1.4) TC4DBSF (TC4CR1.5)	N	TC4IE (EIE2.3)	7T	x
19	0x009B						
20	0x00A3	I2C	I2CTXSF (I2CSF.7) I2CRXSF (I2CSF.6) I2CSTPSF (I2CSF.4) TO1SF(I2CTO1.6) TO2SF(I2CTO2.6) TO3SF(I2CTO3.6)	N	I2CIE (EIE2.5)	By Flag[3]	By Flag[3]
21	0x00AB						
22	0x00B3						
23	0x00BB						
24	0x00C3						
25	0x00CB						
26	0x00D3	UART2	UT2SF2 (UR2S2.0) UR2SF (UR2S2.1) U2ERRSF (UR2S2.2)	N	UART2IE (EIE3.3)	7T	x

Fixed Priority Order	Interrupt Vector	Interrupt Source	Status Flag	Clear by HW	Interrupt Mask	Normal Wake-up Valid Time[1]	
						IDLE	PD
27	0x00DB	PWMC	PWMCDSF (PWMCCR2.6) PWMCPSF (PWMCCR2.7)	N	PWMCIE (EIE3.4)	7T	x
28	0x00E3				(EIE3.5)		
29	0x00EB	External Interrupt 2~9 INT2~9	EXFSF2 (EXSF2.1) EXFSF3 (EXSF2.3) EXFSF4 (EXSF2.5) EXFSF5 (EXSF2.7) EXFSF6 (EXSF3.1) EXFSF7 (EXSF3.3) EXFSF8 (EXSF3.5) EXFSF9 (EXSF3.7) EXRSF2 (EXSF2.0) EXRSF3 (EXSF2.2) EXRSF4 (EXSF2.4) EXRSF5 (EXSF2.6) EXRSF6 (EXSF3.0) EXRSF7 (EXSF3.2) EXRSF8 (EXSF3.4) EXRSF9 (EXSF3.6)	N	EXIE1 (EIE3.6)	5T	5T
30	0x00F3	External Interrupt 10~11 INT10~11	EXFSF10 (EXSF4.1) EXFSF11 (EXSF4.3) EXRSF10 (EXSF4.0) EXRSF11(EXSF4.2)	N	EXIE2 (EIE3.7)	5T	5T
31	0x00FB						

Table 9- 1 Interrupt Sources

IE: Interrupt Enable

Bit	7	6	5	4	3	2	1	0
Name	EA	ICIE	ERTCIE	ES0	ET1	-	ET0	-
Type	R/W	R/W	R/W	R/W	R/W	R	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA8; SFR Page = All Pages

Bit 7: Enable All Interrupts.

Globally enables/disables all interrupts. It overrides individual interrupt mask settings.

0: Disable all interrupt sources.

1: Enable each interrupt according to its individual mask setting.

Bit 6: Input-Change Interrupts.

This bit sets the masking of the Pin Change interrupt.

0: Disable Pin Change interrupt.

1: Enable Pin Change interrupt.

Bit 5: RTC Interrupts.

This bit sets the masking of the RTC interrupt.

0: Disable RTC interrupt.

1: Enable RTC interrupt.

Bit 4: Enable UART0 Interrupt.

This bit sets the masking of the UART0 interrupt.

0: Disable UART0 interrupt.

1: Enable UART0 interrupt.

Bit 3: Enable Timer 1 Interrupt.

This bit sets the masking of the Timer 1 interrupt.

0: Disable all Timer 1 interrupt.

1: Enable Interrupt requests generated by the TF1 flag.

Bit 2: Reserved. Read = 0, Write = Don't Care.

Bit 1: Enable Timer 0 Interrupt.

This bit sets the masking of the Timer 0 interrupt.

0: Disable all Timer 0 interrupt.

1: Enable Interrupt requests generated by the TF0 flag.

Bit 0: Reserved. Read = 0, Write = Don't Care.

EIE1: Extended Interrupt Enable 1

Bit	7	6	5	4	3	2	1	0
Name	TC3IE	ADIE	-	SPIIE	-	CMP1IE	SHIE	HLVDIE
Type	R/W	R/W	R	R/W	R	R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB1; SFR Page = 0

Bit 7: Timer/Counter 3 Interrupt Enable.

This bit sets the masking of the Timer/Counter 3 interrupt.

0: Disable Timer/Counter 3 interrupt.

1: Enable Timer/Counter 3 interrupt.

Bit 6: ADC Conversion Complete Interrupt Enable.

This bit sets the masking of the ADC Conversion Complete interrupt.

0: Disable ADC Conversion Complete interrupt.

1: Enable ADC Conversion Complete interrupt

Bit 5: Reserved. Read = 0, Write = Don't Care.

Bit 4: Serial Peripheral Interface (SPI) Interrupt Enable.

This bit sets the masking of the SPI interrupt.

0: Disable SPI interrupt.

1: Enable SPI interrupt

Bit 3: Reserved. Read = 0, Write = Don't Care.

Bit 2: Comparator1 (CMP1) Interrupt Enable.

This bit sets the masking of the CMP1 interrupt.

0: Disable CMP1 interrupt.

1: Enable CMP1 interrupt

Bit 1: System-Hold Interrupt Enable.

This bit sets the masking of the System-Hold interrupt.

0: Disable System-Hold interrupt.

1: Enable System-Hold interrupt

Bit 0: High/ Low Voltage Detector (LVD) Interrupt Enable.

This bit sets the masking of the HLVD interrupt.

0: Disable HLVD interrupt.

1: Enable HLVD interrupt

EIE2: Extended Interrupt Enable 2

Bit	7	6	5	4	3	2	1	0
Name	-	-	I2CIE	-	TC4IE	-	PWMBIE	PWMAIE
Type	R	R	R/W	R	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB2; SFR Page = 0

Bits 7~6: Reserved. Read = 0, Write = Don't Care.

Bit 5: I2C Interrupt Enable.

This bit sets the masking of the I2C interrupt.

0: Disable I2C interrupt.

1: Enable I2C interrupt

Bit 4: Reserved. Read = 0, Write = Don't Care.

Bit 3: Timer/Counter 4 Interrupt Enable.

This bit sets the masking of the Timer/Counter 4 interrupt.

0: Disable Timer/Counter 4 interrupt.

1: Enable Timer/Counter 4 interrupt.

Bit 2: Reserved. Read = 0, Write = Don't Care.

Bit 1: PWMB Interrupt Enable.

This bit sets the masking of the PWMB interrupt.

0: Disable PWMB interrupt.

1: Enable PWMB interrupt

Bit 0: PWMA Interrupt Enable.

This bit sets the masking of the PWMA interrupt.

0: Disable PWMA interrupt.

1: Enable PWMA interrupt

EIE3: Extended Interrupt Enable 3

Bit	7	6	5	4	3	2	1	0
Name	EXEIE2	EXEIE1	-	PWMCIE	UART2IE	-	-	-
Type	R/W	R/W	R	R/W	R/W	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XB3; SFR Page = 0

Bit 7: EXEIE2 Interrupt Enable.

This bit sets the masking of the INT10~11 Interrupt.

0: Disable INT10~11 Interrupt.

1: Enable INT10~11 Interrupt.

Bit 6: EXEIE1 Interrupt Enable.

This bit sets the masking of the INT2~9 Interrupt.

0: Disable INT2~9 Interrupt.

1: Enable INT2~9 Interrupt.

Bit 5: Reserved. Read = 0, Write = Don't Care.

Bit 4: PWMC Interrupt Enable.

This bit sets the masking of the PWMC interrupt.

0: Disable PWMC interrupt.

1: Enable PWMC interrupt

Bit 3: UART2 Interrupt Enable.

This bit sets the masking of the UART2 interrupt.

0: Disable UART2 interrupt.

1: Enable UART2 interrupt

Bits 2~0: Reserved. Read = 0, Write = Don't Care.

9.2 Interrupt Priority

The interrupt sources can each be programmed individually to low or high priority level (as shown in Table 9-2)below. A high priority interrupt cannot be preempted. However, it can preempt a low priority interrupt service routine. Each interrupt has a related interrupt priority bit in SFR (IP, EIP1, EIP2 or EIP3) used to configure its priority level. The default is Low priority. If two interrupts are recognized concurrently, the interrupt with the higher priority is serviced first. If both interrupts have the same priority level, a fixed priority order is used to determine which to service first, as indicated in Table 9.1

IP (or EIPx)	Interrupt Priority Level
0	Low
1	**High**

Table 9-2 Priority Level Bit Values

IP: Interrupt Priority

Bit	7	6	5	4	3	2	1	0
Name	-	PICIE	PRTCIE	PS0	PT1	PX1	PT0	PX0
Type	R	R/W	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB8; SFR Page = All Pages

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Input-Change Interrupt Priority Control.

This bit sets the priority of the Input-Change interrupt.

0: Input-Change interrupt set to low priority level.

1: Input-Change interrupt set to high priority level.

Bit 5: RTC Interrupt Priority Control.

This bit sets the priority of the RTC interrupt.

0:RTC interrupt set to low priority level.

1: RTC interrupt set to high priority level.

Bit 4: UART0 Interrupt Priority Control.

This bit sets the priority of the UART0 interrupt.

0: UART0 interrupt set to low priority level.

1: UART0 interrupt set to high priority level.

Bit 3: Timer 1 Interrupt Priority Control.

This bit sets the priority of the Timer 1 interrupt.

0: Timer 1 interrupt set to low priority level.

1: Timer 1 interrupt set to high priority level.

Bit 2: External Interrupt 1 Priority Control.

This bit sets the priority of the External Interrupt 1 interrupt.

0: External Interrupt 1 set to low priority level.

1: External Interrupt 1 set to high priority level.

Bit 1: Timer 0 Interrupt Priority Control.

This bit sets the priority of the Timer 0 interrupt.

0: Timer 0 interrupt set to low priority level.

1: Timer 0 interrupt set to high priority level.

Bit 0: External Interrupt 0 Priority Control.

This bit sets the priority of the External Interrupt 0 interrupt.

0: External Interrupt 0 set to low priority level.

1: External Interrupt 0 set to high priority level.

EIP1: Extended Interrupt Priority 1

Bit	7	6	5	4	3	2	1	0
Name	PTC3	PAD	-	PSPI	-	PCMP1	PSH	PHLVD
Type	R/W	R/W	R	R/W	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB9; SFR Page = 0

Bit 7: Timer/Counter 3 Interrupt Priority Control.

This bit sets the priority of the Timer/Counter 3 interrupt.

0: Timer/Counter 3 interrupt set to low priority level.

1: Timer/Counter 3 interrupt set to high priority level.

Bit 6: ADC Conversion Complete Interrupt Priority Control.

This bit sets the priority of the ADC interrupt.

0: ADC interrupt set to low priority level.

1: ADC interrupt set to high priority level.

Bit 5: Reserved. Read = 0, Write = Don't Care.

Bit 4: SPI Interrupt Priority Control.

This bit sets the priority of the ADC interrupt.

0: SPI interrupt set to low priority level.

1: SPI interrupt set to high priority level.

Bits 3: Reserved. Read = 0, Write = Don't Care.

Bit 2: Comparator1 (CMP1) Interrupt Priority Control.

This bit sets the priority of the CMP1 interrupt.

0: CMP1 interrupt set to low priority level.

1: CMP1 interrupt set to high priority level.

Bit 1: System-Hold Interrupt Priority Control.

This bit sets the priority of the System-Hold interrupt.

0: System-Hold interrupt set to low priority level.

1: System-Hold interrupt set to high priority level.

Bit 0: High Low Voltage Detector(HLVD) Interrupt Priority Control.

This bit sets the priority of the HLVD interrupt.

0: HLVD interrupt set to low priority level.

1: HLVD Hold interrupt set to high priority level.

EIP2: Extended Interrupt Priority 2

Bit	7	6	5	4	3	2	1	0
Name	-	-	PI2C	-	PTC4	-	PPWMB	PPWMA
Type	R	R	R/W	R	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBA; SFR Page = 0

Bits 7~6 : Reserved. Read = 0, Write = Don't Care.

Bit 5: I2C Interrupt Priority Control.

This bit sets the priority of the I2C interrupt.

0: I2C interrupt set to low priority level.

1: I2C interrupt set to high priority level.

Bit 4: Reserved. Read = 0, Write = Don't Care.

Bit 3: Timer/Counter 4 Interrupt Priority Control.

This bit sets the priority of the Timer/Counter 4 interrupt.

0: Timer/Counter 4 interrupt set to low priority level.

1: Timer/Counter 4 interrupt set to high priority level.

Bit 2: Reserved. Read = 0, Write = Don't Care.

Bit 1: PWMB Interrupt Priority Control.

This bit sets the priority of the PWMB interrupt.

0: PWMB interrupt set to low priority level.

1: PWMB interrupt set to high priority level.

Bit 0: PWMA Interrupt Priority Control.

This bit sets the priority of the PWMA interrupt.

0: PWMA interrupt set to low priority level.

1: PWMA interrupt set to high priority level.

EIP3: Extended Interrupt Priority 3

Bit	7	6	5	4	3	2	1	0
Name	PEX2	PEX1		PPWMC	PUART2			-
Type	R/W	R/W	R	R/W	R/W	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBB SFR Page = 0

Bit : Reserved. Read = 0, Write = Don't Care.

Bit 7: External INT10~11 Interrupt Priority Control.

This bit sets the priority of the External INT10~11 interrupt.

0: External INT10~11 interrupt set to low priority level.

1: External INT10~11 interrupt set to high priority level.

Bit 6: External INT2~9 Interrupt Priority Control.

This bit sets the priority of the External INT10~11 interrupt.

0: External INT2~9 interrupt set to low priority level.

1: External INT2~9 interrupt set to high priority level.

Bit 5: Reserved. Read = 0, Write = Don't Care.

Bit 4: PWMC Interrupt Priority Control.

This bit sets the priority of the PWMC interrupt.

0: PWMC interrupt set to low priority level.

1: PWMC interrupt set to high priority level.

Bit 3: UART2 Interrupt Priority Control.

This bit sets the priority of the UART2 interrupt.

0: UART2 interrupt set to low priority level.

1: UART2 interrupt set to high priority level.

Bits 2~0: Reserved. Read = 0, Write = Don't Care.

9.3 INT PIN

INT2~11 are external interrupt sources, there are interrupt flags at EXFSF2~11, EXRSF2~11(EXSF2, EXSF3, EXSF4). INT2~11 pins are activated at level or by an edge by setting or clearing EIES2~4. The INTx pins catch port change and set them as an interrupt input event to wakeup MCU (Even when CPU is in Power-Down mode).

When INT2~11, an external interrupt, is configured as edge-sensitive, the corresponding interrupt flag is automatically cleared by hardware when the CPU vectors to the ISR. If configured as level sensitive, the interrupt flag remains at Logic 1 while the input is activated, as defined by the corresponding polarity bit. The flag remains at Logic 0 while the input is inactive. The external interrupt source must keep the input active
(3 system clock + clock source warm up time) until the interrupt request is recognized. Then it must deactivate the interrupt request before execution of the ISR is completed, otherwise another interrupt request will be generated.

EXEN0: External Interrupt Pin Enable.

Bit	7	6	5	4	3	2	1	0
Name	EXE9	EXE8	EXE7	EXE6	EXE5	EXE4	EXE3	EXE2
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB5; SFR Page = 2

Bit 7: INT9 Enable.

Setting this bit activates INT9.

0: Disable INT9.

1: Enable INT9.

Bit 6: INT8 Enable.

Setting this bit activates INT8.

0: Disable INT8.

1: Enable INT8.

Bit 5: INT7 Enable.

Setting this bit activates INT7.

0: Disable INT7.

1: Enable INT7.

Bit 4: INT6 Enable.

Setting this bit activates INT6.

0: Disable INT6.

1: Enable INT6.

Bit 3: INT5 Enable.

Setting this bit activates INT5.

0: Disable INT5.

1: Enable INT5.

Bit 2: INT4 Enable.

Setting this bit activates INT4.

0: Disable INT4.

1: Enable INT4.

Bit 1: INT3 Enable.

Setting this bit activates INT3.

0: Disable INT3.

1: Enable INT3.

Bit 0: INT2 Enable.

Setting this bit activates INT2.

0: Disable INT2.

1: Enable INT2.

EXEN1: External Interrupt Pin Enable.

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	EXE11	EXE10
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB5; SFR Page = 2

Bits 7~2: Reserved. Read = 0, Write = Don't Care.

Bit 1: INT11 Enable.

Setting this bit activates INT11.

0: Disable INT11.

1: Enable INT11.

Bit 0: INT10 Enable.

Setting this bit activates INT10.

0: Disable INT10.

1: Enable INT10.

EIESC1: External Interrupt Edge Select Control 1.

Bit	7	6	5	4	3	2	1	0
Name	EIEDG1	EIEDG0	-	-	-	-	-	-
Type	R/W	R/W	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB6; SFR Page = 0

Bits 7~6: INT Deglitch Time (INT2~11).

EIEDG1	EIEDG0	Deglitch Time (ns)
0	0	50ns
0	1	200ns
1	0	400ns
1	1	Bypass

Bits 5~0: Reserved. Read = 0, Write = Don't Care.

EIES2: External Interrupt Edge Select Control 2.

Bit	7	6	5	4	3	2	1	0
Name	EI5ES1	EI5ES0	EI4ES1	EI4ES0	EI3ES1	EI3ES0	EI2ES1	EI2ES0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBD; SFR Page = 1

Bit 7~6: External Interrupt 5 Edge Select.

EI5ES1	EI5ES0	External Interrupt 5 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

Bit 5~4: External Interrupt 4 Edge Select.

EI4ES1	EI4ES0	External Interrupt 4 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

Bit 3~2: External Interrupt 3 Edge Select.

EI3ES1	EI3ES0	External Interrupt 3 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

Bit 1~0: External Interrupt 2 Edge Select.

EI2ES1	EI2ES0	External Interrupt 2 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

EIES3: External Interrupt Edge Select Control 3.

Bit	7	6	5	4	3	2	1	0
Name	EI9ES1	EI9ES0	EI8ES1	EI8ES0	EI7ES1	EI7ES0	EI6ES1	EI6ES0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBE; SFR Page = 2

Bit 7~6: External Interrupt 9 Edge Select.

EI9ES1	EI9ES0	External Interrupt 9 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

Bit 5~4: External Interrupt 8 Edge Select.

EI8ES1	EI8ES0	External Interrupt 8 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

Bit 3~2: External Interrupt 7 Edge Select.

EI7ES1	EI7ES0	External Interrupt 7 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

Bit 1~0: External Interrupt 6 Edge Select.

EI6ES1	EI6ES0	External Interrupt 6 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

EIES4: External Interrupt Edge Select Control 3.

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	EI11ES1	EI11ES0	EI10ES1	EI10ES0
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBF; SFR Page = 2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bit 3~2: External Interrupt 11 Edge Select.

EI11ES1	EI11ES0	External Interrupt 11 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

Bit 1~0: External Interrupt 10 Edge Select.

EI10ES1	EI10ES0	External Interrupt 10 Edge Select
0	0	Active falling edge triggered
0	1	Active rising edge triggered
1	0	Active rising, falling edge triggered
1	1	Active rising, falling edge triggered

EXSF2: Extended External Interrupt Status Flag.

Bit	7	6	5	4	3	2	1	0
Name	EXFSF5	EXRSF5	EXFSF4	EXRSF4	EXFSF3	EXRSF3	EXFSF2	EXRSF2
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB9; SFR Page = 2

Bit 7: External Interrupt 5 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

Bit 6: External Interrupt 5 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

Bit 5: External Interrupt 4 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

Bit 4: External Interrupt 4 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

Bit 3: External Interrupt 3 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

Bit 2 External Interrupt 3 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

Bit 1 External Interrupt 2 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

Bit 0 External Interrupt 2 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES2 is detected. It can be cleared by software.

EXSF3: Extended External Interrupt Status Flag.

Bit	7	6	5	4	3	2	1	0
Name	EXFSF9	EXRSF9	EXFSF8	EXRSF8	EXFSF7	EXRSF7	EXFSF6	EXRSF6
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBA; SFR Page = 2

Bit 7: External Interrupt 9 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

Bit 6: External Interrupt 9 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

Bit 5: External Interrupt 8 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

Bit 4: External Interrupt 8 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

Bit 3: External Interrupt 7 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

Bit 2 External Interrupt 7 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

Bit 1 External Interrupt 6 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

Bit 0 External Interrupt 6 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES3 is detected. It can be cleared by software.

EXSF4: Extended External Interrupt Status Flag.

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	EXFSF11	EXRSF11	EXFSF10	EXRSF10
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBB; SFR Page = 2

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bit 3: External Interrupt 11 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES4 is detected. It can be cleared by software.

Bit 2 External Interrupt 11 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES4 is detected. It can be cleared by software.

Bit 1 External Interrupt 10 Falling Status Flag.

This flag is set by hardware when an edge of type defined by EIES4 is detected. It can be cleared by software.

Bit 0 External Interrupt 10 Rising Status Flag.

This flag is set by hardware when an edge of type defined by EIES4 is detected. It can be cleared by software.

9.4 Pin-Change

Pin-change is supported to detectthe changes in the pin status, which is possible through logic conversion from high to low and vice versa. Pin-change operations must be enabled by the following: 1) Latch pin status through "MOV direct, direct" instruction; and (2) Set the ICxEN bit. Note that pin-change is invalid when port configuration mode at push-pull output.

ICEN: Input-Change Enable

Bit	7	6	5	4	3	2	1	0
Name	IC3EN	IC2EN	IC1EN	IC0EN	IC3SF	IC2SF	IC1SF	IC0SF
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE9; SFR Page = 0

Bit 7: Input-Change Port 3 Enable.

Setting this bit activates Input-Change Port 3.

0: Disable Input-Change Port 3.

1: Enable Input-Change Port 3.

Bit 6: Input-Change Port 2 Enable.

Setting this bit activates Input-Change Port 2.

0: Disable Input-Change Port 2.

1: Enable Input-Change Port 2.

Bit 5: Input-Change Port 1 Enable.

Setting this bit activates Input-Change Port 1.

0: Disable Input-Change Port 1.

1: Enable Input-Change Port 1.

Bit 4: Input-Change Port 0 Enable.

Setting this bit activates Input-Change Port 0.

0: Disable Input-Change Port 0.

1: Enable Input-Change Port 0.

Bit 3: Input-Change Port 3 Status Flag.

This flag is set by hardware when an input change is detected. It can be cleared by software.

Bit 2: Input-Change Port 2 Status Flag.

This flag is set by hardware when an input change is detected. It can be cleared by software.

Bit 1: Input-Change Port 1 Status Flag.

This flag is set by hardware when an input change is detected. It can be cleared by software.

Bit 0: Input-Change Port 0 Status Flag.

This flag is set by hardware when an input change is detected. It can be cleared by software.

QBODICEN: Ouasi-Bidirectional and Open-Drain Input-Change Enable

Bit	7	6	5	4	3	2	1	0
Name	QBIC3EN	QBIC2EN	QBIC1EN	QBIC0EN	ODIC3EN	ODIC2EN	ODIC1EN	ODIC0EN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xEA; SFR Page = 0

Bit 7: Input-Change Port 3 Enable of Quasi-Bidirectional mode

Setting this bit activates Input-Change Port 3.

0: Disable Input-Change Port 3.

1: Enable Input-Change Port 3.

Bit 6: Input-Change Port 2 Enable of Quasi-Bidirectional mode

Setting this bit activates Input-Change Port 2.

0: Disable Input-Change Port 2.

1: Enable Input-Change Port 2.

Bit 5: Input-Change Port 1 Enable of Quasi-Bidirectional mode

Setting this bit activates Input-Change Port 1.

0: Disable Input-Change Port 1.

1: Enable Input-Change Port 1.

Bit 4: Input-Change Port 0 Enable of Quasi-Bidirectional mode

Setting this bit activates Input-Change Port 0.

0: Disable Input-Change Port 0.

1: Enable Input-Change Port 0.

Bit 3: Input-Change Port 3 Enable of Open-Drain mode

Setting this bit activates Input-Change Port 3.

0: Disable Input-Change Port 3.

1: Enable Input-Change Port 3.

Bit 2: Input-Change Port 2 Enable of Open-Drain mode

Setting this bit activates Input-Change Port 2.

0: Disable Input-Change Port 2.

1: Enable Input-Change Port 2.

Bit 1: Input-Change Port 1 Enable of Open-Drain mode

Setting this bit activates Input-Change Port 1.

0: Disable Input-Change Port 1.

1: Enable Input-Change Port 1.

Bit 0: Input-Change Port 0 Enable of Open-Drain mode

Setting this bit activates Input-Change Port 0.

0: Disable Input-Change Port 0.

1: Enable Input-Change Port 0.

10 Watchdog Timer (WDT)

The Watchdog Timer starts counting upward when the WDTE bit is set to “1” and stops (or WDT is not used) by first clearing WDTE bit. first then, write 0xB1 to WDTKEY in order. The WDT is enabled in default. On the other hand, the clock source of WDT is low speed clock.

WDTCR: WDT Control Register

Bit	7	6	5	4	3	2	1	0
Name	WDTE	-	-	-	WPSR3	WPSR2	WPSR1	WPSR0
Type	R/W	R	R	R	R/W	R/W	R/W	R/W
Reset	1	0	0	0	1	1	1	1

SFR Address = 0xCF; SFR Page = 0

Bit 7: Watchdog Timer Enable Bit.

0: Disable WDT (with WDTKEY = 0xB1).

1: Enable WDT.

Bits 6~4: Reserved. Read = 0, Write = Don't Care.

Bits 3~0: Watchdog pre-scaler.

WPSR3	WPSR2	WPSR1	WPSR0	WDT Rate
0	0	0	0	1:1
0	0	0	1	1:2
0	0	1	0	1:4
0	0	1	1	1:8
0	1	0	0	1:16
0	1	0	1	1:32
0	1	1	0	1:64
0	1	1	1	1:128
1	0	0	0	1:256
1	0	0	1	1:512
1	0	1	0	1:1024
1	0	1	1	1:2048
1	1	0	0	1:4096
1	1	0	1	1:8192
1	1	1	0	1:16384
1	1	1	1	1:32768

During use, such as switching WDT rate, HW will let the WDT timer count again.

WDTKEY: WDT Key

Bit	7	6	5	4	3	2	1	0
Name	WDTKEY.7	WDTKEY.6	WDTKEY.5	WDTKEY.4	WDTKEY.3	WDTKEY.2	WDTKEY.1	WDTKEY.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCE; SFR Page = 0

Writing 0x4E to this register will reset the counter. Before writing 0xB1 to this register, the WDTE must be set to 0. Then WDT will be disabled.

(The WDTKEY will keep 0x4E after resetting the WDT. To disable the WDT, WDTE should be cleared first, then 0xB1 be written to WDTKEY. Note that WDTKEY will not be cleared after disabling the WDT).

11 Inter-Integrated Circuit (I2C)

The MCU supports a bidirectional, 2-wire bus, 7/10-bit addressing, and data transmission protocol. A device that sends data onto the bus is defined as a transmitter, while a device receiving data is defined as a receiver. The bus has to be controlled by a master device which generates the Serial Clock (SCL), controls the bus access, and generates the Start and Stop conditions. Both master and slave can operate as a transmitter or a receiver, but the master device determines which mode is to be activated.

Both SDA and SCL are bi-directional lines, connected to a positive supply voltage via a pull-up resistor. When the bus is free, both lines are HIGH. The output stages of devices connected to the bus must have an open-drain or open-collector to perform the wired-AND function. Data on the I2C-bus can be transferred at rates of up to 100k bit/s in the Standard-mode or up to 400k bit/s in Fast-mode.

The data on the SDA line must remain stable during the HIGH period of the clock. The HIGH or LOW state of the data line can only change when the clock signal on the SCL line is LOW.

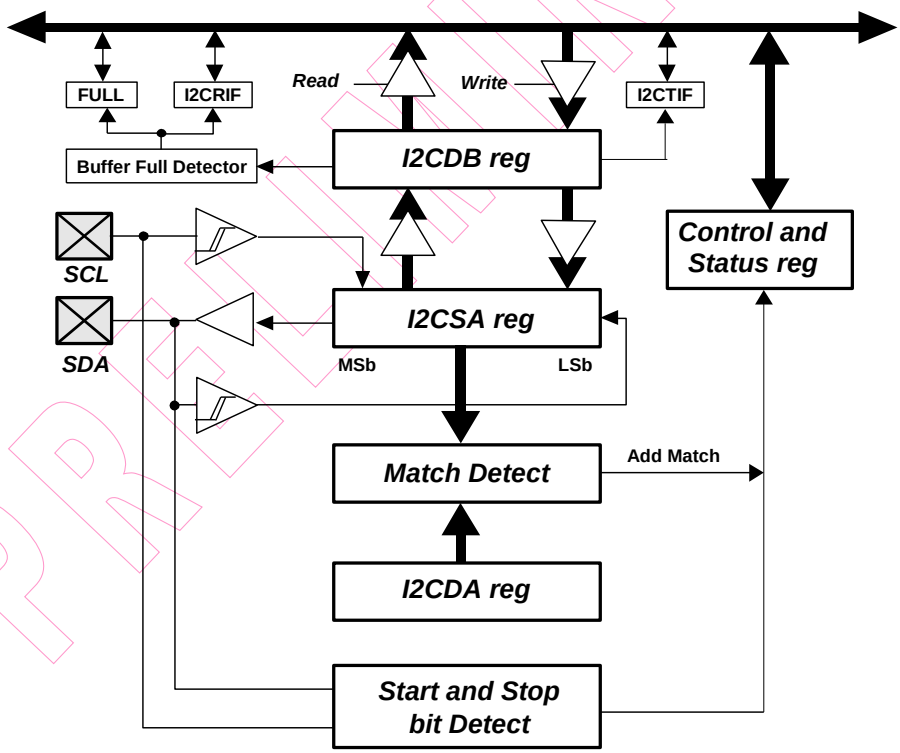


Figure 11- 1 I2C Block Diagram

Table 12.1 I2C interrupt occurs as shown below:

Condition	Master/Slave	Transmit Address	Transmit Data	Stop
Master-transmitter transmits to slave-receiver	Master	Transmit interrupt	Transmit interrupt	Stop interrupt
	slave	Receive interrupt	Receive interrupt	Stop interrupt
Master receiver read slave-transmitter	Master	Transmit interrupt	Receive interrupt	Stop interrupt
	slave	Transmit interrupt	Transmit interrupt	Stop interrupt

Within the procedure of the I2C bus, unique situations arise which are defined as START (S) and STOP (P) conditions. A HIGH to LOW transition on the SDA line while SCL is HIGH is one such unique case. This situation indicates a START condition. A LOW to HIGH transition on the SDA line while SCL is HIGH defines a STOP condition.

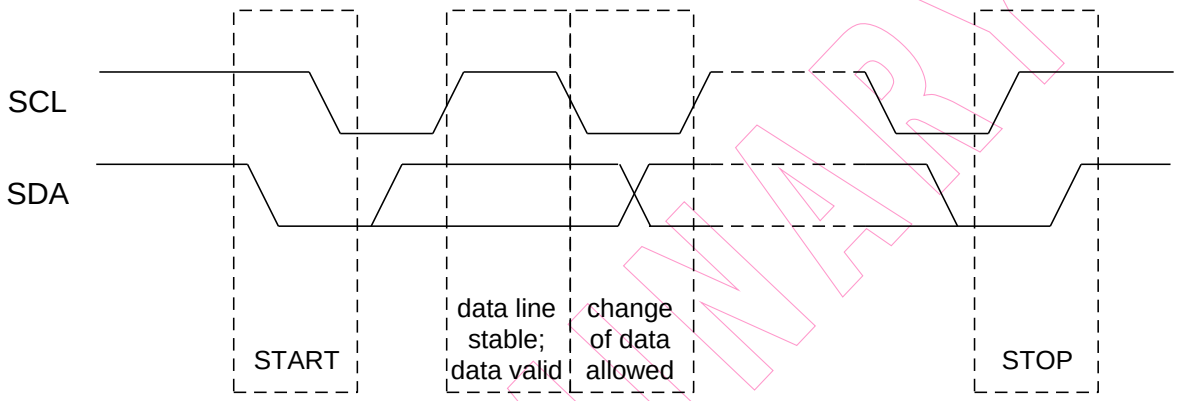


Figure 11- 2 I2C Transfer Condition

11.1 7-Bit Slave Address

Master-transmitter transmits to slave-receiver. The transfer direction is not changed. Master reads slave immediately after first byte. At the moment of the first acknowledge, the master- transmitter becomes a master- receiver and the slave-receiver becomes a slave-transmitter. This first acknowledge is still generated by the slave. The STOP condition is generated by the master, which has previously sent a not-acknowledge (A).

The difference between a master transmitter and a master receiver is their R//W bits. If the R//W bit were “0”, the master device would be a transmitter; the other way, the master device would be a receiver. The master transmitter is described by the Figure 7-Bit Slave Address in Master-Transmitter transmits to Slave-Receiver, and the master receiver is described by Figure 7-Bit Slave Address in Master-Receiver reads the Slave-Transmitter.

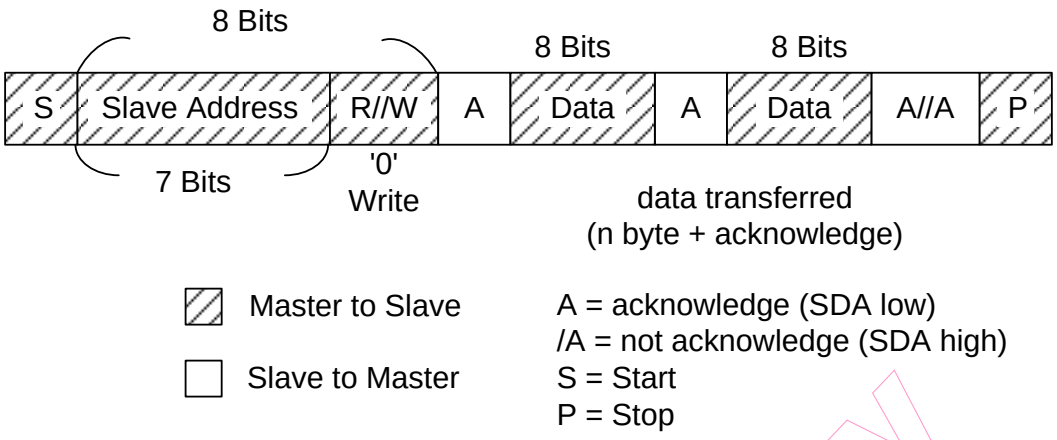


Figure 11- 3 Master-Transmitter transmits to Slave-Receiver (7-Bit Slave Address)

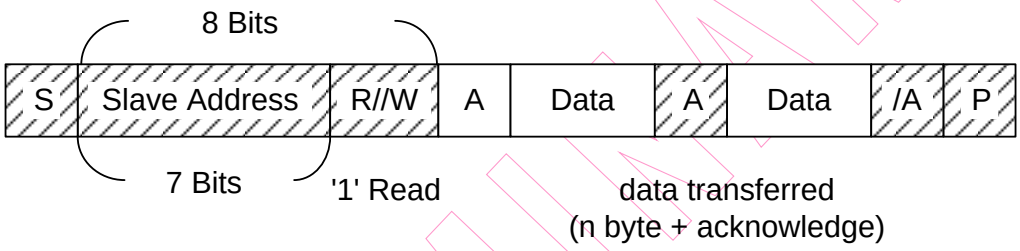


Figure 11- 4 Master Receiver reads Slave-Transmitter (7-Bit Slave Address)

11.2 10-Bit Slave Address

In 10-Bit slave address mode, using 10-Bit for addressing exploits the reserved combination 11110XX for the first seven bits of the first byte following a START(S) or repeated START (Sr) condition. The first seven bits of the first byte are the combination 11110XX of which the last two bits (XX) are the two most-significant bits of the 10-bits address. If the R//W bit were “0”, the second byte after acknowledge would be the eight address bits of 10-bits slave address; in the other way, the second byte would just only be the next transmitted data from a slave to master device. The first bytes 11110XX be transmitted by using the slave address register (I2CSA), and the second bytes XXXXXXXX would be transmitted by using the data buffer (I2CDB).

There are few kinds of different formats that would be explained in Figure 39 ~ Figure 43 in the 10-bit slave address mode. The possible data transfer formats are:

Master-transmitter transmits to slave-receiver with a 10-bit slave address:

When the slave have received the first byte after START bit from master, each slave devices will compare the seven bits of the first byte (11110XX) with its own address and, the eighth bit, R//W, if the R//W bit were “0”, the slave would return the Acknowledge (A1) and it is possible to have more than one slave device returning it. Then all slave device will continue to compare the second address (XXXXXXXX), if the slave device have had a match, that would be only one slave device returning acknowledge. The matching slave device will remain addressed by the master until it receives the STOP condition or a repeated START condition followed by at different slave address.

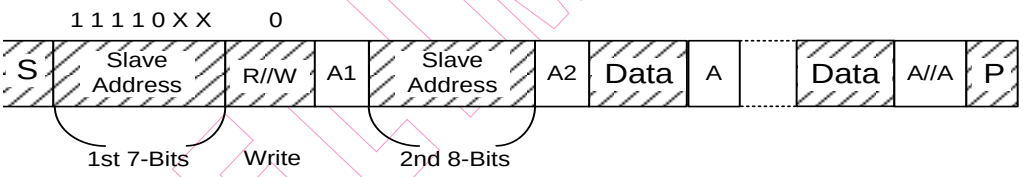


Figure 11- 5Master-transmitter transmits to slave-receiver (10-bits slave address)

Master-Receiver reads Slave-Transmitter with a 10-bit Slave Address:

Up to and including acknowledge bit A2, the procedure is the same as that described for master-transmitter addressing a slave receiver. After the acknowledge A2, a repeated START condition (Sr) is followed by seven bits slave address (11110XX). However, the eighth bit R//W is “1”, the addressed slave device will return the acknowledge A3. If the repeated START(Sr) condition and the seven bits of first byte(11110XX) are received by slave device, all the slave device would compare with their own addresses and test the eighth R//W, but none of the slave device will return the acknowledge because R//W=1.

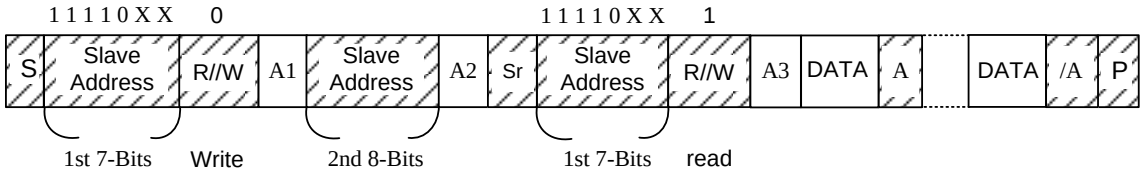


Figure 11- 6 Master-receiver read slave-transmitter (10-bits slave address)

Master addresses a slave with 10-Bits addresses transmits and receives data in the same slave device:

At first the transmitter procedure is the same as the section of the Master-transmitter transmits to slave-receiver with a 10-bits slave address. Then the master device can start to transmit the data to slave device. The slave device will receive either Acknowledge or none acknowledge followed by repeating START (Sr), and repeating the procedure of "Master-receiver read slave-transmitter with a 10-bit slave address".

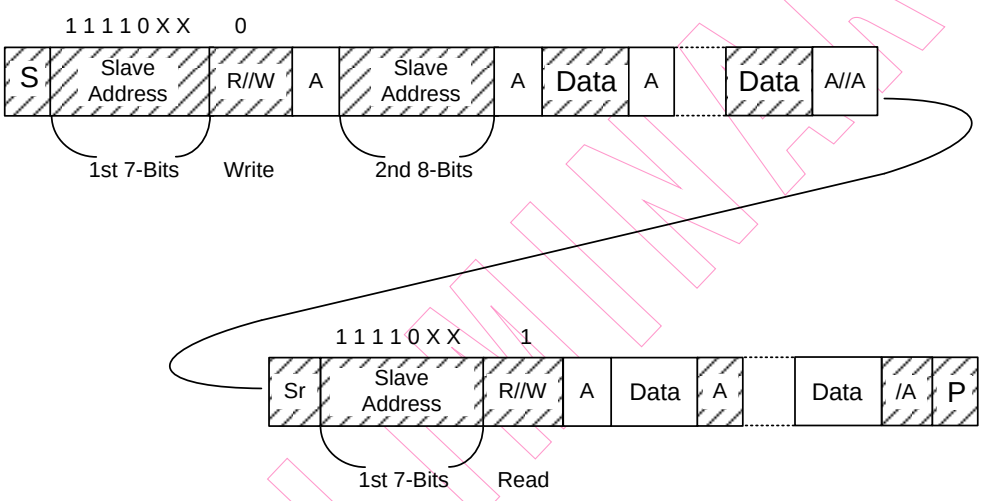


Figure 11- 7Master transmits and receives data in the same slave device (10-Bit Slave Addresses)

Master device transmit data to two or more than two slave device:

The section of "Master-transmitter transmits to slave-receiver with a 10-bit slave address" describes the procedure for transmitting data to slave device, if the master device has finished the transmittal, and want to transmit the data to another device, the master would need to address the new slave device, the address procedure is described in the section of the "Master-transmitter transmits to slave-receiver with a 10-Bit slave address". If the master device wants to transmit the data in 7-Bits slave address mode and transmit the data in 10-Bits slave address mode in the serial transfer, after the START or repeat START conditions, a 7-Bits and 10-Bit address could be transmitted. Figure 11-9 shows how to transmit the data in 7-Bit and 10-Bit address mode in serial transfer..

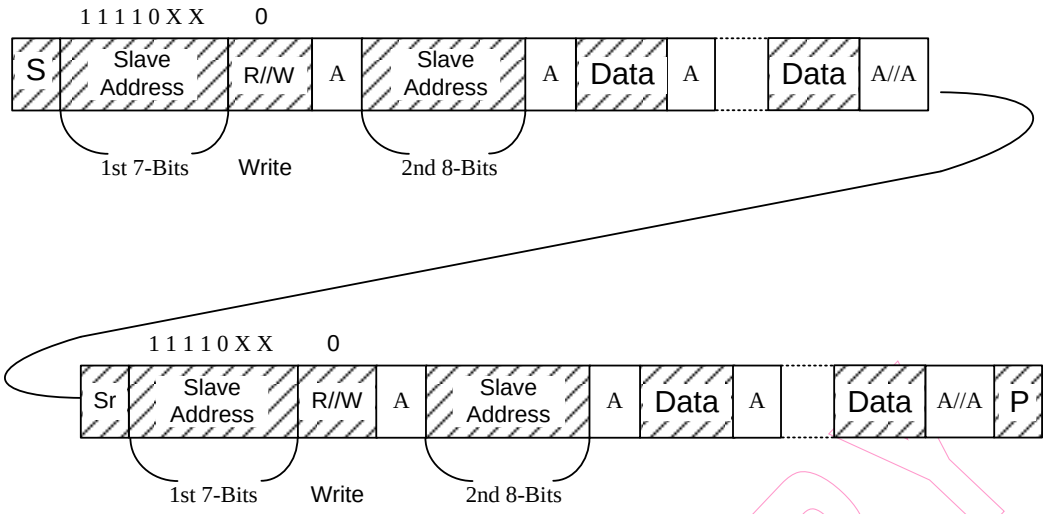


Figure 11- 8 Transmit one more devices (10-bit Slave Address)

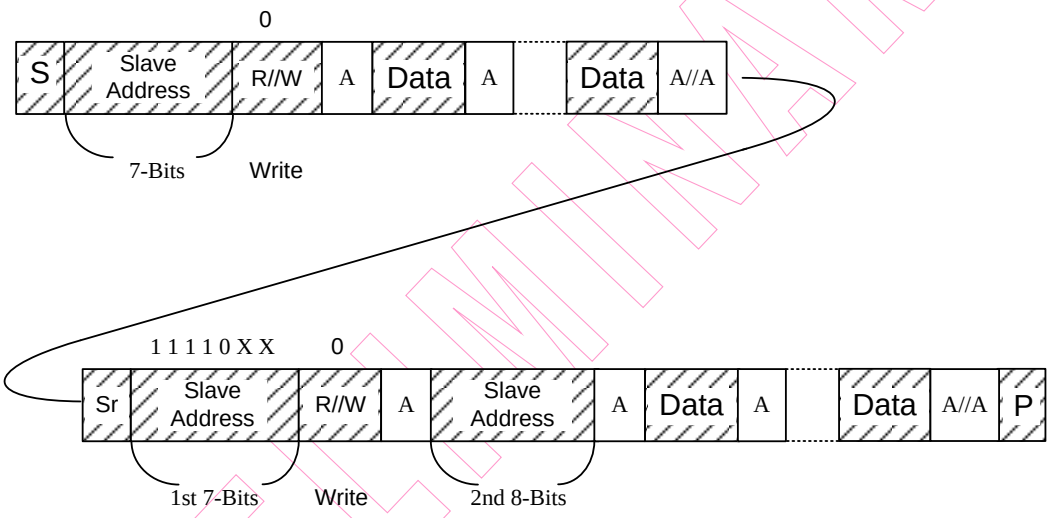


Figure 11- 9 Slave Address Mode with 7 bits and 10 bits

Master Mode:

In transmitting/receiving serial data, the I2C operates as follows:

Step 1: Set I2CTS1~0 and ISS bits to select I2C transmit clock source.

Step 2: Set I2CEN and IMS bits to enable I2C master function.

Step 3: Write slave address into the I2CSA register and IRW bit to select read or write.

Step 4: Set strobe bit will start transmit and then Check I2CTXSF (I2CRXSF) bit.

Step 5: Write 1st data into the I2CDB register, set strobe bit and Check I2CTXSF (I2CRXSF) bit.

Step 6: Write 2nd data into the I2CDB register, set strobe bit, STOP bit and Check I2CTXSF (I2CRXSF) bit.

Slave Mode:

In receiving/transmitting serial data, the I2C operates as follows:

Step 1: Set I2CTS1~0 and ISS bits to select I2C transmit clock source.

Step 2: Set I2CEN and IMS bits to enable I2C slave function.

Step 3: Write device address into the I2CDA register.

Step 4: Check I2CRXSF (I2CTXSF) bit, read I2CDB register (address) and then clear Pend bit.

Step 5: Check I2CRXSF (I2CTXSF) bit, read I2CDB register (1st data) and then clear Pend bit.

Step 6: Check I2CRXSF (I2CTXSF) bit, read I2CDB register (2nd data) and then clear the Pend bit.

Step 7: Check I2CSTPSF bit, end transmission.

I2CCR1: I2C Status and Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	Strobe/Pend	IMS	ISS	STOP	SAR_EMPTY	ACK	FULL	EMPTY
Type	R/W	R/W	R/W	R/W	R	R	R	R
Reset	0	0	0	0	1	0	0	1

SFR Address = 0xB1; SFR Page = 1

Bit 7: In master mode, this bit is used as strobe signal to control I2C circuit to send SCL clock. Reset automatically after receiving or transmitting handshake signal (ACK or NACK). In slave mode, it is used as pending signal, user should clear it after filling data into Tx buffer or get data from Rx buffer to inform slave I2C circuit to release SCL signal.

Bit 6: I2C Master/Slave mode select bit.

0: Slave

1: Master

Bit 5: I2C Fast/Standard mode select bit. (If Fm is 4MHz and I2CTS1~0<0,0>)

0: Standard mode (100K bit/s)

1: Fast mode (400K bit/s)

Bit 4: In Master mode, if STOP=1 and R/nW=1, then MCU must return nACK signal to slave device before sending STOP signal. If STOP=1 and R/nW=0, then MCU send STOP signal after receiving an ACK signal. Reset when MCU send STOP signal to Slave device. In slave mode, if STOP=1 and R/nW=0, then MCU must return nACK signal to master device.

Bit 3: Set when MCU transmit 1 byte data from I2C Slave Address Register and receive ACK (or nACK) signal. Reset when MCU write 1 byte data to I2C Slave Address Register.

Bit 2: The Ack condition bit is set to 1 by hardware when the device responds acknowledge (ACK). Reset when the device responds not-acknowledge (nACK) signal.

Bit 1: Set by hardware when I2C receive buffer register is full. Reset by hardware when MCU read data from I2C receive buffer register.

Bit 0: Set by hardware when I2C transmit buffer register is empty and receive ACK (or nACK) signal. Reset by hardware when MCU write new data to I2C transmit buffer register.

I2CCR2: I2C Status and Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	I2CBF	GCEN	RSTSWN	BBF	I2CTS2	I2CTS1	I2CTS0	I2CEN
Type	R	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	1	0	0	0	0	0

SFR Address = 0xB2; SFR Page = 1

Bit 7: I2C Busy Flag Bit

0: clear to "0" in Slave mode, if receive STOP signal or I2C slave address not match.

1: set when I2C communicate with master in slave mode.

*The bit is set to 1 if STAR signal is received and it is cleaned when I2C slave address is not match, I2C disable or STOP signal is received in slave mode.

Bit 6: I2C General Call Function Enable Bit

0: Disable General Call Function

1: Enable General Call Function

Bit 5: Software Reset Bit.

0: Software reset disabled.

1: Software reset enabled (It is used to reset the FSM and release the SCL).

We always do not set this bit.

Bit 3: Busy Flag Bit. I2C detection is busy in the master mode. Read only.

*The bit is set to 1 if STAR signal is transmitted and it is cleaned when I2C disable or STOP signal is transmitted in master mode.

Bits 3~1: I2C Transmit Clock Select Bits. When using different operating frequency (FHS), these bits must set correctly to let SCL fill in with standard/fast mode.

I2CCR1 Bit 5=1, Fast Mode

I2CTS2	I2CTS1	I2CTS0	SCL CLK	Operating F_{HS} (MHz)
0	0	0	$F_{HS}/10$	*4
0	0	1	$F_{HS}/15$	6
0	1	0	$F_{HS}/20$	8
0	1	1	$F_{HS}/30$	12
1	0	0	$F_{HS}/40$	16
1	0	1	$F_{HS}/50$	-
1	1	0	$F_{HS}/60$	-
1	1	1	$F_{HS}/70$	-

*: 4MHz is the lowest limited system frequency of I2C in Fast Mode.

I2CCR1 Bit 5=0, Standard Mode

I2CTS2	I2CTS1	I2CTS0	SCL CLK	Operating F_{HS} (MHz)
0	0	0	$F_{HS}/40$	4
0	0	1	$F_{HS}/60$	6
0	1	0	$F_{HS}/80$	8
0	1	1	$F_{HS}/120$	12
1	0	0	$F_{HS}/160$	16
1	0	1	$F_{HS}/200$	20
1	1	0	$F_{HS}/240$	-
1	1	1	$F_{HS}/280$	-

Bit 0: I2C Enable Bit

0: Disable I2C mode

1: Enable I2C mode

I2CSA: I2C Slave Address Register

Bit	7	6	5	4	3	2	1	0
Name	SA6	SA5	SA4	SA3	SA2	SA1	SA0	IRW
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB3; SFR Page = 1

Bits 7~1: When MCU is used as master device for I2C application,this is the slave device address register.

Bit 0: When MCU is used as master device for I2C application,this bit is Read/Write transaction control bit.

0: Write

1: Read

I2CDB: I2C Data Buffer Register

Bit	7	6	5	4	3	2	1	0
Name	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB4; SFR Page = 1

Bits 7~0: I2C Receive/Transmit Data Buffer.

I2CDAL: I2C Device Address Register

Bit	7	6	5	4	3	2	1	0
Name	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB5; SFR Page = 1

Bits 7~0: When MCU is used as slave device for I2C application, this register stores the address of MCU. It is used to identify the data on the I2C bus to extract the message delivered to the MCU.

*Slave address 0x77 is reserved for OCD use.

I2CDAH: I2C Device Address Register

Bit	7	6	5	4	3	2	1	0
Name	SCL_I2CDG1	SCL_I2CDG0	SDA_I2CDG1	SDA_I2CDG0	I2CBFEN	ADDR_SYNC	DA9	DA8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB6; SFR Page = 1

Bits 7~6: Select I2C SCL deglitch time select bits.

SCL_I2CDG1	SCL_I2CDG0	Deglitch Time +/-30%
0	0	50ns
0	1	200ns
1	0	400ns
1	1	Bypass

Bits 5~4: Select I2C SDA deglitch time select bits.

SDA_I2CDG1	SDA_I2CDG0	Deglitch Time +/-30%
0	0	50ns
0	1	200ns
1	0	400ns
1	1	Bypass

Bit 3: I2C Buffer Enable

0: single mode

1: Buffer mode Enable

Bit 2: Check address with async/sync SCL/SDA.

0: Check address with async SCL/SDA.

1: Check address with sync SCL/SDA.

Bits 1~0: I2C Device Address Bit 9 and Bit 8.

I2CSF: I2C Status Flag

Bit	7	6	5	4	3	2	1	0
Name	I2CTXSF	I2CRXSF	I2CSTPIEN	I2CSTPSF	-	-	-	-
Type	R/W	R/W	R/W	R/W	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB7; SFR Page = 1

Bit 7 (I2CTXSF): I2C transmit status flag. Set when I2C transmits 1 byte data and receive handshake signal (ACK or NACK). Reset by firmware or I2C disable.

Bit 6 (I2CRXSF): I2C receive status flag. Set when I2C receives 1 byte data and responds ACK signal. Reset by firmware or I2C disable.

Bit 5 (I2CSTOIEN): STOP interrupt enable bit

0: disable

1: enable

Bit 4 (I2CSTPSF): I2C stop status flag. Set when I2C generates stop signal.

Bits 3~0: Reserved. Read = 0, Write = Don't Care.

I2CRXAF: I2C Receiver data Almost Full Buffer Control length

Bit	7	6	5	4	3	2	1	0
Name	I2CRXAFSF	-	-	I2CRA4	I2CRA3	I2CRA2	I2CRA1	I2CRA0
Type	R/W	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XB9; SFR Page = 1

Bit 7: I2C Rx FIFO almost full status bit for buffer mode only.

0: rx fifo data length is less than rx almost full buffer length

1: rx fifo data length is greater or equal to rx almost full buffer length

Bits 6~5: Reserved. Read = 0, Write = Don't Care.

Bits 4~0 (I2CRA4~I2CRA0): I2C RX transmission data almost full length set (length=2~16)

I2CTO1: I2C Time-out 1

Bit	7	6	5	4	3	2	1	0
Name	TO1EN	TO1SF				TOTPRE2	TOTPRE1	TOTPRE0
Type	R/W	R/W	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBA; SFR Page = 1

Bit 7 (TO1EN): I2C time out function enable bit: (standard 35ms)

0: Disable

1: Enable

Counting starts if the SCL signal is in low voltage. When contents of the up-counter match with the TO1TR, interrupt is generated and the counter is cleared. Counter is cleared if SCL is a high voltage signal.

Bit 6 (TO1SF): I2C time out 1 status flag

0: Interrupt trigger

1: Interrupt non-trigger

Bits 5~3: Reserved. Read = 0, Write = Don't Care.

Bits 2~0 (TOTPRE2~ TOTPRE0): Timer pre-scaler selection for 8-bit

TOTPRE2	TOTPRE1	TOTPRE0	Pre-scaler
0	0	0	F_{LS} / 1
0	0	1	F_{LS} / 2
0	1	0	F_{LS} / 4
0	1	1	F_{LS} / 8
1	0	0	F_{LS} / 16
1	0	1	F_{LS} / 32
1	1	0	F_{LS} / 64
1	1	1	F_{LS} / 128

I2CTO1R: I2C Timer Reload 1 Register

Bit	7	6	5	4	3	2	1	0
Name	TO1R.7	TO1R.6	TO1R.5	TO1R.4	TO1R.3	TO1R.2	TO1R.1	TO1R.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBB; SFR Page = 1

Bits 7~0 (TO1R.7~TO1R.0): I2C timer-reload value

I2CTO2: I2C Time-out 2

Bit	7	6	5	4	3	2	1	0
Name	TO2EN	TO2SF	-	-	-	TO2TPRE2	TO2TPRE1	TO2TPRE0
Type	R/W	R/W	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBC; SFR Page = 1

Bit 7 (TO2EN): I2C time out function enable bit: (standard 25ms)

0: Disable

1: Enable

Count from start to stop signal. When contents of the up-counter match with the TO2TR, interrupt is generated and the counter is cleared.

Bit 6 (TO2SF): I2C time out 2 status flag

0: Interrupt trigger

1: Interrupt non-trigger

Bits 5~3: Reserved. Read = 0, Write = Don't Care.

Bits 2~0 (TO2TPRE2~ TO2TPRE0): Timer pre-scaler selection for 8-bit

TO2TPRE2	TO2TPRE1	TO2TPRE0	Pre-scaler
0	0	0	F_{LS} / 1
0	0	1	F_{LS} / 2
0	1	0	F_{LS} / 4
0	1	1	F_{LS} / 8
1	0	0	F_{LS} / 16
1	0	1	F_{LS} / 32
1	1	0	F_{LS} / 64
1	1	1	F_{LS} / 128

I2CTO2R: I2C Timer reloads 2 Register

Bit	7	6	5	4	3	2	1	0
Name	TO2R.7	TO2R.6	TO2R.5	TO2R.4	TO2R.3	TO2R.2	TO2R.1	TO2R.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBD; SFR Page = 1

Bits 7~0 (TO2R.7~TO2R.0): I2C timer-reload value

I2CTO3: I2C Time-out 3

Bit	7	6	5	4	3	2	1	0
Name	TO3EN	TO3SF	-	-	-	TO3TPRE2	TO3TPRE1	TO3TPRE0
Type	R/W	R/W	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBE; SFR Page = 1

Bit 7 (TO3EN): I2C time out function enable bit: (standard 25ms)

0: Disable

1: Enable

Count from START-to-ACK, ACK-to-ACK, or ACK-to-STOP. When contents of the up-counter match with the TO3TR, interrupt is generated and the counter is cleared.

Bit 6 (TO3SF): I2C time out 3 status flag

0: Interrupt trigger

1: Interrupt non-trigger

Bits 5~3: Reserved. Read = 0, Write = Don't Care.

Bits 2~0 (TO3TPRE2~ TO3TPRE0): Timer pre-scaler selection for 8-bit

TO3TPRE2	TO3TPRE1	TO3TPRE0	Pre-scaler
0	0	0	F_{LS} / 1
0	0	1	F_{LS} / 2
0	1	0	F_{LS} / 4
0	1	1	F_{LS} / 8
1	0	0	F_{LS} / 16
1	0	1	F_{LS} / 32
1	1	0	F_{LS} / 64
1	1	1	F_{LS} / 128

I2CTO3R: I2C Timer reload 3 Register

Bit	7	6	5	4	3	2	1	0
Name	TO3R.7	TO3R.6	TO3R.5	TO3R.4	TO3R.3	TO3R.2	TO3R.1	TO3R.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBF; SFR Page = 1

Bits 7~0 (TO2R.7~TO2R.0): I2C timer reload value

I2CTXLEN: I2C Transmit Buffer Length

Bit	7	6	5	4	3	2	1	0
Name	I2CTCC	-	-	I2CTL4	I2CTL3	I2CTL2	I2CTL1	I2CTL0
Type	R/W	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	1	0	0	0	0

SFR Address = 0xC1; SFR Page = 1

Bit 7 (I2CTCC): I2C TX_CNT reset bit

0: Reset TX_CNT when I2CTX_SF active

1: Reset TX_CNT when TX_LEN = TX_CNT

Bits 6~5: Reserved. Read = 0, Write = Don't Care.

Bits 4~0 (I2CTL4~I2CTL0): I2C TX transmission data length (length =2 ~ 16)

I2CRXLEN: I2C Receiver Buffer Length

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	I2CRL4	I2CRL3	I2CRL2	I2CRL1	I2CRL0
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC2; SFR Page = 1

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bits 4~0 (I2CRL4~I2CRL0): I2C RX transmission data length (length =2 ~ 16)

I2CSTASU: I2C start/stop setup time timing register

Bit	7	6	5	4	3	2	1	0
Name	STASU7	STASU6	STASU5	STASU4	STASU3	STASU2	STASU1	STASU0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	1	1

SFR Address = 0XC3; SFR Page = 1

Bit 7~0: I2C start condition setup time detection.

tSU; STA = (STASU + 1) / system clock

※Note : tSU; STA = tHD; STO (min)

Standard mode = 4.7us

Fast mode = 0.6us

I2CSTAHD: I2C start/stop hold time timing register

Bit	7	6	5	4	3	2	1	0
Name	STAHD7	STAHD6	STAHD5	STAHD4	STAHD3	STAHD2	STAHD1	STAHD0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	1	1

SFR Address = 0XC4; SFR Page = 1

Bit 7~0: I2C start condition hold time detection.

tHD; STA = (STAHD + 1) / system clock

※Note : tHD; STA = tSD; STO (min)

Standard mode = 4.7us

Fast mode = 0.6us

I2CSCLFIR: I2C SCL digital filter

Bit	7	6	5	4	3	2	1	0
Name	SCL_DF_TYPE	SCL_DF6	SCL_DF5	SCL_DF4	SCL_DF3	SCL_DF2	SCL_DF1	SCL_DF0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XC5; SFR Page = 1

Bit 7: Filter noise level select on bus.

0: filter noise at High and Low level on bus.

1: only filter noise at High or Low level on bus.

Bit 6~0: Digital filter counter for the low level noise of SCL (should be set 0x00 before sleep).

I2CSDAFIR: I2C SDA digital filter

Bit	7	6	5	4	3	2	1	0
Name	SDA_DF_TYPE	SDA_DF6	SDA_DF5	SDA_DF4	SDA_DF3	SDA_DF2	SDA_DF1	SDA_DF0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XC6; SFR Page = 1

Bit 7: Filter noise level select on bus.

0: filter noise at High and Low level on bus.

1: only filter noise at High or Low level on bus.

Bit 6~0: Digital filter counter for the low level noise of SDA (should be set 0x00 before sleep).

I2CCR4: I2C status and control register 4

Bit	7	6	5	4	3	2	1	0
Name	SDA_IN_DEL1	SDA_IN_DEL0	SCL_IN_DEL1	SCL_IN_DEL0	SCL_FIL_LEV	SDA_FIL_LEV	-	-
Type	R/W	R/W	R/W	R/W	R/W	R/W	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XC7; SFR Page = 1

Bit 7~6: Select SDA input delay time for hold time requirement. (used in Power-Down mode).

SDA_IN_DEL1	SDA_IN_DEL0	Deglitch Time +/-30%
0	0	Bypass
0	1	30ns
1	0	100ns
1	1	300ns

Bit 5~4: Select SCL input delay time for hold time requirement. (used in Power-Down mode).

SCL_IN_DEL1	SCL_IN_DEL0	Deglitch Time +/-30%
0	0	Bypass
0	1	30ns
1	0	100ns
1	1	300ns

Bit 3: Filter noise select of SCL.

0: filter noise at Low level on bus.

1: filter noise at High level on bus.

Bit 2: Filter noise select of SDA

0: filter noise at Low level on bus.

1: filter noise at High level on bus.

12 UART0

UART0 has two associated SFRs: Serial Port 0 Control Register 0 (SCON0) and Serial Port 0 Data Buffer (SBUF0). The SBUF0 consists of two separate registers: transmit and receive registers. A data writes into the SBUF0 sets this data in UART0 output register and starts a transmission. A data reads from SBUF0 reads data from the UART0 receive register.

The serial port can operate in 4 modes: one synchronous and three asynchronous modes as shown in Table 12-1. Mode 2 and 3 haves a special feature for multiprocessor communications. This feature is enabled by setting MCE0 bit in SCON0 register. The master processor first sends out an address byte, which identifies the target slave. An address byte differs from a data byte in that the 9th bit is 1 in an address byte and 0 in a data byte. With MCE0 = 1, no slave will be interrupted by a data byte. An address byte will interrupt all slaves. The addressed slave will clear its MCE0 bit and prepare to receive the data bytes that will be coming. The slaves that were not being addressed leave their MCE0 set and ignoring the incoming data.

Mode	Description
0	Synchronous
1	8-bit UART (Variable Baud Rate)
2	9-bit UART (Fixed Baud Rate)
3	9-bit UART (Variable Baud Rate)

Table 12- 1 UART0 Modes

12.1 UART0 Mode 0: Synchronous

A total of 10 bits per data byte is used by the 8-bit UART mode, with one Start bit, eight Data bits (LSB first), and one Stop bit. Data bits or LSB are transmitted first from the TX0 pin and received at the RX0 pin. Upon receiving, the Stop bit goes into RB08 (SCON0.2) and the eight Data bits are stored in SBUF0.

When the software writes a data byte to the SBUF0 register, data transmission starts. The TI0 Transmit Interrupt Flag (SCON0.1) is set at the end of the transmission, which is the beginning of the stop-bit time. Data reception can start any time after theREN0 Receive Enable bit (SCON0.4) is set to Logic 1. After receiving the stop bit, data byte will be loaded into the SBUF0 receiving register if the following conditions are satisfied: RI0 must be Logic 0, and if MCE0 is Logic 1, the stop bit must be Logic 1. When there's data overrun, the first 8 bits received are latched into the SBUF0 receiving register and the subsequent overrun data bits are lost.

If the conditions are satisfied, the 8 bits of data will be stored in SBUF0, the Stop bit will be stored in RB08 and the R10 flag is set. However, on the contrary, if the conditions are not satisfied, SBUF0 and RB08 will not be loaded and the RI0 flag will not be set. When either T10 or R10 is set, and if enabled, an interrupt will occur.

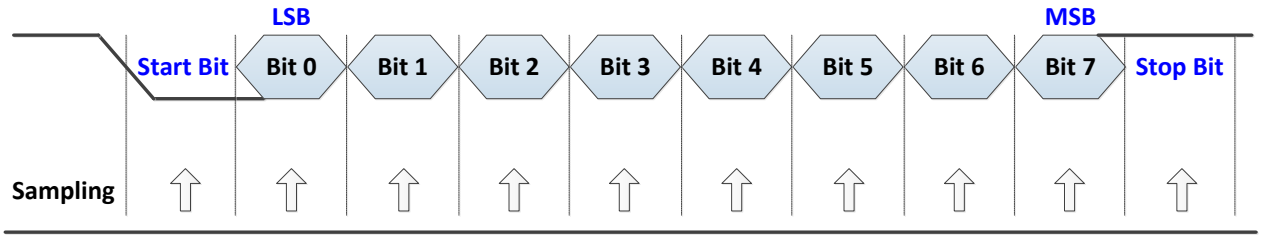


Figure 12-1 UART0 8-bit Timing

12.2 UART0 Mode 2: 9-BIT UART (Fixed Baud Rate)

A total of 11 bits per data byte is used by the 9-bit UART mode, with one Start bit, eight Data bits (LSB first), a programmable 9th data bit, and one Stop bit. The value in TB08 (SCON0.3), which is assigned by user's software, determines the status of the 9th transmit Data bit. For error detection, or for use in multiprocessor communications, it can be assigned the value of the parity flag (bit P in register PSW). Upon receiving, the 9th bit goes into RB08 (SCON0.2) and the Stop bit is ignored.

When an instruction writes a data byte to the SBUF0 register, data transmission starts. The TI0 Transmit Interrupt Flag (SCON0.1) is set at the end of the transmission, which is the beginning of the stop-bit time. Data reception can start any time after the REN0 Receive Enable bit (SCON0.4) is set to Logic 1. After receiving the stop bit, data byte will be loaded into the SBUF0 receiving register if the following conditions are satisfied: 1) RI0 must be Logic 0, and 2) If MCE0 is Logic 1, the 9th bit must be Logic 1 (when MCE0 is Logic 0, the status of the 9th Data bit is not important).

If the conditions are satisfied, the 8 bits of data will be stored in SBUF0, the 9th bit will be stored in RB08 and the R10 flag will be set to "1". However, on the contrary, if the conditions are not satisfied, SBUF0 and RB08 will not be loaded and the RI0 flag will not be set to "1". When either T10 or R10 is set to "1", a UART0 interrupt will occur if enabled

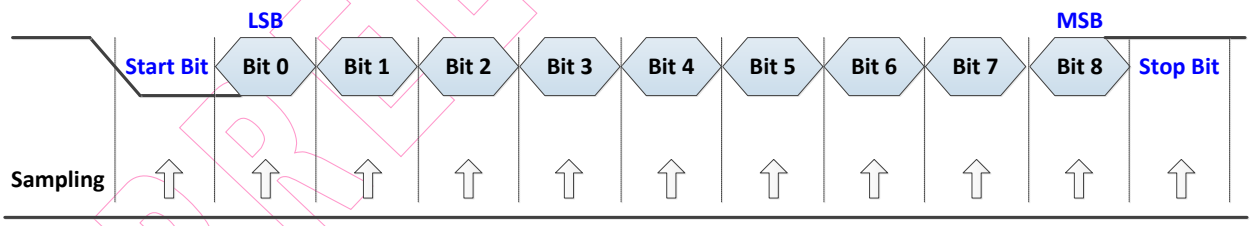


Figure 12-2 UART0 9-bit Timing

12.3 UART0 Multiprocessor Communications

By special use of the 9th data bit, 9-Bit UART mode can support multiprocessor communication between a master processor and one or more slave processors. When a master processor transmits to one or more slaves, it first sends an address byte to select the target(s). The difference between a Data byte and an Address byte is that in a Data byte, the 9th bit is always set to Logic 0, whereas the 9th bit of an Address byte is Logic 1.

Setting the Slave processor's MCE0 bit (SCON0.5) configures its UART in a way that upon receiving a Stop bit, the UART will generate an interrupt, that is only if the 9th bit is Logic 1 (RB80 = 1), which signifies that an Address byte has been received. In handling the UART interrupt, the software will compare the received address with that of the Slave's own assigned 8-bit address. If the addresses match, the slave will clear its MCE0 bit to enable interrupts upon reception of the subsequent Data byte(s). Slaves that were not addressed leave their MCE0 bits set and will not generate interrupts upon reception of the subsequent data bytes, thus ignoring the data. When the entire message is received, the addressed Slave resets its MCE0 bit to ignore all transmissions until it receives the next Address byte.

A single Slave can be assigned with multiple addresses or multiple Slaves can be assigned with a single Address, by that, simultaneously enabling "broadcast" transmissions to more than one Slave. The Master processor can be configured to receive all transmissions or a protocol can be implemented in such a way that the Master/Slave role is temporarily reversed to enable half-duplex transmission between the original Master and Slave(s).

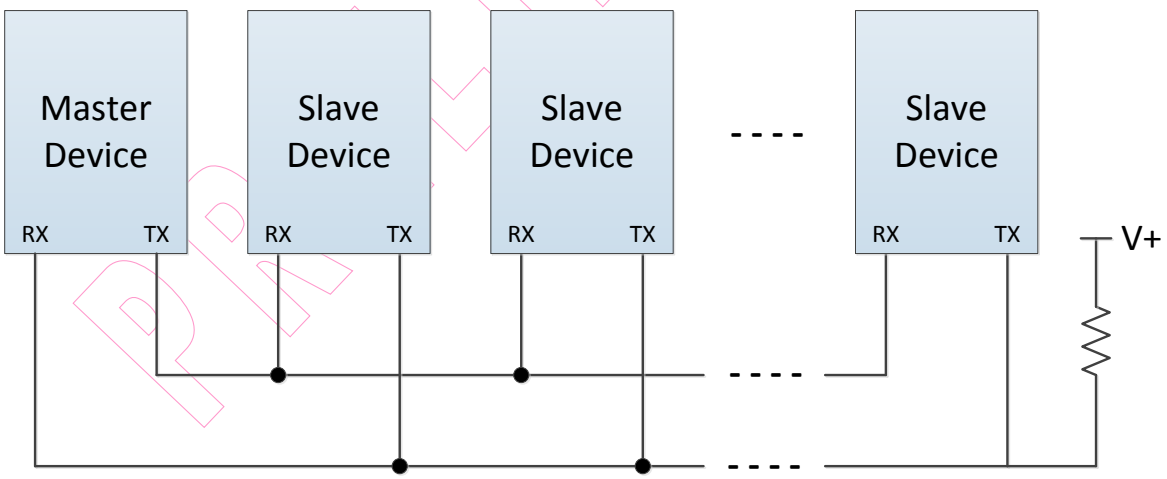


Figure 12-3 UART0 Multiprocessor Communications

12.4 UART0 Baud Rates

UART0 Baud Rates are generated by Timer 1 in 8-bit auto-reload mode. TX clock is generated by TL1 and RX clock is generated by a copy of TL1, which is not user-accessible. To generate the TX and RX baud rates, the Timer overflow of TX and RX are divided into two. The RX Timer runs when Timer 1 is enabled, and the same reload value (TH1) is used. But an RX Timer reload is forced when a START condition is detected on the RX pin. This allows receiving to commence any time a START is detected, regardless of the TX Timer state.

Timer 1 should be configured for Mode 2, 8-bit auto-reload, and its reload value should be set such that overflows will occur at twice the desired UART baud rate frequency. It should be noted that Timer 1 may be clocked by one of the six sources: SYSCLK, SYSCLK/4 or SYSCLK/12. For any given Timer 1 clock source, the UART0 Baud Rate is determined by Equation A, B and C.

UART0 Mode	Baud Rate
Mode 0	SYSCLK/12
Mode 1, 3	Timer 1 overflow rate ($T1_{OV}$) SMOD0 = 0 $T1_{OV}$/32 SMOD0 = 1 $T1_{OV}$/16
Mode 2	SMOD0 = 0 SYSCLK/64 SMOD0 = 1 SYSCLK/32

Table 12- 2 UART0 Baud Rates

Details of each UART0 in Mode Baud Rate are calculated with formulas shown as follows:

Mode 0 Baud Rate:

$$Baud\ Rate = \frac{SYSCLK}{12} \quad \text{(A)}$$

Mode 2 Baud Rate :

$$Baud\ Rate = \frac{2^{SMOD0}}{64} \times SYSCLK \quad \text{(B)}$$

Mode 1/3 Baud Rate:

$$Baud\ Rate = \frac{2^{SMOD0}}{32} \times T1_{OV} = \frac{2^{SMOD0}}{32} \times \frac{Timer1_{CLK}}{[256 - (TH1)]}$$

$$\Rightarrow TH1 = 256 - \frac{2^{SMOD0} \times Timer1_{CLK}}{32 \times Baud\ Rate} \quad \text{(C)}$$

SBUF0: Serial Port 0 Data Buffer

Bit	7	6	5	4	3	2	1	0
Name	SBUF0.7	SBUF0.6	SBUF0.5	SBUF0.4	SBUF0.3	SBUF0.2	SBUF0.1	SBUF0.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x99; SFR Page = All Pages

SCON0: Serial Port 0 Control

Bit	7	6	5	4	3	2	1	0
Name	SM00	SM01	MCE0	REN0	TB08	RB08	TI0	RI0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x98; SFR Page = All Pages

Bits 7~6: Serial Port 0 Operation Mode.

SM00	SM01	Description	Baud Rate
0	0	Synchronous	SYSCLK/12
0	1	8-bit UART	Variable
1	0	9-bit UART	SYSCLK /32 or /64
1	1	9-bit UART	Variable

Bit 5: Multiprocessor Communication Enable.

0: Enable a multiprocessor communication feature.

1: Disable a multiprocessor communication feature.

Bit 4: Receive Enable.

0: UART0 reception disabled.

1: UART0 reception enabled.

Bit 3: Ninth Transmission Bit.

The 9th transmitted data bit in Modes 2 and 3. Set or cleared by the CPU, depending on the function it performs (parity check, multiprocessor communication etc.)

Bit 2: Ninth Receive Bit.

In Modes 2 and 3, it is the 9th data bit received. In Mode 1, if MCE0 is 0, RB08 is the stop bit. In Mode 0, this bit is not used.

Bit 1: Transmit Interrupt Flag.

Transmit interrupt flag, set by hardware after completion of a serial transfer. It must be cleared by software.

Bit 0: Receive Interrupt Flag.

Receive interrupt flag, set by hardware after completion of a serial reception. It must be cleared by software.

PCON: Power Control

Bit	7	6	5	4	3	2	1	0
Name	SMOD0	-	-	-	DPS	-	PD	IDLE
Type	R/W	R	R	R	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x87; SFR Page = All Pages

Bit 7: UART0 double baud rate bit.

0: Disable double Baud rate of the UART0.

1: Enable double Baud rate of the UART0 in mode 1, 2, or 3.

Bits 6~4: Reserved. Read = 0, Write = Don't Care.

Bit 3: Data Pointer Select.

This bit is used to switch between DPTR and DPTR1.

0: Select DPTR.

1: Select DPTR1.

Bits 2: Reserved. Read = 0, Write = Don't Care.

Bit 1: Power-down control bit.

Setting this bit activates Power-down operation. The PD bit is cleared automatically by hardware when waking up from power-down.

Bit 0: Idle mode control bit.

Setting this bit activates Idle mode operation. The IDL bit is cleared automatically by hardware when waking up from Idle (If PD and IDLE is 1, CPU enters into Power-Down Mode).

13 UART2

In UART2, each transmitted or received character is individually synchronized by framing it with a Start bit and Stop bit. Full duplex data transfer is possible since theUART2 has independent transmit and receive sections. Double buffering for both sections allows theUART2 to be programmed for continuous data transfer.

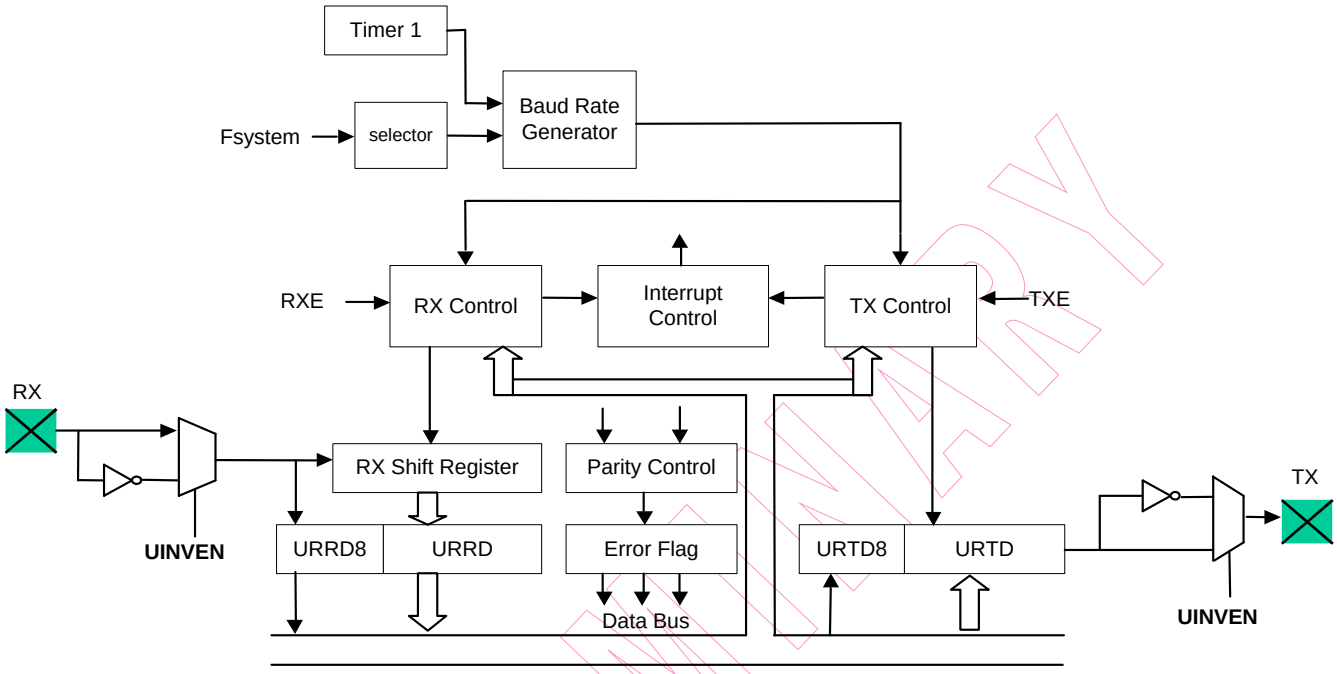


Figure 13-1 UART2 Function Block

The Figure 14.2 shows the general format of one character sent or received. The communication channel is normally held in the marked state (high). Character transmission or reception starts with a transition to the space state (low). The first bit transmitted or received is the start bit (low). It is followed by the data bits, in which the least significant bit (LSB) comes first. The data bits are followed by the parity bit. If present, then the stop bit or bits (high) confirm the end of the frame. In receiving, the UART2 synchronizes on the falling edge of the start bit. When two or three “0” are detected during three samples, it is recognized as normal start bit and the receiving operation is started.

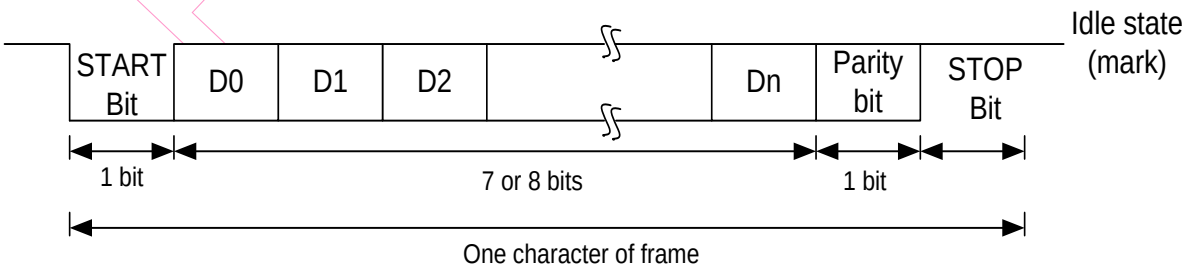


Figure 13- 2 Data Format in UART2

13.1 UART2 Mode

There are three UART2 modes. Mode 1 (7 bits data) and Mode 2 (8 bits data) allow the addition of a parity bit. The parity bit addition is not available in Mode 3. Figure 11 below shows the data format in each mode.

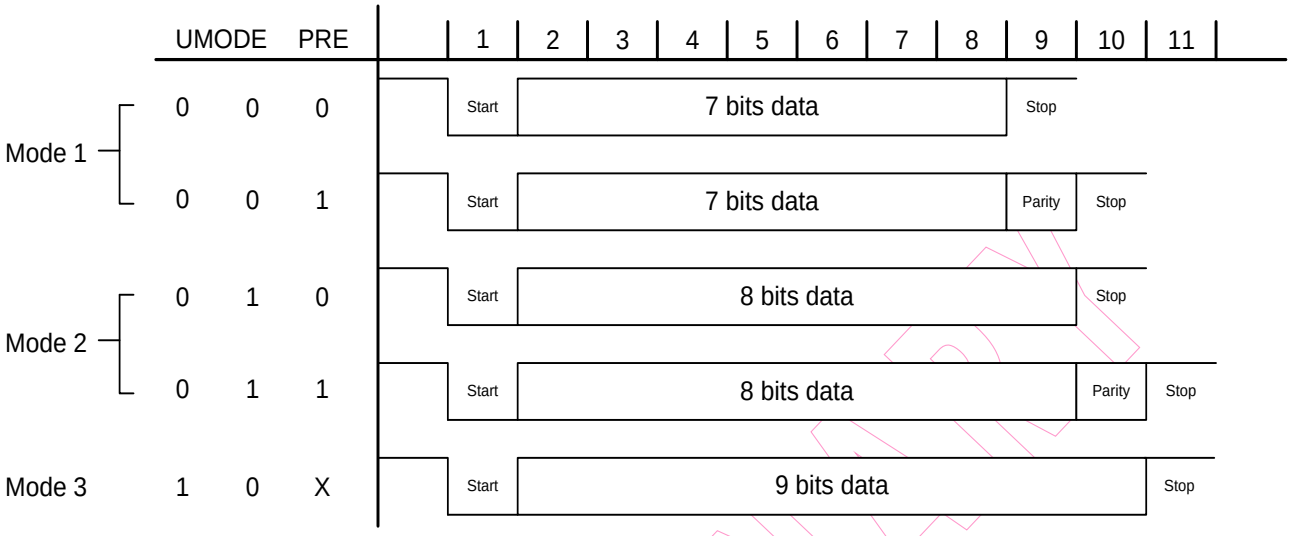


Figure 13- 3 UART2 Mode

13.2 UART2 Transmitting

In transmitting serial data, the UART2 operates as follows:

Step 1: Set the TXE bit of the URCR register to enable the UART2 transmission function.

Step 2: Write data into the URTD register and the UTBF bit of the URCR register will be cleared by hardware.

Step 3: Then start transmitting.

Step 4: Serially transmitted data are transmitted in the following order from the TX pin.

Step 5: Start bit: one “0” bit is output.

Step 6: Transmit data: 7, 8 or 9 bits data are output from the LSB to the MSB.

Step 7: Parity bit: one parity bit (odd or even selectable) is output.

Step 8: Stop bit: one “1” bit (stop bit) is output.

Mark state: output “1” continues until the start bit of the next transmitted data. After transmitting the stop bit, the UART2 generates a UTSF interrupt (if enabled).

13.3 UART2 Receiving

In receiving, the UART2 operates as follows:

Step 1: Set RXE bit of the URS register to enable the UART2 receiving function. The UART2 monitors the RX pin and synchronizes internally when it detects a start bit.

Step 2: Receive data is shifted into the URRD register in the order from LSB to MSB.

Step 3: The parity bit and the stop bit are received. After one character is received, the URBF bit of the URS register will be set to 1. This means UART2 interrupt will occur.

Step 4: The UART2 makes the following checks:

Parity check: The number of 1 of the received data must match the even or odd parity setting of the EVEN bit in the URS register.

Frame check: The start bit must be 0 and the stop bit must be 1.

Overrun check: The URBF bit of the URS register must be cleared (that means the URRD register should be read out) before the next received data is loaded into the URRD register.

If any checks failed, the UERRSF interrupt will be generated (if enabled), and an error flag is indicated in PRERR, OVERR or FMERR bit. The error flag should be cleared by software otherwise, UERRSF interrupt will occur when the next byte is received.

Step 5: Read received data from URRD register. And URBF bit will be set by hardware.

13.4 UART2 Timing

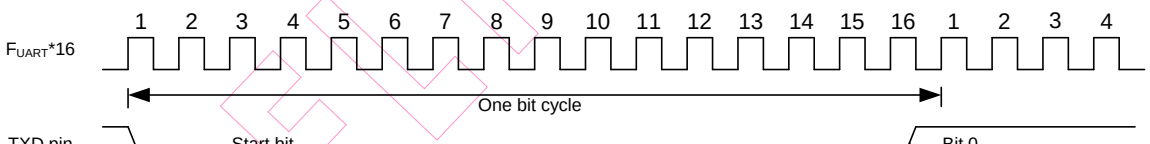


Figure 13- 4 Transmission Counter Timing

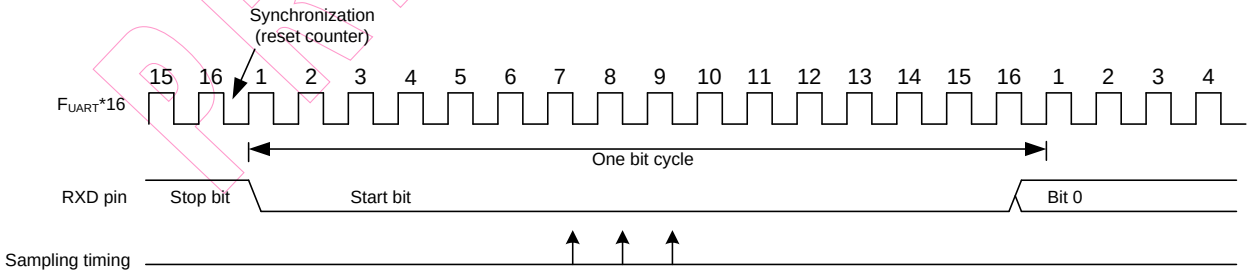
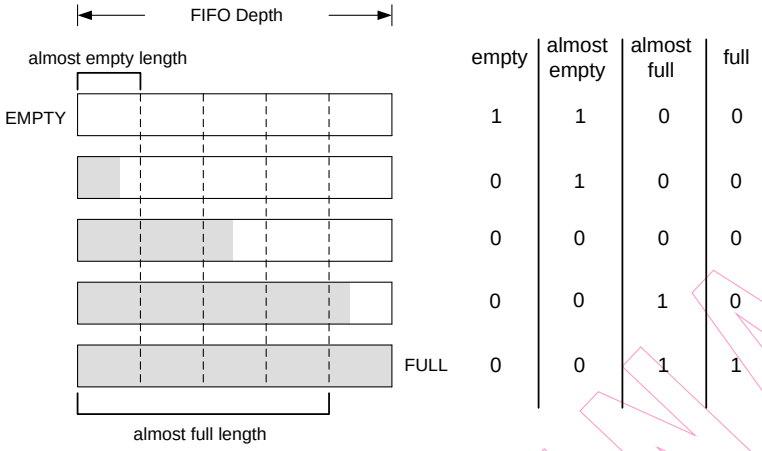


Figure 13- 5 Receiving Counter Timing

13.5 UART2 Baud Rate Generator

The baud rate generator is comprised of a circuit that generates a clock pulse to determine the transfer speed for transmission/reception in the UART2. By setting URBRS[10:0] bits, the desired baud rate can be determined.

13.6 UART2 Buffer Mode



UR2CR: UART2 Control Register

Bit	7	6	5	4	3	2	1	0
Name	U2INV EN	U2MODE1	U2MODE0	UR2BRS.10	UR2BRS.9	UR2BRS.8	U2TBE	U2TXE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	1	0

SFR Address = 0xF1; SFR Page = 2

Bit 7: Enable UART2 TXD and RXD Port Inverse Output Bit.

0: Disable TXD and RXD port inverse output.

1: Enable TXD and RXD port inverse output.

Bits 6~5: UART2 mode select bits.

UMODE1	UMODE0	UART2 mode
0	0	Mode1: 7-bit
0	1	Mode2: 8-bit
1	0	Mode3: 9-bit
1	1	Reserved

Bits 4~2: UART2 baud rate setting

$$\text{UART1 baud rate} = \frac{F_{HS} \text{ or } F_{LS}}{((\text{URBRS}[10:0]+1)\times 16)}$$

Bit 1: UART2 transfer buffer empty flag. Set to 1 when transfer buffer is empty. Reset to 0 automatically when write into URTD register. UTBF bit will be cleared by hardware when enabling transmission. And UTBF bit is read only. Therefore, write URTD register is necessary to start transmitting shifting.

Bit 0: Enable transmission.

0: Disable.

1: Enable.

UR2S: UART2 Status Register

Bit	7	6	5	4	3	2	1	0
Name	UR2TD8	U2EVEN	U2PRE	U2PRERR	U2OVERR	U2FMERR	U2RBF	U2RXE
Type	W	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF2; SFR Page = 2

Bit 7: UART2 transmit data bit 8. Write only.

Bit 6: select parity check.

0: Odd parity.

1: Even parity.

Bit 5: enable parity addition.

0: Disable.

1: Enable.

Bit 4: Parity error flag. Set to 1 when parity error occurs, and clear to 0 by software.

Bit 3: Over running error flag. Set to 1 when overrun error occurs, and clear to 0 by software.

Bit 2: Framing error flag. Set to 1 when framing error occurs, and clear to 0 by software.

Bit 1: UART2 read buffer full flag. Set to 1 when one character is received. Reset to 0 automatically when read from URRDL register. URBF will be cleared by hardware when enabling receiving. And URBF bit is read-only. Therefore, read URS register is necessary to avoid overrun error.

Bit 0: Enable receiving.

0: Disable.

1: Enable.

URTD: UART2 Transmit Data Buffer Register

Bit	7	6	5	4	3	2	1	0
Name	UR2TD.7	UR2TD.6	UR2TD.5	UR2TD.4	UR2TD.3	UR2TD.2	UR2TD.1	UR2TD.0
Type	W	W	W	W	W	W	W	W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF3; SFR Page = 2

URRDL: UART2 Receive Data Low Buffer Register

Bit	7	6	5	4	3	2	1	0
Name	UR2RD.7	UR2RD.6	UR2RD.5	UR2RD.4	UR2RD.3	UR2RD.2	UR2RD.1	UR2RD.0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF4; SFR Page = 2

URRDH: UART2 Receive Data High Buffer Register

Bit	7	6	5	4	3	2	1	0
Name	UR2RD.8	-	-	-	-	-	-	-
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF5; SFR Page = 2

Bit 7: UART2 Receive data bit 8

Bit 6~0: Reserved. Read = 0, Write = Don't Care.

URS2: UART2 Status Register 2

Bit	7	6	5	4	3	2	1	0
Name	UR2DG1	UR2DG0	UR2CSS	BUFMODE	-	U2ERRSF	U2RSF	U2TSF
Type	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
Reset	0	0	1	0	0	0	0	0

SFR Address = 0xF6; SFR Page = 2

Bits 7~6 (URDG1~URDG0): UART2 deglitch time select bits.

URDG1	URDG0	Deglitch Time +/-30%
0	0	50ns
0	1	200ns
1	0	400ns
1	1	Bypass

Bit 5: UART2 clock source selection

0: Use FLS as UART2 clock source

1: Use FHS as UART2 clock source

Bit 4(BUFMODE): UART1 buffer mode enable bit

0: single mode

1: buffer mode(While in buffer mode, OVERR、UTBE and URBF are invalid)

Bit 3: Reserved. Read = 0, Write = Don't Care.

Bit 2: UART2 error status flag, set by hardware when error happens. It must be cleared by software.

Bit 1: UART2 receive status flag, set by hardware after completion of a serial reception. It must be cleared by software.

Bit 0: UART2 transmit status flag, set by hardware after completion of a serial transfer. It must be cleared by software.

URBRS: UART2 Baud Rate Setting Register

Bit	7	6	5	4	3	2	1	0
Name	UR2BRS.7	UR2BRS.6	UR2BRS.5	UR2BRS.4	UR2BRS.3	UR2BRS.2	UR2BRS.1	UR2BRS.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF7; SFR Page = 2

Bits 7~0: UART2 baud rate setting.

$$\text{UART2 baud rate} = \frac{F_{HS} \text{ or } F_{LS}}{((URBRS\,[10:0]+1)\times 16)}$$

UR2BUFTXSF: UART2 Buffer mode Tx Status Flag

Bit	7	6	5	4	3	2	1	0
Name	TXBUF_START	-	TX_FIFO_RST	TX_FULL	TX_AFULL	TX_AEMPTY	TX_EMPTY	TX_ERR
Type	R/W	R	R/W	R	R	R	R	R
Reset	0	0	0	0	0	1	1	0

SFR Address = 0XE9; SFR Page = 2

Bit 7 (TXBUF_START): Tx buffer mode transmission start bit. When Tx FIFO is empty, it would be cleared by hardware to stop transfer.

Bit 6: Not used bits. Fixed to “0” all the time.

Bit 5 (TX_FIFO_RST): Tx FIFO reset bit. When Tx FIFO error occurs, it must be set to ‘’1” and the Tx FIFO would recover to initial state.

Bit 4 (TX_FULL): Tx FIFO full bit. Set to “1” when Tx FIFO is full.

Bit 3 (TX_AFULL): Tx FIFO almost full bit. Set to “1” when Tx FIFO is almost full. The almost full length is decided by URTXLEN_AF[2:0].

Bit 2 (TX_AEMPTY): Tx FIFO almost empty bit. Set to “1” when Tx FIFO is almost empty. The almost empty length is decided by URTXLEN_AE[2:0].

Bit 1 (TX_EMPTY): Tx FIFO empty bit. Set to “1” when Tx FIFO is empty.

Bit 0 (TX_ERR): Tx FIFO error bit. Set to “1” when Tx FIFO is underflow/overflow.

UR2BUFRXSF: UART2 Buffer mode Rx Status Flag

Bit	7	6	5	4	3	2	1	0
Name	-	-	RX_FIFO_RST	RX_FULL	RX_AFULL	RX_AEMPTY	RX_EMPTY	RX_ERR
Type	R	R	R/W	R	R	R	R	R
Reset	0	0	0	0	0	1	1	0

SFR Address = 0xEA; SFR Page = 2

Bits 7~6: Not used bits. Fixed to “0” all the time.

Bit 5 (RX_FIFO_RST): Rx FIFO reset bit. When Rx FIFO error occurs, it must be set to ‘’1” and the Rx FIFO would recover to initial state.

Bit 4 (RX_FULL): Rx FIFO full bit. Set to “1” when Rx FIFO is full.

Bit 3 (RX_AFULL): Rx FIFO almost full bit. Set to “1” when Rx FIFO is almost full. The almost full length is decided by URRXLEN_AF[2:0].

Bit 2 (RX_AEMPTY): Rx FIFO almost empty bit. Set to “1” when Rx FIFO is almost empty. The almost empty length is decided by URRXLEN_AE[2:0].

Bit 1 (RX_EMPTY): Rx FIFO empty bit. Set to “1” when Rx FIFO is empty.

Bit 0 (RX_ERR): Rx FIFO error bit. Set to “1” when Rx FIFO is underflow/overflow.

UR2BUFTXIM : UART2 Buffer mode Tx Interrupt Mask Flag

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	TX_FULLIE	TX_AFULLIE	TX_AEMPTYIE	TX_EMPTYIE	TX_ERRIE
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xEB; SFR Page = 2

Bits 7~5: Not used bits. Fixed to “0” all the time.

Bit 4 (TX_FULLIE): Interrupt mask of Tx FIFO full bit.

0: Disable.

1: Enable.

Bit 3 (TX_AFULLIE): Interrupt mask of Tx FIFO almost full bit.

0: Disable.

1: Enable.

Bit 2 (TX_AEMPTYIE): Interrupt mask of Tx FIFO almost empty bit.

0: Disable.

1: Enable.

Bit 1 (TX_EMPTYIE): Interrupt mask of Tx FIFO empty bit.

0: Disable.

1: Enable.

Bit 0 (TX_ERRIE): Interrupt mask of Tx FIFO error bit.

0: Disable.

1: Enable.

UR2BUFRXIM: UART2 Buffer mode Rx Interrupt Mask Flag

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	RX_FULLIE	RX_AFULLIE	RX_AEMPTYIE	RX_EMPTYIE	RX_ERRIE
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xEC; SFR Page = 2

Bits 7~5: Not used bits. Fixed to “0” all the time.

Bit 4 (RX_FULLIE): Interrupt mask of Rx FIFO full bit.

0: Disable.

1: Enable.

Bit 3 (RX_AFULLIE): Interrupt mask of Rx FIFO almost full bit.

0: Disable.

1: Enable.

Bit 2 (RX_AEMPTYIE): Interrupt mask of Rx FIFO almost empty bit.

0: Disable.

1: Enable.

Bit 1 (RX_EMPTYIE): Interrupt mask of Rx FIFO empty bit.

0: Disable.

1: Enable.

Bit 0 (RX_ERRIE): Interrupt mask of Rx FIFO error bit.

0: Disable.

1: Enable.

UR2TXLEN: UART2 Tx length Register

Bit	7	6	5	4	3	2	1	0
Name	URTXLEN_AF[3]	URTXLEN_AF[2]	URTXLEN_AF[1]	URTXLEN_AF[0]	URTXLEN_AE[3]	URTXLEN_AE[2]	URTXLEN_AE[1]	URTXLEN_AE[0]
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	0	0	0	1

SFR Address = 0xED; SFR Page = 2

Bits 7~4 (URTXLEN_AF[2:0]): The Tx FIFO almost full length set.

Bits 3~0 (URTXLEN_AE[2:0]): The Tx FIFO almost empty length set.

UR2RXLEN: UART2 Rx length Register

Bit	7	6	5	4	3	2	1	0
Name	URRXLEN_AF[3]	URRXLEN_AF[2]	URRXLEN_AF[1]	URRXLEN_AF[0]	URRXLEN_AE[3]	URRXLEN_AE[2]	URRXLEN_AE[1]	URRXLEN_AE[0]
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	0	0	0	1

SFR Address = 0xEE; SFR Page =2

Bits 7~4 (URRXLEN_AF[3:0]): The Rx FIFO almost full length set.

Bits 3~0 (URRXLEN_AE[3:0]): The Rx FIFO almost empty length set.

14 Timers

The MCU has five Counters/Timers, two of which are 16-bit Counters/Timers compatible with those in the Standard 8051. Such Timers can be used to measure time intervals, count external events and generate periodic Interrupt requests. Timer 0 and Timer 1 are almost identical and they have four main modes of operation.

Mode	Function Description
0	13-bit counter/timer
1	16-bit counter/timer
2	8-bit counter/timer with auto-reload
3	Two 8-bit counter/timers (Timer 0 only)

Table 14- 1 Timer 0/1 Modes

Timer 0 and Timer 1 can be clocked by any of the five sources, as determined by the Clock Scale bits (TxSC and TxM) and the Timer Mode Select bits (TxM1–TxM0). The Clock Scale bits define a pre-scaled clock by which Timer 0 and/or Timer 1 can be clocked. Timer 0 and Timer 1 can then be configured to use the pre-scaled clock signal or the system clock.

Timer 0 and Timer 1 can also be used as Counters. When operating as a Counter, a Counter/Timer Register is incremented by each high-to-low transition at the chosen Input pin (T0 or T1). Events with a Frequency of up to a quarter of the System Clock can be counted. The Input signal has to be periodic, but it should be at a certain level for at least two full system clock cycles, in order to ensure that the level is properly sampled.

14.1 Timer 0 and Timer 1

Every Timer is used as a 16-bit register accessed as two separate bytes namely, High byte (TH0 or TH1) and Low byte (TL0 or TL1). The Counter/Timer Control register (TCON) is used to indicate the status and to enable Timer 0 and Timer 1. Timer 0 interrupts can be enabled by setting the ET0 bit in the IE register. Timer 1 interrupts can be enabled by setting the ET1 bit in the IE register. Both Counter/Timers in one among the four main modes selected by setting the Mode Select Bits T1M1–T0M0 in the Counter/Timer Mode register (TMOD). Each Timer can be separately configured. The various operating modes are described in the following subsections.

14.1.1 Timer 0/1 Mode 0: 13-bit Counter/Timer

Timer 0 and Timer 1 function as 13-bit Counter/Timers in Mode 0. The THx register containsthe eight MSB of the 13-bit Counter/Timer. The TLx register contains thefive LSB in bit positions (TLx.4–TLx.0). TLx has three upper bits (TLx.7–TLx.5) which cannot be determined and should be masked out or ignored when reading. When the 13-bit register is incremented and an overflow occurred from 0x1FFF (all ones) to 0x0000, the Timer overflow flag TFx in TCON is set, and if Timer 0 or Timer 1 Interrupts are enabled, an Interrupt will occur.

If C/Tx is set to High, high-to-low transitions at the Timer input pin (TCx) will increment the Timer/Counter Data Register. When C/Tx is cleared to low, the system clock is selected to increment the Timer/Counter Data Register.

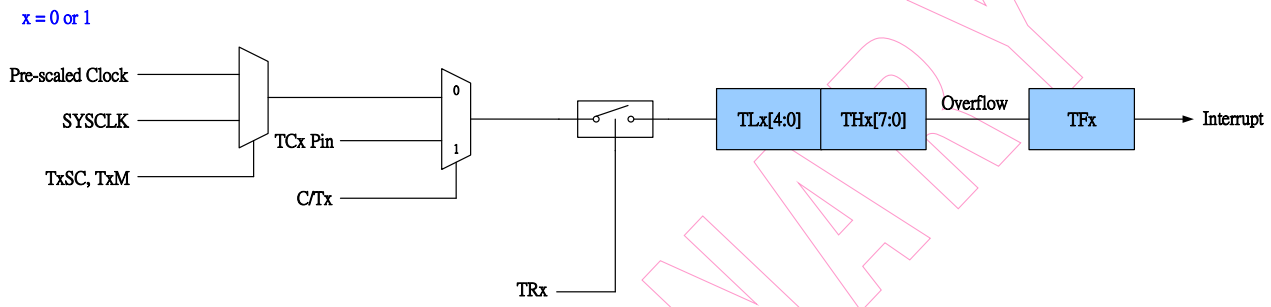


Figure 14- 1 Timer 0/1 Mode 0 Block Diagram

14.1.2 Timer 0/1 Mode 1: 16-bit Counter/Timer

Mode 1 and Mode 0 operate in the same manner, except that the Counter/Timer registers use all 16 bits. The Counter/Timers are enabled and configured in both Mode 1 and Mode 0.

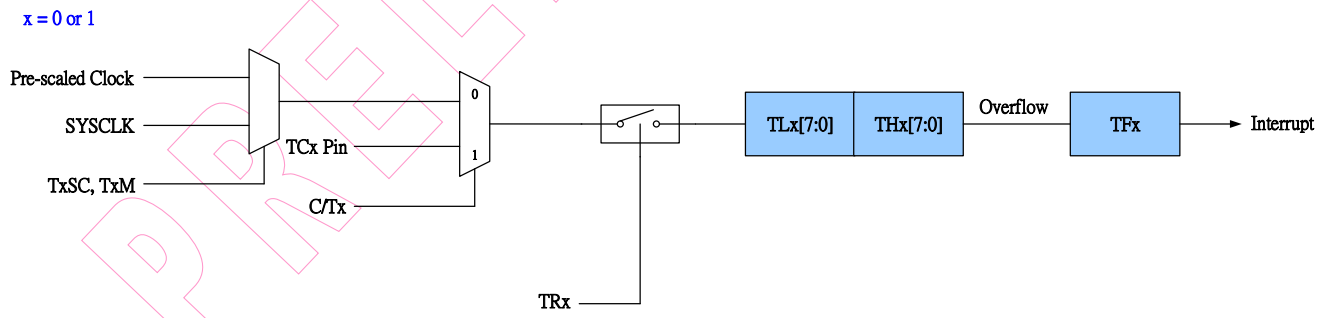


Figure 14- 2Timer 0/1 Mode 1 Block Diagram

14.1.3 Timer 0/1 Mode 2: 8-bit Counter/Timer with Auto-Reload

Timer 0 and Timer 1 are configured in Mode 2 to operate as 8-bit Counter/Timers with auto reload value starting at 21. TLx keeps the Count and THx contain the reload value. When an overflow occurs from all ones to 0x00 in the counter in TLx, the Timer Overflow flag TFx in the TCON register is set and the Counter in TLx is reloaded from THx. If Timer 0 Interrupts are enabled, an interrupt will occur when the TFx flag is set. The reload value in THx will not be changed. Both Counter/Timers are enabled and configured in Mode 2 in the same way as in Mode 0.

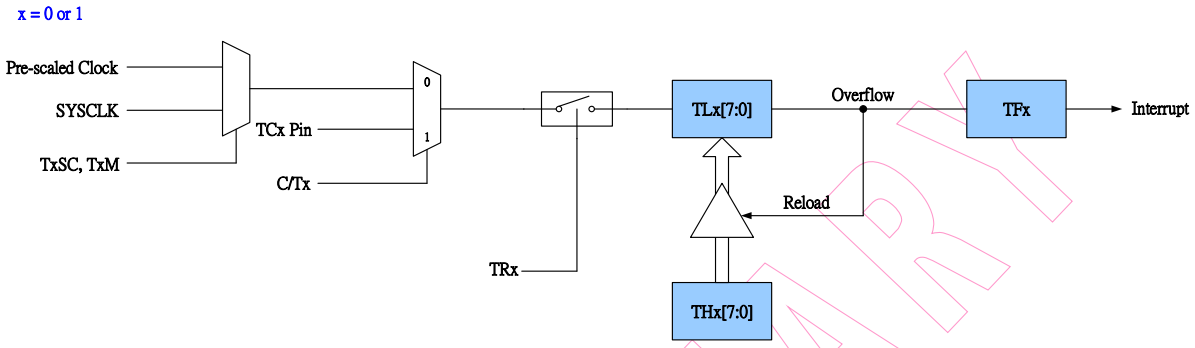


Figure 14- 3Timer 0/1 Mode 2 Block Diagram

14.1.4 Timer 0 Mode 3 Two 8-bit Counter/Timers

Timer 0 is configured in Mode 3 as two separate 8-bit Counter/Timers kept in TL0 and TH0. TL0 can be configured as Timer or Counter, and it is controlled using the Timer 0 Control/Status bits in TCON and TMOD: TR0, C/T0 and TF0. TH0 can only be configured as a Timer, and it is enabled using the Timer 1 Run Control Bit TR1. TH0 sets the Timer 1 Overflow Flag to overflow, and so it controls the Timer 1 Interrupt.

When Timer 0 is operating in Mode 3, Timer 1 is not activated and it can only be operated in Mode 0, Mode 1 or Mode 2, but it cannot generate an interrupt and set the TF1 Flag. Timer 1 overflow can be used to generate Baud Rate of serial ports for UART0. TH1 and TL1 can only be used as Timer. Timer 1 Run Control is handled through its mode settings, since TR1 is used by Timer 0.

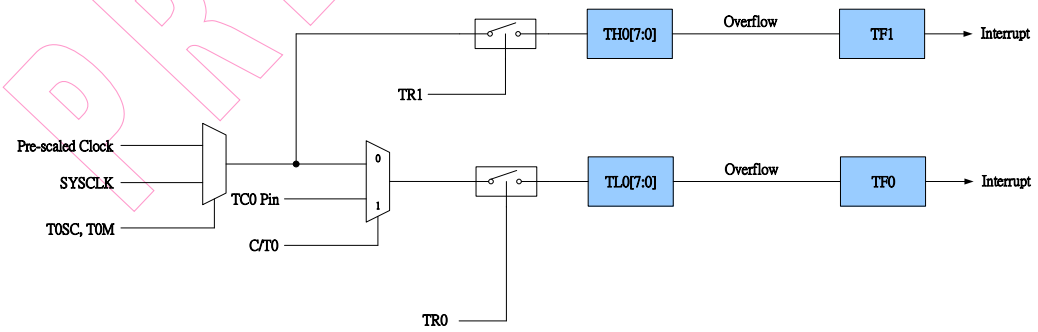


Figure 14- 4 Timer 0 Mode 3 Block Diagram

CKCON0: Clock Control 0

Bit	7	6	5	4	3	2	1	0
Name	-	T1SC	T0SC	T1M	T0M	LSstable	HSstable	HSPD
Type	R/W	R/W	R/W	R/W	R/W	R	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8E; SFR Page = All Pages

Bits 7: Not used bits. Fixed to “0” all the time.

Bit 6, 4: Timer 1 Clock Select.

Select the clock source supplied to Timer 1. Ignored when C/T1 is set to 1.

T1SC	T1M	Description
0	0	System clock divided by 12
0	1	System clock divided by 4
1	X	System clock

Bit 5, 3: Timer 0 Clock Select.

Select the clock source supplied to Timer 0. Ignored when C/T0 is set to 1.

T0SC	T0M	Description
0	0	System clock divided by 12
0	1	System clock divided by 4
1	X	System clock

Bit 2: Reserved. Read = 0, Write = Don't Care.

TCON: Timer Control

Bit	7	6	5	4	3	2	1	0
Name	TF1	TR1	TF0	TR0	-	-	-	-
Type	R/W	R/W	R/W	R/W	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x88; SFR Page = All Pages

Bit 7: Timer 1 Overflow Flag.

Set to 1 by hardware when Timer 1 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 1 interrupt service routine.

Bit 6: Timer 1 Run Control.

Timer 1 is enabled by setting this bit to 1.

Bit 5: Timer 0 Overflow Flag.

Set to 1 by hardware when Timer 0 overflows. This flag can be cleared by software but is automatically cleared when the CPU vectors to the Timer 0 interrupt service routine.

Bit 4: Timer 0 Run Control.

Timer 0 is enabled by setting this bit to 1.

Bit 3~0: Reserved. Read = 0, Write = Don't Care.

TMOD: Timer Mode

Bit	7	6	5	4	3	2	1	0
Name	-	C/T1	T1M1	T1M0	-	C/T0	T0M1	T0M0
Type	R	R/W	R/W	R/W	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x89; SFR Page = All Pages

Bit 7: Reserved. Read = 0, Write = Don't Care.

Bit 6: Counter/Timer 1 Select.

0: Timer: Timer 1 is incremented by clock defined by T1M[1:0] in register CKCON0.

1: Counter: Timer 1 is incremented by high-to-low transitions on external pin (TC1).

Bits 5~4: Timer 1 Mode Select.

T1M1	T1M0	Mode
0	0	13-bit Counter/Timer
0	1	16-bit Counter/Timer
1	0	8-bit Counter/Timer with Auto-Reload
1	1	Timer 1 Inactive

Bit 3: Reserved. Read = 0, Write = Don't Care.

Bit 2: Counter/Timer 0 Select.

0: Timer: Timer 0 is incremented by clock defined by T0M[1:0] bit in register CKCON0.

1: Counter: Timer 0 is incremented by high-to-low transitions on external pin (TC0).

Bits 1~0: Timer 0 Mode Select.

T0M1	T0M0	Mode
0	0	13-bit Counter/Timer
0	1	16-bit Counter/Timer
1	0	8-bit Counter/Timer with Auto-Reload
1	1	Two 8-bit Counter/Timers

TL0: Timer 0 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TL0.7	TL0.6	TL0.5	TL0.4	TL0.3	TL0.2	TL0.1	TL0.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8A; SFR Page = All Pages

TL1: Timer 1 Low Byte

Bit	7	6	5	4	3	2	1	0
Name	TL1.7	TL1.6	TL1.5	TL1.4	TL1.3	TL1.2	TL1.1	TL1.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8B; SFR Page = All Pages

TH0: Timer 0 High Byte

Bit	7	6	5	4	3	2	1	0
Name	TH0.7	TH0.6	TH0.5	TH0.4	TH0.3	TH0.2	TH0.1	TH0.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8C; SFR Page = All Pages

TH1: Timer 1 High Byte

Bit	7	6	5	4	3	2	1	0
Name	TH1.7	TH1.6	TH1.5	TH1.4	TH1.3	TH1.2	TH1.1	TH1.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x8D; SFR Page = All Pages

14.2 Timer 2: Real Time Clock (RTC)

The RTC module is intended for applications in which accurate time must be maintained for extended periods of time with minimal or no CPU intervention.

If system hold event occurs, do not hold RTC Clock.

RTCCON: Timer Control

Bit	7	6	5	4	3	2	1	0
Name	RTCEN	RTCWREN	RTCOEN	RTCCALEN	-	RTCCLK	RTCPTR1	RTCPTR0
Type	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
Reset	0*	0	0	0	0	0	0	0

SFR Address = 0x91; SFR Page = 1

*:only for power-on-reset. This bit would not be changed by another reset.

Bit 7(RTCEN): RTC Run enable.

0: RTC disable

1: let RTC counter start running from zero.

Bit 6(RTCWREN): RTC data register write enable.

0: RTCDATAH and RTCDATAL registers are locked out from being written by the user

1: RTCDATAH and RTCDATAL registers can be written by the user when RTCKEY=0x35

Bit 5(RECOEN): RTCOUT pin enable.

0: RTCOUT pin disable.(RTCO pinchange phase every second

1: RTCOUT pin enable

Bit 4(RTCCALLEN): RTC calibration enable.

0: RTC calibration function disable.

1: RTC calibration function enable. The RTC counter would be updated by adding/subtracting counter numbers according to RTCCAL register every minute or second.

Bit 3: Reserved. Read = 0, Write = Don't Care.

Bit 2(RTCCLK): RTC clock source select.

0: Low Speed Frequency (FLS).

1: RTCIN

Bit 1~0(RTCPTR1~0): RTC data register pointer bits

RTCPTR1	RTCPTR0	RTCDATAH	RTCDATAL
0	0	Minutes	Seconds
0	1	Weekday	Hours
1	0	Month	Day
1	1	Reserved	Year

RTCDATAH: RTC High Byte Data

Bit	7	6	5	4	3	2	1	0
Name	Minutes							
	Weekday							
	Month							
	Reserved							
Type	R/W							
Reset	0							

SFR Address = 0x92; SFR Page = 1

RTCDATAL: RTC Low Byte Data

Bit	7	6	5	4	3	2	1	0
Name	Seconds							
	Hours							
	Day							
	Year							
Type	R/W							
Reset	0							

SFR Address = 0x93; SFR Page = 1

The format of time is:

RTCDATAH/ RTCDATAL	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	RANGE
Seconds	0	10 seconds(default: 000)(0~5)			seconds(default: 0000)(0~9)				0-59
Minutes	0	10 minutes(default: 000)(0~5)			minutes(default: 0000)(0~9)				0-59
Hours	0	0	10 hours(default: 00)(0~2)		hours(default: 0000)(0~9)				0-23
Weekday	0	0	0	0	0	WDAY(default: 001)(1~7)			1-7
Day	0	0	10 date(default:00)(0~3)		date(default: 0001)(0~9)				1-31
Month	0	0	0	10 months (default:0) (0~1)	month(default: 0001)(0~9)				1-12
Year	10 years(default: 0000)(0~9)				year(default: 0000)(0~9)				0-99**
Reserved	U	U	U	U	U	U	U	U	

*: Unimplemented bit, read as '0'

**:value "0" means year-2000.

If the filled value exceeded the range, it will be cleared to the initial value.

ALARM: Alarm Control Register

Bit	7	6	5	4	3	2	1	0
Name	ALARMEN	ALARMF	-	-	-	ALARMCF2	ALARMCF1	ALARMCF0
Type	R/W	R/W	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x94; SFR Page =1

Bit 7(ALARMEN): ALARM enable.

0: disable

1: enable.

Bit 6(ALARMF): ALARM flag.

0: no alarm event according to ALARMCF[2:0] happens,

1: alarm event according to ALARMCF[2:0] happens.

Clear by Software.

Bits 5~3: Reserved. Read = 0, Write = Don't Care.

Bit 2~0(ALARMCF2~0): alarm event configuration bits. If the corresponding bit of interrupt is enabled, CPU will jump to corresponding interrupt vector as alarm event triggers.

ALARMCF2	ALARMCF1	ALARMCF0	event
0	0	0	1s
0	0	1	10s(0s,10s,20s,30s, 40s, 50s)
0	1	0	1min
0	1	1	10mins(0min, 10min,20min,30min, 40min, 50min)
1	0	0	1h
1	0	1	1day
1	1	0	1week(Saturday to Sunday)
1	1	1	1month

RTCCAL1: RTC Calibration Register 1

Bit	7	6	5	4	3	2	1	0
Name	RTCCAL.7	RTCCAL.6	RTCCAL.5	RTCCAL.4	RTCCAL.3	RTCCAL.2	RTCCAL.1	RTCCAL.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x95; SFR Page = 1

Bit 7~0(RTCCAL7~0): RTC Drift Calibration bits

RTCCAL2: RTC Calibration Register 2

Bit	7	6	5	4	3	2	1	0
Name	FebSel1	FebSel0	CalSel	RTCCAL.12	RTCCAL.11	RTCCAL.10	RTCCAL.9	RTCCAL.8
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x96; SFR Page = 1

Bit 7~6(FebSel1~0): February Select.

FebSel1	FebSel0	Days
0	0	28
0	1	29
1	0	-
1	1	-

Bit 5(CalSel): Calibration Select

1: Adjustment RTC clock pulses every second.

0: Adjustment RTC clock pulses every minute.

Bit 4~0(RTCCAL12~8): RTC Drift Calibration bits

RTCCAL12~0: RTC Drift Calibration bits

0111111111111= Maximum positive adjustment; add 4095 RTC clock pulses every minute or second.

0000000000001= Minimum positive adjustment; add 1 RTC clock pulses every minute or second.

0000000000000= No adjustment

1000000000000= No adjustment

1000000000001= Minimum negative adjustment; subtract 1 RTC clock pulses every minute or second.

1111111111111= Maximum negative adjustment; subtract 4095 RTC clock pulses every minute or second.

RTCKEY: RTC Key

Bit	7	6	5	4	3	2	1	0
Name	RTCKEY.7	RTCKEY.6	RTCKEY.5	RTCKEY.4	RTCKEY.3	RTCKEY.2	RTCKEY.1	RTCKEY.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x97; SFR Page = 1

Bit 7~0(RTCKEY): RTCKEY=0x35 and RTCWREN=1, the time (seconds/minutes/hours/weekday/day/month/year) can be set, RTCKEY will be cleared Hardware every time a RTCDATAL or RTCDATAH is written. Hence, RTCKEY needs to be filed in prior RTCDATAL or RTCDATAH writing.

14.3 Timers 3-4

Timer 3, 4 can be used as one 16-bit up-counter.

14.3.1 Timers 3-4: Timer/Counter Mode

In Timer/Counter mode, counting up is performed using internal clock or TCx pin. When the contents of up-counter are matched the TCxDA, then interrupt is generated and counter is cleared. Counting up resumes after the counter is cleared. The current contents of up-counter are loaded into TCxDB by setting TCxRC to “1”.

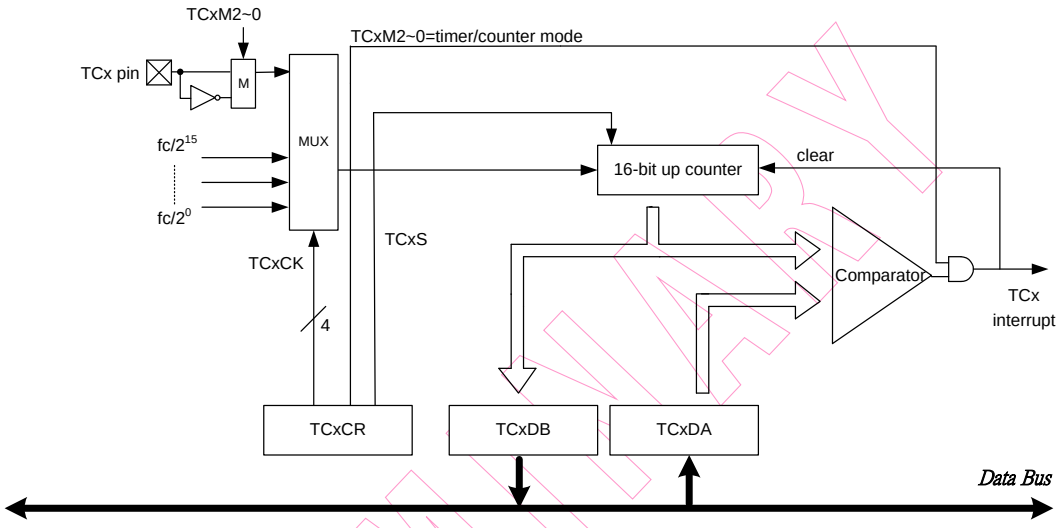


Figure 14-5 Timer/Counter Mode

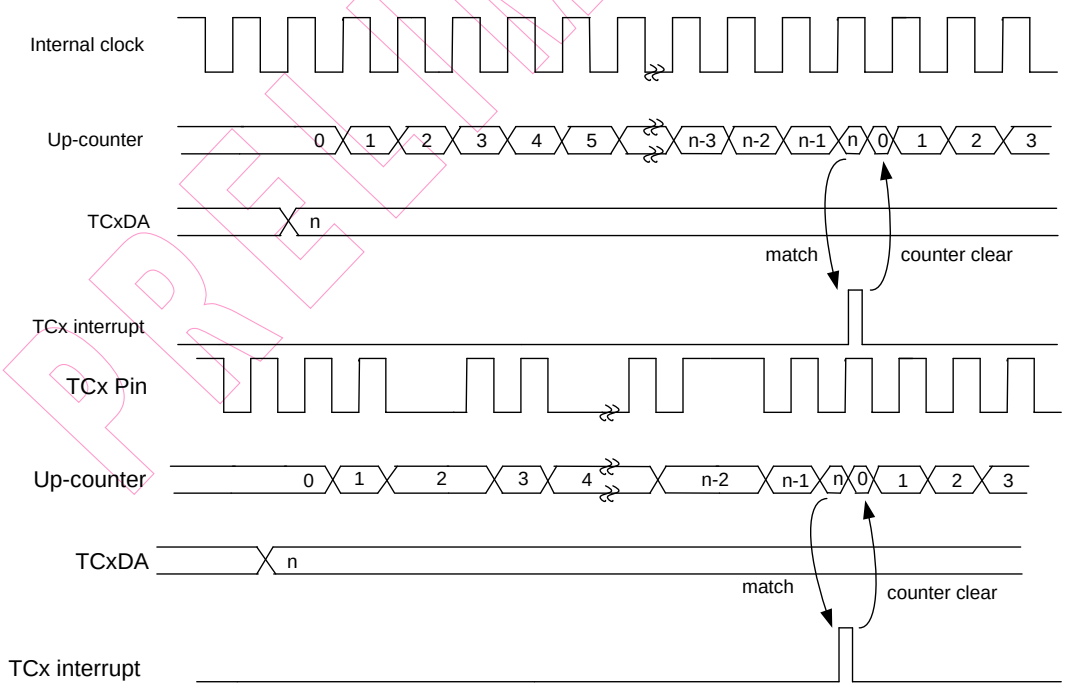


Figure 14- 6 Timer/Counter Mode Waveform

14.3.2 Timers 3-4: Window Mode

In Window mode, countering up is performed on rising edge of the pulse that is logical AND of an internal clock and the TCx pin (window pulse). When the contents of up-counter are matched the TCxDA, then interrupt is generated and counter is cleared. The frequency (window pulse) must be slower than the selected internal clock.

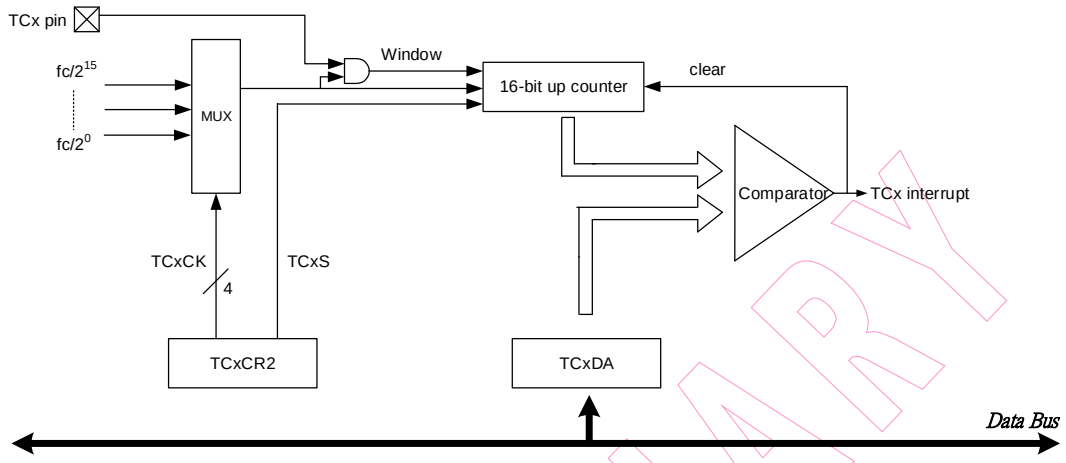


Figure 14-7 Window Mode

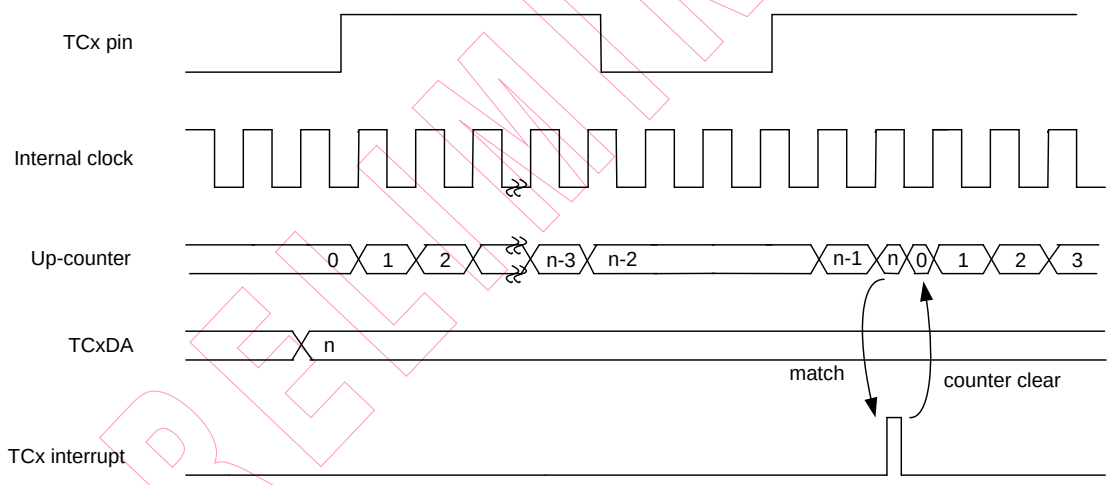


Figure 14-8 Window Mode Waveform

14.3.3 Timers 3-4: Capture Mode

In Capture mode, the pulse width, period and duty of the TCx input pin are measured, and can be used to decode the remote control signal. The counter is running free by the internal clock. On the rising (falling) edge of TCx pin, the contents of counter are loaded into TCxDA, then the counter is cleared and interrupt is generated. On the falling (rising) edge of TCx pin, the contents of counter are loaded into TCxDB. At this time, the counter is still countering. Once the next rising edge of TCx pin is triggered, the contents of counter are loaded into TCxDA, the counter is cleared and interrupt is generated again. If an overflow before the edge is detected, the FFH is loaded into TCxDA and the overflow interrupt is generated. During interrupt processing, it can be determined whether there is an overflow or not by checking if the TCxDA value is FFH.

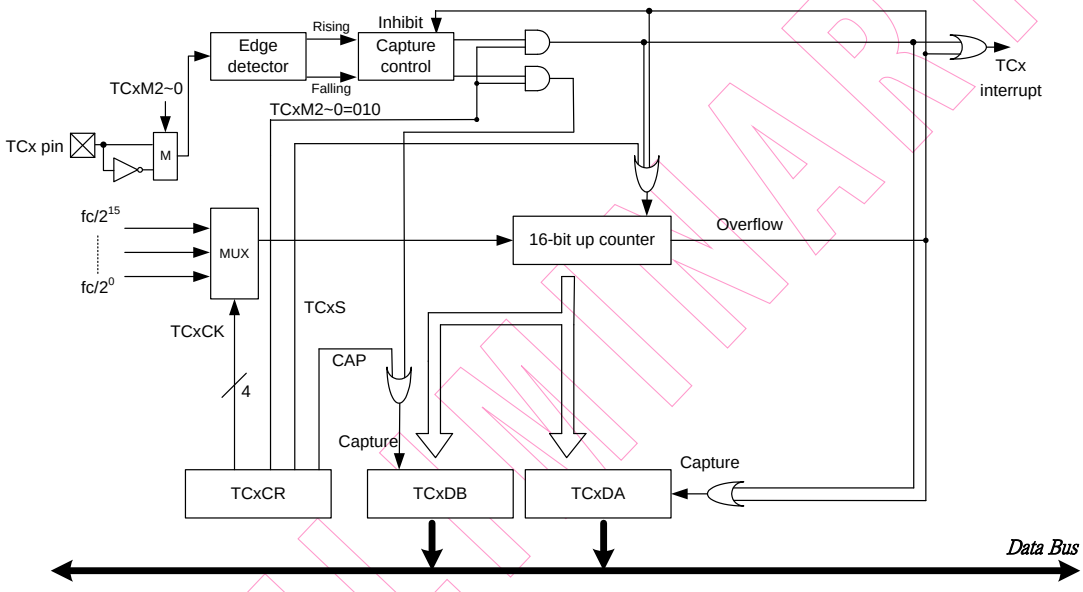


Figure 14-9 Capture Mode

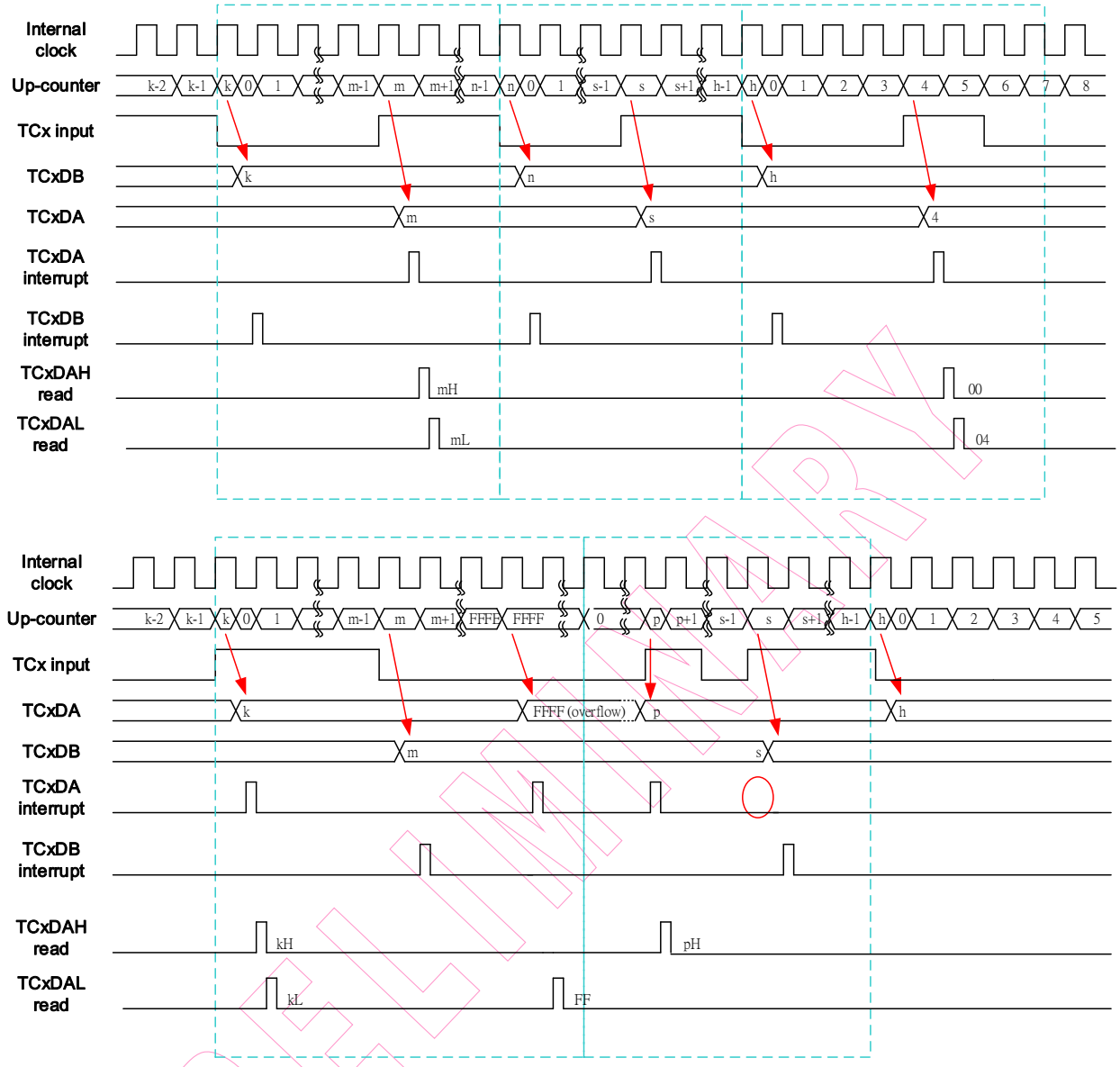


Figure 14-10 Capture Mode Waveform (1)(2)

14.3.4 Timers 3-4: Programmable Divider Output Mode and Pulse Width Modulation Mode

In Programmable Divider Output (PDO) mode, counting up is performed using internal clock. The contents of TCxDA are compared with the contents of up-counter. The F/F output is toggled and the counter is cleared each time a match is found. The F/F output is inverted and output to PDO pin. This mode can generate 50% duty pulse output. The PDO pin is initialized to “0” during reset. A TCx interrupt is generated each time the PDO output is toggled.

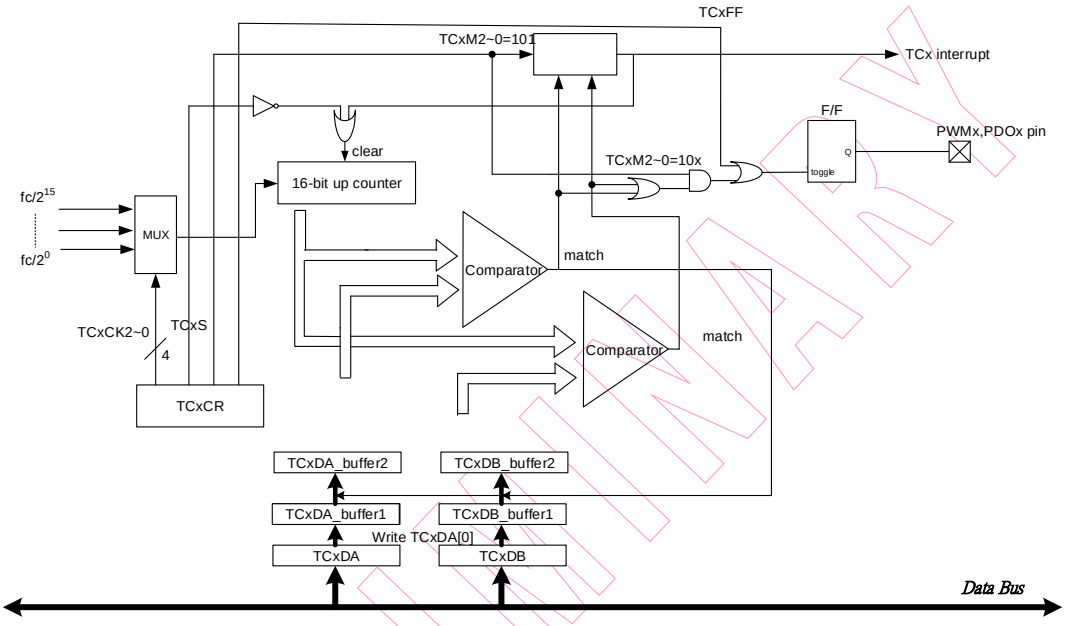


Figure 14- 11PWM/PDO mode

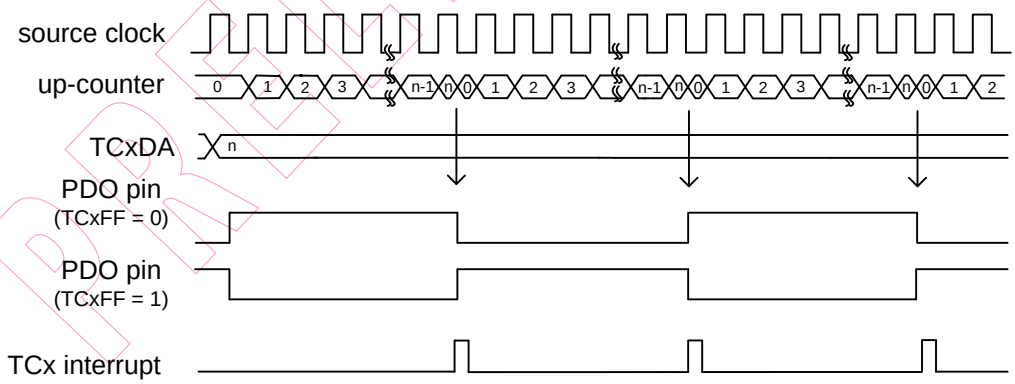


Figure 14- 12PDO Mode Waveform

In Pulse Width Modulation (PWM) Output mode, counting up is performed using the internal clock with prescaler. The Duty of PWMx control by TCxDB, and the period of PWMx control by TCxDA. The pulse at the PWMx pin is held to high level as long as TCxS=1 or timerx matches TCxDA, while the pulse is held to low level as long as timerx matches TCxDB. Once TCxFF is set to 1, the signal of PWMx is inverted. A TCx interrupt is generated and defined by TCxIS. On the other hand, the TCxDA and TCxDB can be written anytime, but the data of TCxDA and TCxDB are latched only at writing TCxDA[0]. Therefore, the new duty and new period of PWM appear at the PMW pin at the last period–match.

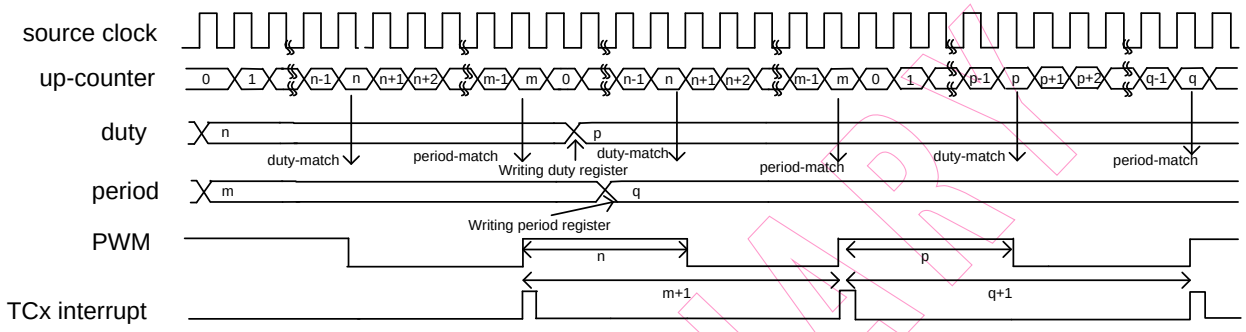


Figure 14- 13 PWM Mode Waveform

14.3.5 Timers 3-4: Buzzer Mode

TCx pin output the clock after dividing frequency.

14.3.6 Description of Timers 3-4 Control Registers

TC3CR1: Timer/Counter 3 Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	TC3S	TC3RC	TC3DBSF	TC3DASF	TC3FF	TC3OMS	TC3IS1	TC3IS0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD1; SFR Page = 0

Bit 7: Timer/Counter 3 start control (main switch for all modes).

0: Stop and clear counter.

1: Start.

Bit 6: Timer 3 Read Control Bit.

0: When this bit is set to 0, can't read data from TC3DB.

1: When this bit is set to 1, data read from TC3DB is a number of counting.

Bit 5: 16-bit Timer/Counter 3 Data B (Duty) status flag, cleared by software.

Bit 4: 16-bit Timer/Counter 3 Data A (Period) status flag, cleared by software.

Bit 3: inversion for Timer/Counter 3 as PWM or PDO mode.

0: Duty is Logic 1.

1: Duty is Logic 0.

Bit 2: Timer Output Mode Select Bit.

0: Repeating mode.

1: One–shot mode (One-shot mode means the timer only counts a cycle).

Bits 1~0: Timer 3 Interrupt Type Select Bits.

These two bits are used when the Timer operates in PWM mode.

TC3IS1	TC3IS0	Timer 3 Interrupt Type Select
0	0	TC3DA(period) matching
0	1	TC3DB(duty) matching
1	×	TC3DA and TC3DB matching

TC3CR2: Timer/Counter 3 Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	TC3M2	TC3M1	TC3M0	-	TC3CK3	TC3CK2	TC3CK1	TC3CK0
Type	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD2; SFR Page = 0

Bits 7~5: Timer/Counter 3 operation mode select.

TC3M2	TC3M1	TC3M0	Operating Mode Select
0	0	0	Timer/Counter Rising Edge
0	0	1	Timer/Counter Falling Edge
0	1	0	Capture Mode Rising Edge
0	1	1	Capture Mode Falling Edge
1	0	0	Window mode
1	0	1	Programmable Divider output
1	1	0	Pulse Width Modulation output
1	1	1	Buzzer (output timer/counter clock source. The duty cycle of clock source must be 50/50)

Bit 4: Not used bits. Fixed to “0” all the time.

Bits 3~0: Timer/Counter 3 clock source prescaler select.

TC3CK3	TC3CK2	TC3CK1	TC3CK0	Clock Source Normal	Resolution 8 MHz F_C=8M	Max time 8 MHz F_C=8M	Resolution 16kHz F_C=16K	Max time 16kHz F_C=16K
0	0	0	0	F_C	125ns	32µs	62.5µs	16ms
0	0	0	1	$F_C/2$	250ns	64µs	125µs	32ms
0	0	1	0	$F_C/2^2$	500ns	128µs	250µs	64ms
0	0	1	1	$F_C/2^3$	1µs	256µs	500µs	128ms
0	1	0	0	$F_C/2^4$	2µs	512µs	1ms	256ms
0	1	0	1	$F_C/2^5$	4µs	1024µs	2ms	512ms
0	1	1	0	$F_C/2^6$	8µs	2048µs	4ms	1024ms
0	1	1	1	$F_C/2^7$	16µs	4096µs	8ms	2048ms
1	0	0	0	$F_C/2^8$	32µs	8192µs	16ms	4096ms
1	0	0	1	$F_C/2^9$	64µs	16384µs	32ms	8192ms
1	0	1	0	$F_C/2^{10}$	128µs	32768µs	64ms	16384ms
1	0	1	1	$F_C/2^{11}$	256µs	65536µs	128ms	32768ms
1	1	0	0	$F_C/2^{12}$	512µs	131072µs	256ms	65536ms
1	1	0	1	$F_C/2^{13}$	1.024ms	262144µs	512ms	131072ms
1	1	1	0	$F_C/2^{14}$	2.048ms	524.288ms	1.024s	262144ms
1	1	1	1	$F_C/2^{15}$	4.096ms	1.048s	2.048s	524288ms

TC3DAL: Timer/Counter 3 DATA Buffer A

Bit	7	6	5	4	3	2	1	0
Name	TC3DA.7	TC3DA.6	TC3DA.5	TC3DA.4	TC3DA.3	TC3DA.2	TC3DA.1	TC3DA.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD3; SFR Page = 0

Bits 7~0: Data buffer A of timer/counter 3

TC3DAH: Timer/Counter 3 DATA Buffer A

Bit	7	6	5	4	3	2	1	0
Name	TC3DA.15	TC3DA.14	TC3DA.13	TC3DA.12	TC3DA.11	TC3DA.10	TC3DA.9	TC3DA.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XD4; SFR Page = 0

Bits 7~0: Data buffer A of timer/counter 3

TC3DBL: Timer/Counter 3 Data Buffer B

Bit	7	6	5	4	3	2	1	0
Name	TC3DB.7	TC3DB.6	TC3DB.5	TC3DB.4	TC3DB.3	TC3DB.2	TC3DB.1	TC3DB.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD5; SFR Page = 0

Bits 7~0: Data buffer B of timer/counter 3.

TC3DBH: Timer/Counter 3 Data Buffer B

Bit	7	6	5	4	3	2	1	0
Name	TC3DB.15	TC3DB.14	TC3DB.13	TC3DB.12	TC3DB.11	TC3DB.10	TC3DB.9	TC3DB.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD6; SFR Page = 0

Bits 7~0: Data buffer B of timer/counter 3.

When Timer / Counter x is used as PWM mode, the duty value stored at register TCxDB must be smaller less than or equal to the period value stored at register TCxDA.,i.e. duty ≦ period. And then the PWM waveform is generated. If duty is greaterlarger than period, the PWM output waveform keeps remains inat high voltage level.

The period value set by users is extra plus 1 in inner circuit, for example::

If the period value is set as 0x4F, the circuit actually processes 0x50 period length.

If the period value is set as 0xFF, the circuit actually processes 0x100 period length.

TC3CR3: Timer/Counter 3 Control Register 3

Bit	7	6	5	4	3	2	1	0
Name	TC3SCS1	TC3SCS0	-	-	-	-	TC3SS1	TC3SS0
Type	R/W	R/W	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XD7; SFR Page = 0

Bit 7~6: Timer/Counter 3 start control source selection.

TC3SCS1	TC3SCS0	Timer/Counter 3 start control source selection
0	0	Control by TC3S(TC3CR3 bit7) bit.
0	1	SPI (Start Timer 3 upon receiving the first Byte, no matter if Buffer mode has started or not)
1	0	UART (Start Timer 3 upon receiving the first Byte, no matter if Buffer mode has started or not)
1	1	I2C (Start Timer 3 after correct Address check, no matter if Buffer mode has started or not)

Bit 5~2: Not used bits. Fixed to “0” all the time.

Bit 1~0: Timer/Counter 3 clock source select bits.

TC3SS1	TC3SS0	Timer 3 clock Source Select
0	0	FLS
0	1	FHS
1	0	PLL(48MHz)
1	1	External TC3 pin as count source (Fc). It is used only for timer/counter mode.

TC4CR1: Timer/Counter 4 Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	TC4S	TC4RC	TC4DBSF	TC4DASF	TC4FF	TC4MOS	TC4IS1	TC4IS0
Type	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD9; SFR Page = 0

Bit 7: Timer/Counter 4 start control (main switch for all modes).

0: Stop and clear counter.

1: Start.

Bit 6: Timer 4 Read Control Bit.

0: When this bit is set to 0, can't read data from TC4DB.

1: When this bit is set to 1, data read from TC4DB is a number of counting.

Bit 5: 16-bit Timer/Counter 4 Data B(Duty) status flag, cleared by software.

Bit 4: 16-bit timer/counter 4 Data A(Period) status flag, cleared by software.

Bit 3: Inversion for Timer/Counter 4 as PWM or PDO mode

0: Duty is logic 1.

1: Duty is logic 0.

Bit 2: Timer Output Mode Select Bit.

0: Repeating mode.

1: One–shot mode (One-shot mode means the timer only counts a cycle).

Bits 1~0: Timer 4 Interrupt Type Select Bits.

These two bits are used when the Timer operates in PWM mode.

TC4IS1	TC4IS0	Timer 4 Interrupt Type Select
0	0	TC4DA(period) matching
0	1	TC4DB(duty) matching
1	×	TC4DA and TC4DB matching

TC4CR2: Timer/Counter 4 Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	TC4M2	TC4M1	TC4M0	-	TC4CK3	TC4CK2	TC4CK1	TC4CK0
Type	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDA; SFR Page = 0

Bits 7~5: Timer/Counter 4 operation mode select.

TC4M2	TC4M1	TC4M0	Operating Mode Select
0	0	0	Timer/Counter Rising Edge
0	0	1	Timer/Counter Falling Edge
0	1	0	Capture Mode Rising Edge
0	1	1	Capture Mode Falling Edge
1	0	0	Window mode
1	0	1	Programmable Divider output
1	1	0	Pulse Width Modulation output
1	1	1	Buzzer (output timer/counter clock source. The duty cycle of clock source must be 50/50)

Bit 4: Not used bits. Fixed to “0” all the time.

Bits 3~0: Timer/Counter 4 clock source pre-scaler select.

TC4CK3	TC4CK2	TC4CK1	TC4CK0	Clock Source Normal	Resolution 8MHZ F_C=8M	Max time 8MHz F_C=8M	Resolution 16KHZ F_C=16K	Max time 16KHz F_C=16K
0	0	0	0	F_C	125ns	32us	62.5us	16ms
0	0	0	1	$F_C/2$	250ns	64us	125us	32ms
0	0	1	0	$F_C/2^2$	500ns	128us	250us	64ms
0	0	1	1	$F_C/2^3$	1us	256us	500us	128ms
0	1	0	0	$F_C/2^4$	2us	512us	1ms	256ms
0	1	0	1	$F_C/2^5$	4us	1024us	2ms	512ms
0	1	1	0	$F_C/2^6$	8us	2048us	4ms	1024ms
0	1	1	1	$F_C/2^7$	16us	4096us	8ms	2048ms
1	0	0	0	$F_C/2^8$	32us	8192us	16ms	4096ms
1	0	0	1	$F_C/2^9$	64us	16384us	32ms	8192ms
1	0	1	0	$F_C/2^{10}$	128us	32768us	64ms	16384ms
1	0	1	1	$F_C/2^{11}$	256us	65536us	128ms	32768ms
1	1	0	0	$F_C/2^{12}$	512us	131072us	256ms	65536ms
1	1	0	1	$F_C/2^{13}$	1.024ms	262144us	512ms	131072ms
1	1	1	0	$F_C/2^{14}$	2.048ms	524.288ms	1.024s	262144ms
1	1	1	1	$F_C/2^{15}$	4.096ms	1.048s	2.048s	524288ms

TC4DAL: Timer/Counter 4 Data Buffer A

Bit	7	6	5	4	3	2	1	0
Name	TC4DA.7	TC4DA.6	TC4DA.5	TC4DA.4	TC4DA.3	TC4DA.2	TC4DA.1	TC4DA.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDB; SFR Page = 0

Bits 7~0: Data buffer A of timer/counter 4.

TC4DAH: Timer/Counter 4 Data Buffer A

Bit	7	6	5	4	3	2	1	0
Name	TC4DA.15	TC4DA.14	TC4DA.13	TC4DA.12	TC4DA.11	TC4DA.10	TC4DA.9	TC4DA.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDC; SFR Page = 0

Bits 7~0: Data buffer A of timer/counter 4.

TC4DBL: Timer/Counter 4 Data Buffer B

Bit	7	6	5	4	3	2	1	0
Name	TC4DB.7	TC4DB.6	TC4DB.5	TC4DB.4	TC4DB.3	TC4DB.2	TC4DB.1	TC4DB.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDD; SFR Page = 0

Bits 7~0: Data buffer B of 8 bit timer/counter 4

TC4DBH: Timer/Counter 4 Data Buffer B

Bit	7	6	5	4	3	2	1	0
Name	TC4DB.15	TC4DB.14	TC4DB.13	TC4DB.12	TC4DB.11	TC4DB.10	TC4DB.9	TC4DB.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDE; SFR Page = 0

Bits 7~0: Data buffer B of 8 bit timer/counter 4

TC4CR3: Timer/Counter 4 Control Register 3

Bit	7	6	5	4	3	2	1	0
Name	TC4SCS1	TC4SCS0	-	-	-	-	TC4SS1	TC4SS0
Type	R/W	R/W	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDF; SFR Page = 0

Bit 7~6: Timer/Counter 4 start control source selection.

TC4SCS1	TC4SCS0	Timer/Counter 4 start control source selection
0	0	Control by TC3S(TC3CR3 bit7) bit.
0	1	SPI (Start Timer 4 upon receiving the first Byte, no matter if Buffer mode has started or not)
1	0	UART (Start Timer 4 upon receiving the first Byte, no matter if Buffer mode has started or not)
1	1	I2C (Start Timer 4 after correct Address check, no matter if Buffer mode has started or not)

Bit 5~2: Not used bits. Fixed to “0” all the time.

Bit 1~0: Timer/Counter 4clock source select bits.

TC4SS1	TC4SS0	Timer 4 clock Source Select
0	0	FLS
0	1	FHS
1	0	PLL(48MHz)
1	1	External TC3 pin as count source (Fc). It is used only for timer/counter mode.

15 Operational Amplifier (OP)/Comparator (CMP)

The MCU has a comparator comprising of five analog inputs and one output. CMP1 can also be used as OP. In order to achieve MCU pins usability, the MCU provides a variety of internal wiring connections to minimize the needs of external wiring connections. The comparator can be utilized to wake up the MCU from green mode. The comparator and OP circuit illustrations are shown below.

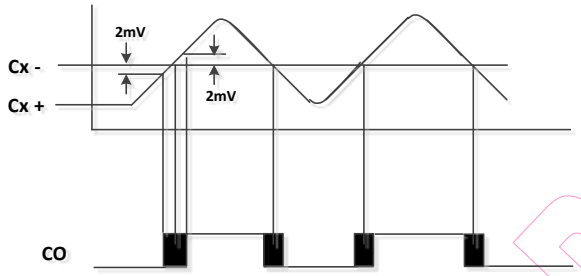
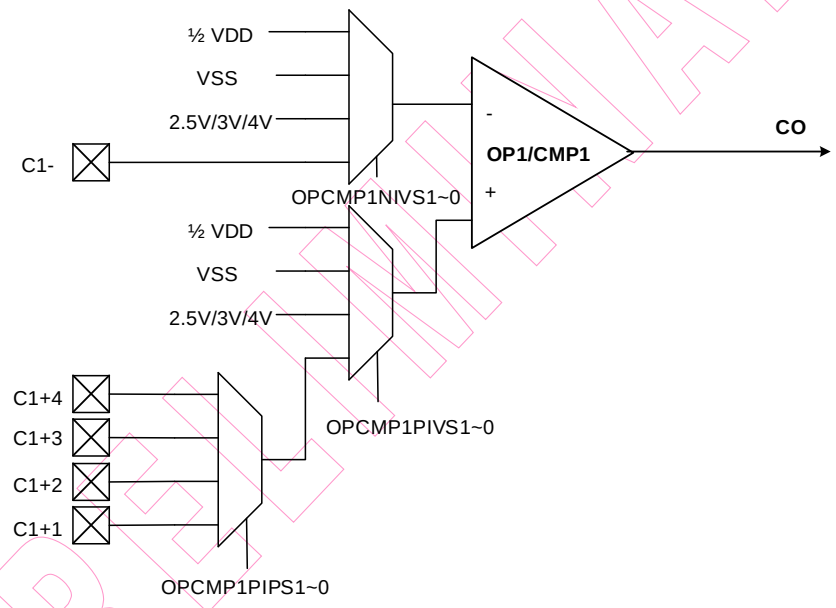


Figure 15-1 CMP Input Voltage Offset Illustration



OPCMP1CR1: OP/CMP 1 Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	OPCMP1En	OPCMP1SW	CMP1OUT_SEL	CMP1OUT	-	-	SC1PWM edge	-
Type	R/W	R/W	R/W	R	R	R	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xEE; SFR Page = 0

Bit 7: OP/CMP1 function enable bit

0: Disable

1: Enable

Bit 6: OP/CMP1 function switch bit

0: Switch to CMP

1: Switch to OP. CO1 is used as OP1OUT

Bit 5: CMP1 output selection. This bit is valid when OPCMP1SW = 0.

0: Do not output to pin. At this time, the pin is used as other function pin.

1: Output to pin. At this time, the pin is dedicated CMP1OUT pin

Bit 4 : CMP1 outputs result. This bit is valid when OPCMP1SW = 0.

0: The result of comparator1 is logic low (0).

1: The result of comparator1 is logic high (1).

Bits 3~2:Reserved. Read = 0, Write = Don't Care.

Bit 1: Edge selection of CMP1 comparing result for shutting down PWM function

0: Rising edge. When CMP1 output result changes from 0 to 1, the PWM function is shut down.

1: Falling edge. When CMP1 output result changes from 1 to 0, the PWM function is shut down.

Bit 0:Reserved. Read = 0, Write = Don't Care.

OPCMP1CR2: OP/CMP 1 Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	-	-	OPCMP1 PIPS1	OPCMP1 PIPS0	OPCMP1 PIVS1	OPCMP1 PIVS0	OPCMP1 NIVS1	OPCMP1 NIVS0
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xED; SFR Page = 0

Bits 7~6: Reserved. Read = 0, Write = Don't Care.

Bit 5~4: Pin Selection for OP/CMP1 Positive Input.

OPCMP1 PIPS1	OPCMP1 PIPS0	OP/CMP1 Positive Input
0	0	CIP1+1
0	1	CIP1+2
1	0	CIP1+3
1	1	CIP1+4

Bit 3~2: Voltage Reference Selection for OP/CMP1 Positive Input.

OPCMP1 PIVS1	OPCMP1 PIVS0	OP/CMP1 Positive Input
0	0	To pad
0	1	VSS
1	0	Internal Voltage Reference(by VPIS2~0 bits)
1	1	1/2VDD

Bit 1~0: Voltage Reference Selection for CMP1 Negative Input.

OPCMP1NIVS1	OPCMP1NIVS0	OP/CMP1 Negative Input
0	0	To pad
0	1	VSS
1	0	Internal Voltage Reference(by VPIS2~0 bits)
1	1	1/2VDD

SDPWMCR1: Shutting Down PWM Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	SC1PWMCEN	SC1PWMBEN	SC1PWMAEN
Type	R	R	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE6; SFR Page = 0

Bits 7~3: Reserved. Read = 0, Write = Don't Care.

Bit 2: Shutting Down PWM enable bit. Use CMP1 output result to shut down PWMC function.

0: Disable

1: Enable

Bit 1: Shutting Down PWM enable bit. Use CMP1 output result to shut down PWMB function.

0: Disable

1: Enable

Bit 0: Shutting Down PWM enable bit. Use CMP1 output result to shut down PWMA function.

0: Disable

1: Enable

COSF1: CMP Status Flag Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	CMP1RFS	-	-	-	CMP1SF
Type	R	R	R	R/W	R	R	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xEF; SFR Page = 0

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bit 4: Set which edge of CMP1OUT used to trigger an interrupt.

0: Rising edge (0->1)

1: Falling edge (1->0)

Bits 3~1: Reserved. Read = 0, Write = Don't Care.

Bit 0: CMP1 interrupt status flag. Set by H/W and cleared by S/W.

0: no interrupt occurs.

1: interrupt occurs.

CMPNRCR1: CMP Rising/Falling edge Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	NRC1	NRC0	-	-	-	-		CMP1NREN
Type	R	R	R	R	R	R	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE7; SFR Page =0

Bits 7~6: Noise rejection cycle count

NRC[1:0]	Cycle
00	2
01	4
10	8
11	16

Bits 5~1: Not used bits. Fixed to “0” all the time.

Bit 0: Noise rejection enable bit of CMP1

0: Disable (default)

1: Enable

CO1TRL: CMP1 TRIM Low Register

Bit	7	6	5	4	3	2	1	0
Name	ITR1E	CMP1TS	-	-	CMP1T.3	CMP1T.2	CMP1T.1	CMP1T.0
Type	R/W	R/W	R	R-	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9A; SFR Page = 1

Bit 7: Internally trimmed enable bit of comparator1

0: Disable (default)

1: Enable. Positive and negative input ends are disconnected from pad and connected by internal switch, and then connected to internal reference voltage.

Bit 6: The trimmed sign bit of comparator1

Bits 5~4: Reserved. Read = 0, Write = Don't Care.

Bits 3~0: The trimmed bits of comparator1

16 12-Bit ADC

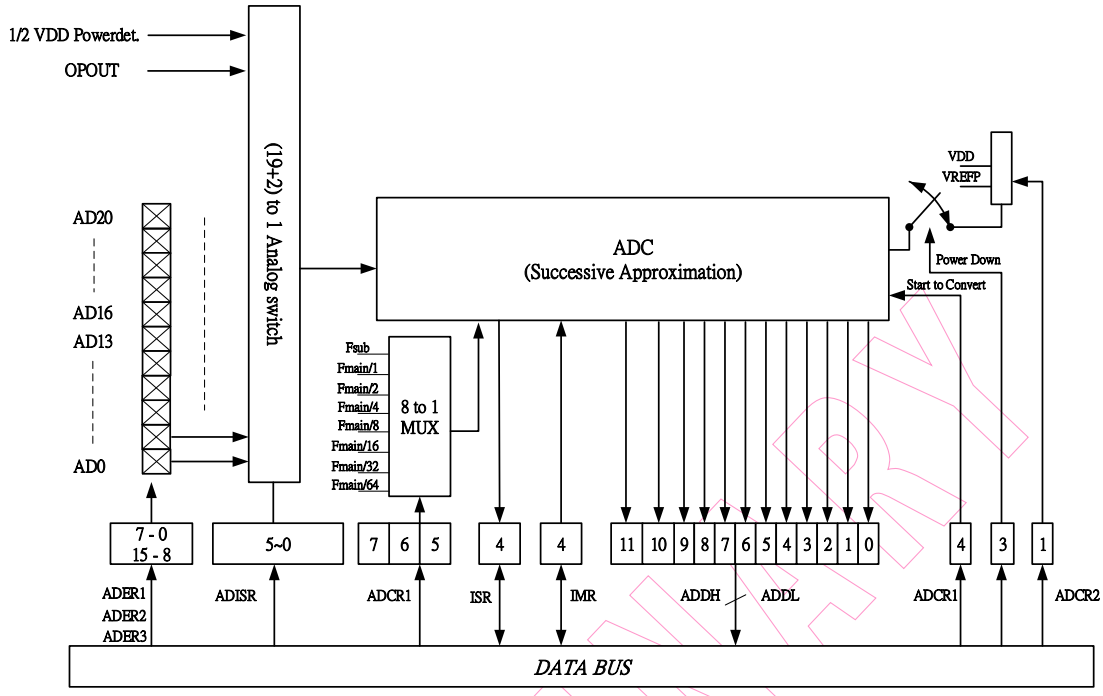


Figure 16- 1AD Converter

This is a 12-bit successive approximation register analog to digital converter (SAR ADC). There are two reference voltages for SAR ADC. The positive reference voltage can select internal AVDD, internal voltage sources or external input pin by setting the VREFP & VPIS[1:0] bits in ADCR2. Connecting to external positive reference voltage provides more accuracy than using internal AVDD.

16.1 ADC Data Register

When the AD conversion is completed, the result is loaded to the ADDH and ADDL. And the ADSF is set if ADIE is enabled.

16.2 A/D Sampling Time

The accuracy, linearity, and speed of the successive approximation AD converter are dependent on the properties of the ADC. The source impedance and the internal sampling impedance directly affect the time required to charge the sample and hold capacitor. The application program controls the length of the sample time to meet the specified accuracy. Generally speaking, the program should wait for 2 µs for each kilo ohms of the analog source impedance and at least 2 µs for the low-impedance source. The maximum recommended impedance for the analog source is 10k ohms at VDD = 5V. After the analog input channel is selected; this acquisition time must be done before AD conversion can be started.

16.3 A/D Conversion Time

CKR[2:0] select the conversion time (TCT), in terms of instruction cycles. This allows the MCU to run at maximum frequency without sacrificing accuracy of AD conversion. For the MCU, the conversion time per bit is about 1µs. The following table shows the relationship between TCT and the maximum operating frequencies.

System Mode	CKR2~0	Clock Rate	Max. system operation frequency in 2.7~5V	Max. system operation frequency in 3.0~5.5V
Normal Mode	000	F_{HS}/16	8 MHz	20MHz
	001	F_{HS}/8	4 MHz	16 MHz
	010	F_{HS}/4	2 MHz	8 MHz
	011	F_{HS}/2	1 MHz	4 MHz
	100	F_{HS}/64	20 MHz	20 MHz
	101	F_{HS}/32	16 MHz	20 MHz
	110	F_{HS}/1	0.5 MHz	2 MHz
	111	F_{LS}	F_{LS}	F_{LS}
Slow/Green Mode	xxx	F_{LS}	F_{LS}	F_{LS}

16.4 ADC Operation during Sleep Mode

In order to obtain a more accurate ADC value and reduce power consumption, the AD conversion remains operational during sleep mode. As the SLEP instruction is executed, all the MCU operations will stop except for the Oscillator, Timer0~1, Timer3~4, PWM timers and AD conversion.

1. The AD Conversion is considered completed as determined by:
2. The ADRUN bit is cleared to “0”.
3. The ADSF bit is set to “1”.

Wake up and enters into Interrupt vector if the ADIE bit and the global interrupt is enabled.

The results are fed into the ADDL and ADDH registers when the conversion is completed.

16.5 Programming Process/Considerations

Follow these steps to obtain data from the ADC:

1. Write to the twenty one bits (ADE[20:0]) on the ADER1~3 registers to define the characteristics of IO used as digital I/O, analog channels, or voltage reference pin

2. Write to the ADC register to configure the AD module:
   a) Select the ADC input channel (ADIS[5:0))
   b) Define the AD conversion clock rate (CKR[2:0])
   c) Select the VREFS input source of the ADC
   d) Set the ADP bit to 1 to begin sampling
3. Set the ADIE bit and enable global interrupt, if the interrupt or wake-up functions are employed
4. Set the ADRUN bit to 1
5. Write “SLEP” instruction or polling.
6. Wait for wake-up or for the ADRUN bit to be cleared to “0” , status flag (ADSF) is set “1,” or ADC interrupt occurs.
7. Read the ADDL and ADDH conversion data registers. If the ADC input channel changes at this time, the ADDL and ADDH values can be cleared to ‘0’.
8. Clear the status flag (ADSF).
9. For next conversion, go to Step 1 or Step 2 as required. At least two TCT is required before the next acquisition starts.

NOTE
In order to obtain accurate values, it is necessary to avoid any data transition on I/O pins during AD conversion

16.6 Programming Process for Detecting Internal VDD and OPAmp

In detecting the internal VDD and the operational amplifier output operation, a difference from the previous chapter is that before starting the ADC conversion operation, the VDD detection and operational amplifier should first be prepared.

VDD Detection：

The only thing to note in VDD detection is that, prior starting the AD conversion operation, switch the channel to ½ VDD channel. When the dividing resistor is initiated, AD conversion can be performed immediately. If user wants to get accurate conversion values, many approaches can be taken such as adding capacitance to the VDD Pin, doing conversions more than twice, and taking averages of several data. These can all increase data reliability.

Note that when not detecting VDD ordinarily, be sure not to keep on switching the channel to detect ½ VDD channel, because there is always a DC current consumption. When an analog multiplexer must switch to other channel, it will turn off the resistor divider.

Detection of the OPAmp Output：

When detecting OPAmp output signal, OPAmp must be let to work before AD conversion. Therefore, before performing AD conversion, user must first confirm that the settings and operation of OPAmp is normal, in order to get the correct value. As for the AD conversion control program, the process is similar to the previous section. User should pay attention as to whether the correct channel is selected, to prevent getting incorrect values.

In the process of dynamic signal conversion, note that both input signal bandwidth and ADC conversion time need to confirm to the Nyquist Theorem. Otherwise, the conversion will be unable to respond or result in distorted digital signals.

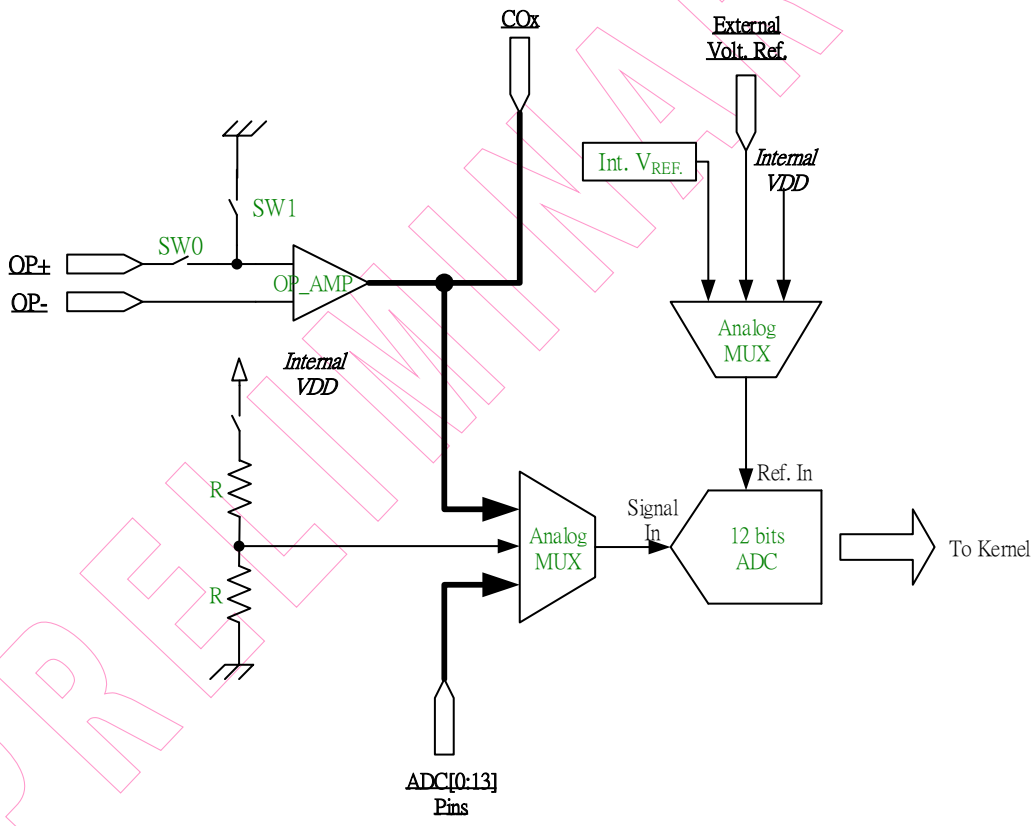


Figure 16- 2 Block diagram of ADC,CMP & OPAmp, and VDD Detection.

(With ½ VDD resistor divider added, the SIP actually has only one output to ADC Mux, handled by digital circuit)

16.7 Description of ADC control registers

ADCR1 (ADC Control Register 1)

Bit	7	6	5	4	3	2	1	0
Name	CKR2	CKR1	CKR0	ADRUN	ADP	ADOM	SHS1	SHS0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x91; SFR Page = 0x0

Bit 7~5 (CKR2~0): Clock Rate Selection of ADC

System Mode	CKR2~0	Clock Rate	Max. system operation frequency in 2.7~5V	Max. system operation frequency in 3.0~5V
Normal Mode	000	F_{Main}/16	8 MHz	20 MHz
	001	F_{Main}/8	4 MHz	16 MHz
	010	F_{Main}/4	2 MHz	8 MHz
	011	F_{Main}/2	1 MHz	4 MHz
	100	F_{Main}/64	20 MHz	20 MHz
	101	F_{Main}/32	16 MHz	20 MHz
	110	F_{Main}/1	0.5 MHz	2 MHz
	111	FLS	F_{LS}	FLS
Slow/Green Mode	xxx	FLS	F_{LS}	FLS

Bit 4 (ADRUN): ADC Starts to Run

In single mode:

0: Reset on completion of the conversion by hardware, this bit cannot be reset by software.

1: An A/D conversion starts. This bit can be set by software

In continuous mode:

0: ADC is stopped.

1: ADC is running unless this bit is reset by software

Bit 3 (ADP): ADC Power

0: ADC is in power down mode.

1: ADC is operating normally.

Bit 2 (ADOM): ADC Operation Mode Selection

0: ADC operates in single mode.

1: ADC operates in continuous mode.

Bit 1~0 (SHS1~0): Sample & Hold Timing Selection

SHS[1:0]	Sample & Hold Timing
00	2 x T_{AD}
01	4 x T_{AD}
10	8 x T_{AD}
11	12 x T_{AD}

ADCR2 (ADC Control Register 2)

Bit	7	6	5	4	3	2	1	0
Name	ADSF	ADIM	ADCMS	VPIS2	VPIS1	VPIS0	VREFP	VREFN
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x92; SFR Page = 0x0

Bit 7: ADC status flag. Set when AD conversion is completed, reset by software.

Bit 6 (ADIM): ADC Interrupt Mode

0: Normal mode. Interrupt occurred after AD conversion completed.

1: Compare mode. Interrupt occurred when comparison result conforms the setting of ADCMS bits.

Bit 5 (ADCMS): ADC Comparison Mode Selection.

In compare mode:

0: Interrupt occurred when AD conversion data is greater than data in ADCVL, ADCVH register.

It means when ADD >= ADCD, interrupt occurred.

1: Interrupt occurred when AD conversion data is less than data in ADCVL, ADCVH register.

It means when ADD < =ADCD, interrupt occurred.

In normal mode:

No effect

Bits 4 ~ 2 (VPIS2~0): Internal Positive Reference Voltage Selection.

VPIS[2]	VPIS[1:0]	Reference Voltage
0	00	4.096 V
0	01	3.072 V
0	10	2.560 V
0	11	2.048 V
1	00	AVDD
1	01	AVDD
1	10	AVDD
1	11	AVDD

Bit 1 (VREFP): Positive Reference Voltage Selection

0: Internal positive reference voltage. The actual voltage is set by VPIS[2:0] bits

1: From VREF pin.

Bit 0:(VREFN): Negative Reference Voltage Selection

0: from GNDA pad

1: from GNDA pad (combine VREF's GND to GNDA)

> **NOTE**
> *When using internal voltage reference, users need to wait for at least 50s the first time to enable and stabilize the internal voltage reference circuit. After that, users only need to wait for 6s at least whenever switching voltage references.*

ADISR (Analog to Digital Converter Input Channel Selection Register)

Bit	7	6	5	4	3	2	1	0
Name	-	-	ADIS5	ADIS4	ADIS3	ADIS2	ADIS1	ADIS0
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x93; SFR Page = 0x0

Bits 7~6: Reserved. Read = 0, Write = Don't Care.

Bits 5~0 (ADIS5~0): ADC input channel selection bits

ADISR[5:0]	Selected Channel	ADISR[5:0]	Selected Channel
000000	ADC0	010000	ADC16
000001	ADC1	010001	ADC17
000010	ADC2	010010	ADC18
000011	ADC3	010011	ADC19
000100	ADC4	010100	ADC20
000101	ADC5	010101 `````` 111101	N/A (1/2 VDD PowerDet)
000110	ADC6		
000111	ADC7		
001000	ADC8		
001001	ADC9		
001010	ADC10		
001011	ADC11		
001100	ADC12		
001101	ADC13		
001110	N/A (1/2 VDD PowerDet)	111110	OPOUT
001111	N/A (1/2 VDD PowerDet)	111111	1/2 VDD PowerDet

Note: Internal Vref stable time 4us.

ADER1 (Analog to Digital Converter Input Control Register 1)

Bit	7	6	5	4	3	2	1	0

Name	ADE.7	ADE.6	ADE.5	ADE.4	ADE.3	ADE.2	ADE.1	ADE.0
Type	R/W	R/W	R/W	R/w	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x94; SFR Page = 0x0

Bit 7 (ADE.7): AD converter enable bit.

0: Disable ADC7, used as other function pin.

1: Enable ADC7, use as analog input pin.

Bit 6 (ADE.6): AD converter enable bit

0: Disable ADC6, used as other function pin.

1: Enable ADC6, used as analog input pin

Bit 5 (ADE.5): AD converter enable bit

0: Disable ADC5, used as other function pin.

1: Enable ADC5, used as analog input pin

Bit 4 (ADE.4): AD converter enable bit

0: Disable ADC4, used as other function pin.

1: Enable ADC4, used as analog input pin

Bit 3 (ADE.3): AD converter enable bit

0: Disable ADC3, used as other function pin.

1: Enable ADC3, used as analog input pin

Bit 2 (ADE.2): AD converter enable bit

0: Disable ADC2, used as other function pin.

1: Enable ADC2, used as analog input pin

Bit 1 (ADE.1): AD converter enable bit

0: Disable ADC1, used as other function pin.

1: Enable ADC1, used as analog input pin

Bit 0 (ADE.0): AD converter enable bit

0: Disable ADC0, used as other function pin.

1: Enable ADC0, used as analog input pin

ADER2 (Analog to Digital Converter Input Control Register 2)

Bit	7	6	5	4	3	2	1	0
Name	-	-	ADE.13	ADE.12	ADE.11	ADE.10	ADE.9	ADE.8
Type	R	R	R/W	R/w	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x95; SFR Page = 0x0

Bits 7~6: Reserved. Read = 0, Write = Don't Care.

Bit 5 (ADE.13): AD converter enable bit of P55 pin.

0: Disable ADC13, used as other function pin.

1: Enable ADC13, used as analog input pin

Bit 4 (ADE.12): AD converter enable bit

0: Disable ADC12, used as other function pin.

1: Enable ADC12, used as analog input pin

Bit 3 (ADE.11): AD converter enable bit

0: Disable ADC11, used as other function pin.

1: Enable ADC11, used as analog input pin

Bit 2 (ADE.10): AD converter enable bit

0: Disable ADC10, used as other function pin.

1: Enable ADC10, used as analog input pin

Bit 1 (ADE.9): AD converter enable bit

0: Disable ADC9, used as other function pin.

1: Enable ADC9, used as analog input pin

Bit 0 (ADE.8): AD converter enable bit

0: Disable ADC8, used as other function pin.

1: Enable ADC8, used as analog input pin

ADER3 (Analog to Digital Converter Input Control Register 3)

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	ADE.20	ADE.19	ADE.18	ADE.17	ADE.16
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9C; SFR Page = 0x0

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bit 4 (ADE.20): AD converter enable bit

0: Disable ADC20, used as other function pin.

1: Enable ADC20, used as analog input pin

Bit 3 (ADE.19): AD converter enable bit

0: Disable ADC19, used as other function pin.

1: Enable ADC19, used as analog input pin

Bit 2 (ADE.18): AD converter enable bit

0: Disable ADC18, used as other function pin.

1: Enable ADC18, used as analog input pin

Bit 1 (ADE.17): AD converter enable bit

0: Disable ADC17, used as other function pin.

1: Enable ADC17, used as analog input pin

Bit 0 (ADE.16): AD converter enable bit

0: Disable ADC16, used as other function pin.

1: Enable ADC16, used as analog input pin

ADDL (Low Byte of Analog to Digital Converter Data)

ADFM =0

Bit	7	6	5	4	3	2	1	0
Name	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 7~0 (ADD7~0): Low Byte of AD Data Buffer

ADFM =1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	ADD3	ADD2	ADD1	ADD0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x96; SFR Page = 0x0

Bits 7~4: Reserved. Read = 0, Write = Don't Care.

Bits 3~0 (ADD3~0): Low Byte of AD Data Buffer

ADDH (High Byte of Analog to Digital Converter Data)

ADFM =0

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	ADD11	ADD10	ADD9	ADD8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

Bits 7~5: Reserved. Read = 0, Write = Don't Care.

Bits 3~0 (ADD11~8): High Byte of AD Data Buffer.

ADFM =1

Bit	7	6	5	4	3	2	1	0
Name	ADD.11	ADD.10	ADD.9	ADD.8	ADD.7	ADD.6	ADD.5	ADD.4
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x97; SFR Page = 0x0

Bits 7~0 (ADD11~4): High Byte of AD Data Buffer.

The format of AD data is dependent on code option ADFM. The following table shows how the data justified in different ADFM settings.

ADFM			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADC 12 bits	0	ADDH					ADD11	ADD10	ADD9	ADD8
		ADDL	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
	1	ADDH	ADD11	ADD10	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4
		ADDL					ADD3	ADD2	ADD1	ADD0

Note: Not used Bits are set to “0” by hardware. If ADFM= 0 and when 12-bit resolution,

ADDH<7:4> = 0000

ADCVL (Low Byte of Analog to Digital Comparison)

Bit	7	6	5	4	3	2	1	0
Name	ADCV.7	ADCV.6	ADCV.5	ADCV.4	ADCV.3	ADCV.2	ADCV.1	ADCV.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

+SFR Address = 0x9A; SFR Page = 0x0

Bits 7~0 (ADCV7~0): Low Byte Data for AD Comparison.

User should use the data format as the same as ADDH and ADDL registers. Otherwise, it will yield inaccurate results after AD comparison.

ADCVH (High Byte of Analog to Digital Comparison)

Bit	7	6	5	4	3	2	1	0
Name	ADCV.15	ADCV.14	ADCV.13	ADCV.12	ADCV.11	ADCV.10	ADCV.9	ADCV.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9B; SFR Page = 0x0

Bits 7~0 (ADCV15~8): High Byte Data for AD Comparison

User should use the data format as the same as ADDH and ADDL register. Otherwise, it will yield inaccurate results after AD comparison.

17 Serial Peripheral Interface(SPI)

17.1 Overview and Features

SPI is a Serial Peripheral Interface that provides a simple and efficient method of data exchange between two or more devices. The SPI protocol has master/slave deice. The communication is controlled by master device. If master initial the data frame, slave will be active. As the simple protocol, multiple devices can transmit/receive the data information by this 4-wire Serial interface.

- Two operational modes: Master mode and Slave mode
- Baud rate: 7 different programmable baud rates
  - Periodic SPI clock, where the SPI clock period is an integer multiple of CLK.
- Data Word length: 8 bits. Data must be left aligned when written to the transmit buffer register. Data read back from the receive buffer register is right aligned.
- Full duplex: Simultaneous receive and transmit operation.
- Clocking: 4 programmable clocking schemes.
- Interrupt/polling: Transmit and receive operations are accomplished by either interrupt-driven or polling.
- Support data transfer direction by selecting either MSB first or LSB first.
- Support receive data mode with data overrun status.
- Support data transfer with time delay between packages and the next package.

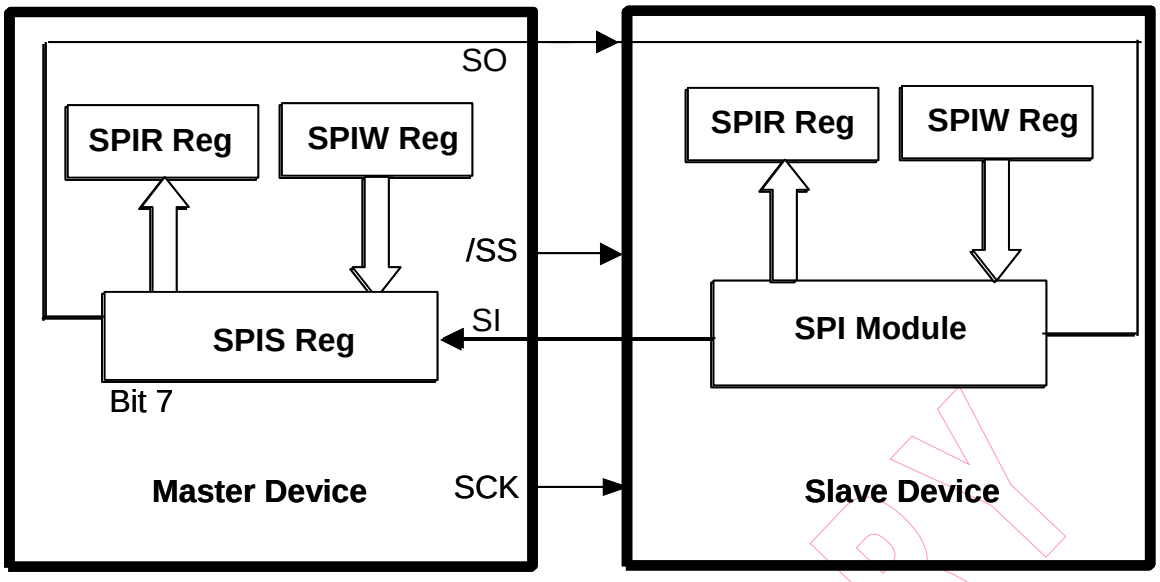


Figure 14 SPI Master/Slave Communication

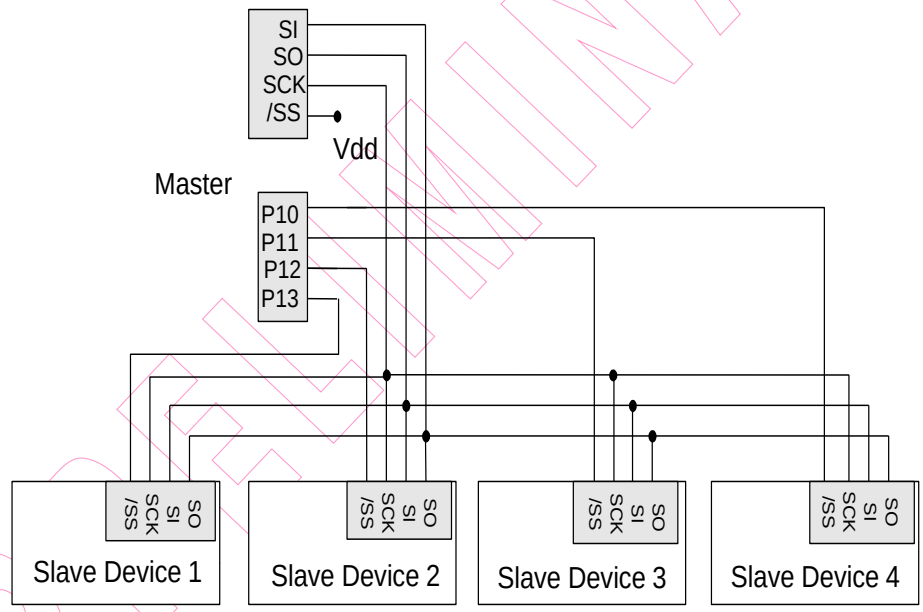


Figure 15 SPI Configuration of Single-Master and Multi-Slave

17.2 SPI Signal

Serial Clock (SCK)

SCK is the Serial Peripheral Interface clock signal. This control signal is driven by the master and controls the rate at which data is transferred. The master may transmit data at a variety of clock rates. SCK will cycle once for each bit that is transmitted. It is an output signal if the device is configured as a master, or as an input signal if the device is configured as Slave.

The clock rate is selected by the SPI clock rate selects Bit SPR[2:0] in the SPICON of the master Device.

The data is always shifted out at one edge of the clock and sampled at the opposite edge. Clock polarity and clock phase relative to the data are programmed into the SPICON control register and define the transfer format.

SPI SCK has wake-up functionality in Sleep modes.

Master Data Out Slave Data In (MOSI)

The MOSI pin is a data transmit (output) pin for transmitting output data when it is in master mode. The MOSI pin is a data receive (input) pin for receiving input data when it is in slave mode.

Master Data In Slave Data Out (MISO)

The MISO pin is a data receive (input) pin for receiving input data when it is in master mode. The MISO pin is a data transmit (output) pin for transmitting output data when it is in slave mode. The MISO pin of a slave device will be placed in the high impedance state if the slave device is not selected.

Slave Select (/SS)

The /SS is the Serial Peripheral Interface Slave Select input signal. This is an active low signal used to enable a slave device. This input-only pin functions like a chip select, and is provided by the master device for the slave devices. For the master device, the /SS pin can be set as GPIO pin.

17.3 SPI Transfer Format

The SPI supports four different combinations of serial clock phase and polarity. The user application code can select any of these combinations using the CPOL and CPHA bits in the Control register.

The clock polarity and the clock phase should be identical for the master device and the slave device involved in the communication link. The transfer format from the master may be changed between transfers to adjust to various requirements of a slave device.

For the master device, a transfer begins when data is written to SPITDBR and ends when TCF is cleared. For slave with CPHA = 0, a transfer starts when Slave Select (/SS) goes low and ends when Slave Select (/SS) returns high. In this case, SPIIF is set at the middle of the last Serial Clock (SCK) cycle when data is transferred from the shifter to the parallel data register, but the transfer will continue until /Slave Select (/SS) goes high.

On the other hand, for slave with CPHA = 1, a transfer starts with the first active edge of Serial Clock (SCK) and ends when TCF is cleared at the sampling edge of the last Serial Clock (SCK) cycle.

When each transfer is completed, the TCF will be cleared and an interrupt will be generated if the SPI interrupt is enabled.

17.4 Shift Data Register(SPISFDR)

The Data Shift Register (SPISFDR) is a 16-bit data shift register (not accessible by software). The SPISFDR is buffered to prevent a write to SPITDBR from overwriting the shift register during an active transfer.

The data in SPISFDR is shifted out (MSB) in the subsequent Serial Clock (SCK) cycles. For every bit (MSB) shifted out of the SPI, a bit is shifted into the LSB end of the shift register.

17.5 Operations

Master Mode Operation

When a device is configured as a master (MSTR = 0), the SPI provides the serial clock on the SCK pin for the entire serial communications network.

The SPR[2:0] in the control register determines both transmit and receive bit transfer rate for the network. The SPI supports 7 different data transfer rates.

Any data written to SPITDBR initiates data transmission on the MO (Master Out) pin if the SPI module is enabled. Simultaneously, the received data is shifted through the MI (Master In) pin into the LSB of SPISFDR. When the selected number of bits has been transmitted, the received data is loaded into the SPIRDBR for the software to read. Data is stored right aligned in SPIRDBR.

When the receive data transfer is completed, which means that the specified number of data bits has been shifted through SPISFDR, the following events will then occur:

The SPISFDR contents are transferred to SPIRDBR.

The TCF bit is cleared to 0.

If the SPI interrupt is enabled, an interrupt is asserted.

Slave Mode Operation

When a device is configured as a slave (MSTR = 1), the SCK pin is used as the input for the serial shift clock which is supplied from the external master.

If data is to be transmitted by the slave simultaneously, and SPITDBR has not been previously loaded, the data must be written to SPITDBR before the beginning of the SCK signal.

The /SS pin operates as the slave-select pin. An active low signal on the /SS pin allows the slave SPI to transfer data to the serial data line. An inactive high signal causes the slave SPI serial Shift register to stop and its serial output pin is placed into high-impedance state. This allows many slave devices to be tied together on the network, although only one slave device is selected at a time.

Master Transmission (Single Mode)

In master mode, it's forbidden to do consecutive TDBR writes or only one instruction between TDBR writes because unexpected error will happen.

However, if three or more TDBR writes are proceeded and the interval is at least 2 instruction cycles, the intermediate TDBR write may not be transmitted. 2nd TDBR write may be overwritten by 3rd TDBR write if 2nd TDBR data is still not sent out serially (ex. During 1st TDBR serial out period) when 3rd TDBR write is done. Therefore, it should be careful to do multiple TDBR writes.

Slave Transmission (Single Mode)

In slave mode, if SCK of 1st transfer doesn't begin to toggle, 2nd TDBR write will overwrite 1st TDBR write no matter how many cycles between 1st and 2nd TDBR writes. When SCK of 1st transfer begins to toggle, 2nd TDBR can be written and will be "delayed" to be loaded into shift registers until 1st transfer finishes.

Transmission with DTDLY Condition

In both master and slave modes, DTDLY can be set to delay the end of the transfer. The last bit of transmission data will be extended as shown in Figure 1.

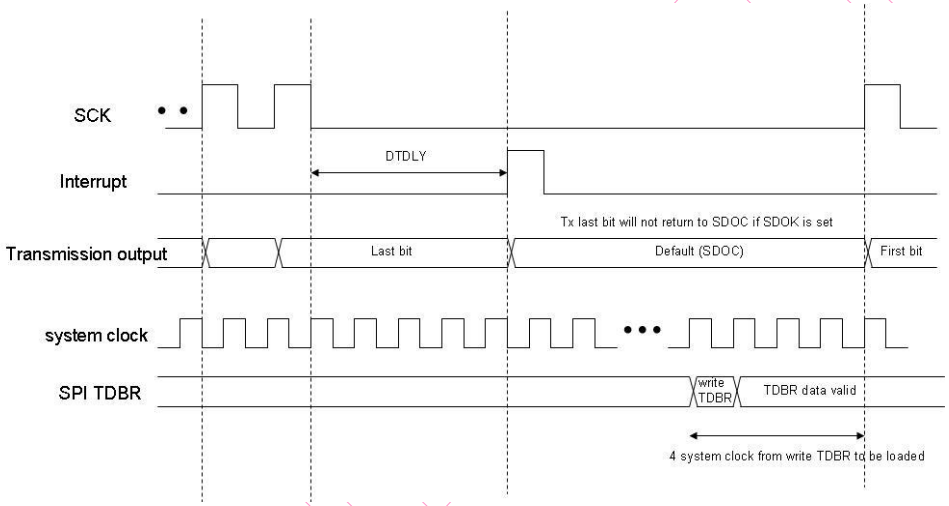


Figure 1 Transmission diagram with DTDLY condition

17.6 SPI Timing Diagram

SPI Master Mode

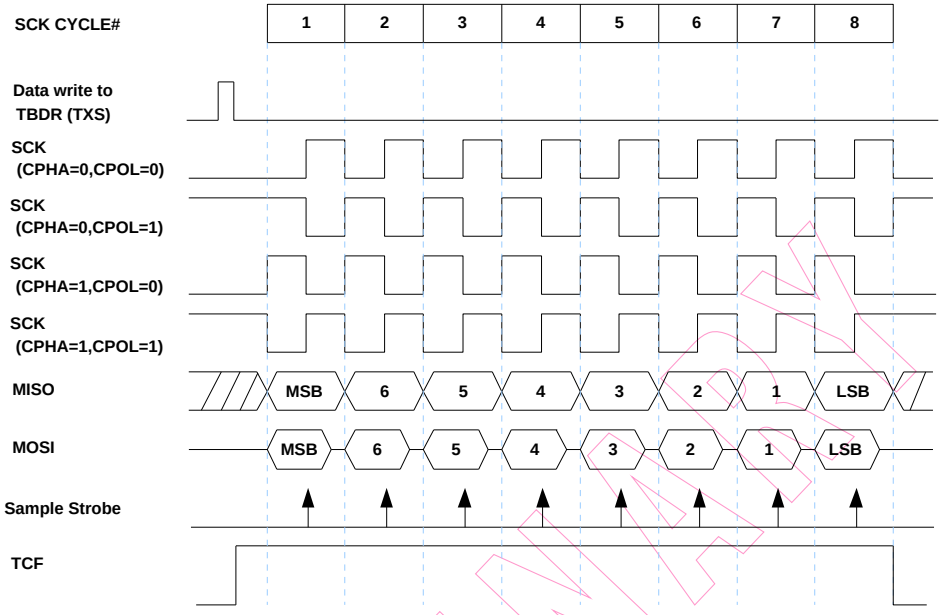


Figure 2 SPI Master Mode Timing Diagram

SPI Slave Mode Timing (CPHA = 1 / CHPA = 0)

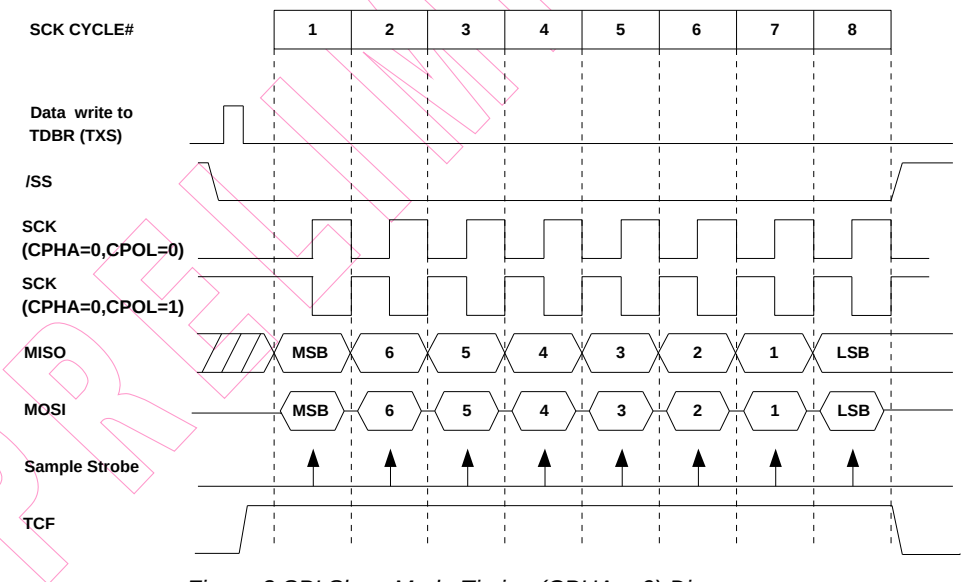


Figure 3 SPI Slave Mode Timing (CPHA = 0) Diagram

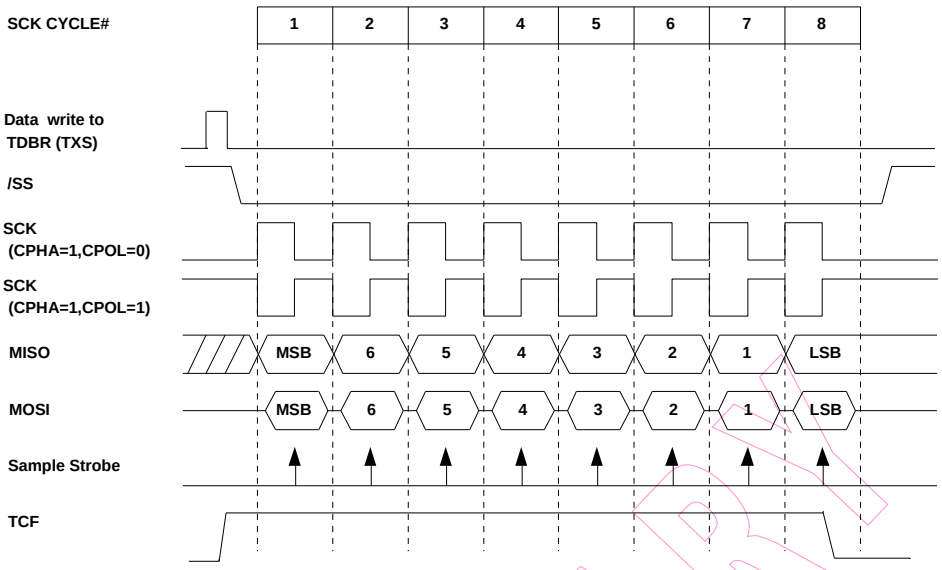


Figure 4 SPI Slave Mode Timing (CPHA = 1) Diagram

Consecutively Receiving Bytes (CPHA = 1 / CHPA = 0)

Figure 5 and Figure 6 are consecutive transmission charts, not real timing diagrams. TXS = 0 means that the buffer is empty, so TXS will be 0 after 8 cycles. "Write Byte 3" will be early before the ninth cycle.

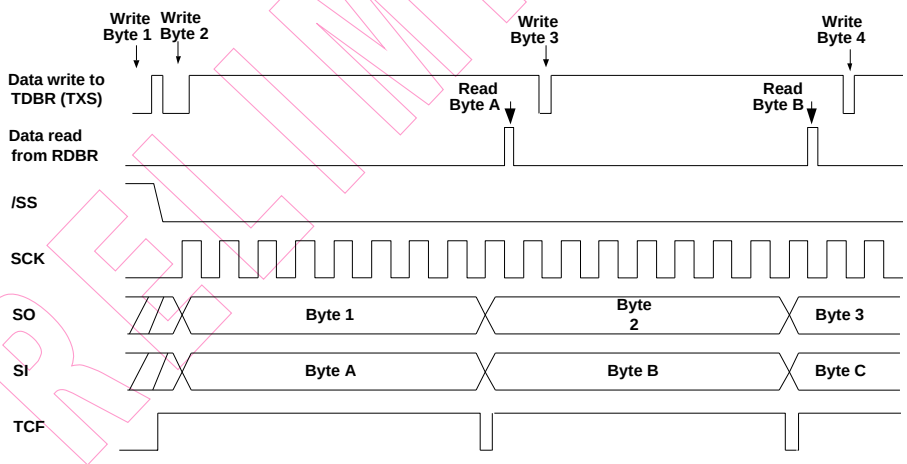


Figure 5 Consecutively Receiving Bytes Timing (Master or Slave Mode CPOL=0; CPHA=1) Diagram

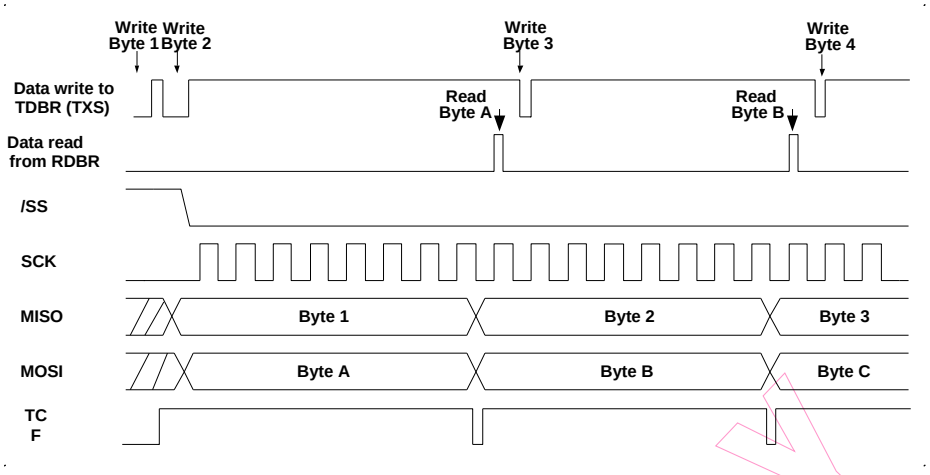


Figure 6 Consecutively Receiving Bytes Timing (Master or Slave Mode CPOL=0; CPHA=0) Diagram

SPI Slave Mode Timing Requirement

Figure 7 shows SPI slave timing requirement. It is necessary to have 10 or 5 system clock cycles between consecutive SCK transfers for single mode or buffer mode, respectively. 2 system cycles are needed for /SS (slave chip select) to be enabled and disabled. The setup and hold time of slave data input (MISO) should be 10ns to make sure master SCK and data timing at PCB design. There are six system cycles entering ISR (Interrupt Service Routine) after last SCK toggling. It needs 4 system cycles to be loaded into Tx shift registers after executing TDBR write instruction.

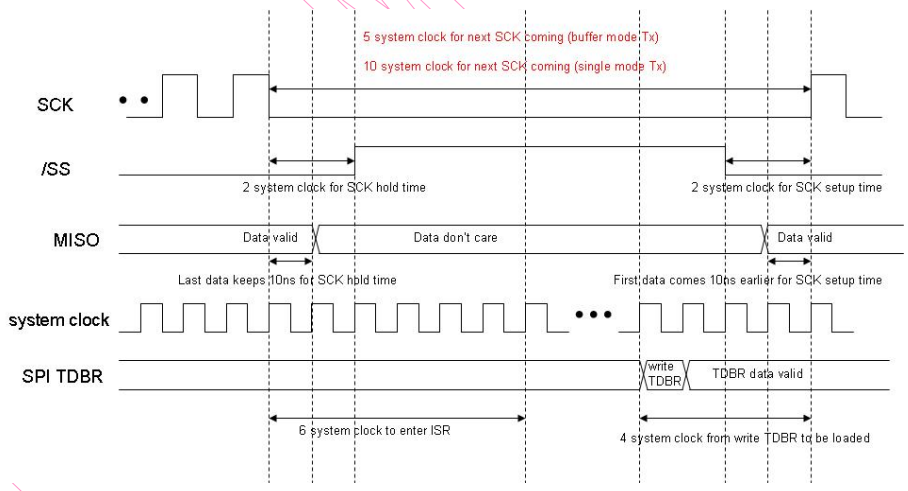


Figure 7 SPI slave timing requirement

In buffer mode, the minimum interval between consecutive transfers is 5 system clock cycles. It can be more than 5 cycles by setting TX_INTVAL when connecting with slower SPI slave devices. In single mode, the minimum interval between consecutive transfers depends on ISR (Interrupt Service Routine) coding. Shown in Figure 8, the interval can be minimal as 10 system cycles if writing TDBR first in ISR.

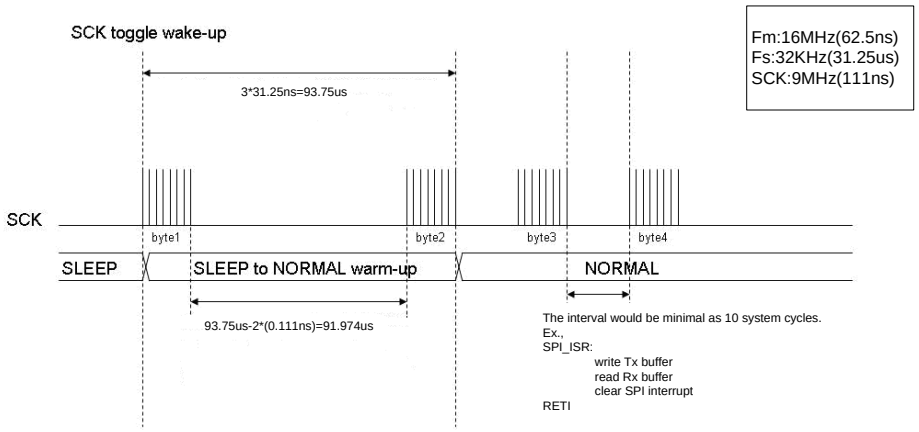


Figure 8 SPI single mode interval between consecutive transfers timing diagram 1

In Figure 8, the interval of SPI slave sleep wake-up transfers are described. If the system needs 3 warm-up cycles (31.25us period), the interval cannot be less than 91.974us to avoid byte1 being overwritten by byte2. The interval becomes so large due to longer system warm-up time after wake-up. It can be improved to use shorter system warm-up time (if it exists) or ignore first wake-up byte1 data as shown in .

17.7 Buffer Mode Function

The buffer mode is used to transmit and receive multiple data bytes in one interrupt to save repetitive ISR (Interrupt Sub-Routine) cycles of single mode. The setting flowchart is shown in Figure 9. Different lengths can be set for Tx and Rx FIFO. When an interrupt happens, the application will enter ISR and first check “RXF/TXF” bits of status register to decide whether it is a Rx or a Tx interrupt. Then the application can decide to read Rx FIFO data or fill new data into Tx FIFO.

For slave transmission condition, the TX FIFO must be written. Therefore, master can begin to toggle SCK to get slave Tx data even if slave Tx FIFO is not filled to the length “TX_BUFLEN”. It means that the internal Tx read pointer can be increased even if the Tx write pointer still doesn’t reach the “TX_BUFLEN” length setting. For slave receive condition, all Rx FIFO data must be read after entering Rx ISR. The error will happen as new Rx data is coming and the previous data in Rx FIFO is not read.

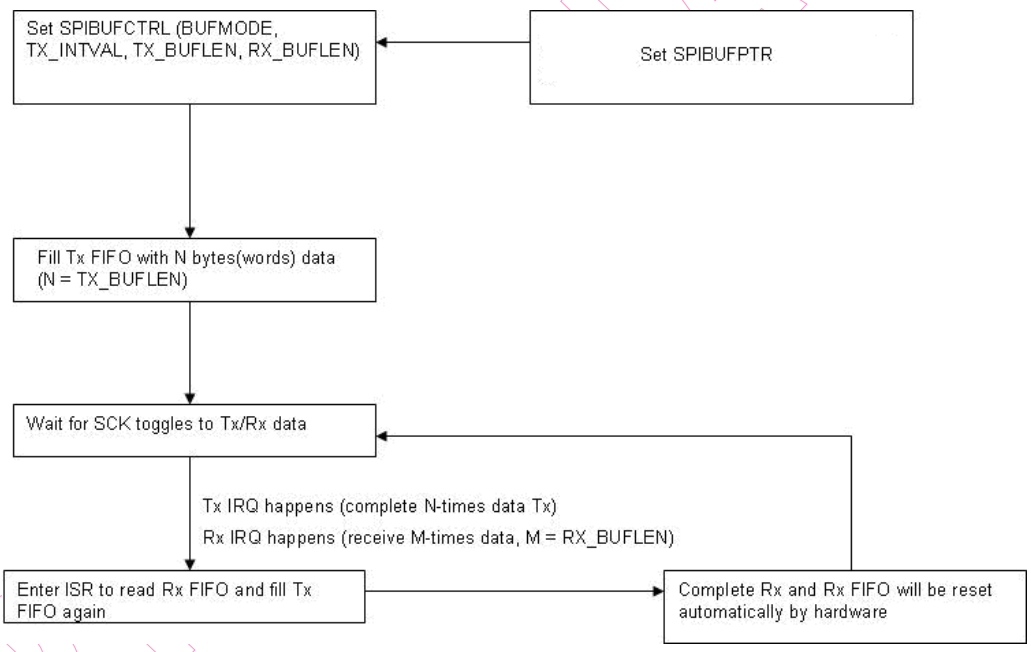


Figure 9 SPI buffer mode flow

In 8-bit mode, the data format in Tx/Rx FIFO is shown in Figure 10. Odd byte will be put in low-byte and even byte in high-byte. No matter the FIFO is read or written, the data will be processed automatically by hardware from or to the corresponding location. For 8-bit mode reading, the data byte is in low-byte of the word and the high-byte is filled with zero.

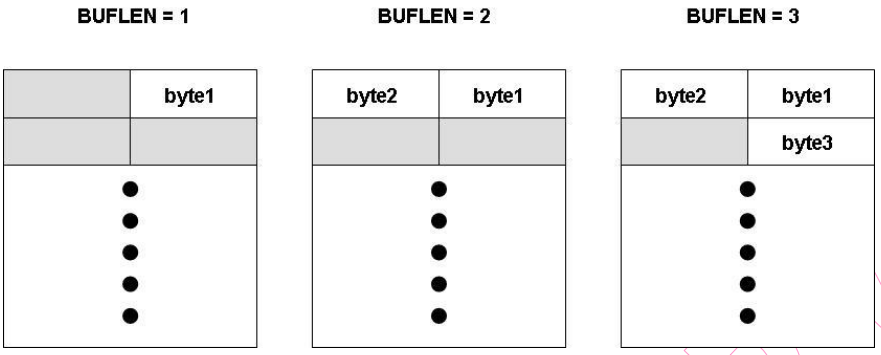


Figure 10 SPI buffer mode FIFO data format

SPI Master Initial Flow Chart

There are two data transfer modes under the SPI master device, namely, Empty and Complete modes. In Empty mode, data transfer process occurs continuously as long as the SPITDBR empty flag (TXS = 0) remains enabled. In Complete mode, data transfer is performed in batches, meaning that writing data to SPITDBR can be done only when the current batch of data being transferred is completed (TCF=0).

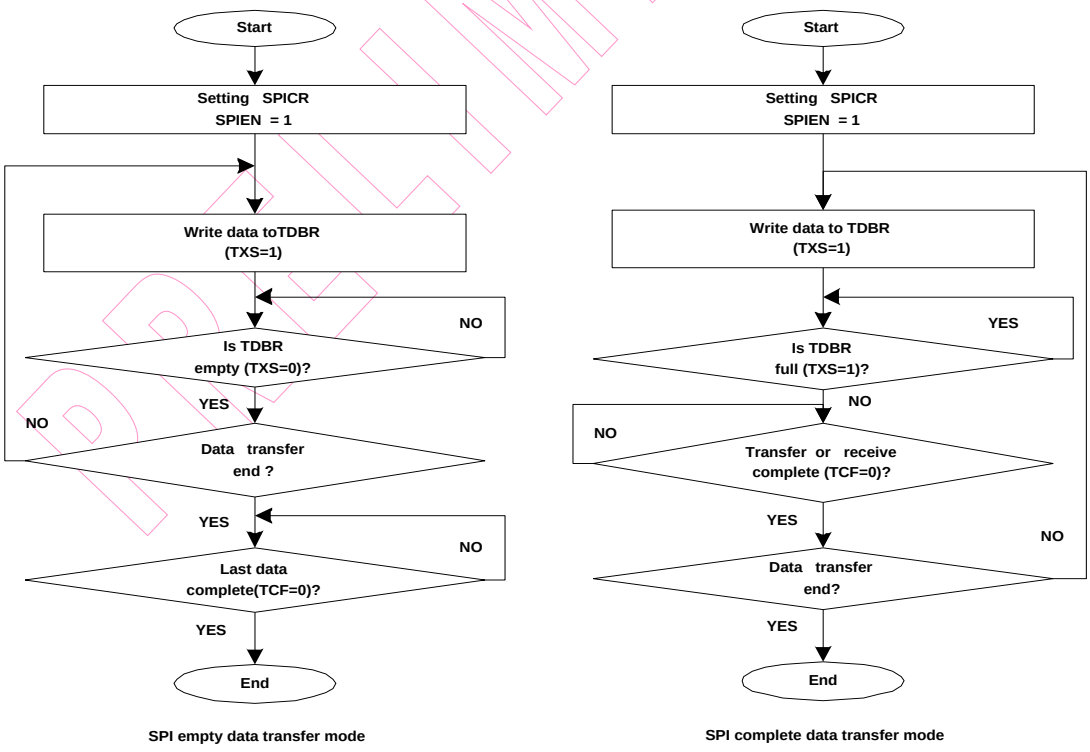


Figure 11 Complete Data Transfer Mode Flow Chart

SPI Application Note

1. To make the transmission operate normally, set the control register (SPICON) first.
2. To control the SPI Slave, the SPI Slave needs at least 4 system clock cycles to handle the protocol after writing SPITDBR, and /SS should be enabled (low-active) at that time. After that, the SPI Master can transmit SCK and MO (Master Out).
3. If the SPI Slave does not transmit data to the Master, 8'b0 should be written to make the transmission operate normally.
4. The SDOOD and SCKOD on SPICON Bit 1 and Bit 0 control the open drain function. These two bits are for output control, so SCKOD will be enabled in master mode when it needs high driving capability because the SCK pin is output in master mode. SDOOD is open drain control for SDO, no matter the SPI is a master or a slave.
5. For only-Rx-slave in single mode, there is no need to write SPITDBR before SCK is toggled by master.

17.8 Description of SPI Control Registers

SPIRDAFBC: Receive data almost full buffer mode control register.

Bit	7	6	5	4	3	2	1	0
Name	RXAFI_EN	-	-	RXBALEN4	RXBALEN3	RXBALEN 2	RXBALEN 1	RXBALEN 0
Type	R/W	R	R	R	R	R	R	R
Reset	0	0	0	0	1	1	1	1

SFR Address = 0xAC; SFR Page = 0x0

Register SPIRDAFBC is the receive data almost full buffer control register.

Bit 7: Tx/Rx Rx FIFO almost full interrupt enable bit

0: Disable. SPI interrupt flag set to 1 “only” when rx fifo data length is full.

1: Enable. SPI interrupt flag set to 1 when rx fifo data length match rx fifo almost full length.

Bit 6~5: Reserved. Read = 0, Write = Don't Care.

Bit 4~0: Rx buffer almost full length in buffer mode. The length is at most 16 bytes for the buffer mode FIFO. These bits can be set from 1 to 16 in decimal.

SPICON1: SPI Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	SPIEN	CPHA	CPOL	MSTR	SPR2	SPR1	SPR0	-
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBD; SFR Page = 0

Bit 7: SPI Enable/Disable.

0: Disable SPI.

1: Enable SPI.

The signal control GPIO share function. When this signal is set high, GPIO will switch to SPI mode automatically.

Bit 6: Clock Phase.

0: SCK toggle starts in the middle of first data bit.

1: SCK toggle starts in the beginning of first data bit.

Bit 5: Clock Polarity

0: SCK active-high

1: SCK active-low

CPHA	CPOL	Clock Scheme Description
0	0	The SPI transmits data one-half (½) cycle ahead of a rising edge of SCK and receives data on a rising edge of SCK.
0	1	The SPI transmits data one-half (½) cycle ahead of a falling edge of SCK and receives data on a falling edge of SCK.
1	0	The SPI transmits data on a rising edge of SCK and receives data on a falling edge of SCK.
1	1	The SPI transmits data on a falling edge of SCK and receives data on a rising edge of SCK.

Bit 4: Master / Slave Mode.

0: SPI is in master mode

1: SPI is in slave mode.

Bit 3~1: SPI Clock Rate Selection = CLK / Divisor (CLK: system clock).

SPR2	SPR1	SPR0	SPI Clock Rate
0	0	0	CLK/2
0	0	1	CLK/4
0	1	0	CLK/8
0	1	1	CLK/16
1	0	0	CLK/32
1	0	1	CLK/64
1	1	0	CLK/128
1	1	1	Reserved

3-bit SPR is used to set the bit transfer rate for a master device. When SPI is configured as a slave, the value written to SPR will be ignored. Note that if the SPI is configured as a slave, the system frequency must be at least eight times greater than (>8X) the master serial clock frequency.

Bit 0: Not used bit. Set to “0” at all time.

SPICON2: SPI Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	DORD	SPIHDEN	SDOOD	SCKOD
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBE; SFR Page = 0

Bit 7~4: Not used bits. Set to “0” at all time.

Bit 3: Data Shift-out Direction at Transmission.

0: MSB first

1: LSB first

Bit 2: The signal controlled by this bit can also increase the driving capability of the I/O pads.

0: Disable high current output

1: Enable high current output

Bit 1: This output enable or disable the open-drain attribute of the SDO pad (either MOSI or MISO)

0: Disable

1: Enable

Bit 0: This output enable or disable the open-drain attribute of the SCK pad.

0: Disable

1: Enable

SPISR1: SPI Status Register 1

Bit	7	6	5	4	3	2	1	0
Name	MB_START	RXF	TXE	RXEMY	SCK_SEL	DEGL_SEL	SDOC	-
Type	R/W	R/W	R/W	R	R/W	R/W	R/W	R
Reset	0	0	0	1	0	0	0	0

SFR Address = 0xBF; SFR Page = 0x0

Bit 7: Master buffer mode start control bit. The data in master Tx FIFO will be loaded into shift register only when this bit is set to 1.

Bit 6: Rx FIFO interrupt status bit for buffer mode only. (HW auto clear after SW read)

Bit 5: Tx FIFO interrupt status bit for buffer mode only. (HW auto clear after SW read)

Bit 4: Rx FIFO Empty status bit for buffer mode only. (HW auto clear after SW read)

Bit 3: SPI Slave SCK Selection. It can be set for slave SCK to pass through a RC delay cell to filter the glitch shorter than 15ns

0: Slave SCK doesn't pass through de-glitch delay cell.

1: Slave SCK passes through de-glitch delay cell.

Bit 2: SPI slave SCK/SDI de-glitch select. It's effective only when SCK_SEL is set to 1

0: SCK/SDI will be delayed 15ns by de-glitch cell.

1: SCK/SDI will be delayed 30ns by de-glitch cell.

Bit 1: SPI SDO default state

0: After finishing serial data output, the SDO output (either spi_so or spi_mo) is kept at Logic 1

1: After finishing serial data output, the SDO output (either spi_so or spi_mo) is kept at Logic 0

Bit 0: Not used bit. Set to “0” at all time.

SPISR2: SPI Status Register 2

Bit	7	6	5	4	3	2	1	0
Name	SPIIF	WKEN	RXAF	RCF	SRO	TXS	RBF	TCF
Type	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
Reset	0	1	0	0	0	0	0	0

SFR Address = 0XC1; SFR Page = 0x0

Bit 7: SPI interrupt flag. Set by Hardware, cleared by Software.

Bit 6: SPI Wake-up enable.

0: Disable

1: If this bit is set and SPI is enabled, SPI can be waked-up from IDLE/SLEEP modes by SCK toggling. The application FW can turn off this bit to prevent unnecessary wake-up when buffer mode.

Bit 5: Rx FIFO almost full status bit for buffer mode only

0: rx fifo data length is less than rx almost full buffer length

1: rx fifo data length is greater or equal to rx almost full buffer length

Bit 4: Receiving progress flag (Slave mode only)

0: The SPI slave is not busy for receiving the data.

1: The SPI slave is busy for receiving the data, MCU should not go to sleep during the flag is high.

Bit 3: Buffer overrun indicator (Slave mode only)

0: No buffer overrun.

1: A new byte of data is received while the previous one is still held at register SPIRDBR . Note that the previous byte of data will be destroyed while this condition occurs.

Bit 2: SPITDBR status flag

0: SPITDBR status flag.

1: SPITDBR is full

The transmit buffer becomes full (TXS = 1) after it is written into. It becomes empty (TXS = 0) when data transfer begins and the transmitted value is loaded into the Shift register (H/W set; H/W cleared). It's kept logic-1 only 1 or 2 system clock cycles after writing Tx buffer. It will become logic-0 when data is moved by HW from Tx buffer to Tx shift register (TCF = 1).

Note

In master mode, Tx buffer can be written when TXS is 0. However, in slave mode, Tx buffer can't be written when TXS is 0 and TCF is 1. The data in Tx buffer will be overwritten. Slave 2nd TDBR data can be written when RCF is 1.

Bit 1: Read Buffer Full Flag.

0: Receiving is not completed, and SPIRDBR has not fully changed.

1: Receiving is completed, and SPIRDBR has fully changed (H/W set; H/W & S/W cleared).

This bit is set by HW and cleared by SW or HW (clear automatically after read). So it will be cleared automatically after reading Rx buffer (SPIRDBR) by CPU or ICE. Therefore, RBF is always 0 in emulation environment.

Bit 0: Transmitting progress flag. This bit is both set and cleared by HW

0: Tx transfer is not proceeding

1: Tx transfer is proceeding

The SPI hardware clears this bit to indicate that it has completed sending and is ready for the next task. This flag causes an interrupt to be requested if the SPI interrupt is enabled.

SPITDBR: SPI Transmit Data Buffer Register

Bit	7	6	5	4	3	2	1	0
Name	SPITD.7	SPITD.6	SPITD.5	SPITD.4	SPITD.3	SPITD.2	SPITD.1	SPITD.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC2; SFR Page = 0x0

Bits 7~0: SPI Tx buffer

The Transmit Data Buffer Register (SPITDBR) is a readable and writeable register. Data is loaded into this register before being transmitted. Just prior to the beginning of a data transfer, the data in SPITDBR is loaded into the Shift Data (SPISFDR) Register.

If multiple writes to SPITDBR occur while a data transfer is in progress, only the last written data will be transmitted. None of the intermediate values written to SPITDBR will be transmitted. Multiple writes to SPITDBR is possible but not recommended.

SPIRDBR: SPI Receive Data Buffer Register

Bit	7	6	5	4	3	2	1	0
Name	SPIRD.7	SPIRD.6	SPIRD.5	SPIRD.4	SPIRD.3	SPIRD.2	SPIRD.1	SPIRD.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XC3; SFR Page = 0x0

Bits 7~0: SPI Rx buffer

The Receive Data Buffer Register (SPIRDBR) is a 8-bit read-only (RO) register. In the end of a data transfer, the data in the shift register is loaded into SPIRDBR.

SPITDBC: SPI transmit data control register

Bit	7	6	5	4	3	2	1	0
Name	TX_INTVAL	-	-	TXBLEN4	TXBLEN3	TXBLEN2	TXBLEN1	TXBLEN0
Type	R/W	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	1	0	0	0	0

SFR Address = 0XC4; SFR Page = 0x0

Bit 7: The system cycle interval between two master Tx transaction. The interval is defined from the last serial SCK toggling of current transfer to the first SCK toggling of next transfer.

0: No system cycles between two Tx transfer. (continuous)

1: 4 system cycles between two Tx transfer.

Bit 6~5: Not used bit. Set to “0” at all time.

Bit 4~0: Tx buffer length in buffer mode. The length is at most 16 bytes for the buffer mode FIFO.

This can be set from 2 to 16 in decimal.

SPIRDBC: SPI receive data buffer control register.

Bit	7	6	5	4	3	2	1	0
Name	BUFMODE	-	-	RXBLEN4	RXBLEN3	RXBLEN2	RXBLEN1	RXBLEN0
Type	R/W	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	1	0	0	0	0

SFR Address = 0xC5; SFR Page = 0x0

Bit 7: SPI buffer mode enable. Register SPIBUFPTR1 is the FIFO counter (pointer) register for buffer mode.

0: non-buffer mode. (single mode)

1: buffer mode.

Bit 6~5: Not used bit. Set to “0” at all time.

Bit 4~0: Rx buffer length in buffer mode. The length is at most 16 bytes for the buffer mode FIFO.

This can be set from 2 to 16 in decimal.

SPIBUFPTR1: FIFO counter (pointer) register 1 for buffer mode.

Bit	7	6	5	4	3	2	1	0
Name	PTRSEL	-	-	RP4	RP3	RP2	RP1	RP0
Type	R/W	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC6; SFR Page = 0x0

Register SPIRBUFPTR is the FIFO counter (pointer) register for buffer mode.

Bit 7: FIFO COUNTER (Pointer) selection for counter (pointer)

0: Tx counter (pointer) can be read in counter (pointer) register.

1: Rx counter (pointer) can be read in counter (pointer) register.

Bit 6~5: Not used bit. Set to “0” at all time.

Bit 4~0: Tx or Rx FIFO read counter (pointer) value.

PTR_SEL=0: Tx FIFO read counter (pointer) value.

PTR_SEL=1: Rx FIFO read counter (pointer) value.

SPIBUFPTR2: FIFO counter (pointer) register 2 for buffer mode.

Bit	7	6	5	4	3	2	1	0
Name	PTRRST	-	-	WP4	WP3	WP2	WP1	WP0
Type	R/W	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC7; SFR Page = 0x0

Register SPIWBUFPTR is the FIFO counter (pointer) register for buffer mode.

Bit 7: Tx/Rx FIFO counter (pointer) reset

0: not reset

1: reset. (hardware)

Bit 6~5: Not used bit. Set to “0” at all time.

Bit 4~0: Tx or Rx FIFO write counter (pointer) value.

PTR_SEL=0: Tx FIFO write counter (pointer) value.

PTR_SEL=1: Rx FIFO write counter (pointer) value.

18 LED Driver

The LED driver contains a controller, a duty cycle generator with 1-8 Common signal pins, and 8 Segment driver pins. Segment 1-8 can also be used as I/O port. When LEDEN bit is set, LED function is enabled. It is controlled by LEDFR, LCDCR1, EIOCOM1, EIOSEG1, EIOSEG11, LCDADDR, and LCDDATA register.

When MCU enters the Power-Down mode, the LED will be turned off.

Fs=16K Hz, Toff= 62.5us

Fs=32K, Toff=31.25us

Fs=128K, Toff =31.25us

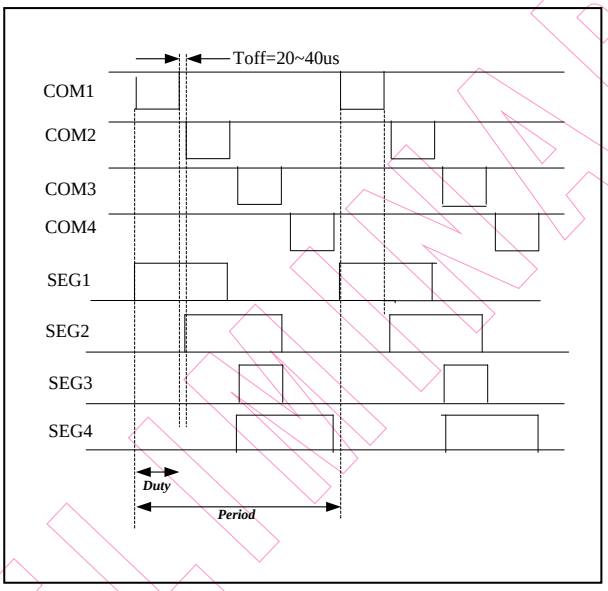


Figure 18-a 1LED Driver

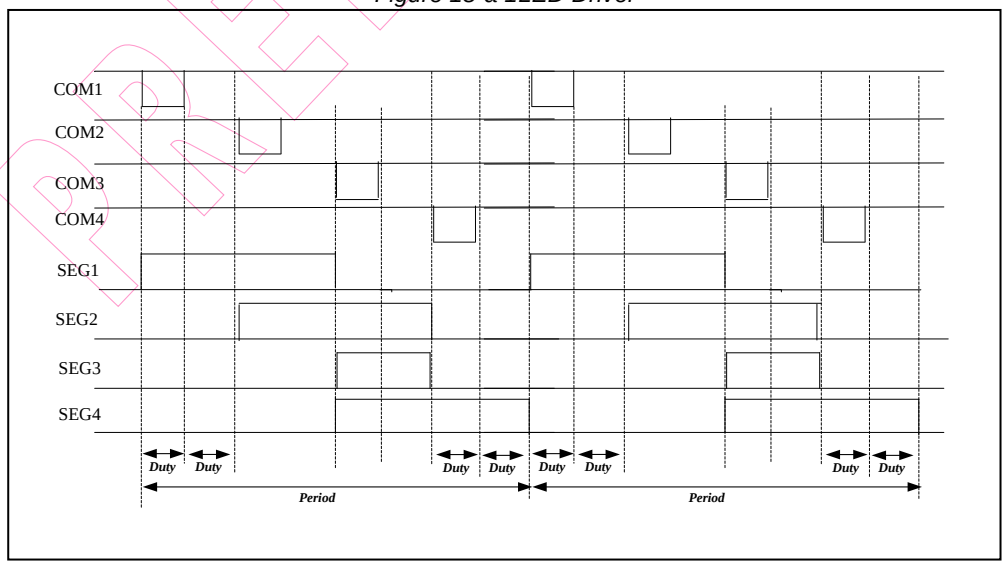


Figure 18-b 1LED Driver

18.1 Description of LED control registers

LEDFR: LED frame rate select bits

Bit	7	6	5	4	3	2	1	0
Name	-	LEDPR.2	LEDPR.1	LEDPR.0	LEDDT.3	LEDDT.2	LEDDT.1	LEDDT.0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XA4; SFR Page = 0x1

Bits 7 Reserved. Read = 0, Write = Don't Care.

Bits 6~4 (LEDPR2~0): LED Period time.

LEDPR[2:0]	Description
000	(LED Duty Time) *(LEDCOM(max)) *1
001	(LED Duty Time) *(LEDCOM(max)) *2
010	(LED Duty Time) *(LEDCOM(max)) *4
011	(LED Duty Time) *(LEDCOM(max)) *6
100	(LED Duty Time) *(LEDCOM(max)) *8
101	(LED Duty Time) *(LEDCOM(max)) *12
110	(LED Duty Time) *(LEDCOM(max)) *14
111	(LED Duty Time) *(LEDCOM(max)) *16

i.e.:

LEDSS = 32KHz

LED Duty Time select =64/FLS = 2ms+delay

LED Duty =1/4 duty

If LEDPR =1 => LED Period time = 2ms*4*1 = 8ms

If LEDPR =16 => LED Period time = 2ms*4*16 = 128ms

Bits 3~0 (LEDDT3~0): LED Duty time.

LEDDT[3:0]	Description
0000	$1/F_{LS}$
0001	$2/F_{LS}$
0010	$4/F_{LS}$
0011	$8/F_{LS}$
0100	$16/F_{LS}$
0101	$32/F_{LS}$
0110	$64/F_{LS}$
0111	$128/F_{LS}$
1xxx	$256/F_{LS}$

LEDCR1: LED Driver Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	LEDEN	-	-	-	-	-	-	-
Type	R/W	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XA9; SFR Page = 0x1

Bit 7 (LEDEN): LED enable bit

0: Disable LED driver. COM/SEG initial pull-low during Power-on

1: Enable LED driver (LED Enable)

Bits 6~0: Not used, set to "0" all the time.

LEDADDR: Address of LED RAM

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	LEDA2	LEDA1	LEDA0
Type	R	R	R	R	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA1; SFR Page = 0x1

Bits 7~3: Not used, set to "0" all the time.

Bits 2~0 (LEDA4~0): Address of LED RAM. This register is used for 00H~08H correspond to SEG0~SEG7.

LEDDATA: Data of LCD RAM

Bit	7	6	5	4	3	2	1	0
Name	LEDD7	LEDD6	LEDD5	LEDD4	LEDD3	LEDD2	LEDD1	LEDD0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA2; SFR Page = 0x1

Bit 7 ~ 0 (LEDD7~0): LED RAM data register

The following table shows the organization of LED RAM and the relation between data in RAM and LED signal.

Address of LCD RAM	Data of LED RAM								SEG
	Bit 7 (LEDD7)	Bit 6 (LEDD6)	Bit 5 (LEDD5)	Bit 4 (LEDD4)	Bit 3 (LEDD3)	Bit 2 (LEDD2)	Bit 1 (LEDD1)	Bit 0 (LEDD0)	
0x00h	-	-	-	-	-	-	-	-	SEG0
0x01h	-	-	-	-	-	-	-	-	SEG1
0x02h	-	-	-	-	-	-	-	-	SEG2
|	|	|	|	|	|	|	|	|	|
0x07h	-	-	-	-	-	-	-	-	SEG7
COM	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0	

EIOCOM1: External IO Pins Function Select for LED COM Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0XC1; SFR Page = 0x2

Bits 7~0 (COM7~0): LED Pin Switch for COM7~COM0

0: Function as normal I/O or other functions (default)

1: Function as LED common pins

EIOSEG1: External IO Pins Function Select for LED SEG Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	SEG4	SEG3	SEG2	SEG1	SEG0
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address =0XC2; SFR Page = 0x2

Bits 7~5: Not used, set to "0" all the time.

Bits 4~0 (SEG4~0): LED Pin Switch for SEG5~SEG0

0: Function as normal I/O or other functions (default)

1: Function as LED segment pins

EIOSEG11: External IO Pins Function Select for LED SEG1 Control Register 11

Bit	7	6	5	4	3	2	1	0
Name	SEG7_1	SEG6_1	SEG5_1	-	-	-	-	-
Type	R/W	R/W	R/W	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address =0xC6; SFR Page = 0x2

Bits 7~5 (SEG7_1~SEG5_1): LED Pin Switch for LEDS7_1_LEDS5_1(SEG7_1~SEG5_1)

0: Function as normal I/O or other functions (default)

1: Function as LED segment pins

Bits 4~0: Not used, set to "0" all the time.

19 PWM

In PWM mode, it produces up to 16-bit resolution PWM output (see. the functional block diagram). A PWM output consists of a time period and a duty cycle, and it keeps the output high. The baud rate of the PWM is the inverse of the time period. Figure 25~28 (PWM Output Timing) depict the relationships between a time period and a duty cycle.

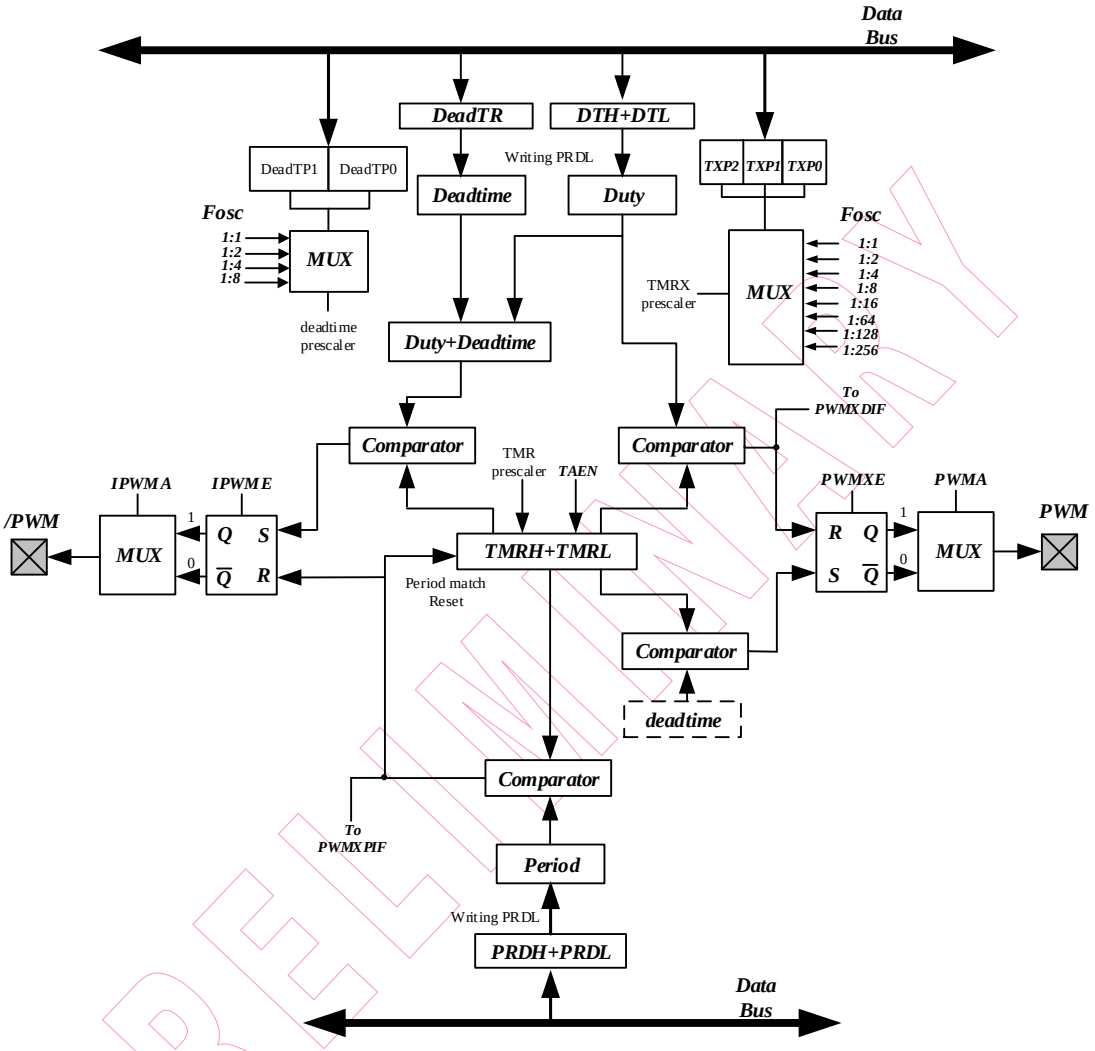


Figure 19- 2 The PWM Functional Block Diagram

PWM and /PWM (inverted PWM) can be used individually or used as dual PWM. When used individually, the definitions of active level between PWM and /PWM are somewhat different.

For example, set period and duty cycle (period > duty), PWME=1/0 and IPWME=0/1, PWMA = 1/0, IPWMA=1/0, and finally set TAEN = 1. The following figures show PWM output timing according to different PWMA and IPWMA settings.

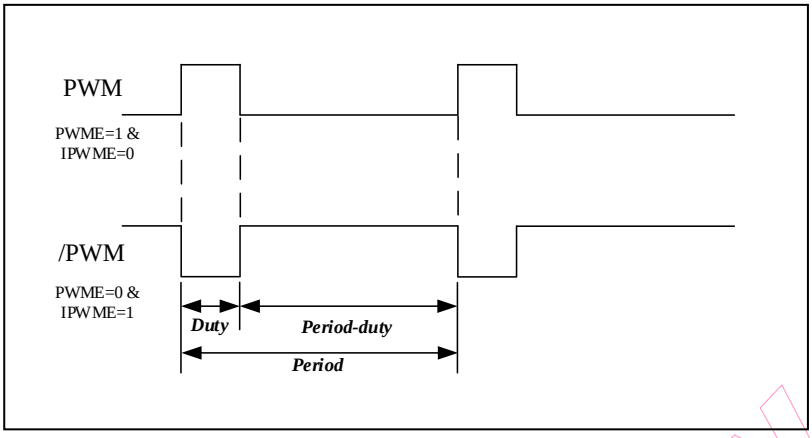


Figure 19- 3PWM Output Timing (PWMA=0 and IPWMA=0)

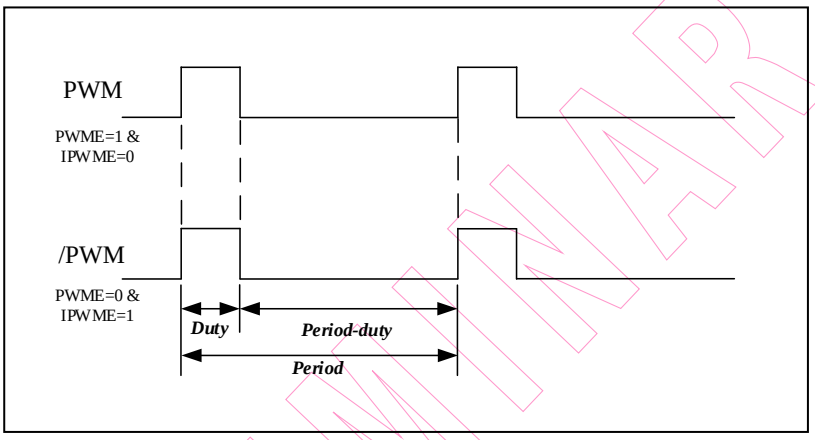


Figure 19- 4PWM Output Timing (PWMA=0 and IPWMA=1)

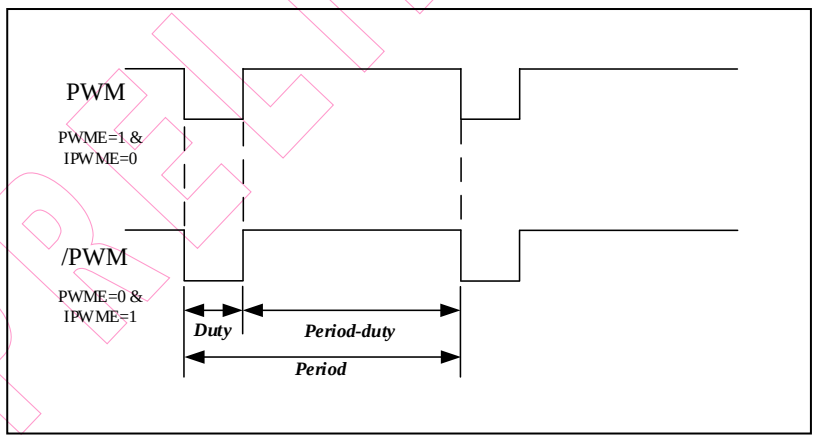


Figure 19- 5PWM Output Timing (PWMA=1 and IPWMA=0)

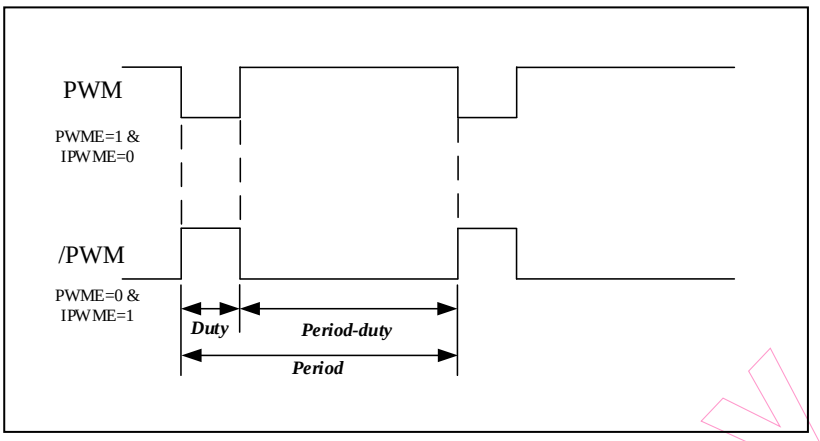


Figure 19- 6PWM Output Timing (PWMA=1 and IPWMA=1)

For shutting off the operating PWM function, please refer to Figure ?.

19.1 Increment Timer Counter (TMRA: TMRAH/TMRAL)

TMRA are 16-bit clock counters with programmable prescalers. It's designed for the PWM module as baud rate clock generators. TMR can be read only. If employed, they can be turned off for power saving by setting the TAEN bit to 0.

TMRA is internal designs and its value cannot be set by user directly.

19.2 PWM Time Period (PRD: PRDAL/H)

The PWM period is 16-bit resolution. The PWM time period is defined by writing to the PRD register. When the value of TMRA is equal to PRD, the following events occur on the next increment cycle:

- TMRA is cleared
- The PWM pin is set to 1

NOTE
The PWM output will not be set, if the duty cycle is 0

- The PWMPSF flag is set to 1

The following formula describes how to calculate the PWM time period:

$$Period = (PRDX+1) \times \left(\frac{1}{F_{OSC}}\right) \times (TMRA\ prescale\ value)$$

Example:

PRDX = 49; Fosc = 4 MHz; TMR prescaler (0, 0, 0) = 1 : 1,

Then

$$Period = (49+1) \times \left(\frac{1}{4M}\right) \times 1 = 12.5\mu s$$

19.3 PWM Duty Cycle (DT: DTAH/DTAL)

The PWM duty cycle is defined by writing to the DT register, and is latched from DT to DLX while TMRX is cleared. When DLX is equal to TMR, the PWM pin is cleared. DT can be loaded anytime. However, it cannot be latched into DLX until the current value of DLX is equal to TMR.

The following formula describes how to calculate the PWM duty cycle:

$$Duty\ cycle\ =\ (DT)\ \times\ \left(\frac{1}{F_{OSC}}\right)\ \times\ (TMRA\ prescale\ value)$$

Example:

DTX = 10; Fosc = 4 MHz; TMR prescaler (0, 0, 0) = 1 : 1,

Then

$$Duty\ cycle\ =\ (10)\ \times\ \left(\frac{1}{4M}\right)\ \times\ 1\ =\ 2.5\mu s$$

19.4 Dual PWM function

It consists of a complementary PWM (i.e. PWM and /PWM), one outputs PWM signal and the other outputs inverted PWM signal. It can output any pulse width signal user want by programming relative control registers.

The dead time mode is supported. It means that the complementary PWM signals can be controlled to get a time interval that the complementary PWM signals won't be intersected.

The following Figures 27 ~ 29 illustrate the dual PWM output waveform.

Disable dead time control (DEADTE = 0). Set period and duty cycle (period > duty). Set PWME & IPWME =1, PWMA = 0/1, IPWMA = 0/1, and finally set TAEN = 1.

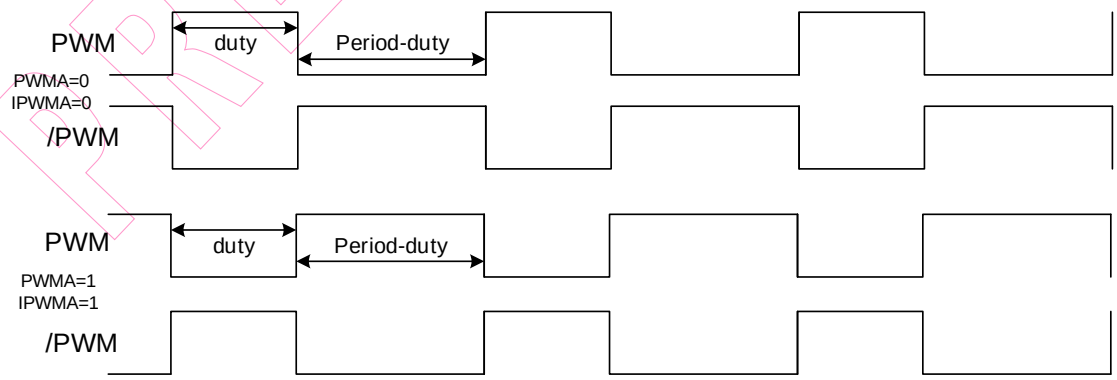


Figure 19- 7Dual PWM output waveform (DEADTE = 0)

Set dead time > 0 (set dead time prescaler if required). Enable dead time control (DEADTE = 1). Set period and duty cycle (period > duty). Set PWME & IPWME =1, PWMA = 0, IPWMA = 0, and finally set TAEN = 1. For the loading new duty, period, and deadtime value at run time, following subchapter "PWM Programming Process/Steps" makes such descriptions.

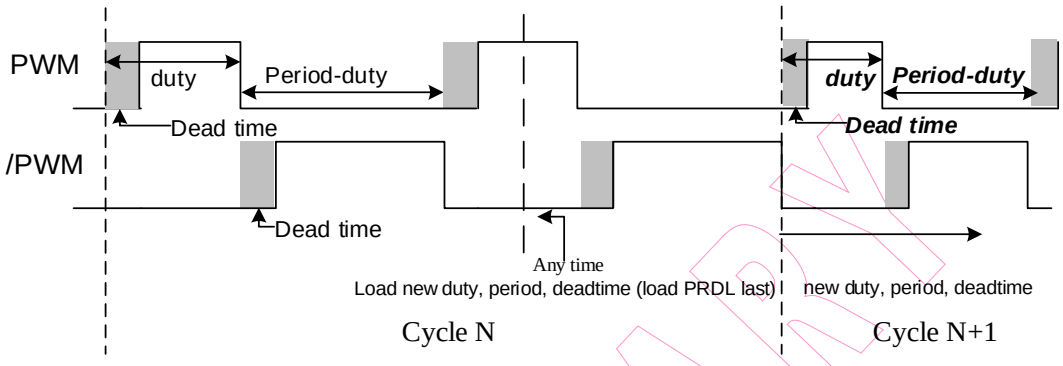


Figure 19- 8Dual PWM output waveform (DEADTE = 1, Dead Time > 0)

User can make use of the falling edge of comparator's output to close dual PWM or single PWM function by setting SDPWMX=1. The following figure illustrates how to shut down dual PWM.

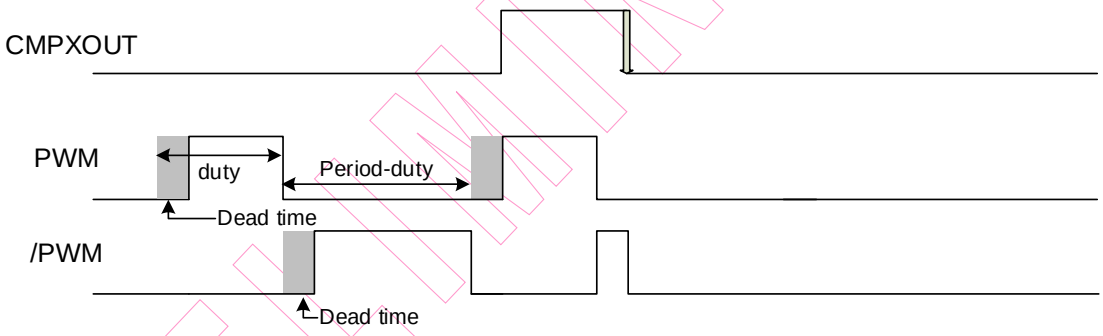


Figure 19- 9Dual PWM output waveform (DEADTE = 1, Dead Time ≧ 0, SDDPWM = 1, SDDPWMedge =1)

19.5 Comparator X

Changing the output status while matching occurs will simultaneously set the TMRXIF flag.

19.6 PWM Programming Process/Steps

1. Load the PWM duty cycle to DT.
2. Load the PWM dead-time cycle (only for dual PWM function).
3. Load the PWM time period to PRDA.
4. Enable the interrupt function by writing PWMIE=1 (EIE2.0), if required.
5. Load a desired value for the timer pre-scaler.

6. Set active level of duty of PWM.
7. Enable PWM function, i.e., enable PWME control bit. (If using dual PWM function, enable IPWME control bit too)
8. Finally, enable TMRA function, i.e., enable TAEN control bit.

If the application needs to change PWM duty, period and dead-time cycle at run time, refer to the following programming steps:

1. Load new duty and dead-time cycle (if using dual PWM function) at any time.
2. Load new period cycle. The order of loading period cycle must be taken care. As the low byte of PWM period cycle is assigned a value, the new PWM cycle is loaded into circuit.
3. The circuit would automatically update the new duty, period, and dead-time cycle to generate new PWM waveform at next PWM cycle.

19.7 Description of PWM control registers

DeadTR: Dead Time Register

Bit	7	6	5	4	3	2	1	0
Name	DEADTR.7	DEADTR.6	DEADTR.5	DEADTR.4	DEADTR.3	DEADTR.2	DEADTR.1	DEADTR.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC9; SFR Page = 0x1

Bits 7~0: The value of register is used as dead-time duration setting.

PWMENCR: PWM Enable Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	DeadTP1	DeadTP0	-	TCEN	TBEN	TAEN
Type	R	R	R/W	R/W	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCA; SFR Page = 0x1

Bits 7~6, 3~1: Reserved. Read = 0, Write = Don't Care.

Bits 5~4 (DeadTP1~DeadTP0): Dead-time pre-scaler

DeadTP1	DeadTP0	Prescale
0	0	1:1 (default)
0	1	1:2
1	0	1:4
1	1	1:8

Bits 3: Reserved. Read = 0, Write = Don't Care.

Bit 2 (TCEN): TMRC enable bit. All PWM functions are valid only as this bit is set

0 = TMRC is off (default)

1 = TMRC is on

Bit 1 (TBEN): TMRB enable bit. All PWM functions are valid only as this bit is set

0 = TMRB is off (default)

1 = TMRB is on

Bit 0 (TAEN): TMRA enable bit. All PWM functions are valid only as this bit is set

0 = TMRA is off (default)

1 = TMRA is on

PWMACR1: PWMA Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	PWMAE	IPWMAE	PWMAA	IPWMAA	-	TAP2	TAP1	TAP0
Type	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XD1; SFR Page = 0x1

Bit 7 (PWMAE): PWMA enable bit

0: Disable (default)

1: Enable. The compound pin is used as PWM pin. Note when using PWM function, do not enable other function, otherwise it will lead to errors.

Bit 6 (IPWMAE): Inverse PWMA enable bit

0: Disable (default)

1: Enable. The compound pin is used as /PWM pin. Note when using /PWM function, do not enable other function, otherwise it will lead to errors.

Bit 5 (PWMAA): Active level of PWMA

0: Duty duration is logic high (default)

1: Duty duration is logic low

Bit 4 (IPWMAA): Active level of inverse PWMA

0: Period-Duty duration is logic high (default)

1: Period-Duty duration is logic low

Bits 3: Reserved. Read = 0, Write = Don't Care.

Bits 2~0 (TAP2~TAP0): Timer A clock pre-scale option bits

TAP2	TAP1	TAP0	Prescale
0	0	0	1:1 (default)
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

PWMAE	TAEN	Function description
0	0	Not used as PWM function; I/O pin or other functional pin.
0	1	Timer function; I/O pin or other function pin.
1	0	PWM function, the waveform keeps at inactive level.
1	1	PWM function, the normal PWM output waveform.

PWMACR2: PWMA Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	PWMAPSF	PWMADSF	PWMACMPLC	PWMASWAP	PWMAOSM	PWMAS1	PWMAS0	DeadTAE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCB; SFR Page = 0x1

Bit 7 (PWMAPSF): Status flag of period-matching for PWMA (Pulse Width Modulation). Set when a selected period is reached, reset by software.

Bit 6 (PWMADSF): Status flag of duty-matching for PWMA (Pulse Width Modulation). Set when a selected duty is reached, reset by software.

Bit 5 (PWMCMPLC): The enable bit of the link control between PWM and CMP.

0: Disable

1: Enable. Before PWM function is active, the CMP must be enabled first. If not, the PWM function would be not active.

The enable order of PWM and CMP: Turn CMP on first and then turn PWM on.

The disable order of PWM and CMP: Turn PWM off first and then turn CMP off.

If CMP is turned off but PWM is turned on, then PWM is also turned off.

The relationship between CMP and PWM is shown in the table below: (PWMEN = 1)

PWMACMPLC	CMPxEn	SC1PWMAEN	TAEN	CMP Function	PWM Function
0	0	x	0	Off	Off
0	0	x	1	Off	On
0	1	x	0	On	Off
0	1	x	1	On	On
1	0	0	0	Off	Off
1	0	0	1	Off	On
1	1	0	0	On	Off
1	1	0	1	On	On
1	0	1	0	Off	Off
1	0	1	1	Off	Off
1	1	1	0	On	Off
1	1	1	1	On	On

Bit 4 (PWMASWAP): To select whether DPWM output is swapped or not.

0: Outputs DPWM waveform is not swapped.

1: Outputs DPWM waveform is swapped.

Bit 3 (PWMAOSM): One-shot mode of PWM. This bit is set by SW and cleared by HW. This bit is valid in DPWM trigger mode. TAEN needs be enabled first to use one-shot mode of PWM.

0: Disable (default)

1: Enable

Bits 2~1 (PWMAS1~PWMAS0): Clock selection for PWMA timer

PWMAS1	PWMAS0	Clock selection for PWMA timer
0	0	F_{LS}
0	1	F_{HS}
1	0	PLL (48MHz)
1	1	NA(PLL (48MHz))

Note

When using crystal oscillator as clock source, user needs to set corresponding frequency divider to obtain 1MHz clock into PLL circuit. The configuration of frequency divider is at code option word

Bit 0 (DeadTAE): Dead-time function enable bit for dual PWM A

0: Disable (default)

1: Enable

PRDAL: Low Byte of PWMA Period

Bit	7	6	5	4	3	2	1	0
Name	PRDA.7	PRDA.6	PRDA.5	PRDA.4	PRDA.3	PRDA.2	PRDA.1	PRDA.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD2; SFR Page = 0x1

Bits 7~0 (PRDA.7~PRDA.0): The contents of the register are low byte of the PWM period

PRDAH: High Byte of PWM Period

Bit	7	6	5	4	3	2	1	0
Name	PRDA.15	PRDA.14	PRDA.13	PRDA.12	PRDA.11	PRDA.10	PRDA.9	PRDA.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD3; SFR Page = 0x1

Bits 7~0 (PRDA.15~PRDA.8): The contents of the register are high byte of the PWM period

DTAL: Low Byte of PWM Duty

Bit	7	6	5	4	3	2	1	0
Name	DTA.7	DTA.6	DTA.5	DTA.4	DTA.3	DTA.2	DTA.1	DTA.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD4; SFR Page = 0x1

Bits 7~0 (DTA.7~DTA.0): The contents of the register are low byte of the PWM duty

DTAH: High Byte of PWM Duty

Bit	7	6	5	4	3	2	1	0
Name	DTA.15	DTA.14	DTA.13	DTA.12	DTA.11	DTA.10	DTA.9	DTA.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD5; SFR Page = 0x1

Bits 7~0 (DTA.15~DTA.8): The contents of the register are high byte of the PWM duty

TMRAL: Low Byte of Timer A

Bit	7	6	5	4	3	2	1	0
Name	TMRA.7	TMRA.6	TMRA.5	TMRA.4	TMRA.3	TMRA.2	TMRA.1	TMRA.0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD6; SFR Page = 0x1

Bits 7~0 (TMRA.7~TMRA.0): The contents of the register are low byte of the PWM timer which is counting. This is read-only.

TMRAH: High Byte of Timer A

Bit	7	6	5	4	3	2	1	0
Name	TMRA.15	TMRA.14	TMRA.13	TMRA.12	TMRA.11	TMRA.10	TMRA.9	TMRA.8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD7; SFR Page = 0x1

Bits 7~0 (TMRA.15~TMRA.8): The contents of the register are high byte of the PWM timer which is counting. This is read-only.

PWMB

PWMBCR1: PWMB Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	PWMBE	IPWMBE	PWMBA	IPWMBA	-	TBP2	TBP1	TAP0
Type	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD9; SFR Page = 0x1

Bit 7 (PWMBE): PWMA enable bit

0: Disable (default)

1: Enable. The compound pin is used as PWM pin. Please note that when using PWM function, do not enable other functions, otherwise it will lead to defaults.

Bit 6 (IPWMBE): Inverse PWMB enable bit

0: Disable (default)

1: Enable. The compound pin is used as /PWM pin. Please note that when using /PWM function, do not enable other functions, otherwise it will lead to defaults.

Bit 5 (PWMBA): Active level of PWMB

0: Duty duration is logic high (default)

1: Duty duration is logic low

Bit 4 (IPWMBA): Active level of inverse PWMB

0: Period-Duty duration is logic high (default)

1: Period-Duty duration is logic low

Bit 3: Not used bits. Fixed to “0” all the time.

Bits 2~0 (TBP2~TBP0): Timer B clock pre-scale option bits

TBP2	TBP1	TBP0	Prescale
0	0	0	1:1 (default)
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

PWMBE	TBEN	Function description
0	0	Not used as PWM function; I/O pin or other functional pin.
0	1	Timer function; I/O pin or other function pin.
1	0	PWM function, the waveform keeps at inactive level.
1	1	PWM function, the normal PWM output waveform.

PWMBCR2: PWMB Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	PWMBPSF	PWMBDSF	PWMBCMPLC	PWMBSWAP	PWMBOSM	PWMBS1	PWMBS0	DeadTBE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCC; SFR Page = 0x1

Bit 7 (PWMBPSF): Status flag of period-matching for PWMB (Pulse Width Modulation). Set when a selected period is reached, reset by software.

Bit 6 (PWMBDSF): Status flag of duty-matching for PWMB (Pulse Width Modulation). Set when a selected duty is reached, reset by software.

Bit 5 (PWMBCMPLC): The enable bit of the link control between PWM and CMP.

0: Disable

1: Enable. Before PWM function is active, the CMP must be enabled first. If not, the PWM function would be not active.

The enable order of PWM and CMP: Turn CMP on first and then turn PWM on.

The disable order of PWM and CMP: Turn PWM off first and then turn CMP off.

If CMP is turned off but PWM is turned on, then PWM is also turned off.

The relationship table between CMP and PWM is shown below: (PWMEN = 1)

PWMBCMPLC	CMPxEn	SC1PWMBEN	TBEN	CMP Function	PWM Function
0	0	x	0	Off	Off
0	0	x	1	Off	On
0	1	x	0	On	Off
0	1	x	1	On	On
1	0	0	0	Off	Off
1	0	0	1	Off	On
1	1	0	0	On	Off
1	1	0	1	On	On
1	0	1	0	Off	Off
1	0	1	1	Off	Off
1	1	1	0	On	Off
1	1	1	1	On	On

Bit 4 (PWMBSWAP): To select whether DPWM output is swapped or not.

0: Outputs DPWM waveform is not swapped.

1: Outputs DPWM waveform is swapped.

Bit 3 (PWMBOSM): One-shot mode of PWM. This bit is set by SW and cleared by HW. This bit is valid in DPWM trigger mode. TBEN needs be enabled first to use one-shot mode of PWM.

0: Disable (default)

1: Enable

Bits 2~1 (PWMBS1~PWMBS0): Clock selection for PWMB timer

PWMBS1	PWMBS0	Clock selection for PWMB timer
0	0	F_{LS}
0	1	F_{HS}
1	0	PLL (48MHz)
1	1	NA(PLL (48MHz))

Note

When using crystal oscillator as clock source, user needs to set corresponding frequency divider to obtain 1MHz clock into PLL circuit. The configuration of frequency divider is at code option word

Bit 0 (DeadTBE): Dead-time function enable bit for dual PWM B

0: Disable (default)

1: Enable

PRDBL: Low Byte of PWMB Period

Bit	7	6	5	4	3	2	1	0
Name	PRDB.7	PRDB.6	PRDB.5	PRDB.4	PRDB.3	PRDB.2	PRDB.1	PRDB.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDA; SFR Page = 0x1

Bits 7~0 (PRDB.7~PRDB.0): The contents of the register are low byte of the PWM period

PRDBH: High Byte of PWM Period

Bit	7	6	5	4	3	2	1	0
Name	PRDB.15	PRDB.14	PRDB.13	PRDB.12	PRDB.11	PRDB.10	PRDB.9	PRDB.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDB; SFR Page = 0x1

Bits 7~0 (PADB.15~PADB.8): The contents of the register are high byte of the PWM period

DTBL: Low Byte of PWM Duty

Bit	7	6	5	4	3	2	1	0
Name	DTB.7	DTB.6	DTB.5	DTB.4	DTB.3	DTB.2	DTB.1	DTB.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDC; SFR Page = 0x1

Bits 7~0 (DTB.7~DTB.0): The contents of the register are low byte of the PWM duty

DTBH: High Byte of PWM Duty

Bit	7	6	5	4	3	2	1	0
Name	DTB.15	DTB.14	DTB.13	DTB.12	DTB.11	DTB.10	DTB.9	DTB.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDD; SFR Page = 0x1

Bits 7~0 (DTB.15~DTB.8): The contents of the register are high byte of the PWM duty

TMRBL: Low Byte of Timer B

Bit	7	6	5	4	3	2	1	0
Name	TMRB.7	TMRB.6	TMRB.5	TMRB.4	TMRB.3	TMRB.2	TMRB.1	TMRB.0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDE; SFR Page = 0x1

Bits 7~0 (TMRB.7~TMRB.0): The contents of the register are low byte of the PWM timer which is counting. This is read-only.

TMRBH: High Byte of Timer B

Bit	7	6	5	4	3	2	1	0
Name	TMRB.15	TMRB.14	TMRB.13	TMRB.12	TMRB.11	TMRB.10	TMRB.9	TMRB.8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDF; SFR Page = 0x1

Bits 7~0 (TMRB.15~TMRB.8): The contents of the register are high byte of the PWM timer which is counting. This is read-only.

PWMC

PWMCCR1: PWMC Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	PWMCE	IPWMCE	PWMCA	IPWMCA	-	TCP2	TCP1	TCP0
Type	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XE1; SFR Page = 0x1

Bit 7 (PWMCE): PWMC enable bit

0: Disable (default)

1: Enable. The compound pin is used as PWM pin. Please note that when using PWM function, do not enable other functions, otherwise it will lead to defaults.

Bit 6 (IPWMCE): Inverse PWMC enable bit

0: Disable (default)

1: Enable. The compound pin is used as /PWM pin. Please note that when using /PWM function, do not enable other functions, otherwise it will lead to defaults.

Bit 5 (PWMCA): Active level of PWMC

0: Duty duration is logic high (default)

1: Duty duration is logic low

Bit 4 (IPWMCA): Active level of inverse PWMC

0: Period-Duty duration is logic high (default)

1: Period-Duty duration is logic low

Bits 3: Reserved. Read = 0, Write = Don't Care.

Bits 2~0 (TCP2~TCP0): Timer C clock pre-scale option bits

TCP2	TCP1	TCP0	Prescale
0	0	0	1:1 (default)
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:64
1	1	0	1:128
1	1	1	1:256

PWMCE	TCEN	Function description
0	0	Not used as PWM function; I/O pin or other functional pin.
0	1	Timer function; I/O pin or other function pin.
1	0	PWM function, the waveform keeps at inactive level.
1	1	PWM function, the normal PWM output waveform.

PWMCCR2: PWMC Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	PWMCPSF	PWMCDSF	PWMCCMPLC	PWMCSWAP	PWMCOSM	PWMCS1	PWMCS0	DeadTCE
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCD; SFR Page = 0x1

Bit 7 (PWMCPSF): Status flag of period-matching for PWMC (Pulse Width Modulation). Set when a selected period is reached, reset by software.

Bit 6 (PWMCDSF): Status flag of duty-matching for PWMC (Pulse Width Modulation). Set when a selected duty is reached, reset by software.

Bit 5 (PWMCCMPLC): The enable bit of the link control between PWM and CMP.

0: Disable

1: Enable. Before PWM function is active, the CMP must be enabled first. If not, the PWM function would be not active.

The enable order of PWM and CMP: Turn CMP on first and then turn PWM on.

The disable order of PWM and CMP: Turn PWM off first and then turn CMP off.

If CMP is turned off but PWM is turned on, then PWM is also turned off.

The relationship table between CMP and PWM shows below: (PWMEN = 1)

PWMCCMPLC	CMPxEn	SC1PWMCEN	TCEN	CMP Function	PWM Function
0	0	x	0	Off	Off
0	0	x	1	Off	On
0	1	x	0	On	Off
0	1	x	1	On	On
1	0	0	0	Off	Off
1	0	0	1	Off	On
1	1	0	0	On	Off
1	1	0	1	On	On
1	0	1	0	Off	Off
1	0	1	1	Off	Off
1	1	1	0	On	Off
1	1	1	1	On	On

Bit 4 (PWMCSWAP): To select whether DPWM output is swapped or not.

0: Outputs DPWM waveform is not swapped.

1: Outputs DPWM waveform is swapped.

Bit 3 (PWMCOSM): One-shot mode of PWM. This bit is set by SW and cleared by HW. This bit is valid in DPWM trigger mode. TCEN needs be enabled first to use one-shot mode of PWM.

0: Disable (default)

1: Enable

Bits 2~1 (PWMCS1~PWMCS0): Clock selection for PWMC timer

PWMCS1	PWMCS0	Clock selection for PWMC timer
0	0	F_{LS}
0	1	F_{HS}
1	0	PLL (48MHz)
1	1	NA(PLL (48MHz))

Note

When using crystal oscillator as clock source, user needs to set corresponding frequency divider to obtain 1MHz clock into PLL circuit. The configuration of frequency divider is at code option word

Bit 0 (DeadTCE): Dead-time function enable bit for dual PWMC

0: Disable (default)

1: Enable

PRDCL: Low Byte of PWMC Period

Bit	7	6	5	4	3	2	1	0
Name	PRDC.7	PRDC.6	PRDC.5	PRDC.4	PRDC.3	PRDC.2	PRDC.1	PRDC.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE2; SFR Page = 0x1

Bits 7~0 (PRDC.7~PRDC.0): The contents of the register are low byte of the PWM period

PRDCH: High Byte of PWM Period

Bit	7	6	5	4	3	2	1	0
Name	PRDC.15	PRDC.14	PRDC.13	PRDC.12	PRDC.11	PRDC.10	PRDC.9	PADC.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE3; SFR Page = 0x1

Bits 7~0 (PRDC.15~PRDC.8): The contents of the register are high byte of the PWM period

DTCL: Low Byte of PWM Duty

Bit	7	6	5	4	3	2	1	0
Name	DTC.7	DTC.6	DTC.5	DTC.4	DTC.3	DTC.2	DTC.1	DTC.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE4; SFR Page = 0x1

Bits 7~0 (DTC.7~DTC.0): The contents of the register are low byte of the PWM duty

DTCH: High Byte of PWM Duty

Bit	7	6	5	4	3	2	1	0
Name	DTC.15	DTC.14	DTC.13	DTC.12	DTC.11	DTC.10	DTC.9	DTC.8
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE5; SFR Page = 0x1

Bits 7~0 (DTC.15~DTC.8): The contents of the register are high byte of the PWM duty

TMRCL: Low Byte of Timer B

Bit	7	6	5	4	3	2	1	0
Name	TMRC.7	TMRC.6	TMRC.5	TMRC.4	TMRC.3	TMRC.2	TMRC.1	TMRC.0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE6; SFR Page = 0x1

Bits 7~0 (TMRC.7~TMRC.0): The contents of the register are low byte of the PWM timer which is counting. This is read-only.

TMRCH: High Byte of Timer B

Bit	7	6	5	4	3	2	1	0
Name	TMRC.15	TMRC.14	TMRC.13	TMRC.12	TMRC.11	TMRC.10	TMRC.9	TMRC.8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE7; SFR Page = 0x1

Bits 7~0 (TMRC.15~TMRC.8): The contents of the register are high byte of the PWM timer which is counting. This is read-only.

20 High Low Voltage Detector(HLVD)

Under unstable power source condition, such as external power noise interference or EMS test condition, a violent power vibration could occur. At the time, the VDD could become unstable as it could be operating below working voltage. When the system supply voltage (VDD) is below operating voltage, the IC kernel will automatically keep all register status.

20.1Description of HLVD control registers

HLVDCR: HLVD Control Register

Bit	7	6	5	4	3	2	1	0
Name	HLVDEN	IRVSF	VDSB	VDM	HLVDS3	HLVDS2	HLVDS1	HLVDS0
Type	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9E; SFR Page = 0x0

Bit 7 (HLVDEN): High / Low Voltage Detector Enable Bit

0= Disable low voltage detector

1= Enable low voltage detector

Bit 6 (IRVSF): Internal Reference Voltage Stable Flag bit

1= Indicate that the voltage detect logic will generate the interrupt flag at the specified voltage range

0= Indicate that the voltage detect logic will not generate the interrupt flag at the specified voltage range and the HLVD interrupt should not be enabled

Bit 5 (VDSB): Voltage Detector State Bit. This is a read only bit.

1= VDD > HVD trip point. (HLVDS<3:0>)

0= VDD < LVD trip point. (HLVDS<3:0>)

Bit 4 (VDM): Voltage Direction Magnitude Select bit

1= Event occurs when voltage equals or exceeds trip point (HLVDS<3:0>)

0= Event occurs when voltage equals or falls trip point (HLVDS<3:0>)

HLVDIE	HLVDEN	VDM	IRVSF	VDSB	HLVDSF	Interrupt
0	1	1	1	0->1	0->1	Not happened
0	1	1	1	1->0	0	Not happened
0	1	0	1	0->1	0	Not happened
0	1	0	1	1->0	0->1	Not happened
1	0	X	X	X	0	Not happened
1	1	X	0	1	0	Not happened
1	1	1	1	0->1	0->1	Happened
1	1	1	1	1->0	0	Not happened
1	1	0	1	0->1	0	Not happened
1	1	0	1	1->0	0->1	Happened

Bit 3~0 (HLVD3~0): High Low Voltage Detector level bits.

HLVDS3~0	HLVD Voltage Interrupt Level
1111	Reserved
1110	Reserved
1101	Reserved
1100	2.5V
1011	2.6V
1010	2.8V
1001	2.9V
1000	3.1V
0111	3.3V
0110	3.5V
0101	3.7V
0100	3.9V
0011	4.1V
0010	4.3V
0001	4.5V
0000	4.7V
XXXX	NA

HLVDCR1: HLVD Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	HLVDSF							
Type	R/W	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9F; SFR Page = 0x0

Bit 7 (HLVDSF): Low Voltage Detector status flag.

HLVDIE	HLVDEN	VDM	IRVSF	VDSB	HLVDSF	Interrupt
0	1	1	1	0->1	0->1	Not happened
0	1	1	1	1->0	0	Not happened
0	1	0	1	0->1	0	Not happened
0	1	0	1	1->0	0->1	Not happened
1	0	X	X	X	0	Not happened
1	1	X	0	1	0	Not happened
1	1	1	1	0->1	0->1	Happened
1	1	1	1	1->0	0	Not happened
1	1	0	1	0->1	0	Not happened
1	1	0	1	1->0	0->1	Happened

The following steps are needed to setup the HLVD function:

1. Set the HLVDEN to “1”, then use Bits 3~0 (HLVDS2~HLVDS0) of Register Bank R to set the HLVD interrupt level

2. Wait for HLVD interrupt to occur

During sleep mode, the HLVD module continues to operate. If the device voltage drops slowly and crosses the detect point, the H/LVDSF bit will be set and the device will not wake up from Sleep mode. Until other wake-up source wakes up EMxxxxxxN, the HLVD interrupt flag will remain in the prior status.

When the system resets, the HLVD flag will be cleared.

The Figure 6-30 shows the HLVD module to detect the external voltage situation.

If VDD does not drop below VLVD, H/LVDSF remain at “0”.

When VDD drops below VLVD, H/LVDSF is set to “1”. If global ENI enables, H/LVDSF will be set to “1”, the next instruction will branch to the interrupt vector. The HLVD interrupt flag is cleared to “0” by software.

21 On-Chip Debug Support (OCDS)

THE MCU devices include an on-chip 2-Wire debug interface to allow Flash programming and simple debugging functions when used with ELAN IDE. The OCDS only need a total of four pins and they are defined in Table 21.1 below.

OCDS pins	Function
POWER	Power Supply
2W_SCL	Serial Clock
2W_SDA	Serial Data
GND	Ground

Table 20- 1OCDS pin definition

21.1 Limitations of On-Chip Debug

During the debugging process, user must take care of the 2W_CLK and 2W_SDA pins for data and clock communication purposes to ensure that no other components are connected to these two pins. Circuit components restriction is shown in Figure 20-1.

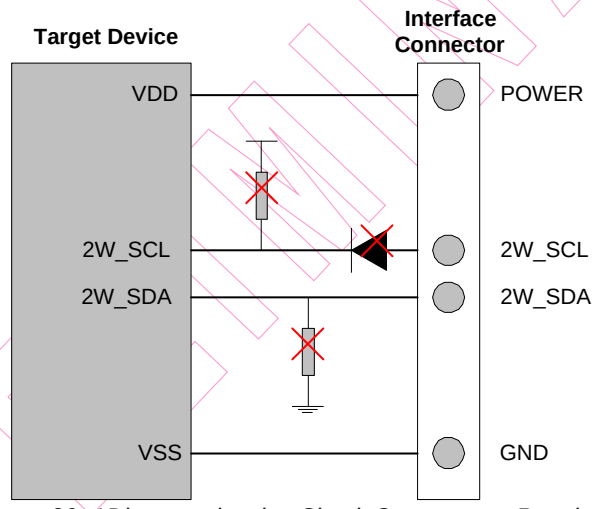


Figure 20- 1Diagram showing Circuit Components Restriction

To ensure efficient debugging, the following are some guidelines to follow:

1. Do not use pull-high and pull-down on the 2W_SCL/2W_SDA pin.
2. Do not use capacitors on the 2W_SCL/2W_SDA pin.
3. Do not use diodes on the 2W_SCL/2W_SDA pin.

During into OCD mode process, user must ensure signal is cleared on the 2W_CLK and 2W_SDA pins. If into OCD mode is completed, the pin-shared function of 2W_CLK/2W_SDA will become invalid. For detailed information on OCDS, refer to the 8051 OCDS User Guide.

22 Code Option

The page size is 64bytes.

Code Option0								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	-	BORT2	BORT1	BORT0	BORS3	BORS2	BORS1	BORS0
1	-	High	High	High	High	High	High	High
0	-	Low	Low	Low	low	low	low	Low
Default	1	1	1	1	1	1	1	1

Bits 6~4 (BORT2~BORT0): BOR sample period. The clock is from 128kHz source.

BORT2~0	BOR Sample Period(us)
000	250
001	500
010	1000
011	Always Off
100	4000
101	8000
110	16000
111	Always On

Bits 3~0 (BORS3~BORS0): High / Low Voltage Detector Level Bits

BORS3~0	BOR Voltage Interrupt Level
1111	Reserved
1110	Reserved
1101	Reserved
1100	2.5V(Default)
1011	2.6V
1010	2.8V
1001	2.9V
1000	3.1V
0111	3.3V
0110	3.5V
0101	3.7V
0100	3.9V
0011	4.1V
0010	4.3V
0001	4.5V
0000	4.7V

Code Option1								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	ADFM	-	-	-	-	RCXT2	RCXT1	RCXT0
1	High	-	-	-	-	High	High	High
0	Low	-	-	-	-	Low	Low	Low
Default	1	1	1	1	1	1	1	1

Bit 7 (ADFM): This bit controls the format of AD data buffer (ADDH & ADDL), Refer to the following table:

ADFM			Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
12 bits	0	ADDH					ADD11	ADD10	ADD9	ADD8
		ADDL	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
	1	ADDH	ADD11	ADD10	ADD9	ADD8	ADD7	ADD6	ADD5	ADD4
		ADDL					ADD3	ADD2	ADD1	ADD0

Note

Not used Bits are set to “0” by hardware. If ADFM= 0 and with 12-bit resolution ,ADDH<7:4> = 0000

Bits 2~0 (RCXT2~RCXT0): High IRC/HXTAL oscillation frequency select.

RCXT[2:0]	IRC (MHz)	XTAL (MHz)
000	4 MHz	1 ~ 4 MHz
111	4MHz	100K ~ 1 MHz

Code Option2								
Bit	7	6	5	4	3	2	1	0
Name	PLL48MOUT	FOSCPS2	FOSCPS1	FOSCPS0	HLFS	RCOUT	HSOSC1	HSOSC0
1	High	High	High	High	HS	ODDIS	High	High
0	Low	Low	Low	Low	LS	ODEN	Low	Low
Default	1	1	1	1	1	1	1	1

Bit 7 (PLL48MOUT): PLL 48MHz out

0: HFOUT acts as PLL 48MHz OUT

1: HFOUT control; by HSOSC1~0.

Bits 6~4: FOSC pre-scaler

FOSCPS2	FOSCPS1	FOSCPS0	Function Description
0	0	0	FOSC/20
0	0	1	FOSC/16
0	1	0	FOSC/12
0	1	1	FOSC/10
1	0	0	FOSC/8
1	0	1	FOSC/4
1	1	0	FOSC/2
1	1	1	FOSC/1(default)

Bit 3 (HLFS): Reset to Normal or Slow Mode Select Bit.

0: CPU is selected as Slow mode when a reset occurs.

1: CPU is selected as Normal mode when a reset occurs (default).

Bit 2 (RCOUT): System Clock Output Enable Bit in IRC mode.

0: HFOUT/LFOUT pin is open drain.

1: HFOUT/LFOUT pin is not open drain (default).

Bit 1~0 (HSOSC1~0): High Speed Clock Source Select

PLL48MOUT	HSOSC1	HSOSC0	Function Description
0	0	0	High IRC mode, HFOUT acts as PLL 48MHz OUT
0	0	1	High XTAL mode. HFOUT acts as PLL 48MHz OUT
0	1	0	External CLK mode. HFOUT acts as PLL 48MHz OUT
0	1	1	High IRC mode, HFOUT acts as PLL 48MHz OUT
1	0	0	High IRC mode, HFOUT acts as clock output pin. The frequency is descripted in chapter 4.1.
1	0	1	High XTAL mode.
1	1	0	External CLK mode. The clock input is HXIN.
1	1	1	High IRC mode, HFOUT acts as I/O pin(Default)

Code Option3								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	LIRCpre1	LIRCpre0	LSS	-	PLLPRE2	PLLPRE1	PLLPRE0	PLLCLKENB
1	High	High	LIRC	-	High	High	High	Disable
0	Low	Low	LXCT	-	Low	Low	Low	enable
Default	1	1	1	1	1	1	1	1

Bit 7,6 (LIRCpre1~0): Frequency select from LIRC.

LIRCpre1	LIRCpre0	LIRC Frquency
0	0	16K
0	1	32K
1	0	NA(16K)
1	1	128K (default)

Bit 5 (LSS): frequency select from low speed.

0: from LXTAL.

1: from LIRC (default).

Bit 3 ~ 1 (PLLPRE2~0): Setting frequency divider of crystal oscillator to obtain 1MHz clock which enters into PLL circuit. It's valid only when x'tal oscillator mode/external CLK mode is used (HSOSC[1:0]=01/10).

PLLPRE2	PLLPRE 1	PLLPRE 0	Function Description
0	0	0	HOSC/20
0	0	1	HOSC/16
0	1	0	HOSC/12
0	1	1	HOSC/10
1	0	0	HOSC/8
1	0	1	HOSC/4
1	1	0	HOSC/2
1	1	1	HOSC/1

Bit 0 (PLLCLKENB): PLL enabled bit for clock source of capture timer.

0: Enable.

1: Disable (default).

Code Option4								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name			LSCLKOUTB	RESETEN		ENWDT		
1			Disable	I/O		Enable		
0			Enable	Reset		Disable		
Default	1	1	1	1	1	1	1	1

Bit 5(LSCLKOUTB): LFOUT acts as clock output pin.

0: Enable.

1: Disable (default).

Bit 4 (RESETEN): P52/RESET pin selection bit.

0: RESET pin Enable.

1: RESET pin Disable and corresponding pin as I/O (default).

Bit 2 (ENWDT): WDT enable bit.

0: Disable.

1: Enable (default).

Code Option5								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name				EFTIM			SHEN	
1				Light			Enable	
0				Heavy			Disable	
Default	1	1	1	1	1	1	1	1

Bit 4 (EFTIM): low pass filter.

0: Heavy.

1: Light (default).

Bit 1 (SHEN): System hold enable.

0: Disable.

1: Enable (default).

> Note
>
> When the system hold circuit is triggered to a system hold circuit release state, digital card SHCLK [1: 0], simply begin the operation of CPU. During LVR detection, circuit output signals will be obscured and this will continue until the CPU starts operating before unmasking. Therefore, to reduce the chances of noise signal triggering the LVR circuit

Code Option10-LIRC trim								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name				LIRCTB4	LIRCTB3	LIRCTB2	LIRCTB1	LIRCTB0
1				High	High	High	High	High
0				Low	Low	Low	Low	Low
Default	1	1	1	1	1	1	1	1

Bits 4~0 (LIRCTB4~0): LIRC Trim bits.

Code Option 13-PLL1								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name					PLL1GAIN2	PLL1GAIN1	PLL1GAIN0	PLL1ENB
1					1	1	1	Disable
0					0	0	0	enable
Default	1	1	1	1	1	1	1	1

Bit 3~1 (PLL1GAIN2~0): PLL Gain.

PLL1GAIN2	PLL1GAIN1	PLL1GAIN0	GAIN
0	0	0	Reserved
0	0	1	Reserved
0	1	0	Reserved
0	1	1	5
1	0	0	4
1	0	1	3
1	1	0	2.5
1	1	1	2

Bit 0 (PLL1ENB): PLL Enable.

0: Enable.

1: Disable.

23 Instruction Set and CPU Core

The MCU, aside from having full compatibility with the Standard 8051 microcontroller, keeping all instruction mnemonics, as well as binary compatibility has integrated great architectural enhancements, allowing high performance CPU execution of instructions.

Mnemonic	Description	Bytes	Clock Cycles
ADD A,Rn	Add register to accumulator	1	1
ADD A,direct	Add direct byte to accumulator	2	2
ADD A,@Ri	Add indirect RAM to accumulator	1	2
ADD A,#data	Add immediate data to accumulator	2	2
ADDC A,Rn	Add register to accumulator with carry flag	1	1
ADDC A,direct	Add direct byte to A with carry flag	2	2
ADDC A,@Ri	Add indirect RAM to A with carry flag	1	2
ADDC A,#data	Add immediate data to A with carry flag	2	2
SUBB A,Rn	Subtract register from A with borrow	1	1
SUBB A,direct	Subtract direct byte from A with borrow	2	2
SUBB A,@Ri	Subtract indirect RAM from A with borrow	1	2
SUBB A,#data	Subtract immediate data from A with borrow	2	2
INC A	Increment accumulator	1	1
INC Rn	Increment register	1	2
INC direct	Increment direct byte	2	3
INC @Ri	Increment indirect RAM	1	3
DEC A	Decrement accumulator	1	1
DEC Rn	Decrement register	1	2
DEC direct	Decrement direct byte	1	3
DEC @Ri	Decrement indirect RAM	2	3
INC DPTR	Increment data pointer	1	1
MUL A,B	Multiply A and B	1	2
DIV A,B	Divide A by B	1	6
DA A	Decimal adjust accumulator	1	3
ANL A,Rn	AND register to accumulator	1	1
ANL A,direct	AND direct byte to accumulator	2	2
ANL A,@Ri	AND indirect RAM to accumulator	1	2
ANL A,#data	AND immediate data to accumulator	2	2
ANL direct,A	AND accumulator to direct byte	2	3
ANL direct,#data	AND immediate data to direct byte	3	3
ORL A,Rn	OR register to accumulator	1	1
ORL A,direct	OR direct byte to accumulator	2	2
ORL A,@Ri	OR indirect RAM to accumulator	1	2
ORL A,#data	OR immediate data to accumulator	2	2
ORL direct,A	OR accumulator to direct byte	2	3

Mnemonic	Description	Bytes	Clock Cycles
ORL direct,#data	OR immediate data to direct byte	3	3
XRL A,Rn	Exclusive OR register to accumulator	1	1
XRL A,direct	Exclusive OR direct byte to accumulator	2	2
XRL A,@Ri	Exclusive OR indirect RAM to accumulator	1	2
XRL A,#data	Exclusive OR immediate data to accumulator	2	2
XRL direct,A	Exclusive OR accumulator to direct byte	2	3
XRL direct,#data	Exclusive OR immediate data to direct byte	3	3
CLR A	Clear accumulator	1	1
CPL A	Complement accumulator	1	1
RL A	Rotate accumulator left	1	1
RLC A	Rotate accumulator left through carry	1	1
RR A	Rotate accumulator right	1	1
RRC A	Rotate accumulator right through carry	1	1
SWAP A	Swap nibbles within the accumulator	1	1
CLR C	Clear carry flag	1	1
CLR bit	Clear direct bit	2	3
SETB C	Set carry flag	1	1
SETB bit	Set direct bit	2	3
CPL C	Complement carry flag	1	1
CPL bit	Complement direct bit	2	3
ANL C,bit	AND direct bit to carry flag	2	2
ANL C,/bit	AND complement of direct bit to carry	2	2
ORL C,bit	OR direct bit to carry flag	2	2
ORL C,/bit	OR complement of direct bit to carry	2	2
MOV C,bit	Move direct bit to carry flag	2	2
MOV bit,C	Move carry flag to direct bit	2	3
MOV A,Rn	Move register to accumulator	1	1
MOV A,direct	Move direct byte to accumulator	2	2
MOV A,@Ri	Move indirect RAM to accumulator	1	2
MOV A,#data	Move immediate data to accumulator	2	2
MOV Rn,A	Move accumulator to register	1	1
MOV Rn,direct	Move direct byte to register	2	3
MOV Rn,#data	Move immediate data to register	2	2
MOV direct,A	Move accumulator to direct byte	2	2
MOV direct,Rn	Move register to direct byte	2	2
MOV direct1,direct2	Move direct byte to direct byte	3	3
MOV direct,@Ri	Move indirect RAM to direct byte	2	3
MOV direct,#data	Move immediate data to direct byte	3	3
MOV @Ri,A	Move accumulator to indirect RAM	1	2
MOV @Ri,direct	Move direct byte to indirect RAM	2	3
MOV @Ri,#data	Move immediate data to indirect RAM	2	2
MOV DPTR,#data16	Load data pointer with a 16-bit constant	3	3

Mnemonic	Description		Bytes	Clock Cycles
MOVC A,@A+DPTR	Move code byte relative to DPTR to accumulator		1	5
MOVC A,@A+PC	Move code byte relative to PC to accumulator		1	4
MOVX A,@Ri	Move external RAM (8-bit address) to A		1	3
MOVX A,@DPTR	Move external RAM (16-bit address) to A		1	2
MOVX @Ri,A	Move A to external RAM (8-bit address)	CODE inside ROM/RAM destination XRAM data	1	4
		all other cases		5
MOVX @DPTR,A	Move A to external RAM (16-bit address)	CODE inside ROM/RAM destination XRAM data	1	3
		all other cases		4
PUSH direct	Push direct byte onto stack		2	3
POP direct	Pop direct byte from stack		2	2
XCH A,Rn	Exchange register with accumulator		1	2
XCH A,direct	Exchange direct byte with accumulator		2	3
XCH A,@Ri	Exchange indirect RAM with accumulator		1	3
XCHD A,@Ri	Exchange low-order nibble indirect RAM with A		1	3
ACALL addr11	Absolute subroutine call		2	4
LCALL addr16	Long subroutine call		3	4
RET	Return from subroutine		1	4
RETI	Return from interrupt		1	4
AJMP addr11	Absolute jump		2	3
LJMP addr16	Long jump		3	4
SJMP rel	Short jump (relative address)		2	3
JMP @A+DPTR	Jump indirect relative to the DPTR		1	5
JZ rel	Jump if accumulator is zero		2	4
JNZ rel	Jump if accumulator is not zero		2	4
JC rel	Jump if carry flag is set		2	3
JNC rel	Jump if carry flag is not set		2	3
JB bit,rel	Jump if direct bit is set		3	5
JNB bit,rel	Jump if direct bit is not set		3	5
JBC bit,direct, rel	Jump if direct bit is set and clear bit		3	5
CJNE A,direct, rel	Compare direct byte to A and jump if not equal		3	5
CJNE A,#data, rel	Compare immediate to A and jump if not equal		3	4
CJNE Rn,#data, rel	Compare immediate to reg. and jump if not equal		3	4
CJNE @Ri,#data, rel	Compare immediate to ind. and jump if not equal		3	5
DJNZ Rn,rel	Decrement register and jump if not zero		2	4
DJNZ direct,rel	Decrement direct byte and jump if not zero		3	5
NOP	No operation		1	1

Table 24.1 Instruction Set Summary

Instruction Set Notes	
Rn	Register R0–R7 of the currently selected register bank.
@Ri	Data RAM location addressed indirectly through R0 or R1.
rel	8-bit, signed (two's complement) offset relative to the first byte of the following instruction. Used by SJMP and all conditional jumps.
Direct	8-bit internal data location's address. This could be a direct-access Data RAM location (0x00–0x7F) or an SFR (0x80–0xFF).
#data	8-bit constant.
#data16	16-bit constant.
Bit	Direct-accessed bit in Data RAM or SFR.
addr11	11-bit destination address used by ACALL and AJMP. The destination must be within the same 2 kB page of Flash Memory as the first byte of the following instruction.
addr16	16-bit destination address used by LCALL and LJMP. The destination may be anywhere within the 8kB Flash Memory space.

23.1 CPU Core Register Descriptions

The processor has an arithmetic section that performs extensive data manipulations and is composed of 1) Arithmetic Logic Unit (ALU) (this circuit performs 8-bit arithmetic and logic operations), 2) an ACC register, 3) B register (this B register is used during multiplication and division operations. In other cases, may be used as normal SFR) and 4) PSW register (The PSW contains several bits that reflect the current state of the CPU). The data pointers (DPTR) are used by some instructions to access external data memory (XRAM).

ACC: Accumulator

Bit	7	6	5	4	3	2	1	0
Name	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xE0; SFR Page = All Pages

B: B Register

Bit	7	6	5	4	3	2	1	0
Name	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xF0; SFR Page = All Pages

DPL: Data Pointer Low Byte

Bit	7	6	5	4	3	2	1	0
Name	DPL.7	DPL.6	DPL.5	DPL.4	DPL.3	DPL.2	DPL.1	DPL.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x82; SFR Page = All Pages

DPH: Data Pointer High Byte

Bit	7	6	5	4	3	2	1	0
Name	DPL.7	DPL.6	DPL.5	DPL.4	DPL.3	DPL.2	DPL.1	DPL.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x83; SFR Page = All Pages

PSW: Program Status Word 0

Bit	7	6	5	4	3	2	1	0
Name	CY	AC	F0	RS1	RS0	OV	F1	PARITY
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD0; SFR Page = All Pages

Bit 7: Carry Flag.

This bit is set when the last arithmetic operation resulted in a carry (addition) or a borrow (subtraction). It is cleared to logic 0 by all other arithmetic operations.

Bit 6: Auxiliary Carry Flag.

This bit is set when the last arithmetic operation resulted in a carry into (addition) or a borrow from (subtraction) the high order nibble. It is cleared to logic 0 by all other arithmetic operations.

Bit 5: User Flag 0.

This is a bit-addressable, general purpose flag for use under software control.

Bit 3~4: Register Bank Select.

These bits select which register bank is used during register accesses.

RS1	RS0	Description
0	0	Bank 0, Addresses 0x00-0x07
0	1	Bank 1, Addresses 0x08-0x0F
1	0	Bank 2, Addresses 0x10-0x17
1	1	Bank 3, Addresses 0x18-0x1F

Bit 2: Overflow Flag.

This bit is set to 1 under the following circumstances:

An ADD, ADDC, or SUBB instruction causes a sign-change overflow.

A MUL instruction results in an overflow (result is greater than 255).

A DIV instruction causes a divide-by-zero condition.

The OV bit is cleared to 0 by the ADD, ADDC, SUBB, MUL, and DIV instructions in all other cases.

Bit 1: User Flag 1.

This is a bit-addressable, general purpose flag for uses under software control.

Bit 0: Parity Flag.

This bit is set to logic 1 if the sum of the eight bits in the accumulator is odd and cleared if the sum is even.

PSW1: Program Status Word 1

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	-	SHSF
Type	R	R	R	R	R	R	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD8; SFR Page = All Pages

(User needs to back up PSW1 by software when CPU into interrupt service routine, because keil compile does not support it)

Bit 7~1: Reserved. Read = 0, Write = Don't Care.

Bit 0: System-Hold Flag.

This flag is set by hardware when a System-Hold is detected. It can be cleared by software.

When this flag is set, the HW will turn off DPWM function automatically.

23.2 Enhanced CPU Core Functions

23.2.1 Dual data pointer

The Standard Data Pointer is known as DPTR, while the New Data Pointer is known as DPTR1, and a single bit is known as DPS (PCON.3) which allows the Program Code to switch in between. Dual Data Pointer Register structure is employed to speed up data block copying. (User needs to back up the DPTR1 by software when CPU goes into interrupt Service Routine, because the keil-C Compiler does not support it)

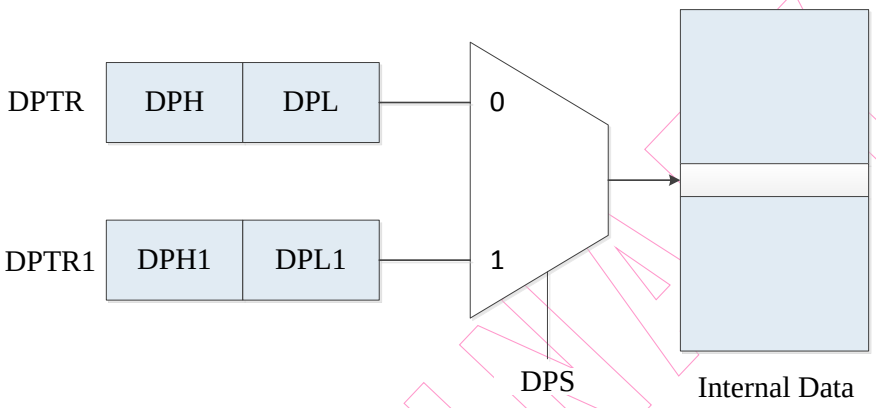


Figure 23.1 Dual Data Pointer Registers

PCON: Power Control

Bit	7	6	5	4	3	2	1	0
Name	SMOD0	-	-	-	DPS	-	PD	IDLE
Type	R/W	R	R	R	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x87; SFR Page = All Pages

Bit 7: UART0 double baud rate bit.

0: Disable double Baud rate of the UART0.

1: Enable double Baud rate of the UART0 in mode 1, 2, or 3.

Bits 6~4: Reserved. Read = 0, Write = Don't Care.

Bit 3: Data Pointer Select.

This bit is used to switch between DPTR and DPTR1.

0: Select DPTR.

1: Select DPTR1.

Bit 2: Reserved. Read = 0, Write = Don't Care.

Bit 1: Power-down control bit.

Setting this bit activates Power-down operation. The PD bit is cleared automatically by hardware when waking up from power-down.

Bit 0: Idle mode control bit.

Setting this bit activates Idle mode operation. The IDL bit is cleared automatically by hardware when waking up from Idle (If PD and IDLE is 1, CPU enters into Power-Down Mode).

DPL1: Data Pointer Low Byte 1

Bit	7	6	5	4	3	2	1	0
Name	DPL.7	DPL.6	DPL.5	DPL.4	DPL.3	DPL.2	DPL.1	DPL.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x84; SFR Page = All Pages

DPH1: Data Pointer High Byte 1

Bit	7	6	5	4	3	2	1	0
Name	DPL.7	DPL.6	DPL.5	DPL.4	DPL.3	DPL.2	DPL.1	DPL.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x85; SFR Page = All Pages

23.2.2 Arithmetic Accelerator

The Standard 8051 only supports 8-bit multiplication and division operations. But the EM85F765N has enhanced arithmetic function that supports 16x16 and 32/16 operand. Note that an extra register is needed for use during 16x16 and 32/16 operand. The registers and their descriptions are shown in Table 23.2

(- : Always invalid.)

Function	Operation	Result					
		MA0	MA1	MA2	MA3	MB1	MB0
DIV 32/16	(MA3~0)/(MB1~0)	Quotient [31:24]	Quotient [23:16]	Quotient [15:8]	Quotient [7:0]	Remainder [15:8]	Remainder [7:0]
MUL 16x16	(MA1~0)*(MB1~0)	Product [31:24]	Product [23:16]	Product [15:8]	Product [7:0]	-	-

Table 23.2 Divide and Multiply Operation Registers

ARITH: Arithmetic Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	F/I	SMEN1	SMEN0	SDEN	MRUN	DRUN
Type	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xA9; SFR Page = 2

Bits 7~6: Reserved. Read = 0, Write = Don't Care.

Bit 5: Fractional / Integer Flag.

F/I is used to indicate Fractional or Integer mode.

0: Integer mode.

1: Fractional mode.

Bit 4~3: Signed multiplication Operation Enable.

SMEN1	SMEN0	Description
0	0	MA15~0(unsigned) X MB(unsigned)
0	1	MA15~0(unsigned) X MB(signed)
1	0	MA15~0(signed) X MB(unsigned)
1	1	MA15~0(signed) X MB(signed)

Bit 2: Signed Division Operation Enable.

0: Unsigned.

1: Signed.

Bit 1: Multiplication Run.

0: Multiplication finish.

1: Multiplication start. If multiplication is finished, the bit is clear to 0 by hardware.

Bit 0: Division Run.

0: Division finish.

1: Division start. If division is finished, the bit is clear to 0 by hardware.

MA0: Multiplier A0

Bit	7	6	5	4	3	2	1	0
Name	MA0.7	MA0.6	MA0.5	MA0.4	MA0.3	MA0.2	MA0.1	MA0.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAA; SFR Page = 2

MA1: Multiplier A1

Bit	7	6	5	4	3	2	1	0
Name	MA1.7	MA1.6	MA1.5	MA1.4	MA1.3	MA1.2	MA1.1	MA1.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAB SFR Page = 2

MA2: Multiplier A2

Bit	7	6	5	4	3	2	1	0
Name	MA2.7	MA2.6	MA2.5	MA2.4	MA2.3	MA2.2	MA2.1	MA2.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAC; SFR Page = 2

MA3: Multiplier A3

Bit	7	6	5	4	3	2	1	0
Name	MA3.7	MA3.6	MA3.5	MA3.4	MA3.3	MA3.2	MA3.1	MA3.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAD; SFR Page = 2

MB0: Multiplier B0

Bit	7	6	5	4	3	2	1	0
Name	MB0.7	MB0.6	MB0.5	MB0.4	MB0.3	MB0.2	MB0.1	MB0.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAE; SFR Page = 2

MB1: Multiplier B1

Bit	7	6	5	4	3	2	1	0
Name	MB1.7	MB1.6	MB1.5	MB1.4	MB1.3	MB1.2	MB1.1	MB1.0
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAF; SFR Page = 2

24 Electrical Characteristics

24.1 Absolute Maximum Specifications

Items	Rating		
Temperature under bias	-40°C	to	85°C
Storage temperature	-65°C	to	150°C
Input voltage	Vss-0.3V	to	Vdd+0.5V
Output voltage	Vss-0.3V	to	Vdd+0.5V
Working Voltage	2.7V	to	5.5V
Working Frequency	32.768KHz	to	20MHz

24.2 DC Characteristics

(Ta=25 °C, VDD=5.0V±5%, VSS=0V)

Symbol	Parameter	Condition	Min	Typ	Max	Unit
IRCE1 (ILRC)	Low internal RC oscillator error per stage			±4		%
IRCE2 (IHRC)	High internal RC oscillator error per stage			±1		%
SFreq	Operation Frequency	-40~+85℃, 2.7~5V	32.768		20,000	KHz
IRC	IRC:VDD to 5V			4		MHz
IIL	Input Leakage Current for input pins	VIN = VDD, VSS	-1	0	1	μA
VIH1	Input High Voltage (Schmitt trigger)	Ports 0~5	0.56Vdd		Vdd+0.3V	V
VIL1	Input Low Voltage (Schmitt trigger)	Ports 0~5	-0.3V		0.44Vdd	V
VIHT1	Input High Threshold Voltage (Schmitt trigger)	/RESET	0.56Vdd		Vdd+0.3V	V
VILT1	Input Low Threshold Voltage (Schmitt trigger)	/RESET	-0.3V		0.44Vdd	V
VIHT2	Input High Threshold Voltage (Schmitt trigger)	Timer0~1, Timer3~4, INT	0.56Vdd		Vdd+0.3V	V
VILT2	Input Low Threshold Voltage (Schmitt trigger)	Timer0~1, Timer3~4, INT	-0.3V		0.44Vdd	V
VIHX1	Clock Input High Voltage (3V)	OSCI in crystal mode	1.52		Vdd+0.3V	V
VILX1	Clock Input Low Voltage (3V)	OSCI in crystal mode	-0.3V		1.12	V
IOH1	Output High Voltage (Ports 0~5)	VOH = 0.9VDD		8		mA
IOH2	Output High Voltage (high drive) (Ports 0,1,2,4,5)	VOH = 0.9VDD VOH = 0.7VDD		16 28		mA
IOH3	Output High Voltage (high drive) (Ports 3)	VOH = 0.7VDD		30		mA
IOL1	Output Low Voltage	VOL = 0.1VDD		14		mA

Symbol	Parameter	Condition	Min	Typ	Max	Unit
	(Ports 0~5)					
IOL2	Output Low Voltage (High sink) (Ports 0,1,2,4,5)	VOL = 0.1VDD VOL=0.3VDD		30 69		mA
IOL3	Output Low Voltage (High sink) (Ports 3)	VOL = 0.3VDD		120		mA
IPH1	Pull-high current (Ports 0~5, except P11, P12, P13, P14, P20, P21)	Pull-high active, input pin at VSS	-56	-75	-100	μA
IPL1	Pull-low current (Ports 0~5)	Pull-low active, input pin at VDD	31	40	51	μA
IPH2	Very weak pull-high current (only used in quasi-bidirectional IO) (Ports 0~5)	Very weak Pull-high active, input pin at VSS	-15	-20	-25	μA
IPH3	Pull-high current (P11, P12, P13, P14, P20, P21)	4.7Kohm at VDD =5V	-0.851	-1.063	-1.276	mA
ISB1	Power down current	WDT disabled			20	μA
ISB2	Power down current	WDT enabled			25	μA
ISB3	Idle mode current	FCPU, FHS off; FLS =128KHz WDT disabled, output pin floating,			150	μA
ISB4	Idle mode current	FCPU, FHS off; FLS =128KHz WDT enabled output pin floating,			155	μA
ISB5	Idle mode current	FCPU off; FHS is IRC 4M; FLS =128KHz WDT enabled output pin floating,			520	μA
ICC1	Operating supply current (Green mode)	FHS off; FCPU = FLS =128KHz WDT disabled output pin floating			715	μA
ICC2	Operating supply current (Green mode)	FHS off; FCPU = FLS = 128KHz WDT enable output pin floating			715	μA
ICC3	Operating supply current (Normal mode)	FCPU = FHS = 4 MHz +PLL*5 (Crystal type), FLS = 128KHz, output pin floating, WDT enabled			3.67	mA
ICC4	Operating supply current (Normal mode)	FCPU = FHS = 4 MHz +PLL*2(Crystal type), FLS = 128KHz, output pin floating, WDT enabled			2.48	mA
ICC5	Operating supply current (Normal mode)	FCPU = FHS = 4 MHz +PLL*5(IRC type), FLS = 128KHz, output pin floating, WDT enabled			2.97	mA
ICC6	Operating supply current (Normal mode)	FCPU = FHS = 4 MHz +PLL*2 (IRC type), FLS = 128KHz, output pin floating, WDT enabled			1.78	mA

Symbol	Parameter	Condition	Min	Typ	Max	Unit
ICC7	Operating supply current (Normal mode)	FCPU = FHS = 4 MHz (Crystal type), FLS = 128KHz, output pin floating, WDT enabled			1.95	mA
ICC8	Operating supply current (Normal mode)	FCPU = FHS = 4 MHz (IRC type), FLS = 128KHz, output pin floating, WDT enabled			1.24	mA

24.3 AC Characteristics

(Ta=25 °C, VDD=5V±5%, VSS=0V)

Symbol	Parameter	Conditions	Min	Type	Max	Unit
Dclk	Input CLK duty cycle		45	50	55	%
Tins	Instruction cycle time	Crystal type	50		DC	ns
		RC type	50		DC	ns
Tdrh	Device reset hold time	Ta = 25°C	11.3	16.2	21.6	ms
Trst	/RESET pulse width	Ta = 25°C	15.6			us
Twdt	Watchdog timer duration	Ta = 25°C	11.3	16.	21.6	ms
Tset	Input pin setup time			0		ns
Thold	Input pin hold time		15	20	25	ns
Tdelay	Output pin delay time	Cload=20pF	45	50	55	ns
Vdd_rr	VDD rise rate		0.05			V/ms

Program Flash Memory Electrical Characteristics(Ta=25 °C, VDD=5V±5%, VSS=0V)

Symbol	Parameter	Conditions	Min	Type	Max	Unit
Tprog	Erase/Write cycle time	Vdd = 5.0V Temperature = -40° C ~ 85° C		6		ms
Treten	Data Retention			10		Years
Tendu	Endurance time		10K			Cycles

AD Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Parameter	Symbol	Test Conditions	Type			Unit	Remark
			Min.	Typ.	Max.		
Operating Range	Vdd	For 5V Fs=100kHz, Fin=1KHz, For 2.7V Fs=25KHz, Fin=1KHz	2.7		5	V	
	V_{REFT}		2.5		Vdd	V	Top reference voltage
	V_{REFB}		Vss	-	V_{REFT}-ΔV_{REF}	V	Bottom reference voltage
	ΔV_{REF}		2.5			V	Reference Voltage Range
1/2 *VDD AD Input	$V_{1/2VDD}$	Vdd=5V	2.475	2.5V	2.525	V	1/2*VDD AD channel Input Voltage
	$T_{1/2VDD}$	Vdd=5V		2.8	4	uS	1/2*VDD Warn up time for ADC sample
	$I_{1/2VDD}$	Vdd=5V		35	42	uA	1/2*VDD Current Consumption
Current Consumption	Ivdd	V_{REFT}= Vdd=5.5V, SVREFT="00", Fs=100KHz, Fin=1KHz			1.4	mA	The ADC top reference voltage is internal Vdd
	Iref				10	uA	
Current Consumption	Ivdd	V_{REFT}= Vdd=5.5V, SVREFT="01"or"10", Fs=100KHz, Fin=1KHz			0.9	mA	The ADC top reference voltage is external VREF
	Iref				0.5	mA	
Standby Current	Isb				0.1	uA	Including voltage reference
ZAI	ZAI				10K	ohm	The external impedance of analog input channel.
SNR	SNR	V_{REFT}= Vdd=5.0V, Fs=100KHz, Fin=1KHz	70			dBc	Signal-To-Noise Ratio
THD	THD	V_{REFT}= Vdd=5.0V, Fs=100KHz, Fin=1KHz			-70	dBc	Total Harmonic Distortion
SNDR	SNDR	V_{REFT}= Vdd=5.0V, Fs=100KHz, Fin=1kHz	68			dBc	Signal-to-Noise & Distortion Ratio

Parameter	Symbol	Test Conditions	Type			Unit	Remark
			Min.	Typ.	Max.		
Worst Harmonic	WH	V_{REFT}= Vdd=5.0V, Fs=100KHz, Fin=1KHz			-73	dBc	
SFDR	SFDR	V_{REFT}= Vdd=5.0V, Fs=100KHz, Fin=1KHz	73			dBc	Spurious Free Dynamic Range
Offset Error	OE	V_{REFT}= Vdd=5.0V, Fs=100KHz			+/-4	LSB	
Gain Error	GE	V_{REFT}= Vdd=5.0V, Fs=100KHz			+/-8	LSB	
DNL	DNL	VREFT= Vdd=5.0v, Fs=100KHz, Fin=1KHz			+/-1	LSB	Differential Nonlinearity
INL	INL	VREFT= Vdd=5.0V, Fs=100KHz, Fin=1KHz			+/-4	LSB	Integral Nonlinearity
Conversion Rate	Fs1	Vdd=3.0~5V, Fin=1KHz			100	K SPS	Including sampling & conversion phase
	Fs2	Vdd=2.7~3.0v, Fin=1KHz			25	K SPS	
Power Supply Rejection Ratio	PSRR	VREFT=2.5V, SVREF="01"or"10", Vdd=2.7V ~ 5.5V, Fs=25K Hz, Vin=0V ~ 2.5V			2	LSB	The change in DC output code from the value with the supply at the Min. limit to the value with the supply at its Max. limit.

OP Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
VOS	Input offset voltage	Vip=0.1V, after trimmed	0	±2	±5	mV
SR	Slew rate	RL=1M ohm, CL=20p, VI(pp)=3V, Av=1	2	2.5		V/μs
IVR*	Input voltage range*		0		3.6	V
VOL	Low-level output voltage	Vip=0V, IL=100uA, Av=1		10	40	mV
		Vip=0V, IL=1mA, Av=1		50	200	mV
VOH	High-level output voltage	Vip=2.5V, IL=100uA, Av=2	4.920	4.980		V
		Vip=2.5V, IL=1mA, Av=2	4.600	4.850		V
ISC_L	Output sink current (short circuit current)		5	10		mA
ISC_H	Output source current (short circuit current)		5	10		mA
IDD	Supply current	No load, Vic=0.1V, Av=1		150	200	μA

PSRR	Power supply rejection ratio	Vdda=2.7V~5V, Vout=0.1V	70	80		dB
CMRR	Common mode reject ratio	0V≦VCM≦(VDD-1.4)	70	80		dB
Avo	Open loop gain	RL=1M ohm, CL=20p,	70	100		dB
PM	Phase margin	RL=1Meg, CL=20p,	60	76		degree
GBP	Gain bandwidth product	RL=1M ohm, CL=20p,	1.2	1.9		MHz

* IVR: Max= Vdda-1.4V

Comparator Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
VOS	Input offset voltage	Vip=0.1V, after trimmed	0	±2	±5	mV
IVR*	Input voltage range*		0		3.6	V
IDD	Supply current			80	100	μA
TRS	Response time	Vin=0.1V, (Note1)		0.5	2	μs
TLRS	Large signal response time	Vin=1.8V, (Note2)		50	100	ns

* IVR: Max= Vdda-1.4V

Note1: The response time specified is a 100mV input step with 10mV overdrive

Note2: The response time specified is a 0V~3.6V input step with 1.8V overdrive.

Vref Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
Vdd	Power Supply		2.7		5	V
Ivdd	DC supply current	Vdd =5.5V,No load		250	400	uA
Tresponse	Response time	Trim bit and VREF select setting time		10	24	us
Vref	Voltage reference output	2.048V 2.560V 3.072V 4.096V			±1	%
Vdd_min	Minimum power supply		Vref+0.1	Vref+0.2*		V

PLLx48 Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
f_{IN}	Input Clock Frequency	Ta= -40~85 [℃]	0.65	4	4.4	MHz
f_{OUT}	Output Clock Frequency	Ta= -40~85 [℃]	7.16		48	MHz
D_{cycle}	Duty cycle		45		55	%
t_{jitter}	Jitter tolerance of output clock	1 sigma		300		ps

PLL1 Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
f_{IN}	Input Clock Frequency	Ta= -40~85 [℃]	3.58	4	4.4	MHz
f_{OUT}	Output Clock Frequency	Ta= -40~85 [℃]	7.16		20	MHz
D_{cycle}	Duty cycle		45		55	%
t_{jitter}	Jitter tolerance of output clock	1 sigma		300		ps

HLVD Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
Vdd	power supply		2.7		5.5	V
I_{lvd}	Current Consumption	HLVD Enable, VDD=5V		9	11	μA
VL	Detected voltage	According HLVDCR Setting	2.5	3.1	4.7	V
VH	Release voltage	According HLVDCR Setting	2.6	3.2	4.8	V
ΔV	VH,VL tolerance			± 0.15		V
VHYST	Hysteresis Voltage		50	100	150	mV
T_VREF	Vref enable time	LVD Enable, VDD=5V		63		uS

BOR Characteristics (Vdd = 5.0V, Vss=0V, Ta=25°C)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
Vdd	power supply		2.7		5.5	V
I_{bor}	Current Consumption	VDD=5V		9	11	μA
VL	Detected voltage	According Code Option0 Setting	2.5	3.1	4.7	V
VH	Release voltage	According Code Option0 Setting	2.6	3.2	4.8	V
ΔV	VH,VL tolerance			± 0.15		V
VHYST	Hysteresis Voltage		50	100	150	mV
T_VREF	Vref enable time	BOR Enable, VDD=5V		63		uS

APPENDIX

A Ordering and Manufacturing Information

EM85F765ND20J

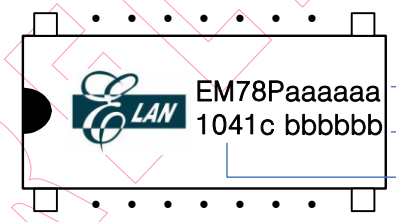
- **Material Type**
  J: RoHS complied
  S: Sony SS-00259 complied
  Contact Elan Sales for details
- **Pin Number**
- **Package Type**
  D: DIP
  SO: SOP
  Check the following section
- **Specific Annotation**
- **Product Number**
- **Product Type**
  F: Flash
- **Elan 8-bit Product**

For example:

EM85F765NSO20S

is EM85F765N with Flash program memory, product,
in 20-pin SOP 300mil package with Sony SS-00259 complied

IC Mark



- **Elan Product Number / Package, Material Type**
- **Batch Number**
- **Manufacture Date**
  "YYWW"
  YY is year and WW is week
  c is Alphabetical suffix code for Elan use only

Ordering Code

EM85F765D20J

- **Material Type**
  Contact Elan Sales for details
- **Package Type / Pin Number**
  Check the following section
- **Elan IC Product Number**

B Package Type

Flash MCU	Package Type	Pin Count	Package Size
EM85F765NQN32	QFN	32	4*4*0.6mm
EM85F765NSO20	SOP	20	300 mil
EM85F765NSO24	SOP	24	300 mil
EM85F765NSO28	SOP	28	300 mil

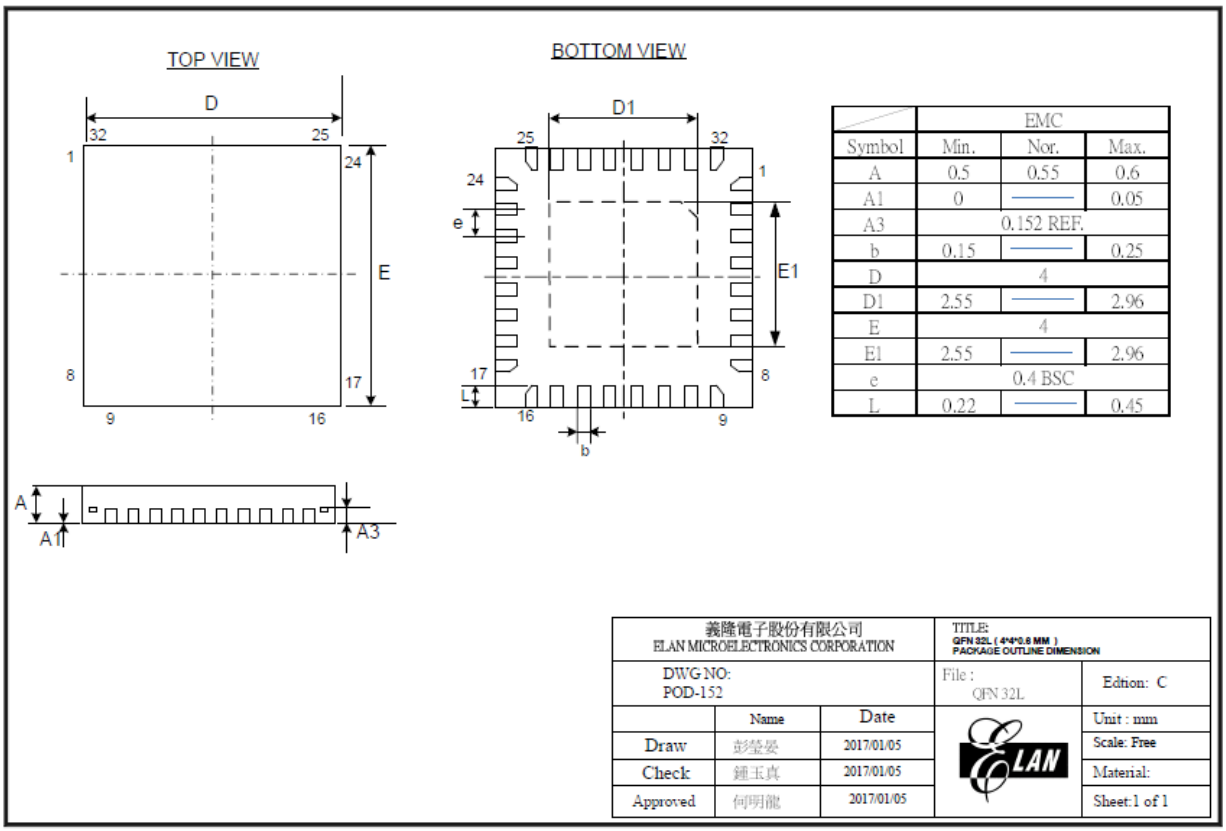
These are Green products which do not contain hazardous substances and comply with the third edition of Sony SS-00259 standard.

Pb contents should be less than 100ppm and complies with Sony specifications.

Part No.	EM85F765NS/J
Electroplate type	Pure Tin
Ingredient (%)	Sn:100%
Melting point (°C)	232°C
Electrical resistivity (μΩ cm)	11.4
Hardness (hv)	8~10
Elongation (%)	>50%

C Package Information

C.1 EM85F765NQN32 4*4*0.6mm

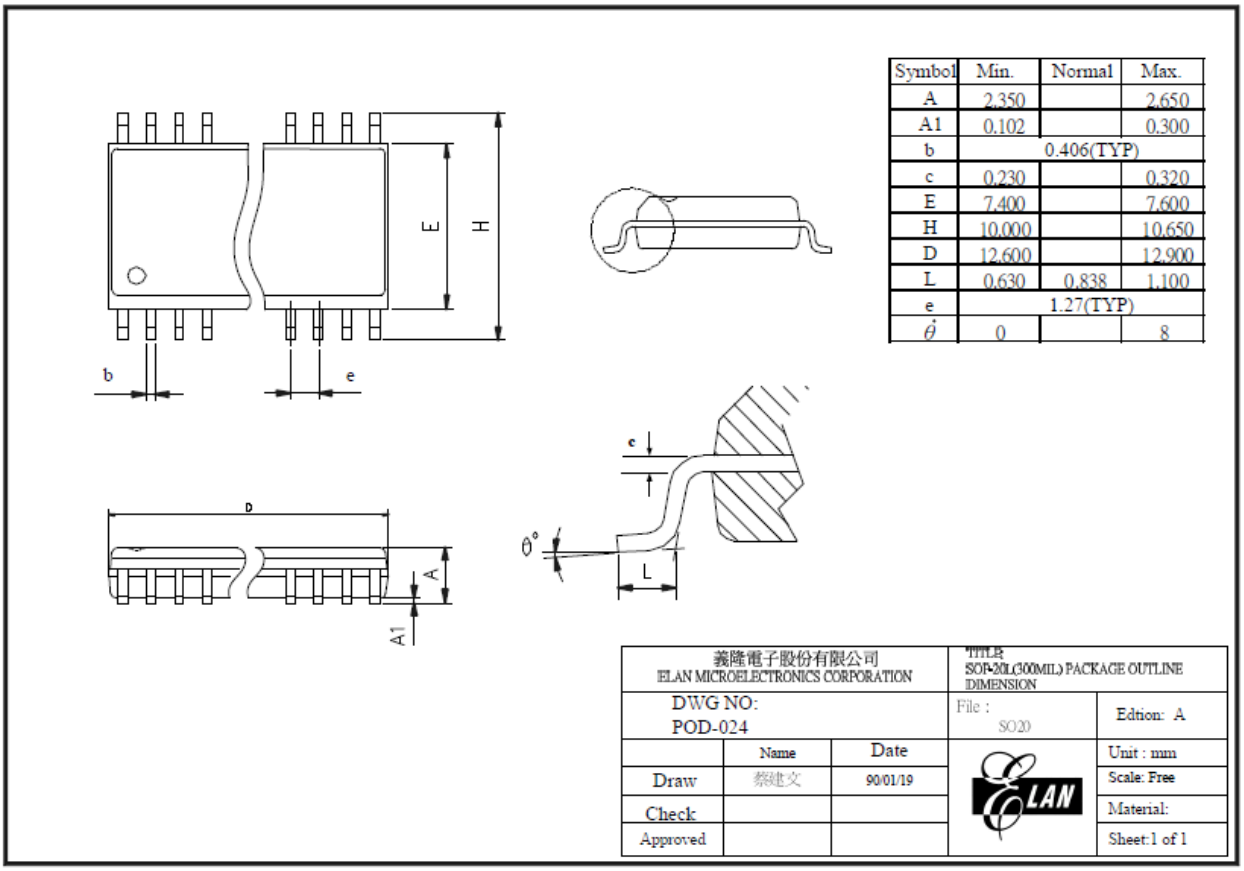


	EMC		
Symbol	Min.	Nor.	Max.
A	0.5	0.55	0.6
A1	0	———	0.05
A3		0.152 REF.	
b	0.15	———	0.25
D		4	
D1	2.55	———	2.96
E		4	
E1	2.55	———	2.96
e		0.4 BSC	
L	0.22	———	0.45

義隆電子股份有限公司 ELAN MICROELECTRONICS CORPORATION			TITLE: QFN 32L (4*4*0.6 MM) PACKAGE OUTLINE DIMENSION	
DWG NO: POD-152			File : QFN 32L	Edtion: C
	Name	Date		Unit : mm
Draw	彭瑩晏	2017/01/05		Scale: Free
Check	鍾玉真	2017/01/05		Material:
Approved	何明龍	2017/01/05		Sheet:1 of 1

Figure C-1 EM85F765N 32-Pin QFN Package Type

C.2 EM85F765NSO20 300mil

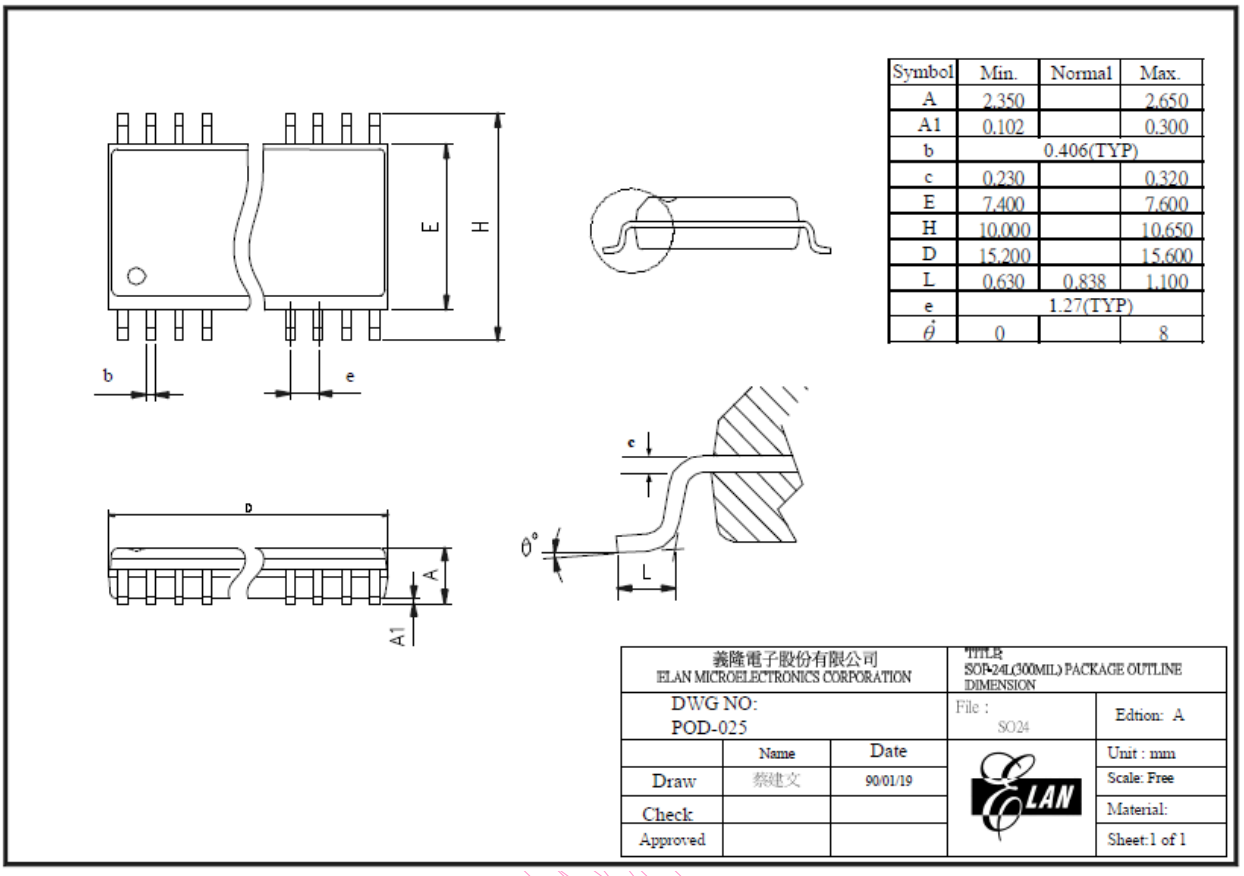


Symbol	Min.	Normal	Max.
A	2.350		2.650
A1	0.102		0.300
b		0.406(TYP)	
c	0.230		0.320
E	7.400		7.600
H	10.000		10.650
D	12.600		12.900
L	0.630	0.838	1.100
e		1.27(TYP)	
θ	0		8

義隆電子股份有限公司 ELAN MICROELECTRONICS CORPORATION			TITLE: SOP-20L(300MIL) PACKAGE OUTLINE DIMENSION	
DWG NO: POD-024			File : SO20	Edtion: A
	Name	Date		Unit : mm
Draw	蔡建文	90/01/19		Scale: Free
Check				Material:
Approved				Sheet:1 of 1

Figure C-2 EM85F765N 20-Pin SOP Package Type

C.3 EM85F765NSO24 300mil

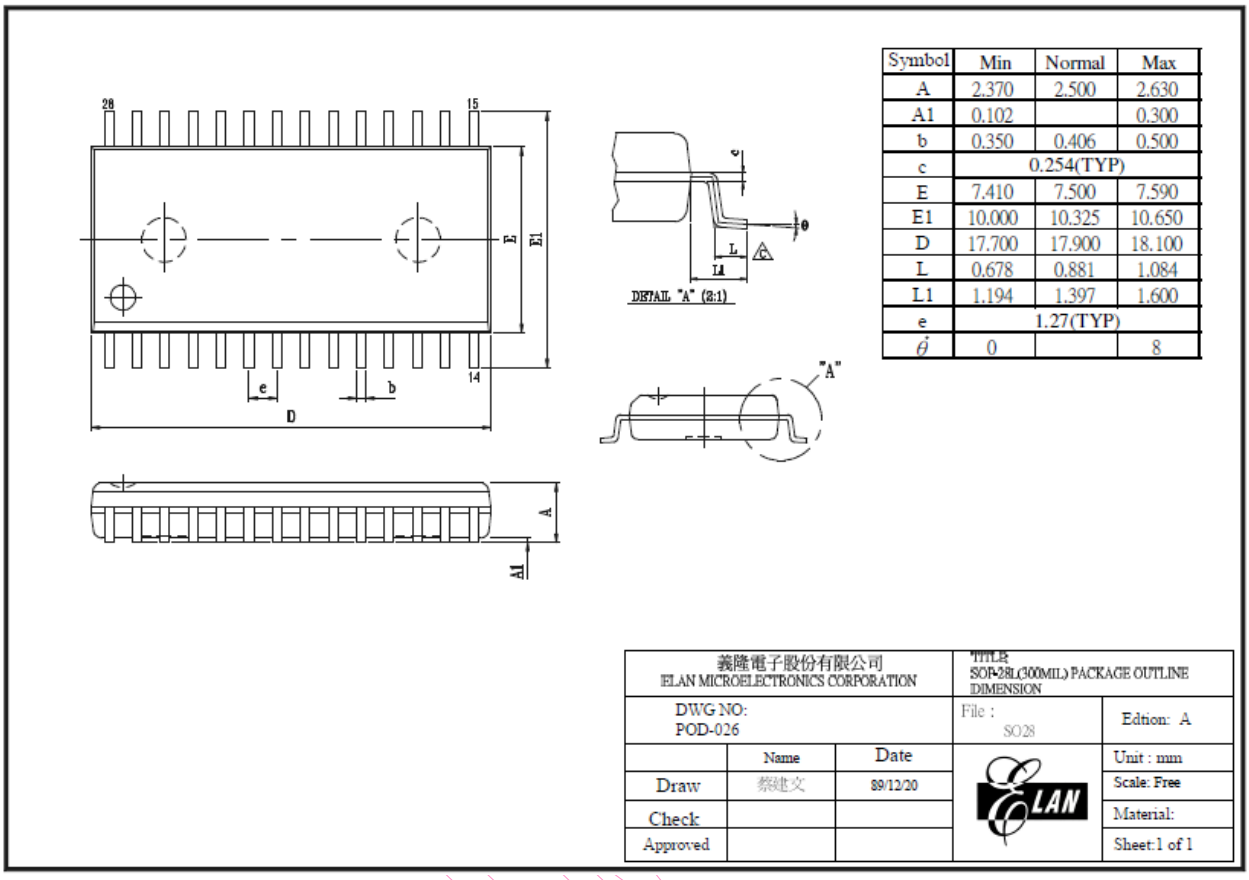


Symbol	Min.	Normal	Max.
A	2.350		2.650
A1	0.102		0.300
b		0.406(TYP)	
c	0.230		0.320
E	7.400		7.600
H	10.000		10.650
D	15.200		15.600
L	0.630	0.838	1.100
e		1.27(TYP)	
θ	0		8

義隆電子股份有限公司 ELAN MICROELECTRONICS CORPORATION			TITLE: SOP-24L(300MIL) PACKAGE OUTLINE DIMENSION
DWG NO: POD-025			File : SO24
	Name	Date	Unit : mm
Draw	蔡建文	90/01/19	Scale: Free
Check			Material:
Approved			Sheet:1 of 1

Figure C-3 EM85F765N 24-Pin SOP Package Type

C.4 EM85F765NSO28 300mil



Symbol	Min	Normal	Max
A	2.370	2.500	2.630
A1	0.102		0.300
b	0.350	0.406	0.500
c		0.254(TYP)	
E	7.410	7.500	7.590
E1	10.000	10.325	10.650
D	17.700	17.900	18.100
L	0.678	0.881	1.084
L1	1.194	1.397	1.600
e		1.27(TYP)	
θ	0		8

義隆電子股份有限公司 ELAN MICROELECTRONICS CORPORATION			TITLE: SOP-28L(300MIL) PACKAGE OUTLINE DIMENSION	
DWG NO: POD-026			File : SO28	Edtion: A
	Name	Date		Unit : mm
Draw	蔡建文	89/12/20		Scale: Free
Check				Material:
Approved				Sheet:1 of 1

Figure C-4 EM85F765N 28-Pin SOP Package Type

D Quality Assurance and Reliability

Test Category	Test Conditions	Remarks
Solderability	Solder temperature=245±5°C, for 5 seconds up to the stopper using a rosin-type flux	–
Pre-condition	Step 1: TCT, 65°C (15 min)~150°C (15 min), 10 cycles	For SMD IC (such as SOP, QFP, SOJ, etc)
	Step 2: Bake at 125°C, TD (endurance)=24 hrs	
	Step 3: Soak at 30°C/60%，TD (endurance)=192 hrs	
	Step 4: IR flow 3 cycles (Pkg thickness ≥ 2.5 mm or Pkg volume ≥ 350 mm^3 ----225±5°C) (Pkg thickness ≤ 2.5 mm or Pkg volume ≤ 350 mm^3 ----240±5°C)	
Temperature cycle test	-65°C (15mins)~150°C (15min), 200 cycles	–
Pressure cooker test	TA =121°C, RH=100%, pressure=2 atm, TD (endurance)= 96 hrs	–
High temperature / High humidity test	TA=85°C , RH=85%，TD (endurance) = 168 , 500 hrs	–
High-temperature storage life	TA=150°C, TD (endurance) = 500, 1000 hrs	–
High-temperature operating life	TA=125°C, VCC = Max. operating voltage, TD (endurance) = 168, 500, 1000 hrs	–
Latch-up	TA=25°C, VCC = Max. operating voltage, 150mA/20V	–
ESD (HBM)	TA=25°C, ≥｜± 8KV｜	IP_ND,OP_ND,IO_ND IP_NS,OP_NS,IO_NS IP_PD,OP_PD,IO_PD, IP_PS,OP_PS,IO_PS, VDD-VSS(+),VDD_VSS(-) mode
ESD (MM)	TA=25°C, ≥ ｜± 700V｜	

D.1 Address Trap Detect

An address trap detect is one of the MCU embedded fail-safe functions that detects MCU malfunction caused by noise or the like. Whenever the MCU attempts to fetch an instruction from a certain section of ROM, an internal recovery circuit is auto started. If a noise-caused address error is detected, the MCU will repeat execution of the program until the noise is eliminated. The MCU will then continue to execute the next program.