EM85F684A_682A

USB Microcontroller

Product Specification

DOC. VERSION 1.1

ELAN MICROELECTRONICS CORP.
March. 2018

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 1st Road
Hsinchu Science Park
Hsinchu, TAIWAN 30076
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

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

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

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

1 General Description ···· 1

2 Features ···· 2

3 Application ···· 4

4 Pin Assignment ···· 4

5 Pin Description ···· 6

6 Block Diagram ···· 10

7 Functional Description ···· 11

7.1 Special Function Registers (SFR) ···· 11

7.1.1 SFR Paging ···· 11

7.1.2 SFR Map ···· 12

7.2 Stack ···· 14

7.3 System Clock ···· 16

7.4 System Reset ···· 17

7.4.1 Power-On Reset ···· 17

7.4.2 Watchdog Timer Reset ···· 17

7.4.3 Software Reset ···· 17

7.4.4 Low Voltage Detector ···· 17

7.4.5 Register ···· 18

7.5 Operation Mode Structure ···· 20

7.5.1 Normal/Green Mode ···· 20

7.5.2 Idle Mode ···· 21

7.5.3 Power-Down Mode ···· 21

7.6 Flash Program Memory ···· 22

7.6.1 In-Circuit Programming (ICP) ···· 23

7.6.2 In-Application Programming (IAP) ···· 23

7.7 Port Configuration ···· 23

7.7.1 Push-Pull Output Mode ···· 28

7.7.2 Input-Only Mode ···· 29

7.8 Interrupts ···· 30

7.8.1 Interrupt Sources ···· 30

7.8.2 Interrupt Priority ···· 36

7.8.3 INT PIN ···· 40

7.8.4 Pin-Change ···· 43

7.9 Watchdog Timer (WDT) ···· 45

7.10 Peripheral Power Down ···· 47

7.11 Universal Serial Bus (USB) 49
7.12 I^{2}C (Inter-Integrated Circuit) 67
7.12.1 7-Bit Slave Address 68
7.12.2 10-Bit Slave Address 69
7.12.3 I2CB Register 73
7.13 UART 83
7.13.1 UART Mode 0: Synchronous 83
7.13.2 UART Mode 1: 8-bit UART (Variable Baud Rate) 84
7.13.3 UART Mode 2: 9-bit UART (Fixed Baud Rate) 84
7.13.4 UART Mode 3: 9-BIT UART (Variable Baud Rate) 85
7.13.5 UART Baud Rates 85
7.14 Timer 88
7.14.1 Timer 0 and Timer 1 88
7.14.1.1 Timer 0/1 Mode 0: 13-bit Counter/Timer 88
7.14.1.2 Timer 0/1 Mode 1: 16-bit Counter/Timer 89
7.14.1.3 Timer 0/1 Mode 2: 8-bit Counter/Timer with Auto-Reload 89
7.14.1.4 Timer 0 Mode 3 Two 8-bit Counter/Timers 90
7.13.2 Timer 3 Register 94
7.15 PWM 98
7.15.1 Increment Timer Counter (TMRX) 99
7.15.2 PWM Time Period (PRDX) 99
7.15.3 PWM Duty Cycle (DTX) 99
7.15.4 PWM Programming Process/Steps 100
7.15.5 PWM Register 101
7.16 SPI 120
7.16.1 SPI Signal 120
7.16.1.1 Serial Clock (SCK) 120
7.16.1.2 Master Data Out Slave Data-In (MOSI) 120
7.16.1.3 Master Data in Slave Data Out (MISO) 120
7.16.1.4 Slave Select (/SS) 120
7.16.2 SPI Transfer Format 120
7.16.3 Shift Data Register (SPISFDR) 121
7.16.4 Operations 121
7.16.4.1 Master Mode Operation 121
7.16.4.2 Slave Mode Operation 121
7.16.5 Timing Diagram 122
7.16.5.1 SPI Master Mode 122
7.16.5.2 SPI Slave Mode Timing (CPHA=1 / CHPA=0) 123
7.16.5.3 Consecutively Receiving Bytes (CPHA=1 / CHPA=0) 124
7.16.5.4 SPI Slave Timing Requirement 125
7.16.6 Buffer mode function 127
7.16.7 SPI Register 128
7.17 EEPROM 135
7.18 On-Chip Debug Support (OCDS) 137
7.18.1 Limitations of On-Chip Debug 137
7.19 Code Option 139

7.20 CPU Register Descriptions 141
7.21 Instruction Set 145

8 Absolute Maximum Ratings 149

9 DC Electrical Characteristics 150

10 Power Sequence 152

11 Application Circuit 153

12 Package Information 154

Specification Revision History

Version	Revision Description	Date
1.0	Preliminary version	2018/04/02
1.1	Modify EPOUTPREHOLD & EPINPREHOLD bit Modify Application Circuit	2018/05/07

1 General Description

EM85F684A/EM85F682A is a high-performance Flash version of the 80C52 single-chip 8-bit microcontroller with full speed USB functions.

EM85F684A/EM85F682A features a full-speed USB module compatible with the USB specifications Version 2.0. This module integrates the USB transceivers with a 3.3V voltage regulator and the Serial Interface Engine (SIE). USB Event detection logic (Reset and Suspend/Resume) and FIFO buffers supporting the mandatory control Endpoint (EP0) and up to four versatile Endpoints (EP1/EP2/EP3/EP4) with minimum software overhead are also part of the USB module. EM85F684A/EM85F682A retains the features of the 80C52 with 256 bytes RAM, 2048/1024 bytes XRAM, two 16-bit Timers/Counters (T0/T1), a full duplex UART and an on-chip oscillator. EM85F684A/EM85F682A also has built-in peripheral functions such as SPI, two I2C, 6 PWMs, etc.

2 Features

- Operating Voltage**:**
  - USB: 4.25V ~ 5.5V at 0°C~70°C (commercial)
  - Non-USB: 3V ~ 5.5V at 0°C ~70°C (commercial)
- Memory Configuration:
  - Flash ROM size:
    - EM85F684A : 32K x 8 bits (Including ISP boot code 8K bytes)
    - EM85F682A : 16K x 8 bits (Including ISP boot code 8K bytes)
  - 2K x 8 bits for EEPROM
    - EM85F684A : When use EEPROM function, Flash ROM size = 30K x 8 bits.
    - EM85F682A : When use EEPROM function, Flash ROM size = 16K x 8 bits.
  - 10,000 write/erase cycles
  - More than 10 years data retention
  - RAM:
    - EM85F684A : 2048 x 8 bits XRAM / EM85F682A : 1024 x 8 bits XRAM
    - 256 x 8 bits RAM
    - USB FIFO: 320 bytes (64 bytes for EP0 and 256 bytes for EP1~EP4).
- USB Specification Compliance:
  - Full-speed Universal Serial Bus Specification Version 2.0.
  - P75 (D+) has an internal pull-high resistor (1.5 KΩ).
  - 1 Control Endpoint (EP0).
  - 4 Bulk/Interrupt/Isochronous Endpoint (EP1~EP4).
  - Programmable EP1~EP4 FIFO depth.
- I/O Ports
  - Each GPIO pin of Ports 5, 6, 7, 8, 9, has an internal programmable pull-high resistor.
  - Pin change wakeup: Ports 5, 6, 9
  - Output Ports have high drive/sink mode
- Interrupt:
  - Internal: TIMER0/1, Timer3, WDT, I2CA/B, SPI, PWMA /B/C/D/E/F, USB, UART
  - External: INT0~INT3, I/O pin change interrupt
- Two 16-bit Timer Counters: T0/T1
- One 8-bit Timer: T3
- Operation Frequency:
  - XTAL Mode: 8/16 MHz
  - PLL24M : Source Clock From XTAL 8/16MHz.
  - Internal High RC Mode: 8/16/24 MHz

  - Internal Low RC Mode: 256K/32K/4K/500Hz

- Operation Mode:
  - Normal Mode: Both high clock and low clock active.
  - Green Mode: Low clock only
  - Idle Mode: CPU clock halt, peripheral clock still in oscillation.
  - Power-Down Mode: Both high clock and low clock stop.

- Serial Peripheral Interface (SPI)
  - 4-wire synchronous with 8-byte buffer
  - 2~12 MHz Master/Slave

- Two sets Inter-Integrated Circuit (I2CA/I2CB).
  - Master/Slave with 7/10 bits address and 8 bits data transmit/receive.
  - Baud rate: 400k/100kbps.

- Six sets Pulse Width Modulation (PWM).
  - Maximum: 6 channels PWM output.
  - 16-bit resolution PWM output.

- Package Type:
  - ◆ 48-pin QFN (5 x 5 x 0.6mm) EM85F684AAW / EM85F682AAW
  - ◆ 24-pin QFN (4 x 4 x 0.8mm) EM85F684ADW

3 Application

USB Gamming Keyboard

USB Joystick

USB Docking

USB Docking+ Battery Charger

4 Pin Assignment

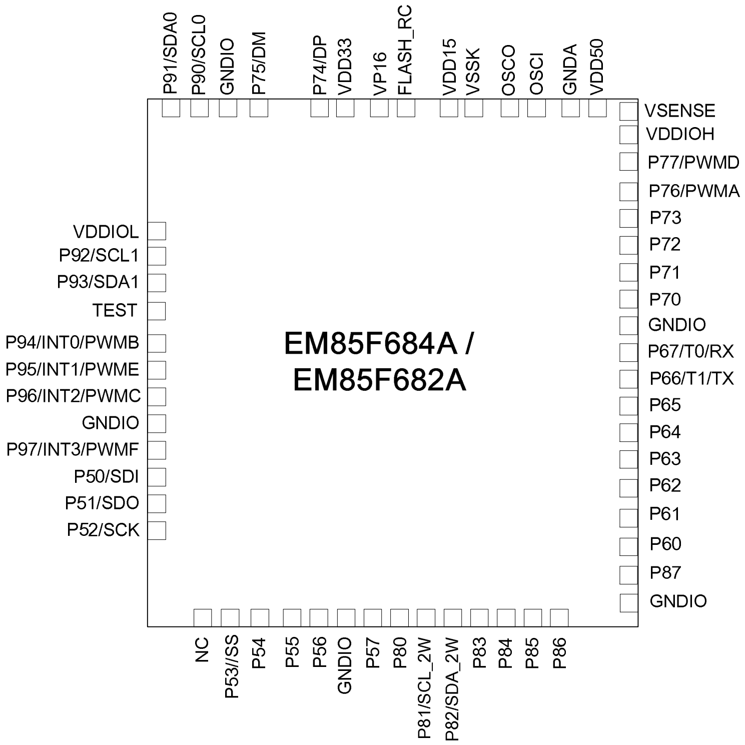
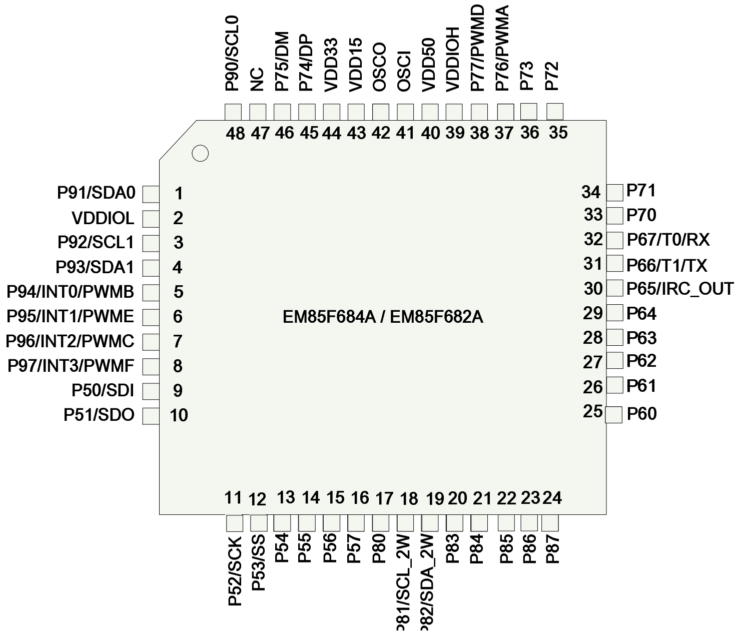


Figure 4-1 DIE

(The Test pin must be bonded to the ground)



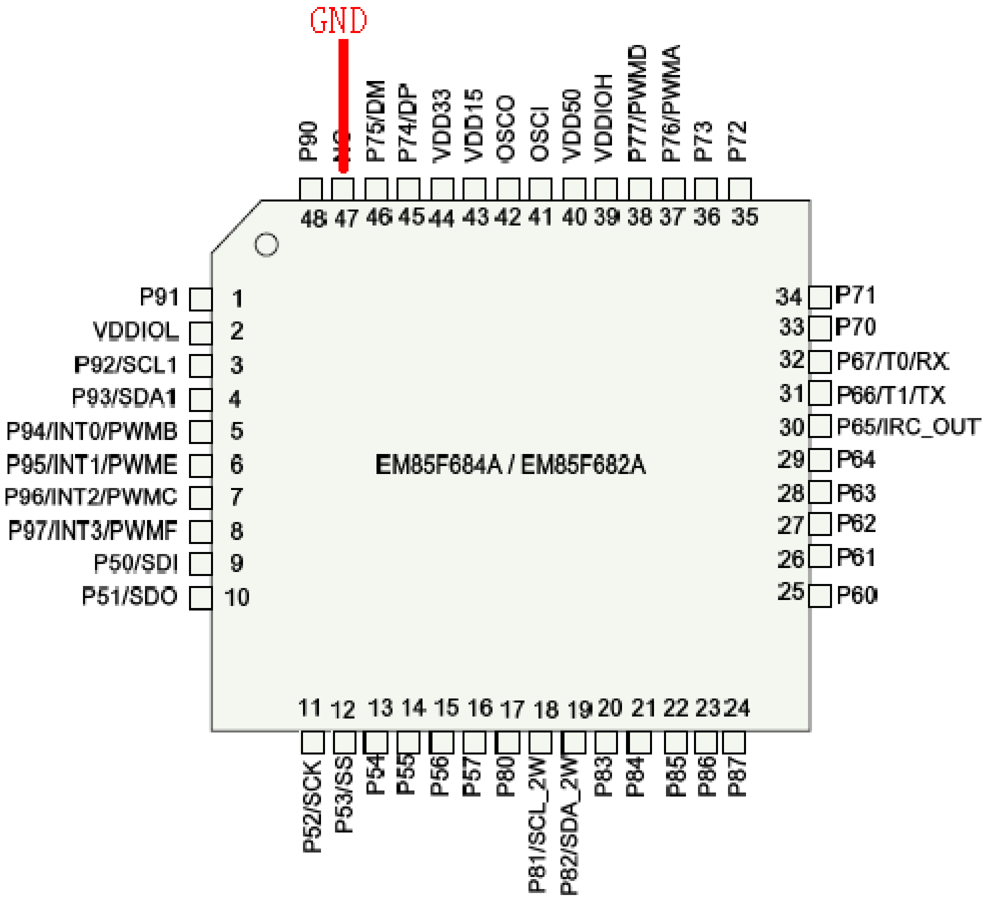


*Figure 4-2 EM85F684A/EM85F682A QFN48 (5*5*0.6mm)*

5 Pin Description

Name	Function	Input Type	Description
VDD50	VDD	Power	Power supply input pins for digital, analog and USB circuit.
VDD50A	VDD	Power	Power supply input pins for analog circuit.
VDD15	V1p5	Power	V1.5 output
VDD33	V33	Power	V33 output (Provide P74/75, P90/91 power)
VSS	Vss	I	Ground input. The package substrate must be connected to Ground.
VDDIOH	I/O VDD	Power	Power supply input pins for the I/O power (P5/6/P70~73,76,77/P8, P94~97 I/O power input)
VDDIOL	I/O VDD	Power	Power supply input pins for the I/O power of P92, P93
VSSK	VSS	Power	Kernel Ground
GNDIO	I/O Ground	Power	I/O Ground
GNDA	GNDA	Power	Analog and USB Ground
OSCI	OSCI	I	Crystal input
OSCO	OSCO	O	Crystal output
P50/SDI (MOSI)	P50	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor.
	DSI	ST	SPI serial data input.
P51/SDO(MISO)	P51	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor.
	SDO	ST	SPI serial data output
P52/SCK	P52	ST	Bi-directional I/O pin with programmable pull-down, pull-high, open-drain, and pin change wake-up
	SCK	ST	SPI serial clock input/output
P53//SS	P53	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor.
	SS	ST	SPI slave mode enable
P54	P54	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P55	P55	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P56	P56	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P57	P57	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor

Name	Function	Input Type	Description
P60	P60	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
P61	P61	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
P62	P62	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
P63	P63	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
P64	P64	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
P65/ IRC_OUT	P65	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	IRC_OUT	ST	IRC clock output pin
	RESET	ST	Hardware Reset Pin. Select from Code Option 3.
P66	P66	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	T1	ST	T1 external input pin
	TX	ST	UART TX port
	RESET	ST	Hardware Reset Pin. Select from Code Option 3.
P67	P6V7	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	T0	ST	T0 external input pin
	RX	ST	UART RX port
P70	P70	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P71	P71	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P72	P72	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor

Name	Function	Input Type	Description
P73	P73	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P74/ DP	P74	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor (3.3V IO)
	DP	ST	USB Differential D+ signal line.
P75/DM	P75	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor (3.3V IO)
	DM	ST	USB Differential D- signal line.
P76 / PWMA	P76	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
	PWMA	ST	Pulse width modulation output A.
P77 / PWMD	P77	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
	PWMD	ST	Pulse width modulation output D.
P80	P80	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
	USB IAP	ST	**USB IAP pin**. Input pull-high, when this pin is low level voltage after power on, entry boot code mode.
P81 / 2W_CLK	P81	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor **UBRG debug pin，this pin must set input**
	2W_CLK	ST	OCD clock line.
P82 / 2W_DATA	P82	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor **UBRG debug pin，this pin must set input**
	2W_DATA	ST	OCD data line.
P83	P83	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P84	P84	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P85	P85	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor

Name	Function	Input Type	Description
P86	P86	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P87	P87	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor
P90	P90	ST	3.3V Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	SCL0	ST	I^{2}CA serial clock line. It is open-drain.
	RESET	ST	Hardware Reset Pin. Select from Code Option 3.
P91	P91	ST	3.3V Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	SDA0	ST	I^{2}CA serial data line. It is open-drain
P92/SCL1	P92	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	SCL1	ST	I2CB serial clock line. It is open-drain.
P93/SDA1	P93	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	SDA1	ST	I2CB serial data line. It is open-drain
P94/INT0/PWMB	P94	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	INT0	ST	External interrupt pin.
	PWMB	ST	Pulse width modulation Output B.
P95/INT1/PWME	P95	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	PWME		Pulse width modulation Output E.
	INT1	ST	External interrupt pin.
P96/INT2/PWMC	P96	ST	Bi-direction pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	INT2	ST	External interrupt pin.
	PWMC	ST	Pulse width modulation output C.
	RESET	ST	Hardware Reset Pin. Select from Code Option 3.
P97/INT3/PWMF	P97	ST	Bi-directional pin. Schmitt trigger structure as input mode. Built-in pull-up resistor and pin change wake-up
	INT3	ST	External interrupt pin.
	PWMF	ST	Pulse width modulation Output F.

Note: P81 and 82 are UBRG debug pins, user must set these two pins as input floating when using OCD debug.

6 Block Diagram

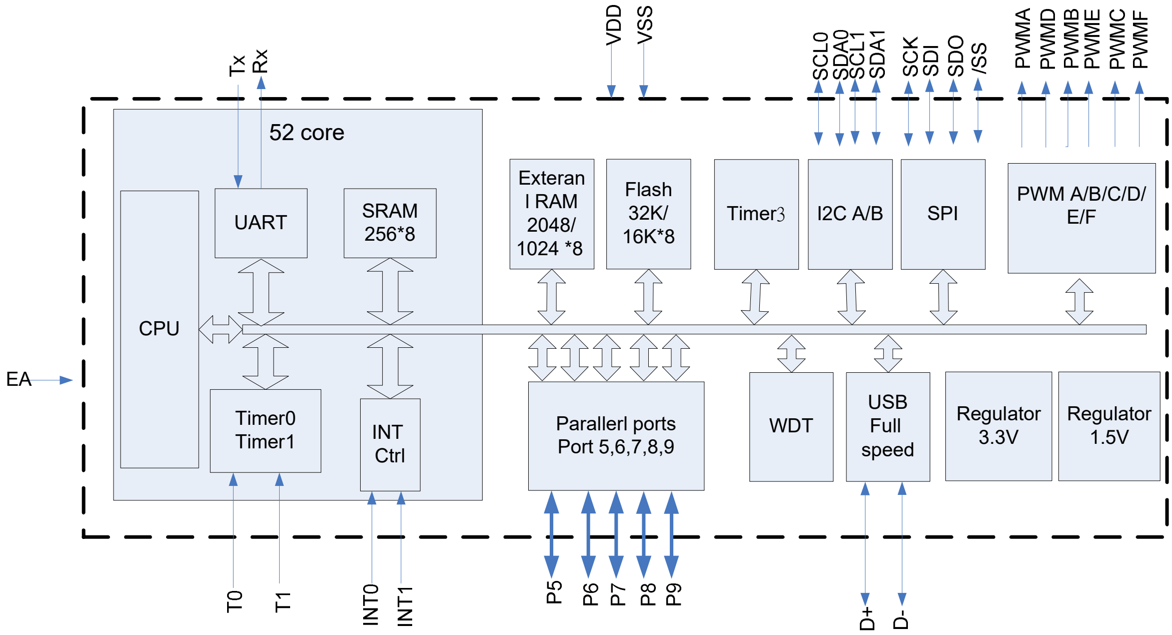


Figure 6-1 EM85F684A/682A Functional Block Diagram

7 Functional Description

7.1 Special Function Registers (SFR)

The direct-access data memory locations from 0x80 to 0xFF constitute the special function registers (SFRs). The SFR registers are accessed anytime the direct addressing mode is used to access memory locations from 0x80 to 0xFF. SFRs with addresses ending in 0x0 or 0x8 are bit-addressable as well as byte-addressable.

7.1.1 SFR Paging

EM85F684A/EM85F682A implements a 'paged' SFR scheme which can expand the number of available SFR addresses, user can switch SFR page by setting PAGESW register (0x86) Bit 0.

PAGESW: Page Switch

Bit	7	6	5	4	3	2	1	0
Name	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	PAGE SW.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

7.1.2 SFR Map

PAGE0	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)
F8	-	P5M0	P6M0	P7M0	P8M0	PHCON0	-	PHDSC0
F0	B	-	-	-	-	PHCON1	-	PHDSC1
E8	P9	-	-	-	-	-	WDTKEY	WDTCR
E0	ACC	I2CBCR4	-	-	-	-	-	-
D8	PSW1	I2CBCR1	I2CBCR2	I2CBSA	I2CBDB	I2CBDAL	I2CBDAH	I2CBCR3
D0	PSW0	-	-	-	-	-	-	I2CBINT
C8	-	-	-	LVDCR	LVDCR1	DEVPD1	DEVPD2	DEVPD3
C0	RSTSC	RSTSC	I2CACR1	I2CACR2	I2CASA	I2CADB	I2CADAL	I2CADAH
B8	IP	EIP1	EIP2	EIP3	-	PRST	-	-
B0	P8	EIE1	EIE2	EIE3	-	EXEN	EIESC1	EIESC2
A8	IE	TC3CR1	TC3CR2	TC3DA	TC3DB	-	ICEN	EEXSF
A0	P7	SPICON1	SPICON2	SPITDBR	SPIRDBR	SPISR1	SPISR2	SPITX
98	SCON	SBUF	SPIRX	SPIBUFPTR1	SPIBUFPTR2	-	-	-
90	P6	-	-	-	-	-	-	XPAGE
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON0	CKCON1
80	P5	SP	DPL	DPH	DPL1	DPH1	PAGESW	PCON

Figure 7.1.1 SFR Memory Map in Page 0

PAGE1	0 (8)	1 (9)	2 (A)	3 (B)	4 (C)	5 (D)	6 (E)	7 (F)
F8	-	P9M0	-	-	-	PHCON2	-	PHDSC2
F0	B	PWMCCR	PRDCL	PRDCH	DTCL	DTCH	TMRCL	TMRCH
E8	P9	PWMBCR	PRDBL	PRDBH	DTBL	DTBH	TMRBL	TMRBH
E0	ACC	PWMACR	PRDAL	PRDAH	DTAL	DTAH	TMRAL	TMRAH
D8	PSW1	-	USBCTRL	-	-	-	PWMTER	PWMSF
D0	PSW0	UDCEP4DATAOUTCNT	UDCEP1BUFDEPTH	UDCEP2BUFDEPTH	UDCEP3BUFDEPTH	UDCEP4BUFDEPTH	PHYTEST0	-
C8	-	UDCEP1DATAINCNT	UDCEP1DATAOUTCNT	UDCEP2DATAINCNT	UDCEP2DATAOUTCNT	UDCEP3DATAINCNT	UDCEP3DATAOUTCNT	UDCEP4DATAINCNT
C0	RSTSC	UDCEP1BUFDATA	UDCEP2BUFDATA	UDCEP3BUFDATA	UDCEP4BUFDATA	UDCBUFSTA	-	-
B8	IP	UDCINT0STA	UDCINT1STA	UDCINT2STA	UDCEPCTRL	UDCEPBUFCTRL0	UDCEPBUFCTRL1	UDCEP0BUFDATA
B0	P8	UDCCTRL	UDCSTA	UDCCFSTA	UDCCFDATA	UDCINT0EN	UDCINT1EN	UDCINT2EN
A8	IE	-	-	-	-	-	-	-
A0	P7	-	-	-	-	-	-	-
98	SCON	SBUF	UDCEPBUFCTRL2	UDCEPBUFCTRL3	UDCEPBUFCTRL4	STDCMDFLG	-	-
90	P6	-	-	-	-	EEPCTRL	EEPAREASEL	-
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON0	CKCON1
80	P5	SP	DPL	DPH	DPL1	DPH1	PAGESW	PCON

Figure 7.1.2 SFR Memory Map in Page 1

PAGE2	0 (8)	1 (9)	2 (A)	3 (B)	4 (C)	5 (D)	6 (E)	7 (F)
F8	-	-	-	-	-	-	-	-
F0	B	PWMFCR	PRDFL	PRDFH	DTFL	DTFH	TMRFL	TMRFH
E8	P9	PWMECR	PRDEL	PRDEH	DTEL	DTEH	TMREL	TMREH
E0	ACC	PWMDCR	PRDDL	PRDDH	DTDL	DTDH	TMRDL	TMRDH
D8	PSW1	-	-	-	-	-	PWMTER1	PWMSF1
D0	PSW0	-	-	-	-	-	-	-
C8	-	-	-	-	-	-	-	-
C0	RSTSC	-	-	-	-	-	-	-
B8	IP	UDCEP1AB UFDATA	UDCEP2AB UFDATA	UDCEP3AB UFDATA	UDCEP4AB UFDATA	UDCINT4E N	UDCINT4S TA	-
B0	P8	UDCINT3E N	UDCINT3S TA	UDCEPBU FCTRL5	UDCEPBU FCTRL6	UDCEPBU FCTRL7	UDCEPBU FCTRL8	UDCEPBU FCTRL9
A8	IE	-	-	-	-	-	-	-
A0	P7	-	-	-	-	-	-	-
98	SCON	SBUF	-	-	-	-	-	-
90	P6	-	-	-	-	-	-	-
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON0	CKCON1
80	P5	SP	DPL	DPH	DPL1	DPH1	PAGESW	PCON

Figure 7.1.3 SFR Memory Map in Page 2

7.2 Stack

A programmer's stack can be located anywhere in the 256-byte data memory. The stack area is designated using the Stack Pointer (SP, 0x81) SFR. The SP will point to the last location used. The next value pushed on the stack is placed at SP+1 and then SP is incremented. (The Stack Pointer is incremented before data is stored during PUSH and CALL execution and decremented after data is popped during POP, RET and RETI execution) A reset initializes the stack pointer to Location 0x07, as shown in Figure 7.2.1. Therefore, the first value pushed on the stack is placed at Location 0x08, which is also the first register (R0) of register bank 1. Thus, if more than one register bank is to be used, the SP should be initialized to a location in the data memory not being used for data storage. The stack depth can extend up to 256 bytes.

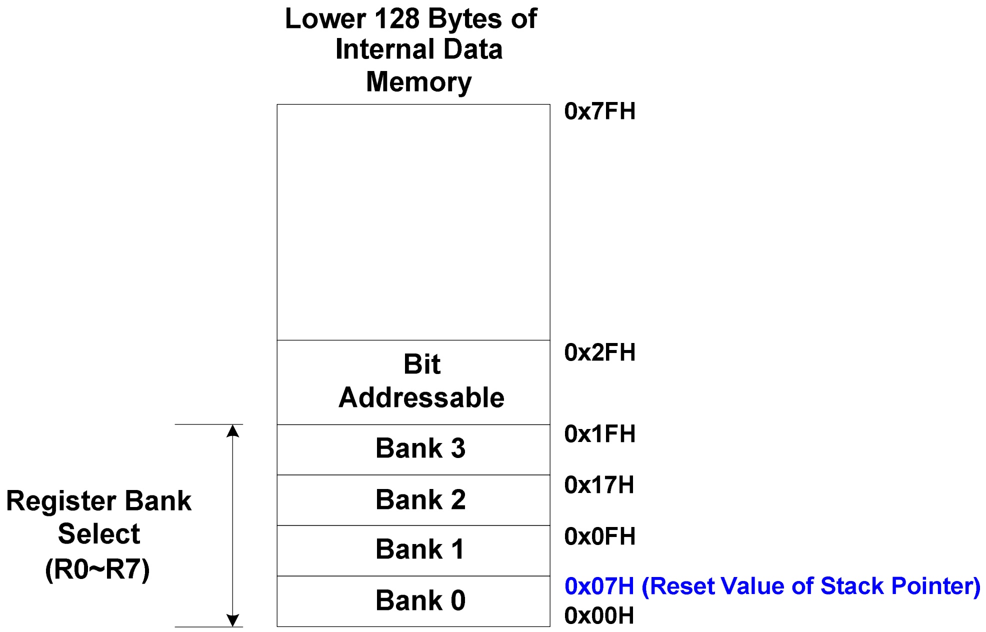

Lower 128 Bytes of Internal Data Memory
0x7FH
Bit Addressable
0x2FH
Bank 3
0x1FH
Bank 2
0x17H
Bank 1
0x0FH
Bank 0
0x07H (Reset Value of Stack Pointer)
0x00H
Register Bank Select (R0~R7)

Figure 7.2.1 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

7.3 System Clock

- EM85F684A/EM85F682A is a dual clock system including high-speed and low-speed clocks. The high-speed clock includes internal high-speed oscillator and external oscillators selected by “FSOSC” code option. The low-speed clock is from internal low-speed oscillator controlled by “LOWFR1,0” bit of CKCON1 register. Both high-speed clock and low-speed clock can be system clock source.

  - High Speed clock source:

    1. External 8/16MHz Crystal
    2. Internal 8/16/24 MHz RC Oscillator (default)
    3. PLL24MHz : Clock source from External 8/16MHz Crystal.

  - Low speed clock:

    Internal RC Oscillator 512kHz divide to 500Hz/4K/32K/256kHz for CPU clock

  - Peripherals clock source:

    - UART/ Time0&1/SPI: System clock
    - I2C: F_{HS}
    - Timer3/ PWM: F_{HS}, F_{512kHz}

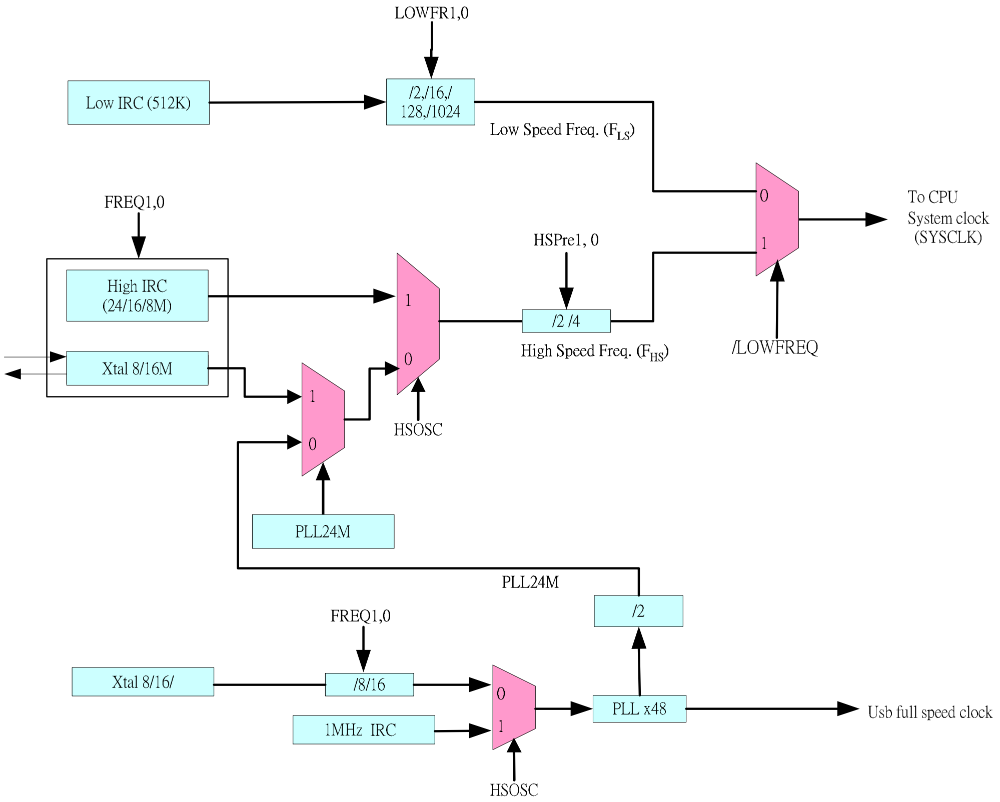


Figure 7.3.1 Clock system

7.4 System Reset

When entering the Reset state, the CPU stop operating and the Interrupt and Timer are disabled, System Reset all of SFR are initialized to the default value, the Program Counter (PC) is reset, the program is executed by the Reset Vector (0000H), during the Reset period, the internal data memory content does not change (stored data before Reset will remain unchanged), all I/O registers are set to an initial value (I/O as the Default mode). EM85F684A/EM85F682A has three sources of reset: power-on reset, watchdog reset and software reset.

7.4.1 Power-On Reset

Power-On Reset (POR) during power-up will carry out System Reset. The CPU will stop operating and maintain the Reset state until VDD rises to above POR voltage. When VDD drops to below POR voltage then it will again carry out Reset System to return to the Reset state, but a POR occurs and PORSF (RSTSC.0) will be set to 1, it can help user to confirm whether to start operating from the next CPU Power-On condition. PORSF bit is set by hardware, cleared by Software.

7.4.2 Watchdog Timer Reset

When WDT is enabled to start the counter, WTSF will be set by WDT overflow. If WDTE (WDTCR.7) is enabled, the WDT overflow will force a system reset that causes CPU to restart. After exiting from reset status, the software can read the WTSF to recognize that WDT reset occurred. For WDT details, refer to Section 10 Watchdog Timer. Note that when POR occurs, WTSF will be cleared.

7.4.3 Software Reset

The SWRSF (RSTSC.7) is set to 1 will force to generate a System Reset, when Reset is caused by a Software Reset, after exiting the Reset state, SWRSF (RSTSC.7) is set to 1. Note that when POR occurs, SWRSF will be cleared.

7.4.4 Low Voltage Detector

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。

7.4.5 Register

RSTSC: Reset Source

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

SFR Address = 0xC0; SFR Page = All Pages

Bit 7: Software Reset Force and Flag.

Write:

0: No effect

1: System reset

Read:

0: No software reset that occurs.

1: Software reset caused a system reset.

Bits 6~4, 1, 3 : Reserved. Read = 0, Write = Don't Care.

Bit 2: Watchdog Reset Flag.

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

Bit 0: Power-On Flag.

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

P: Previous status before reset

Reset Event	Reset Flag		
	SWRSF	WTSF	PORSF
Power-On Reset	0	0	1
Watchdog Reset	P	1	P
Software Reset	1	P	P
OCD Reset	0	0	1

Table 7.4 Reset Flag Status after Reset Event Occur

PRST: Peripheral Reset

Bit	7	6	5	4	3	2	1	0
Name	--	--	--	--	UARTRST	SPIRST	I2CBRST	I2CARST
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBD; SFR Page = 0

Bit 3: UART reset bit. Reset UART registers and pin state as default, initialize state of the machine.

Bit 2: SPI reset bit. Reset SPI registers and pin state as default, initialize state of the machine.

Bit 1: I2CB reset bit. Reset I2CB registers and pin state as default, initialize state of the machine.

Bit 0: I2CA reset bit. Reset I2CA registers and pin state as default, initialize state of the machine.

LVDCR: LVD Control Register

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

SFR Address = 0xCB; SFR Page = 0
Bit 7: Low voltage detector enable bit.

0: Disable

1: Enable
*PS : In Sleep Mode, Low voltage occur than Wake-up MCU.

LVDCR1: LVD Control Register 1

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

SFR Address = 0xCC; SFR Page = 0
Bit 7: Low Voltage Detector Status Flag.

7.5 Operation Mode Structure

The EM85F684A/EM85F682A supports three states: Normal State, Idle State and Power-Down state. In normal state, system clock can be used high-speed clock (Normal Mode) or low-speed oscillator (Green Mode) to reduce power consumption

Figure 7.5.1 presents the operation state structure in EM85F684A/EM85F682A. Definition 7.5.1 describes the Power Control Register (PCON) used to control the EM85F684A/EM85F682A Power-down and Idle power management modes.

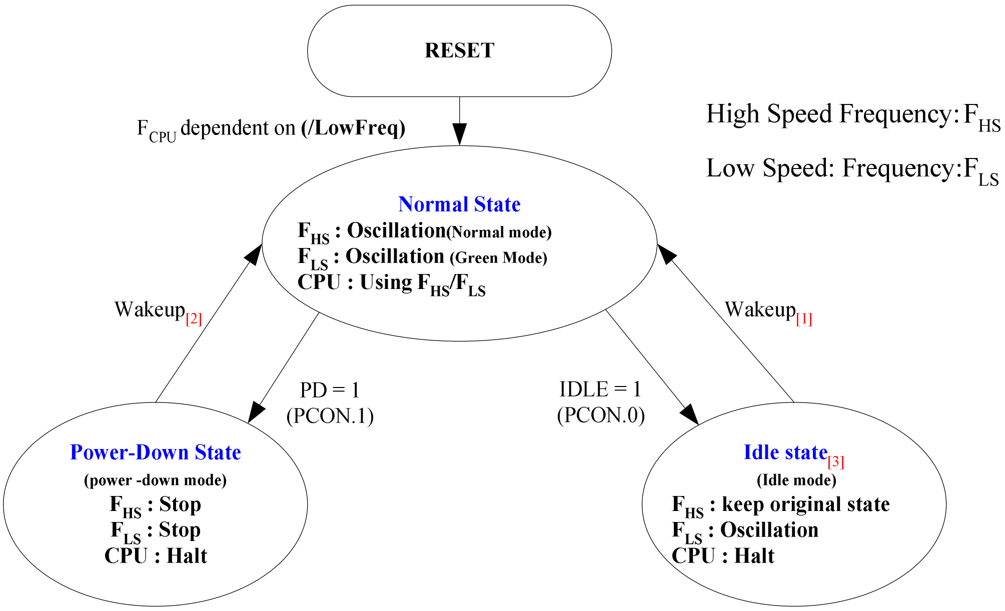

RESET
F_{CPU} dependent on (/LowFreq)
High Speed Frequency: F_{HS}
Low Speed: Frequency: F_{LS}
Normal State
F_{HS} : Oscillation(Normal mode)
F_{LS} : Oscillation (Green Mode)
CPU : Using F_{HS}/F_{LS}
Wakeup[2]
Wakeup[1]
PD = 1 (PCON.1)
IDLE = 1 (PCON.0)
Power-Down State
(power -down mode)
F_{HS} : Stop
F_{LS} : Stop
CPU : Halt
Idle state[3]
(Idle mode)
F_{HS} : keep original state
F_{LS} : Oscillation
CPU : Halt

Figure 7.5.1 Operation Mode Structure

[1].Any enable interrupt source or reset.
[2].External interrupts, Pin change, POR, WDT, USB, etc.
[3].CPU is halt, but system clock does not stop.

7.5.1 Normal/Green Mode

In normal/green mode, CPU and peripherals continue to execute. When setting the LowFreqB (CKCON1.0), it will cause the controller core to enter low-speed oscillator (green mode).

Setting LowFreqB (CKCON1.0) allows System clock to switch between high speed and low speed. When the LowFreqB is set to “0”, F_{HS} oscillator source is automatically stopped. When the LowFreqB is set to “1”, the System clock will switch to F_{HS}.

Normal / Green Mode Features:

1. All analog and digital peripheral remains active.
2. System clock can choose to operate at high speed (normal mode) or low-speed mode (green mode)

7.5.2 Idle Mode

Setting the Idle mode select bit (PCON.0) entering the mode, this mode will stop the CPU. However the digital and analog peripherals continue to operate. In this mode, all of enabled Interrupt and Reset can wake up.

Idle Mode Features:

1. The CPU is halt, all analog and digital peripheral remains active.
2. **CPU can wake up by Reset or any enabled Interrupt.**
3. RAM and SFR contents remain unchanged.

7.5.3 Power-Down Mode

Setting the Power-Down mode select bit (PCON.1) can enter this mode. In this mode, the CPU and the digital peripheral will halt automatically. (Except INT, WDT, and except USB, if these function are enabled.)

Power-Down Mode Features:

1. The CPU, analog and digital peripheral stop (Exceptions: INT, Pin-change, WDT and USB function. These functions / interrupts remain active if they are enabled before entering power down mode.)
2. When WDT is enabled, F_{LS} will not stop and will cause WDT Timeout reset.

Wakeup Source:

1. WDT timeout (WDTE = 1)
2. Pin-change interrupt is enabled (including INT0 and INT1).
3. When Power on reset occurs.
4. When I2C is enabled, and receiving a valid I2C address.
5. When USB is enabled and receives USB resume signal.

Note: NOP instruction must be added after Idle/ power down command.

7.6 Flash Program Memory

The Flash Program Memory is the location where the user code or program is stored.

Two ways to program Flash Memory:

(1)By using ICP (ISP)

(2)By using Elan IAP Tool

Once cleared to Logic 0, a Flash bit must be erased to set it back to Logic 1. Typically, Flash bytes are erased (set to 0xFF) before being reprogrammed.

7.6.1 In-Circuit Programming (ICP)

In-Circuit Programming allows programming and reprogramming the microcontroller. This is simply done with two lines for clock and data and two other lines for power and ground.

7.6.2 In-Application Programming (IAP)

The EM85F684A/EM85F682A supports In-Application Programming (IAP), allowing the program memory to be modified during execution.

7.7 Port Configuration

All port pins of the EM85F684A/EM85F682A may be configured in one of the two modes as follows:

(1)Output (Internal pull high automatically disconnected when switching output)

(2)Input-Only

Port modes may be assigned in software on a pin-by-pin basis as shown in Table 7.7.1 The data registers is listed in Table 7.7.2, it can set the Port Latch logic value or 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 as other functions, changing the IOCX register value to change the I/O mode is not allowed. The corresponding register is allowed to change the I/O mode only when other functions are disabled.

Table 7.7.1 Port Configuration Modes

PxM0	Port Mode
0	Push-Pull Output
1	Input Only (High Impedance)

(x = 5, 6, 7, 8, 9; y = 0, 1, 2, 3, 4, 5, 6, 7)

0: Push-Pull Output

1: Input-Only

Table 7.7.2 Port Configuration Mode 0 Registers

Register	7	6	5	4	3	2	1	0
P5M0 (0xF9)	P5M0.7	P5M0.6	P5M0.5	P5M0.4	P5M0.3	P5M0.2	P5M0.1	P5M0.0
P6M0 (0xFA)	P6M0.7	P6M0.6	P6M0.5	P6M0.4	P6M0.3	P6M0.2	P6M0.1	P6M0.0
P7M0 (0xFB)	P7M0.7	P7M0.6	P7M0.5	P7M0.4	P7M0.3	P7M0.2	P7M0.1	P7M0.0
P8M0 (0xFC)	P8M0.7	P8M0.6	P8M0.5	P8M0.4	P8M0.3	P8M0.2	P8M0.1	P8M0.0
P9M0 (0xF9)*	P9M0.7	P9M0.6	P9M0.5	P9M0.4	P9M0.3	P9M0.2	P9M0.1	P9M0.0
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

***:The Register P9M0 is in PAGE 1.**

Table 7.7.3 Port Data Registers

Register	7	6	5	4	3	2	1	0
P5 (0x80)	P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0
P6 (0x90)	P6.7	P6.6	P6.5	P6.4	P6.3	P6.2	P6.1	P6.0
P7 (0xA0)	P7.7	P7.6	P7.5	P7.4	P7.3	P7.2	P7.1	P7.0
P8 (0xB0)	P8.7	P8.6	P8.5	P8.4	P8.3	P8.2	P8.1	P8.0
P9 (0xE8)	P9.7	P9.6	P9.5	P9.4	P9.3	P9.2	P9.1	P9.0
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

1. When writing to the data register,
0: Set the output latch to Logic LOW,
1: Set the 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	P5PH.7	P5PH.6	P5PH.5	P5PH.4	P5PH.3	P5PH.2	P5PH.1	P5PH.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 = 0xFD; SFR Page = 0

Bits 7~0: Set these bits to enable Port 5 Pull-High.

PHCON1: Pull-High Control 1

Bit	7	6	5	4	3	2	1	0
Name	P6PH.7	P6PH.6	P6PH.5	P6PH.4	P6PH.3	P6PH.2	P6PH.1	P6PH.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 = 0xF5; SFR Page = 0

Bits 7~0: Set these bits to enable Port 6 Pull-High.

PHCON2: Pull-High Control 2

Bit	7	6	5	4	3	2	1	0
Name	--	--	P9PH1	P9PH0	P8PH1	P8PH0	P7PH1	P7PH0
Type	--	--	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xFD; SFR Page = 1

Bit 5: Port 9 pull-high enable control on high nibble.

0: Disable the P9.7 ~ P9.4 pull-high

1: Enable the P9.7 ~ P9.4 pull-high

Bit 4: Port 9 pull-high enable control on low nibble.

0: Disable the P9.3 ~ P9.0 pull-high

1: Enable the P9.3 ~ P9.0 pull-high

Bit 3: Port 8 pull-high enable control on high nibble.

0: Disable the P8.7 ~ P8.4 pull-high

1: Enable the P8.7 ~ P8.4 pull-high

Bit 2: Port 8 pull-high enable control on low nibble.

0: Disable the P8.3 ~ P8.0 pull-high

1: Enable the P8.3 ~ P8.0 pull-high

Bit 1: Port 7 pull-high enable control on high nibble.

0: Disable the P7.7 ~ P7.4 pull-high

1: Enable the P7.7 ~ P7.4 pull-high

Bit 0: Port 7 pull-high enable control on low nibble.

0: Disable the P7.3 ~ P7.0 pull-high

1: Enable the P7.3 ~ P7.0 pull-high

PHDSC0: Port High Drive/Sink Control 0

Bit	7	6	5	4	3	2	1	0
Name	P5PHS.7	P5PHS.6	P5PHS.5	P5PHS.4	P5PHS.3	P5PHS.2	P5PHS.1	P5PHS.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 = 0xFF; SFR Page = 0

Bits 7~0: Set these bits to enable Port 5 High Drive/Sink.

PHDSC1: Port High Drive/Sink Control 1

Bit	7	6	5	4	3	2	1	0
Name	P6PHS.7	P6PHS.6	P6PHS.5	P6PHS.4	P6PHS.3	P6PHS.2	P6PHS.1	P6PHS.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 = 0

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

PHDSC2: Port High Drive/Sink Control 2

Bit	7	6	5	4	3	2	1	0
Name	-	-	P9PHS1	P9PHS0	P8PHS1	P8PHS0	P7PHS1	P7PHS0
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xFF; SFR Page = 1

Bit 7: Port 5 high drive/sink enable control on high nibble.

0: Disable the P5.7 ~ P5.4 high drive/sink.

1: Enable the P5.7 ~ P5.4 high drive/sink.

Bit 6: Port 5 high drive/sink enable 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 9 high drive/sink enable control on high nibble.

0: Disable the P9.7 ~ P9.4 high drive/sink.

1: Enable the P9.7 ~ P9.4 high drive/sink

Bit 4: Port 9 high drive/sink enable control on low nibble.

0: Disable the P9.3 ~ P9.0 high drive/sink.

1: Enable the P9.3 ~ P9.0 high drive/sink.

Bit 3: Port 8 high drive/sink enable control on high nibble.

0: Disable the P8.7 ~ P8.4 high drive/sink.

1: Enable the P8.7 ~ P8.4 high drive/sink.

Bit 2: Port 8 high drive/sink enable control on low nibble.

0: Disable the P8.3 ~ P8.0 high drive/sink.

1: Enable the P8.3 ~ P8.0 high drive/sink.

Bit 1: Port 7 high drive/sink enable control on high nibble.

0: Disable the P7.7 ~ P7.4 high drive/sink.

1: Enable the P7.7 ~ P7.4 high drive/sink.

Bit 0: Port 7 high drive/sink enable control on low nibble.

0: Disable the P7.3 ~ P7.0 high drive/sink.

1: Enable the P7.3 ~ P7.0 high drive/sink.

7.7.1 Push-Pull Output Mode

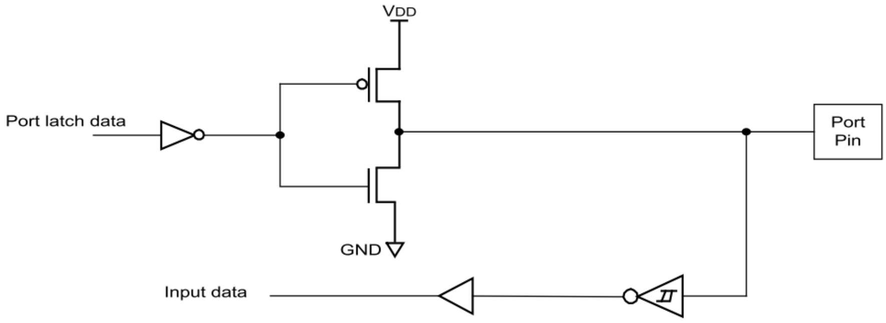


Figure 7.7.1 Push-Pull Output Mode

7.7.2 Input-Only Mode

In Input-Only mode port is input only, no output capability. Input-Only mode port structure diagram is shown in Figure 7.7.2 Input-Only Mode.

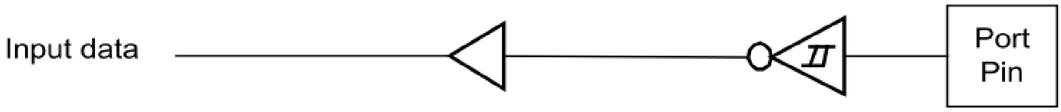


Figure 7.7.2 Input-Only Mode

(This diagram is for reference only, based on the actual circuit design)

7.8 Interrupts

EM85F684A/EM85F682A include an extended interrupt system supporting multiple interrupt sources with two priority levels. The allocation of interrupt sources between on-chip peripherals and external inputs pins varies according to the specific version of the device. Each interrupt source has one or more associated interrupt flag located in an SFR. When a peripheral or external source meets a valid interrupt condition, the associated interrupt flag is set to Logic 1.

If interrupts are enabled for the source, an interrupt request is generated when the interrupt flag is set. As soon as execution of the current instruction is complete, the CPU generates an LCALL to a predetermined address to begin execution of an interrupt service routine (ISR). Each ISR must end with an RETI instruction, which returns program execution to the next instruction that would have been executed if the interrupt request had not occurred. If interrupts are not enabled, the interrupt flag is ignored by the hardware and program execution continues as normal (The interrupt flag is set to Logic 1 regardless of the interrupt's enable/disable state).

Each of the interrupt sources can be individually enabled or disabled through the SFR relevant enabled interrupt. At the same time, the EA (IE.7) must also be set to “1” to activate the enabled interrupt master switch, so that each of the interrupt sources that have been enabled becomes effective. No matter what is bit setting of each enabled interrupt, clearing EA will disable all the interrupts. When CPU enters ISR, a few number of interrupt flags are automatically cleared by hardware while most of them can be cleared only with software while remaining in ISR state. If the interrupt flag setting remains at “1” after the CPU completes the return from interrupt (RETI) instruction, it will immediately generate a new interrupt request. After executing the next instruction, the CPU will then re-enter ISR.

7.8.1 Interrupt Sources

All the interrupt sources are listed in Table 7.8. If Interrupt Mask is activated and an Interrupt Source occurs, the system will generate an interrupt request and set the Interrupt Flag, the CPU will jump to the address corresponding to the ISR (Interrupt Vector), not through the hardware automatically cleared Interrupt Flag, must use software manually cleared.

Priority Order	Interrupt Vector	Interrupt Source	Interrupt Flag	Clear by HW	Interrupt Mask
Top	0x0000	Reset	None	N/A	Always Enabled
0	0x0003	External Interrupt 0 INT0	IE0 (TCON.1)	Y	EX0 (IE.0)
1	0x000B	Timer 0 Overflow	TF0 (TCON.5)	Y	ET0 (IE.1)
2	0x0013	External Interrupt 1 INT1	IE1 (TCON.3)	Y	EX1 (IE.2)
3	0x001B	Timer 1 Overflow	TF1 (TCON.7)	Y	ET1 (IE.3)
4	0x0023	UART	RI0 (SCON0.0) TI0 (SCON0.1)	N	ES0 (IE.4)
5	0x002B	-	-	-	-
6	0x0033	Pin Change Interrupt	IC5SF (ICEN.0) IC6SF (ICEN.1) IC9SF (ICEN.3)	N	ICIE (IE.6)
7	0x003B	LVD	LVDSF (LVDCR1.7)	N	LVDIE(EIE1.0)
8	0x0043	System-Hold	SHSF (PSW1.0)	N	SHIE (EIE1.1)
9	0x004B	-	-	-	-
10	0x0053	External Interrupt 2~3 INT2~3	EXSF2 (EEXSF.0) EXSF3 (EEXSF.1)	N	EEXIE (EIE1.3)
11	0x005B	SPI	SPISF (SPISR2.7)	N	SPIE (EIE1.4)
12	0x0063	PWMD	PWMDDSF (PWMSF1.0) PWMDPSF (PWMSF1.1)	N	PWMDIE (EIE1.5)
13	0x006B	-	-	-	-
14	0x0073	Timer 3	TC3SF(TC3CR1.4)	N	TCIE(EIE1.7)
15	0x007B	PWMA	PWMADSF (PWMSF.0) PWMAPSF (PWMSF.1)	N	PWMAIE (EIE2.0)
16	0x0083	PWME	PWMEDSF (PWMSF1.2) PWMEPSF (PWMSF1.3)	N	PWMDIE (EIE2.1)
17	0x008B	USB	SOFSF (UDCINT0STA.7) SUSPSF (UDCINT0STA.6) RSTSF (UDCINT0STA.5) LPMRESSF (UDCINT0STA.4) RESSF (UDCINT0STA.3) SETUPSF (UDCINT0STA.1) EXTPKGSF(UDCINT0STA.0) EP3OUTSF (UDCINT1STA.7) EP3INSF (UDCINT1STA.6) EP2OUTSF (UDCINT1STA.5) EP2INSF (UDCINT1STA.4)	N	USBIE (EIE2.2)

			EP1OUTSF (UDCINT1STA .3) EP1INSF (UDCINT1STA.2) EP0OUTSF (UDCINT1STA.1) EP0INSF (UDCINT1STA.0) EP4INEMPSF (UDCINT2STA.7) EP3INEMPSF (UDCINT2STA.6) EP2INEMPSF (UDCINT2STA.5) EP1INEMPSF (UDCINT2STA.4) EP0INEMPSF (UDCINT2STA.3) EP4OUTSF (UDCINT2STA.3) EP4INSF (UDCINT2STA.2) ERRSF(UDCINT2STA.0)		
18	0x0093	PWMF	PWMFDSF (PWMSF1.4) PWMFPSF (PWMSF1.5)	N	PWMEIE (EIE2.3)
19	0x009B	-	-	-	-
20	0x00A3	I2CA	I2CATSF (I2CASF.0) I2CARSF (I2CASF.1) I2CASTPSF (I2CASF.2)	N	I2CAIE (EIE2.5)
21	0x00AB	-	-	-	-
22	0x00B3	-	-	-	-
23	0x00BB	PWMB	PWMBDSF (PWMSF.2) PWMBPSF (PWMSF.3)	N	PWMBIE (EIE3.0)
24	0x00C3	PWMC	PWMCDSF (PWMSF.4) PWMCPSF (PWMSF.5)	N	PWMCIE (EIE3.1)
25	0x00CB	-	-	-	-
26	0x00D3	-	-	-	-
27	0x00DB	-	-	-	-
28	0x00E3	I2CB	I2CBTXSF(I2CBINT.7) I2CBRXSF (I2CBINT.6) I2CBSTPSF (I2CBINT.4) TO1SF (SMbusTO1.6) TO2SF (SMbusTO2.6) TO3SF (SMbusTO3.6)	N	I2CBIE (EIE3.5)

Table 7-9 Interrupt Sources

IE: Interrupt Enable

Bit	7	6	5	4	3	2	1	0
Name	EA	ICIE	-	ES0	ET1	EX1	ET0	EX0
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 = 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 Interrupt

This bit sets the masking of the Pin Change interrupt.

0: Disable Pin Change interrupt.

1: Enable Pin Change interrupt.

Bit 5: Reserved

Bit 4: Enable UART Interrupt

This bit sets the masking of the UART interrupt.

0: Disable UART interrupt.

1: Enable UART 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: Enable External Interrupt 1.

This bit sets the masking of External Interrupt 1.

0: Disable external Interrupt 1.

1: Enable Interrupt requests generated by the INT1 input.

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: Enable External Interrupt 0.

This bit sets the masking of External Interrupt 0.

0: Disable external Interrupt 0.

1: Enable Interrupt requests generated by the INT0 input.

EIE1: Extended Interrupt Enable 1

Bit	7	6	5	4	3	2	1	0
Name	TCIE	-	PWMDIE	SPIIE	EEXIE	-	SHIE	LVDIE
Type	R/W	R	R/W	R/W	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB1; SFR Page = Page 0

Bits 6, 2: Reserved

Bit 7: Timer 3 Overflow Interrupt Enable.

0: Disable Timer 3 Overflow Interrupt

1: Enable Timer 3 Overflow Interrupt

Bit 5: PWMD Interrupt Enable.

This bit sets the masking of the PWMD interrupt.

0: Disable PWMD interrupt.

1: Enable PWMD interrupt

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: Extended External Interrupt Enable (INT2~3).

This bit sets the masking of the Extended External interrupt.

0: Disable Extended External interrupt.

1: Enable Extended External 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: Low Voltage Detector (LVD) Interrupt Enable.

This bit sets the masking of the LVD interrupt.

0: Disable LVD interrupt.

1: Enable LVD interrupt

EIE2: Extended Interrupt Enable 2

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	PWMFIE	USBIE	PWMEIE	PWMAIE
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 =Page 0

Bits 7~4: Reserved

Bit 3: PWMF Interrupt Enable.

This bit sets the masking of the PWMF interrupt.

0: Disable PWMF interrupt.

1: Enable PWMF interrupt

Bit 2: USB Interrupt Enable.

This bit sets the masking of the USB interrupt.

0: Disable USB interrupt.

1: Enable USB interrupt

Bit 1: PWME Interrupt Enable.

This bit sets the masking of the PWME interrupt.

0: Disable PWME interrupt.

1: Enable PWME 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	-	-	I2CBIE	-	-	-	PWMCIE	PWMBIE
Type	R	R	R/W	R	R	R	R/W	R/W
Reset	0	0	R/W	0	0	0	0	0

SFR Address = 0xB3; SFR Page =Page 0

Bits 7, 6, 4, 3, 2: Reserved

Bit 5: I2CB Interrupt Enable.

This bit sets the masking of the I2CB interrupt.

0: Disable I2CB interrupt.

1: Enable I2CB interrupt

Bit 1: PWMC Interrupt Enable.

This bit sets the masking of the PWMC interrupt.

0: Disable PWMC interrupt.

1: Enable PWMC interrupt

Bit 0: PWMB Interrupt Enable.

This bit sets the masking of the PWMB interrupt.

0: Disable PWMB interrupt.

1: Enable PWMB interrupt

7.8.2 Interrupt Priority

IP: Interrupt Priority

Bit	7	6	5	4	3	2	1	0
Name	-	PIC	-	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

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

Bit 4: UART Interrupt Priority Control.

This bit sets the priority of the UART interrupt.

0: UART interrupt set to low priority level.

1: UART 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	-	-	PSP	PEEX	-	PSH	-
Type	R/W	R	R	R/W	R/W	R	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB9; SFR Page = 0

Bits 5, 2, 0: Reserved

Bit 7: Timer 3 Interrupt Priority Control.

This bit sets the priority of the Timer3 interrupt.

0: Timer 3 interrupt set to low priority level.

1: Timer 3 interrupt set to high priority level.

Bit 6: Reserved

Bit 4: Serial Peripheral Interface (SPI) Interrupt Priority Control.

This bit sets the priority of the SPI interrupt.

0: SPI interrupt set to low priority level.

1: SPI interrupt set to high priority level.

Bit 3: Extended External Interrupt Priority Control (INT2~3).

This bit sets the priority of the Extended External interrupt.

0: Extended External interrupt set to low priority level.

1: Extended External 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.

EIP2: Extended Interrupt Priority 2

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

SFR Address = 0xBA; SFR Page = 0

Bits 7, 6, 4, 3, 1: Reserved

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 2: USB Interrupt Priority Control.

This bit sets the priority of the USB interrupt.

0: USB interrupt set to low priority level.

1: USB 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	-	-	PI2CB	-	-	-	PPWMC	PPWMB
Type	R	R	R/W	R	R	R	R/W	R/W
Reset	0	0	R/W	0	0	0	0	0

SFR Address = 0xBB; SFR Page = 0

Bit 7~4, 3~2: Reserved. Read = 0, Write = Don't Care.

Bit 5: I2CB Interrupt Priority Control.

This bit sets the priority of the I2CB interrupt.

0: I2CB interrupt set to low priority level.

1: I2CB interrupt set to high priority level.

Bit 1: 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 0: 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.

7.8.3 INT PIN

The INT0~3 are external interrupt sources. The INT0 and INT1 pins are activated at level or by an edge by setting or clearing bit IT0 (TCON.0) or IT1 (TCON.2), note the INT2~3 is enabled by EXENx in EXEN register. EIESC1 and EIESC2 configure the port change detection level on falling or rising event. 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).

TCON: Timer Control

Bit	7	6	5	4	3	2	1	0
Name	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
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 = 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: External Interrupt 1 Status Flag.

This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 1 service routine.

Bit 2: External Interrupt 1 Type Select.

This bit selects whether the configured INT1 interrupt will be edge or level sensitive. INT1 is configured falling edge triggered or rising edge triggered by the EIES1 bit in register EIESC1.

0: INT1 is low level triggered.

1: INT1 is edge triggered.

Bit 1: External Interrupt 0 Status Flag.

This flag is set by hardware when an edge/level of type defined by IT0 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 0 service routine.

Bit 0: External Interrupt 0 Type Select.

This bit selects whether the configured INT0 interrupt will be edge or level sensitive. INT0 is configured falling edge triggered or rising edge triggered by the EIES0 bit in register EIESC1.

0: INT0 is low level triggered.

1: INT0 is edge triggered.

EXEN: External Interrupt Pin Enable

Bit	7	6	5	4	3	2	1	0
Name	--	--	--	--	--	--	EXEN3	EXEN2
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 =Page0

Bits 7~2: Reserved.

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

EIESC1: External Interrupt Edge Select Control 1

Bit	7	6	5	4	3	2	1	0
Name	EIEDG1	EIEDG0	-	-	-	EIES1	-	EIES0
Type	R/W	R/W	R	R	R	R	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB6; SFR Page = Page 0

Bits 6~7: INT Deglitch Time.

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

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

Bit 2: INT1 Edge Select bit.

0: INT1 input is active falling edge triggered.

1: INT1 input is active rising edge triggered.

Bit 0: INT0 Edge Select bit.

0: INT0 input is active falling edge triggered.

1: INT0 input is active rising edge triggered.

EIESC2: External Interrupt Edge Select Control 2

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	EIES3	EIES2
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB7; SFR Page = 0

Bits 7~2: Reserved

Bit 1: External Interrupt 3 Edge Select.

0: INT3 input is active falling edge triggered.

1: INT3 input is active rising edge triggered.

Bit 0: External Interrupt 2 Edge Select.

0: INT2 input is active falling edge triggered.

1: INT2 input is active rising edge triggered.

EEXSF: Extended External Interrupt Status Flag

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	EXSF3	EXSF2
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAF; SFR Page = 0

Bits 7~2: Reserved

Bit 1: External Interrupt 3 Status Flag.

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

Bit 0: External Interrupt 2 Status Flag.

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

7.8.4 Pin-Change

The EM85F684A/EM85F682A Pin-Change function covers three Ports, namely, IC5EN, IC6EN and IC9EN, to control whether the Pin-Change of each Port is enabled, once enabled, the Port of any I/O will the corresponding IC0SF as long as there is a signal like Trigger. IC1SF, IC2SF or IC3SF flag is set to “1” and an interrupt occurs.

ICEN: Input-Change Enable

Bit	7	6	5	4	3	2	1	0
Name	IC9EN	Reserved	IC6EN	IC5EN	IC9SF	Reserved	IC6SF	IC5SF
Type	R/W	R	R/W	R/W	R/W	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAE; SFR Page = 0

Bit 7: Input-Change Port 9 Enable.

Setting this bit activates Input-Change Port 9.

0: Disable Input-Change Port 9.

1: Enable Input-Change Port 9.

Bit 6: Reserved

Bit 5: Input-Change Port 6 Enable.

Setting this bit activates Input-Change Port 6.

0: Disable Input-Change Port 6.

1: Enable Input-Change Port 6.

Bit 4: Input-Change Port 5 Enable.

Setting this bit activates Input-Change Port 5.

0: Disable Input-Change Port 5.

1: Enable Input-Change Port 5.

Bit 3: Input-Change Port 9 Status Flag.

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

Bit 2: Reserved

Bit 1: Input-Change Port 6 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 5 Status Flag.

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

7.9 Watchdog Timer (WDT)

WDTCR: WDT Control Register

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

SFR Address = 0xEF; 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 2-0: Watchdog prescaler

Watchdog Timer base on 8ms

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	1.128
⋮	⋮	⋮	⋮	⋮
1	1	1	1	1.128

WDTKEY: WDT Key

Bit	7	6	5	4	3	2	1	0
Name	Key.7	Key.6	Key.5	Key.4	Key.3	Key.2	Key.1	Key.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 = 0xEE; SFR Page = 0

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

WDTKEY written 0xB1: Disable Watchdog Timer (WDTE is first set to “0”)

WDTKEY written 0x4E: Clear Watchdog Timer

7.10 Peripheral Power Down

The EM85F789N 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 Register

Bit	7	6	5	4	3	2	1	0
Name	-	PWMAPD	TC3PD	-	SPIPD	-	LVDPD	-
Type	R	R/W	R/W	R	R/W	R	R/W	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCD; SFR Page = 0

Bits 7, 4, 2, 0: Reserved.

Bit 6: PWMA Power Down Mode.

0: Normal.

1:Power Down.

Bit 5: Timer3 Power Down Mode.

0: Normal.

1:Power Down.

Bit 3: SPI Power Down Mode.

0: Normal.

1:Power Down.

Bit 1: LVD Power Down Mode.

0: Normal.

1:Power Down.

DEVPD2: Device2 Power Down Register

Bit	7	6	5	4	3	2	1	0
Name	-	PWMCPD	I2CBPD	-	PWMBPD	-	-	I2CAPD
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

Bits 7, 4, 2, 1: Reserved.

Bit 6: PWMC Power Down Mode.

0: Normal.

1:Power Down.

Bit 5: I2CB Power Down Mode.

0: Normal.

1:Power Down.

Bit 3: PWMB Power Down Mode.

0: Normal.

1:Power Down.

Bit 0: I2CA Power Down Mode.

0: Normal.

1:Power Down.

DEVPD3: Device3 Power Down Register

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	-	-	-	-	-	PWMFPD	PWMEPD	PWMDPD
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 = 0xCF; SFR Page = 0

Bits 7, 6, 5, 4, 3: Reserved.

Bit 2: PWMF Power Down Mode.

0: Normal.

1:Power Down.

Bit 1: PWME Power Down Mode.

0: Normal.

1:Power Down.

Bit 0: PWMD Power Down Mode.

0: Normal.

1:Power Down.

7.11 Universal Serial Bus (USB)

USBCTRL: USB Control Register

Bit	7	6	5	4	3	2	1	0
Name	IOUSB	EPOUTPRE HOLD	USBSLEP RESUME	--	--	REG33PD	PHYPD	PLLPD
Type	R/W	R/W	R/W	--	--	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

SFR Address = 0xDA ; SFR Page = 1

Bit 7: IOUSB: 1: USB, 0: GPIO

Bit 6: EPx out buffer verification to simultaneously sharing access problem

Bit 5: UDC Suspend changes from high to low, H/W is set to “1” and notifies the MCU must wake up (even under sleep command), wake up again by F/W clear

Bit 2: REG33PD. 1: power down

Bit 1: PHY power down. High active. When chip uses USB PHY I/O function, PHY_PD must always be set to “L”. (Stable time needs 5 μs) (When not using USB, must be set to “1”)

Bit 0: PLL power down. 1: power down

STDCMDFLG: USB Standard Command Flag Register

Bit	7	6	5	4	3	2	1	0
Name	--	--	--	--	SETADD R	SETFEA TURE	CLRFEA TURE	SETCON FIG
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9D ; SFR Page = 1

Bits 7~4: Reserved

Bit 3: SET ADDRESS command status flag

Bit 2: SET FEATURE command status flag

Bit 1: CLEAR FEATURE command status flag

Bit 0: SET CONFIGURATION command status flag

UDCCTRL: UDC Control Register

Bit	7	6	5	4	3	2	1	0
Name	--	UDCEN	DEVRES UME	STALL	EPINPR EHOLD	UDCSUS P	UDCRST RDY	CLK48O FF
Type	--	R/W	R/W	R/W	R/W	R/W	R	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB1 ; SFR Page = 1

Bit 7: Reserved.

Bit 6: UDC enable signal for timer warm-up and power control. The control bit works in “System Clock”.

1: UDC Device Enable

0: UDC Device Disable

Bit 5: When the device wants to resume bus activity, this bit is activated (ud_resume = DEV_Resume generate).

The control bit works in “System Clock”.

Bit 4: Stall the control endpoint request

Bit 3: EPx in buffer verification to simultaneously share accessing problems

Bit 2: UDC suspend status. High active.

Bit 1: USB reset ready. It is activated when both USB PLL reset and device reset signals are pulled-down.

Bit 0 : This bit can turn off USB 48 MHz clock input for power control. The control bit works in “System Clock”.

0 : Clock On

1 : Clock off

UDCSTA: UDC Status Register

Bit	7	6	5	4	3	2	1	0
Name	--	--	CURIF1	CURIF0	CURALT	CUREP2	CUREP1	CUREP0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XB2 ; SFR Page = 1

Bits 7~6: Reserved

Bits 5~4: Current state of interface

Bit 3: Current state of alternate

Bits 2~0: Current state of endpoint

UDCCFSTA: UDC Load Configuration Status Register

Bit	7	6	5	4	3	2	1	0
Name	EPCFGRDY	EPCFGDN	--	--	--	--	--	--
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xB3; SFR Page = 1

Bit 7: EPINFO block configuration ready for next (clear after read)

Bit 6: EPINFO block configuration done

Bits 5~0: Reserved

UDCCFDATA: UDC Load Configuration Data Register

Bit	7	6	5	4	3	2	1	0
Name	**CFEPDT7**	**CFEPDT6**	**CFEPDT5**	**CFEPDT4**	**CFEPDT3**	**CFEPDT2**	**CFEPDT1**	**CFEPDT0**
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: Configure data to EPINFO block

UDC Configuration:

UDCCFDATA needs to fill a total of 5 Bytes set value, as follows

1. First Byte : Endpoint 1 set value

    (1) Bits 7~6 : EP1 configuration value

    (2) Bits 5~4 : EP1 interface value

    (3) Bits 3~2 : EP1 alternate value

    (4) Bits 1~0 : EP1-In transfer type

        (a) 11 : interrupt mode

        (b) 10 : bulk mode

        (c) 01 : isochronous mode

        (d) 00 : control mode

2. Second Byte : Endpoint 2 set value

    (1) Bits 7~6 : EP1 configuration value

    (2) Bits 5~4 : EP1 interface value

(3) Bits 3~2 : EP1 alternate value

(4) Bits 1~0 : EP1-In transfer type

(a) 11 : interrupt mode

(b) 10 : bulk mode

(c) 01 : isochronous mode

(d) 00 : control mode

3.Third Byte : Endpoint 3 set value

(1) Bits 7~6 : EP1 configuration value

(2) Bits 5~4 : EP1 interface value

(3) Bits 3~2 : EP1 alternate value

(4) Bits 1~0 : EP1-In transfer type

(a) 11 : Interrupt mode

(b) 10 : Bulk mode

(c) 01 : Isochronous mode

(d) 00 : Control mode

4. Fourth Byte : Endpoint 4 set value

(1) Bits 7~6 : EP1 configuration value

(2) Bits 5~4 : EP1 interface value

(3) Bits 3~2 : EP1 alternate value

(4) Bits 1~0 : EP1-In transfer type

(a) 11 : Interrupt mode

(b) 10 : Bulk mode

(c) 01 : Isochronous mode

(d) 00 : Control mode

5. Fifth Byte : Endpoint 1~4 set value

(1) Bits 7~6 : EP1-Out transfer type

(a) 11 : Interrupt mode

(b) 10 : Bulk mode

(c) 01 : Isochronous mode

(d) 00 : Control mode

(2) Bits 5~4 : EP2-Out transfer type

(a) 11 : Interrupt mode

(b) 10 : Bulk mode

(c) 01 : Isochronous mode

(d) 00 : Control mode

(3) Bits 3~2 : EP3-Out transfer type

(a) 11 : Interrupt mode

(b) 10 : Bulk mode

(c) 01 : Isochronous mode

(d) 00 : Control mode

(4) Bits 1~0 : EP4-Out transfer type

(a) 11 : Interrupt mode

(b) 10 : Bulk mode

(c) 01 : Isochronous mode

(d) 00 : Control mode

3 Set the Configuration usage metho

```
; EP1 : Bulk Mode
        MOV     UDCCFDATA, #0x42
        CALL    WaitCfgEpSetting

; EP2 : Bulk Mode
        MOV     UDCCFDATA, #0x42
        CALL    WaitCfgEpSetting

; EP3 : Interrupt Mode
        MOV     UDCCFDATA, #0x43
        CALL    WaitCfgEpSetting

; EP4 : Interrupt Mode
        MOV     UDCCFDATA, #0x43
        CALL    WaitCfgEpSetting

; EP1/2/ Out : Bulk Mode, EP3/4 Out : Interrupt Mode
        MOV     UDCCFDATA, #0xAF

WaitCfgEpSetting_Done:
        MOV     A, UDCCFSTA
        JNB     ACC.bmEpCfgDone, WaitCfgEpSetting_Done


WaitCfgEpSetting:
```

```
WaitCfgEpRdy_High:
        MOV     A, UDCCFSTA
        JNB     ACC.bmEpCfgRdy, WaitCfgEpRdy_High

WaitCfgEpRdy_Low:
        MOV     A, UDCCFSTA
        JB      ACC.bmEpCfgRdy, WaitCfgEpRdy_Low
        RET
```

UDCINT0EN: UDC Interrupt 0 Enable Register

Bit	7	6	5	4	3	2	1	0
Name	SOFINT EN	SUSPIN TEN	RSTINT EN	LPMSUS PINTEN	RESINT EN	RESERVE D	SETUP INTEN	EXTPKG INTEN
Type	R/W	R/W	R/W	R/W	R/W	--	R/W	R/W
Reset	0	1	1	0	0	0	1	0

SFR Address = 0xB5 ; SFR Page = 1

Bit 7: USB SOF in Interrupt Enable

1 : Enable

0 : Disable

Bit 6: Suspend interrupt request enable

1 : Enable

0 : Disable

Bit 5: Host reset interrupt request enable

1 : Enable

0 : Disable

Bit 4: LPM Resume interrupt enable

1 : Enable

0 : Disable

Bit 3: Resume interrupt enable

1 : Enable

0 : Disable

Bit 2: Reserved

Bit 1: EP0 setup interrupt enable

1 : Enable

0 : Disable

Bit 0: Extpckg interrupt enable

1 : Enable

0 : Disable

UDCINT1EN: UDC Interrupt 1 Enable Register

Bit	7	6	5	4	3	2	1	0
Name	EP3OUT INTEN	EP3ININ TEN	EP2OUT INTEN	EP2ININ TEN	EP1OUT INTEN	EP1ININ TEN	EP0OUT INTEN	EP0ININ TEN
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 = 0xB6; SFR Page = 1

Bit 7: EP3 transfer out interrupt enable

1 : Enable

0 : Disable

Bit 6: EP3 transfer in interrupt enable

1 : Enable

0 : Disable

Bit 5: EP2 transfer out interrupt enable

1 : Enable

0 : Disable

Bit 4: EP2 transfer in interrupt enable

1 : Enable

0 : Disable

Bit 3: EP1 transfer out interrupt enable

1 : Enable

0 : Disable

Bit 2: EP1 transfer in interrupt enable

1 : Enable

0 : Disable

Bit 1: EP0 transfer out interrupt enable

1 : Enable

0 : Disable

Bit 0: EP0 transfer in interrupt enable

1 : Enable

0 : Disable

UDCINT2EN: UDC Interrupt 2 Enable Register

Bit	7	6	5	4	3	2	1	0
Name	EP4INEMPINTEN	EP3INEMPINTEN	EP2INEMPINTEN	EP1INEMPINTEN	EP4OUTINTEN	EP4ININTEN	--	ERRINTEN
Type	RW	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 = 0xB7 ; SFR Page = 1

Bit 7: EP4 transfer in empty interrupt enable

1 : Enable

0 : Disable

Bit 6: EP3 transfer in empty interrupt enable

1 : Enable

0 : Disable

Bit 5: EP2 transfer in empty interrupt enable

1 : Enable

0 : Disable

Bit 4: EP1 transfer in empty interrupt enable

1 : Enable

0 : Disable

Bit 3: EP4 transfer out interrupt enable

1 : Enable.

0 : Disable.

Bit 2: EP4 transfer in interrupt enable

1 : Enable

0 : Disable

Bit 1: Reserved

Bit 0: Transition error interrupt enable

1 : Enable

0 : Disable

UDCINT0STA: UDC Interrupt0 Status Register

Bit	7	6	5	4	3	2	1	0
Name	SOFSF	SUSPSF	RSTSF	LPMRESSF	RESSF	Reserved	SETUPSF	EXTPKGSF
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 = 1

Bit 7: USB SOF generation status flag

Bit 6: Suspend request status flag

Bit 5: Reset request status flag

Bit 4: LPM Resume status flag

Bit 3: Resume status flag

Bit 2: Reserved

Bit 1: EP0 setup status flag

Bit 0: Extpckg status flag

**UDCINT0STA hardware set high and clear by FW.*

UDCINT1STA: UDC Interrupt 1 Status Register

Bit	7	6	5	4	3	2	1	0
Name	EP3OUTSF	EP3INSF	EP2OUTSF	EP2INSF	EP1OUTSF	EP1INSF	EP0OUTSF	EP0INSF
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 = 1

Bit 7: EP3 transfer out status flag

Bit 6: EP3 transfer in status flag

Bit 5: EP2 transfer out status flag

Bit 4: EP2 transfer in status flag

Bit 3: EP1 transfer out status flag

Bit 2: EP1 transfer in status flag

Bit 1: EP0 transfer out status flag

Bit 0: EP0 transfer in status flag

**UDCINT1STA hardware set high and clear by FW.*

UDCINT2STA: UDC Interrupt 2 Status Register

Bit	7	6	5	4	3	2	1	0
Name	EP4INEM PSF	EP3INEM PSF	EP2INEM PSF	EP1INEM PSF	EP4OUTSF	EP4INSF	--	ERRSF
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 = 1

Bit 7: EP4 in empty status flag

Bit 6: EP3 in empty status flag

Bit 5: EP2 in empty status flag

Bit 4: EP1 in empty status flag

Bit 3: EP4 transfer out status flag

Bit 2: EP4 transfer in status flag

Bit 1: Reserved

Bit 0: Transition error status flag

**UDCINT2STA hardware set high and clear by FW.*

UDCEPCTRL: Device End-point Control Register

Bit	7	6	5	4	3	2	1	0
Name	EP4STALL	EP43STALL	EP2STALL	EP1STALL	EP4EN	EP3EN	EP2EN	EP1EN
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 = 0xBC ; SFR Page = 1

Bit 7: EP4 stall enable selection

1 : Enable

0 : Disable

Bit 6: EP3 stall enable selection

1 : Enable

0 : Disable

Bit 5: EP2 stall enable selection

1 : Enable

0 : Disable

Bit 4: EP1 stall enable selection

1 : Enable

0 : Disable

Bit 3: EP4 enable selection

1 : Enable

0 : Disable

Bit 2: EP3 enable selection

1 : Enable

0 : Disable

Bit 1: EP2 enable selection

1 : Enable

0 : Disable

Bit 0: EP1 enable selection

1 : Enable

0 : Disable

UDCEPBUFCTRL0: Device End-point Buffer Control 0 Register

Bit	7	6	5	4	3	2	1	0
Name	--	--	--	--	--	--	EP0DAT ARDY	EP0BUF CLR
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xBD ; SFR Page = 1

Bit 7~2: Reserved

Bit 1: EP0 data out ready (clear by buffer empty)

Bit 0: EP0 buffer clear (High active)

UDCEPBUFCTRL1: Device End-point Buffer 1 Control 1 Register

Bit	7	6	5	4	3	2	1	0
Name	EP4NULL PKGSET	EP3NULL PKGSET	EP2NULL PKGSET	EP1NULL PKGSET	--	--	EP4DAT ARDY	EP4BUF CLR
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 = 1

Bit 7: EP4 empty packet set (cleared by hardware)

Bit 6: EP3 empty packet set (cleared by hardware)

Bit 5: EP2 empty packet set (cleared by hardware)

Bit 4: EP1 empty packet set (cleared by hardware)

Bits 3~2: Reserved

Bit 1: EP4 data out ready (cleared by buffer empty)

Bit 0: EP4 buffer clear (High active)

UDCEPBUFCTRL2: Device End-point Buffer Control 2 Register

Bit	7	6	5	4	3	2	1	0
Name	EP4OUTBUFEMPTY	EP3OUTBUFEMPTY	EP2OUTBUFEMPTY	EP1OUTBUFEMPTY	EP0OUTBUFEMPTY	--	EP1DATARDY	EP1BUFCLR
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9A ; SFR Page = 1

Bit 7: EP4 out buffer empty

Bit 6: EP3 out buffer empty

Bit 5: EP2 out buffer empty

Bit 4: EP1 out buffer empty

Bit 3: EP0 out buffer empty

Bit 2: Reserved

Bit 1: EP1 data out ready (cleared by buffer empty)

Bit 0: EP1 buffer clear (High active)

UDCEPBUFCTRL3: Device End-point Buffer Control 3 Register

Bit	7	6	5	4	3	2	1	0
Name	EP4INBUFFULL	EP3INBUFFULL	EP2INBUFFULL	EP1INBUFFULL	EP0INBUFFULL	--	EP2DATARDY	EP2BUFCLR
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9B ; SFR Page = 1

Bit 7: EP4 in buffer full

Bit 6: EP3 in buffer full

Bit 5: EP2 in buffer full

Bit 4: EP1 in buffer full

Bit 3: EP0 in buffer full

Bit 2: Reserved

Bit 1: EP2 data out ready (cleared by buffer empty)

Bit 0: EP2 buffer clear (High active)

UDCEPBUFCTRL4: Device End-point Buffer Control 4 Register

Bit	7	6	5	4	3	2	1	0
Name	EP4OUT BUFFUL L	EP3OUT BUFFUL L	EP2OUT BUFFUL L	EP1OUT BUFFUL L	EP0OUT BUFFUL L	--	EP3DAT ARDY	EP3BUF CLR
Type	R	R	R	R	R	R	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9C ; SFR Page = 1

Bit 7: EP4 out buffer full

Bit 6: EP3 out buffer full

Bit 5: EP2 out buffer full

Bit 4: EP1 out buffer full

Bit 3: EP0 out buffer full

Bit 2: Reserved

Bit 1: EP3 data out ready (cleared by buffer empty)

Bit 0: EP3 buffer clear (High active)

UDCEP0BUFDATA: UDC Endpoint 0 Data Register

Bit	7	6	5	4	3	2	1	0
Name	EP0DATA7	EP0DATA6	EP0DATA5	EP0DATA4	EP0DATA3	EP0DATA2	EP0DATA1	EP0DATA0
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: EP0 Data

UDCEP1BUFDATA: UDC Endpoint 1 Data Register

Bit	7	6	5	4	3	2	1	0
Name	EP1DATA7	EP1DATA6	EP1DATA5	EP1DATA4	EP1DATA3	EP1DATA2	EP1DATA1	EP1DATA0
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 = 1

Bits 7~0: EP1 Data

UDCEP2BUFDATA: UDC Endpoint 2 Data Register

Bit	7	6	5	4	3	2	1	0
Name	EP2DAT A7	EP2DAT A6	EP2DAT A5	EP2DAT A4	EP2DAT A3	EP2DAT A2	EP2DAT A1	EP2DAT A0
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 = 1

Bits 7~0: EP2 Data

UDCEP3BUFDATA: UDC Endpoint 3 Data Register

Bit	7	6	5	4	3	2	1	0
Name	EP3DATA7	EP3DATA6	EP3DATA5	EP3DATA4	EP3DATA3	EP3DATA2	EP3DATA1	EP3DATA0
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 = 1

Bits 7~0: EP3 Data

UDCEP4BUFDATA: UDC Endpoint 4 Data Register

Bit	7	6	5	4	3	2	1	0
Name	EP4DATA7	EP4DATA6	EP4DATA5	EP4DATA4	EP4DATA3	EP4DATA2	EP4DATA1	EP4DATA0
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 = 0xC4 ; SFR Page = 1

Bits 7~0: EP4 Data

UDCBUFSTA: UDC Buffer Status Register

Bit	7	6	5	4	3	2	1	0
Name	--	--	--	EP4INBUFEMPTY	EP3INBUFEMPTY	EP2INBUFEMPTY	EP1INBUFEMPTY	EP0INBUFEMPTY
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

Bits 7~5: Reserved

Bit 4: EP4 in buffer empty

Bit 3: EP3 in buffer empty

Bit 2: EP2 in buffer empty

Bit 1: EP1 in buffer empty

Bit 0: EP0 in buffer empty

UDCEP1DATAIN CNT: UDC Endpoint 1 Data-In Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP1DATI NCNT7	EP1DATI NCNT6	EP1DATI NCNT5	EP1DATI NCNT4	EP1DATI NCNT3	EP1DATI NCNT2	EP1DATI NCNT1	EP1DATI NCNT0
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 = 1

Bits 7~0: EP1 Data-In Count (Count = N + 1)

UDCEP1DATAOUT CNT: UDC Endpoint 1 Data-Out Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP1DAT OUTCNT 7	EP1DAT OUTCNT 6	EP1DAT OUTCNT 5	EP1DAT OUTCNT 4	EP1DAT OUTCNT 3	EP1DAT OUTCNT 2	EP1DAT OUTCNT 1	EP1DAT OUTCNT 0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCA ; SFR Page = 1

Bits 7~0: EP1 Data-Out Count (Count = N + 1)

UDCEP2DATAIN CNT: UDC Endpoint 2 Data-In Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP2DATI NCNT7	EP2DATI NCNT6	EP2DATI NCNT5	EP2DATI NCNT4	EP2DATI NCNT3	EP2DATI NCNT2	EP2DATI NCNT1	EP2DATI NCNT0
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 = 1

Bits 7~0: EP2 Data-In Count (Count = N + 1)

UDCEP2DATAOUT CNT: UDC Endpoint 2 Data-Out Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP2DATO UTCNT7	EP2DATO UTCNT6	EP2DATO UTCNT5	EP2DATO UTCNT4	EP2DATO UTCNT3	EP2DATO UTCNT2	EP2DATO UTCNT1	EP2DATO UTCNT0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCC ; SFR Page = 1

Bits 7~0: EP2 Data-Out Count (Count = N + 1)

UDCEP3DATAIN CNT: UDC Endpoint 3 Data-In Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP3DATI NCNT7	EP3DATI NCNT6	EP3DATI NCNT5	EP3DATI NCNT4	EP3DATI NCNT3	EP3DATI NCNT2	EP3DATI NCNT1	EP3DATI NCNT0
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 = 1

Bits 7~0: EP3 Data-In Count (Count = N + 1)

UDCEP3DATAOUT CNT: UDC Endpoint 3 Data-Out Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP3DAT OUTCN T7	EP3DAT OUTCN T6	EP3DAT OUTCN T5	EP3DAT OUTCN T4	EP3DAT OUTCN T3	EP3DAT OUTCN T2	EP3DAT OUTCN T1	EP3DAT OUTCN T0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xCE ; SFR Page = 1

Bits 7~0: EP3 Data Out Count (Count = N + 1)

UDCEP4DATAIN CNT: UDC Endpoint 4 Data-In Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP4DATI NCNT7	EP4DATI NCNT6	EP4DATI NCNT5	EP4DATI NCNT4	EP4DATI NCNT3	EP4DATI NCNT2	EP4DATI NCNT1	EP4DATI NCNT0
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 = 0xCF ; SFR Page = 1

Bits 7~0: EP4 Data-In Count (Count = N + 1)

UDCEP4DATAOUT CNT: UDC Endpoint4 Data-Out Count Register

Bit	7	6	5	4	3	2	1	0
Name	EP4DAT OUTCN T7	EP4DAT OUTCN T6	EP4DAT OUTCN T5	EP4DAT OUTCN T4	EP4DAT OUTCN T3	EP4DAT OUTCN T2	EP4DAT OUTCN T1	EP4DAT OUTCN T0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD1 ; SFR Page = 1

Bits 7~0: EP4 Data-Out Count (Count = N + 1)

UDCEP1BUFDEPTH: UDC Endpoint1 Buffer Depth Register

Bit	7	6	5	4	3	2	1	0
Name	EP1BUF DPTH7	EP1BUF DPTH6	EP1BUF DPTH5	EP1BUF DPTH4	EP1BUF DPTH3	EP1BUF DPTH2	EP1BUF DPTH1	EP1BUF DPTH0
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 = 1

Bits 7~0: EP1 Buffer Depth (Depth = N + 1)

UDCEP2BUFDEPTH: UDC Endpoint 2 Buffer Depth Register

Bit	7	6	5	4	3	2	1	0
Name	EP2BUF DPTH7	EP2BUF DPTH6	EP2BUF DPTH5	EP2BUF DPTH4	EP2BUF DPTH3	EP2BUF DPTH2	EP2BUF DPTH1	EP2BUF DPTH0
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 = 1

Bits 7~0: EP2 Buffer Depth (Depth = N + 1)

UDCEP3BUFDEPTH: UDC Endpoint 3 Buffer Depth Register

Bit	7	6	5	4	3	2	1	0
Name	EP3BUF DPTH7	EP3BUF DPTH6	EP3BUF DPTH5	EP3BUF DPTH4	EP1BUF DPTH3	EP3BUF DPTH2	EP3BUF DPTH1	EP3BUF DPTH0
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 = 1

Bits 7~0: EP3 Buffer Depth (Depth = N + 1)

UDCEP4BUFDEPTH: UDC Endpoint 4 Buffer Depth Register

Bit	7	6	5	4	3	2	1	0
Name	EP4BUF DPTH7	EP4BUF DPTH6	EP4BUF DPTH5	EP4BUF DPTH4	EP4BUF DPTH3	EP4BUF DPTH2	EP4BUF DPTH1	EP4BUF DPTH0
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 = 1

Bits 7~0: EP4 Buffer Depth (Depth = N + 1)

PHYTEST0: PHY Test 0 Mode Register

Bit	7	6	5	4	3	2	1	0
Name	PHYTST EN	PHY_RS W	USBWA KEUPEN	NEWPID CLR	DEVRES UMESEL	DEVRES TIME1	DEVRES TIME0	PHYRST
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	1	0	0	0	0

SFR Address = 0xD6 ; SFR Page = 1

Bit 7 : PHY test mode enable

1 : Enable

0 : Disable

Bit 6 : PHY connect enable

1 : Enable

0 : Disable (PHY unconnected)

Bit 5 : USB Wake-up enable

1 : Enable

0 : Disable

Bit 4 : New PID command clear enable (Ext: LPM command)

1 : Enable

0 : Disable

*clear UDC send two byte dummy data

Bit 3 : Device resume select

1 : LPM

0 : normal

Bits 2~1 : Device resume time select

11 : 200 μs

10 : 150 μs

01 : 100 μs

00 : 50 μs

Bit 0 : PHY reset signal. High active.

7.12 I^{2}C (Inter-Integrated Circuit)

EM85F690A 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 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 transmitter or receiver, but the master device determines which mode is 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 100 kbit/s in the Standard-mode or up to 400 kbit/s in Fast-mode.

The data on the SDA line must be 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.

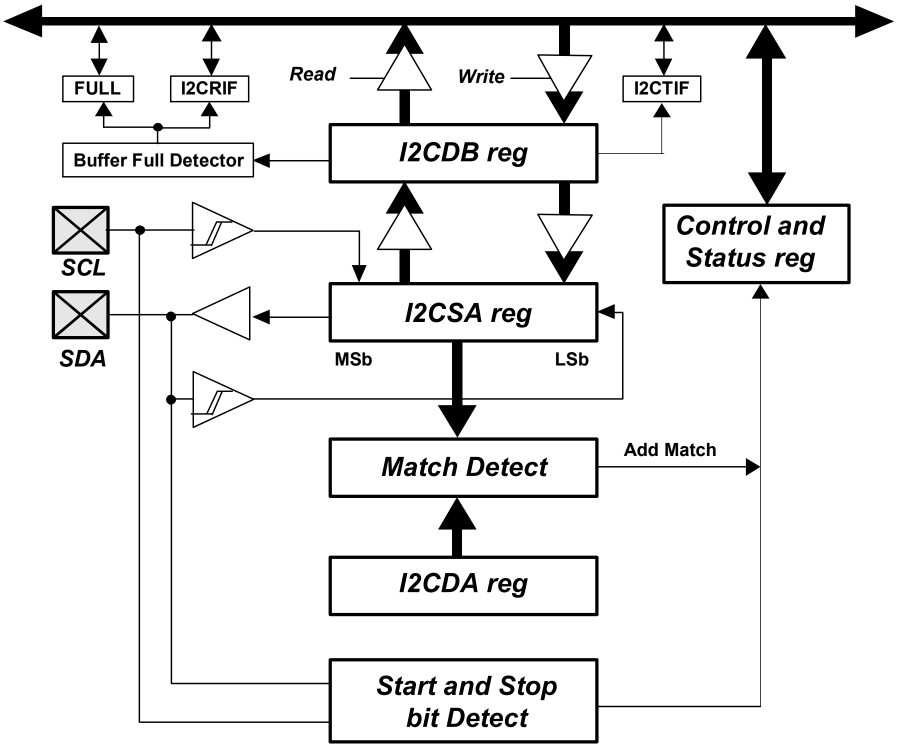


Figure7.12.1. I2C Block Diagram

Table 7.12 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.

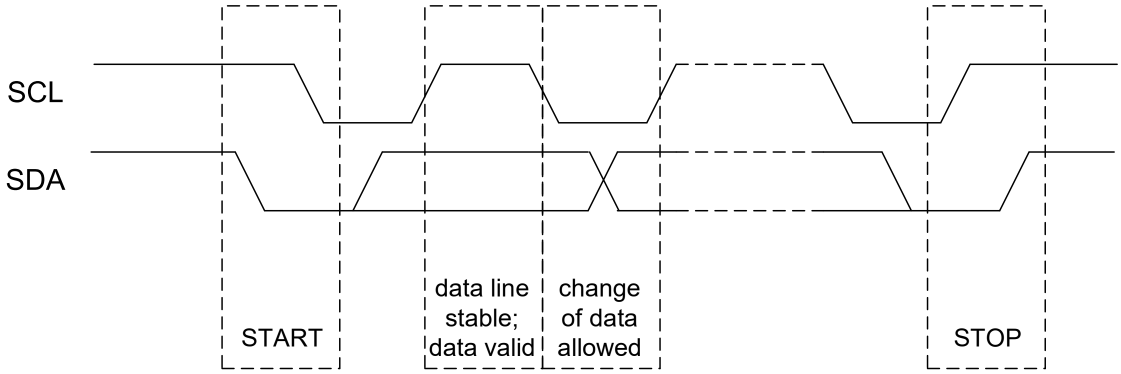


Figure 7.12.2 I2C Transfer Condition

7.12.1 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).

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 master transmitter with master receiver is only in R//W bit, if the R//W bit were “0”, the master device would be transmitter; the other way, the master device would be receiver. The master transmitter is described by the Figure “7-Bits slave address in Master-transmitter transmits to slave -receiver”, and the master receiver is described by Figure “7-Bit slave address in Master-receiver read slave -transmitter”.

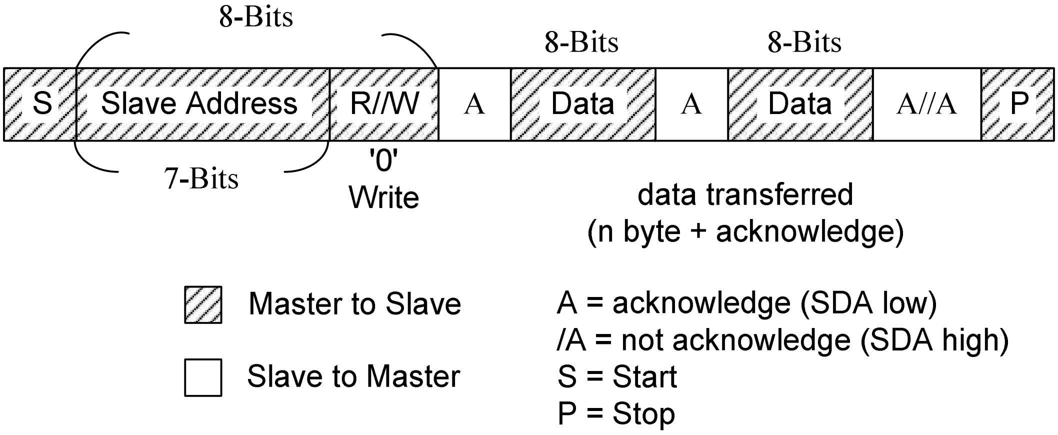


Figure 7.12.3 Master-transmitter transmits to slave-receiver (7-Bits slave address)

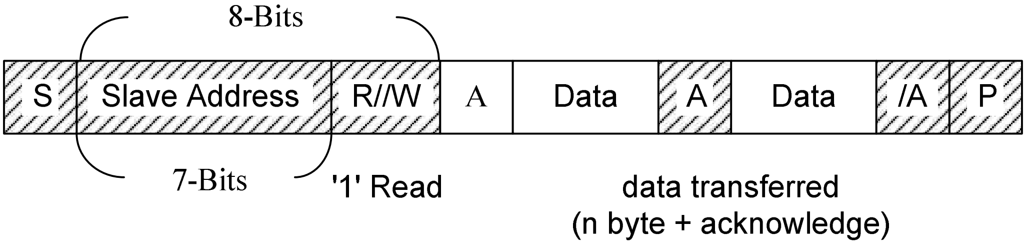


Figure 7.12.4 Master receiver read slave-transmitter (7-Bits slave address)

7.12.2 10-Bit Slave Address

In 10-Bits slave address mode, using 10 Bits 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-bit address. If the R//W bit were “0”, the second byte after acknowledge would be the eight address bits of 10-bit 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 kind of difference formats that would be explained in Fig.7-12.5 ~ Fig.7-12.6 in the 10-bits 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 their own address and the eighth bit, R//W, if the R//W bit were "0", the slave would return the acknowledge(A1) and that would be possible more than one slave device return it. Then all slave device will continue to compare the second address (XXXXXXXX), if the slave device have matched, that would be only one slave device return acknowledge. The matching slave device will remain addressed by the master until it receives the STOP condition or a repeated START condition followed by the different slave address.

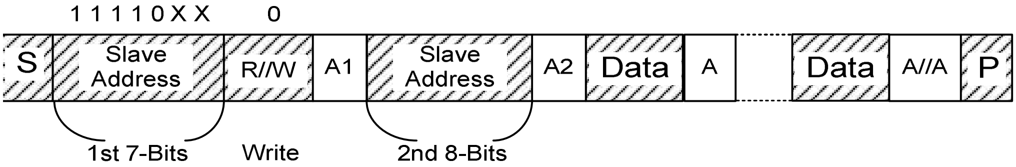


Figure7.12.5 Master-transmitter transmits to slave-receiver (10-bit slave address)

Master-receiver read 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) followed by seven bits slave address (11110XX) but 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) received by slave device, all the slave device would compare with their own address and test the eighth R//W, but none of all slave device return the acknowledge because R//W=1.

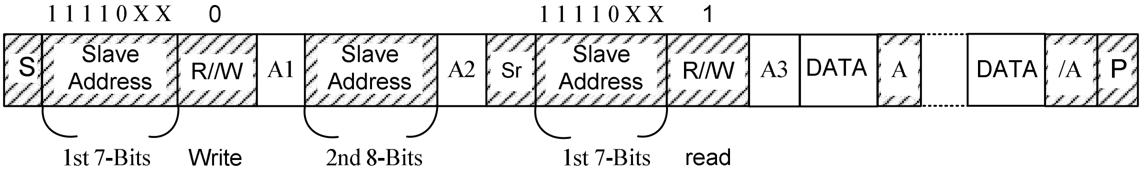


Figure7.12.6 Master-receiver read slave-transmitter (10-bit slave address)

Master addresses a slave with 10-Bit 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-bit slave address", then the master device can start to transmit the data to slave device. If the slave device have received the acknowledge or none acknowledge which were followed by repeat START(Sr) and repeat the procedure of the section of "Master-receiver read slave-transmitter with a 10-bits slave address".

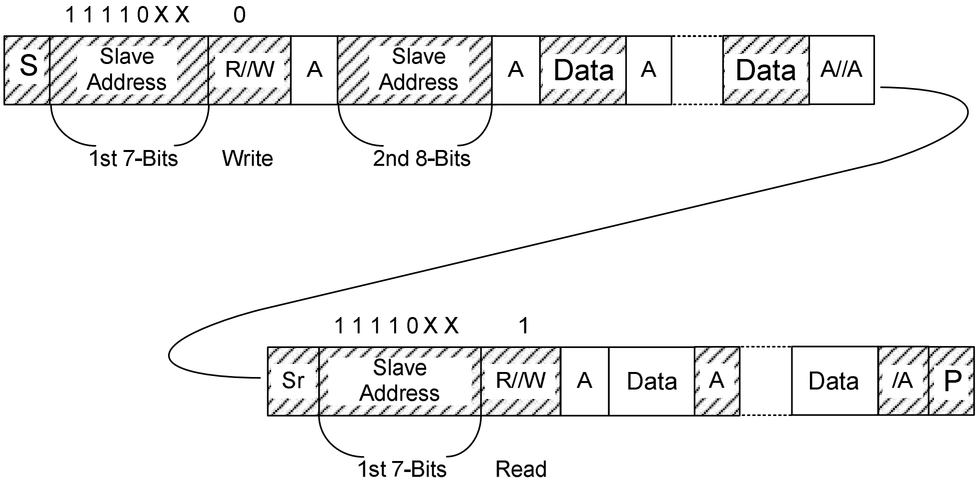


Figure 7.12.7 Master transmits and receives data in the same slave device (10-bit slave addresses)

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

The section of "Master-transmitter transmits to slave-receiver with a 10-bits slave address" describe the procedure how to transmit the data to slave device, if the master device have 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 by the section of the "Master-transmitter transmits to slave-receiver with a 10-Bits 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-Bits address could be transmitted. The Fig.7-12.8 ~ Fig.7-12.9 shows how to transmit the data in 7-Bits and 10-Bits address mode in serial transfer.

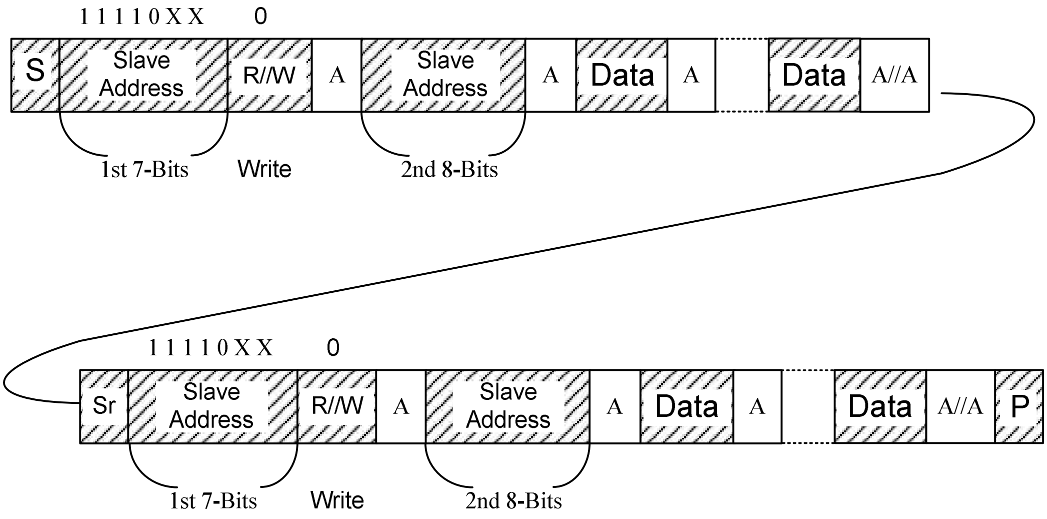


Figure 7-12.8   Transmit one more devices (10-bit slave address)

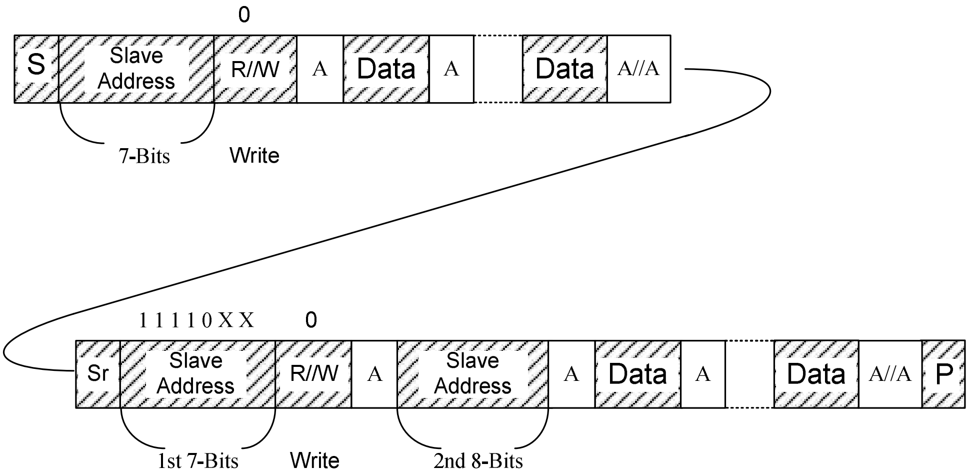


Figure 7.12.9 Slave address mode with 7-bit and 10-bit

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 transmitting and then Check I2CTSF (I2CTSF) bit.

Step 5: Write the 1st data into the I2CDB register, set strobe bit and Check I2CTSF (I2CRSF) bit.

Step 6: Write 2nd data into the I2CDB register, set strobe bit, STOP bit and Check I2CTSF (I2CRSF) 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 I2CRSF (I2CTSF) bit, read I2CDB register (address) and then clear the Pend bit.

Step 5: Check I2CRSF (I2CTSF) bit, read I2CDB register (1st data) and then clear the Pend bit.

Step 6: Check I2CRSF (I2CTSF) bit, read I2CDB register (2st data) and then clear the Pend bit.

Step 7: Check I2CSTPSF bit, end transmission.

7.12.3 I2CA Register

I2CACR1: I2C Status and Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	Strobe/Pe	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	0	0	0	0

SFR Address = 0xC1; SFR Page = Page 0

Bit 7: In master mode, it 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 fill 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 (Default).

1: Master.

Bit 5: I2C Fast/Standard mode select bit. (If FHS is 4 MHz and I2CTS2~0<0, 0, 0>)

0: Standard mode (100K bit/s).

1: Fast mode (250K~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 send STOP signal. If STOP=1 and R/nW=0 then MCU send STOP signal after receive 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.

I2CACR2 : I2C Status and Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	I2CBF	--	RST_SW_N	BBF	I2CATS2	I2CATS1	I2CATS0	I2CAEN
Type	R	--	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	1	0	0	0	0	0

SFR Address = 0xC2; SFR Page = Page 0

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.

*Set when STAR signal, clear when I2C disable or STOP signal for Slave mode.

Bit 6: reserved. Don't use

Bit 5: Software reset bit, low active. It is used to reset the FSM and release the scl pad

Bit 4: Busy Flag Bit. I2C detection is busy in the master mode. Read only.

*Set when STAR signal, clear when STOP signal for Master mode.

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

I2CACR1 Bit 5=1, fast mode

I2CATS2	I2CATS1	I2CATS0	SCL CLK	Operating FHS (MHz)
0	0	0	FHS /10	4
0	0	1	FHS /15	6
0	1	0	FHS /20	8
0	1	1	FHS /25	10
1	0	0	FHS /30	12
1	0	1	FHS /40	16
1	1	0	FHS /50	20
1	1	1	FHS /60	24

I2CACR1 Bit 5=0, standard mode

I2CATS2	I2CATS1	I2CATS0	SCL CLK	Operating FHS (MHz)
0	0	0	FHS /40	4
0	0	1	FHS /60	6
0	1	0	FHS /80	8
0	1	1	FHS /100	10
1	0	0	FHS /120	12
1	0	1	FHS /160	16
1	1	0	FHS /200	20
1	1	1	FHS /240	24

Bit 0: I2CA Enable Bit.

0: Disable I2CA mode (Default).

1: Enable I2CA mode.

I2CAEN Bit =1 : Initialize state machine, release pin state, (register is unchanged).

I2CASA : 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 = 0xC3; SFR Page = Page 0

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

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

0: Write

1: Read

I2CADB : 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 = 0xC4; SFR Page = Page 0

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

I2CADAL : 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 = 0xC5; SFR Page = Page 0

Bits 7~0: When MCU used as slave device for I2C application, this register store 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.**

I2CADAH : I2C Device Address Register

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	-	DA9	DA8
Type	-	-	-	-	-	-	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC6; SFR Page = Page 0

Bits 7~2: Reserved.

Bits 1~0: 1~0: I2CA Device Address Bit 9 and Bit 8.

I2CASF: I2CA Status Flag

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	I2CASTPSF	I2CARSF	I2CATSF
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xC7; SFR Page = 0

Bits 7~3: Reserved. Don't use.

Bit 2 (I2CASTPSF): I2CA stop status flag. Set when I2C occurs stop signal.

Bit 1 (I2CARSF): I2C receive status flag. Set when I2C receiving 1byte data and responds ACK signal. Reset by firmware or I2C disable.

Bit 0 (I2CATSF): I2C transmit status flag. Set when I2C transmits 1 byte data and receive handshake signal (ACK or NACK). Reset by firmware or I2C disable.

7.11.4 I2CB Register

I2CBINT: I2CB interrupt Status and Control Register

Bit	7	6	5	4	3	2	1	0
Name	I2CBTXSF	I2CBRXSF	I2CBSTPIEN	STOP	-	-	-	-
Type	R/W	R/W	R/W	R/W	-	-	-	-
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xD7; SFR Page = Page 0

Bit 7: I2C tx interrupt flag.

Bit 6: I2C rx interrupt flag.

Bit 5: STOP interrupt enable bit.

0: Disable

1: Enable

Bit 4: I2C stop interrupt flag. Set and clear by hardware which determine by STOP, condition is available or not when I2CTX or I2CRX interrupt is triggered.

Bits 3~0: Reserved. Don't use.

I2CBCR1: I2CB Status and Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	Strobe / Pend	MODE (IMS)	SPEED (ISS)	I2CBST PSF	SAR_EM PTY	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 = 0xD9; SFR Page = Page 0

Bit 7: In Master mode, it is used as strobe signal to control I2C circuit in sending SCL clock. Will 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 data are filled into Tx buffer or after obtaining data from Rx buffer to inform slave 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.

0: Standard mode (100K bits/s).

1: Fast mode (400K bits/s).

Bit 4: In Master mode, if STOP=1 and R/nW=1; then I2C module must return NACK signal to slave device before sending STOP signal. If STOP=1 and R/nW=0, then I2C module will send STOP signal after receiving an ACK signal. Will reset when I2C module sent STOP signal to Slave device. In Slave mode, if STOP=1 and R/nW=0; the I2C module must return NACK signal to master device.

Bit 3: Will set when I2C transmits 1 byte data from I2C Slave Address Register and receives ACK (or NACK) signal. Will reset when MCU writes 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). Will reset when the device response is non-acknowledge (NACK) signal.

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

Bit 0: Set by hardware when the I2C transmitted buffer register is empty and receives ACK (or NACK) signal. Reset by hardware when MCU writes new data to I2C transmitted buffer register.

I2CBCR2: I2CB Status and Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	I2CBF	-	RST_SW_N	BBF	I2CBTS2	I2CBTS1	I2CBTS0	I2CBEN
Type	R	-	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	1	0	0	0	0	0

SFR Address = 0xDA; SFR Page = Page 0

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.

Bit 6: Reserved. Don’t use.

Bit 5: Software reset bit, low active. It is used to reset the FSM and release the scl pad.

Bit 4: Busy flag bit. I2C detection is busy in the master mode. Read only.

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

I2CBCR1 Bit 5=1, fast mode

I2CBTS2	I2CBTS1	I2CBTS0	SCL CLK	Operating FHS (MHz)
0	0	0	FHS /10	4
0	0	1	FHS /15	6
0	1	0	FHS /20	8
0	1	1	FHS /25	10
1	0	0	FHS /30	12
1	0	1	FHS /40	16
1	1	0	FHS /50	20
1	1	1	FHS /60	24

I2CBCR1 Bit 5=0, standard mode

I2CBTS2	I2CBTS1	I2CBTS0	SCL CLK	Operating FHS (MHz)
0	0	0	FHS /40	4
0	0	1	FHS /60	6
0	1	0	FHS /80	8
0	1	1	FHS /100	10
1	0	0	FHS /120	12
1	0	1	FHS /160	16
1	1	0	FHS /200	20
1	1	1	FHS /240	24

Bit 0: I2C controller enable bit.

0: Disable I2CB mode (Default).

1: Enable I2CB mode.

I2CBSA: I2CB Slave Address Register

Bit	7	6	5	4	3	2	1	0
Name	SAR6 (SA6)	SAR5 (SA5)	SAR4 (SA4)	SAR3 (SA3)	SAR2 (SA2)	SAR1 (SA1)	SAR0 (SA0)	R/nW (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	1

SFR Address = 0xDB; SFR Page = Page 0

Bits 7~1: Applies when I2C module is used as master device for I2C application. This is a slave device address register.

Bit 0: Applies when I2C module is used as master device for I2C application.

I2CBDDB: I2CB Data Buffer Register

Bit	7	6	5	4	3	2	1	0
Name	BX7 (DB7)	BX6 (DB6)	BX5 (DB5)	BX4 (DB4)	BX3 (DB3)	BX2 (DB2)	BX1 (DB1)	BX0 (DB0)
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

SFR Address = 0xDC; SFR Page = Page 0

Bits 7~0: Transmitting mode: Next byte to transmit via I2C Receiving mode: Last byte received via I2C.

I2CBDAL: I2CB Device Address Register (Low byte)

Bit	7	6	5	4	3	2	1	0
Name	ADDR7 (DA7)	ADDR6 (DA6)	ADDR5 (DA5)	ADDR4 (DA4)	ADDR3 (DA3)	ADDR2 (DA2)	ADDR1 (DA1)	ADDR0 (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 = 0xDD; SFR Page = Page 0

Bits 7~0: When I2C module 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.

I2CBDAH: I2CB Device Address Register (High byte)

Bit	7	6	5	4	3	2	1	0
Name	-	-	-	-	-	ADDR _SYNC	ADDRH9 (DA9)	ADDRH8 (DA8)
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDE; SFR Page = Page 0

Bits 7~3: Reserved. Don’t use.

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

0: Chcek address with async SCL/SDA.

1: Chcek address with sync SCL/SDA.

Bits 1~0: I2CB Device Address bit 9 and Bit 8.

7.13 UART

UART has two associated SFRs: Serial Port 0 Control Register 0 (SCON) and Serial Port 0 Data Buffer (SBUF). The SBUF consists of two separate registers: transmit and receive registers. UATR function is enabled by setting UARTEN bit high (PCON.6).A data writes into the SBUF sets this data in UART output register and starts a transmission. A data reads from SBUF, reads data from the UART receive register.

The serial port can operate in 4 modes: one synchronous and three asynchronous modes as shown in Table7.13.1 Mode 2 and 3 have a special feature for multiprocessor communications (Enable by SMOD0 bit). This feature is enabled by setting SM02 bit in SCON 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 SM02 = 1, no slave will be interrupted by a data byte. An address byte will interrupt all slaves. The addressed slave will clear its SM02 bit and prepare to receive the data bytes that will be coming. The slaves that were not being addressed leave their SM02 set and ignoring the incoming data.

	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 7.13.1 UART Modes

7.13.1 UART Mode 0: Synchronous

In Mode 0, the transmit and receive serial data are performed utilizing RXD Pin, while TXD Pin function as the output of the shift pulse wave, this pulse wave baud rate is fixed at 1/12 System clock, as long as the execution of data will be written to SBUF instruction will lead to action transmission, when data transfer is completed, the CPU will set the TI bit in SCON to 1, and notify the system to generate a serial interrupt.

In terms of the data received, at the start must set the software SCON REN0 bit to “1” and then execute the Instruction to clear the RI0 bit to “0”. The serial port will receive data in accordance with the operation timing, then release when data is complete, the CPU will set the RI0 bit of SCON to “1”, and notify the interrupt system to generate a serial interrupt.

7.13.2 UART Mode 1: 8-bit UART (Variable Baud Rate)

When in Mode 1, every time the CPU transmits or receives 10-bit data, this 10-bit is composed of three parts, the Start bit (which accounts for 1 bit, fixed to 0), data bits (accounting for 8 bits, LSB first pass) and Stop bit (which accounts for 1 bit, fixed to 1). When the serial port operates in Mode 1, data transmission baud rate is set by Timer 1, and the setting method is shown in Table 7.13.2. After the serial communication setting is completed, as long as the CPU executes writing instructions to SBUF, it will start data transmission operation. When writing SBUF data transmission is completed, the CPU will set the SCON TI0 bit to “1”, and notify the Interrupt system to generate a serial port interrupt.

Beginning to receive data, the CPU will sequentially receive 10-bit data from when the RXD Pin signal changes from “1” to “0”. When receiving a Stop Bit 1, the CPU will receive 8-bit data stored in SBUF as well as the Stop bit is stored in RB08 bit, and RI0 bit is set to “1” and notify the Interrupt system to generate a serial port interrupt.

UART Mode	Baud Rate
Mode 0	SYSCLK/12

Modes 1, 3	SMOD0 = 0
	SMOD0 = 1

Mode 2	SMOD0 = 0 SYSCLK/64
	SMOD0 = 1 SYSCLK/32

Table 7.13.2 UART baud rates

7.13.3 UART Mode 2: 9-bit UART (Fixed Baud Rate)

When the serial port operates in Mode 2dya, the CPU will transmit or receive 11 bits of data each time. This 11-bit is composed of 4 parts; the Start Bit (made-up of one bit which is fixed at “0”), Data Bits (made-up of 8 bits, LSB transmitted first), Programmable Data Bits (made-up of one bit, TB08, RB08) and Stop Bit (made-up of one bit which fixed at “1”). The Programmable Data Bits may also be used as parity check bit. Transmission baud rate can be set to 1/32 or 1/64 of the System clock. In terms of data transfer, the TB08 bit value in SCON must be set by software first, then execute the command to write data into SBUF to start the data transfer operation. The serial port will sequentially transmit data starting from Start Bit, 8 Data Bits, TB08 Bit and Stop Bit. When data transfer is completed, the CPU sets the TI bit in SCON to “1” and the interrupt notification system generates a serial port interrupt.

On aspect of receiving data, the data received operation starts when the RXD Pin signal switches from 1 to 0 and the CPU will sequentially receive 11 bits of data. When it receives the Programmable Data Bit 1, CPU will store the received 8 data bits in SBUF and the store the Stop Bit in RB08 bit of SCON. Then it sets the RI0 bit to “1” and the interrupt notification system generates serial port interrupt.

7.13.4 UART Mode 3: 9-BIT UART (Variable Baud Rate)

Except different baud rate settings, all other features of Mode 3 and Mode 2 are identical. Mode 3 baud rate is variable and controlled by Timer 1. For baud rate setting details, please refer to Section 7.11.5 Baud Rates.

When receiving the data, accuracy of baud rate is very important. Hence, when applying Mode 1 and Mode 3, Timer 1 must be started to work first. The proper steps of using serial port are listed below:

Step 1 Set Timer 1 mode and set the TH1 and TL1 according to the transmission baud rate.

Step 2 Determine whether SMOD0 bit is 0 or 1.

Step 3 Set Serial Port mode and clear RI0, TI0 bit to “0”, and set the RENO bit to “1”.

Step 4 Enable serial port interrupt.

Step 5 Activate Timer 1.

Under all four modes, as long as "MOV SBUF, A" instruction is used, it can and will start the serial port transmission operation. When receiving, Mode 0 must set RI0 = 0 and REN0 = 1 to start the serial port receive. Except for RI0 = 0 and REN0 = 1, other modes must receive the Start Bit before it can start receiving serial port operation.

7.13.5 UART Baud Rates

Under each mode, setting method and counting device of baud rate are as follows:

Mode 0 Baud Rate:

$$BoudRate = \frac{SYSCLK}{12}$$

Mode 2 Baud Rate:

$$BoudRate = \frac{2}{64} \times SYSCLK$$

Mode 1/3 Baud Rate:

$$BoudRate = \frac{2^{SMOD0}}{32} \times \frac{SYSCLK}{X[256-(TH1)]}$$

X : Timer 1 Selects the clock source (reference Table 7-13.1)

T1SC	T1M	Description	X
0	0	System clock divided by 12 (8051)	12
0	1	System clock divided by 4 (8051)	4
1	x	System clock	1

Table 7-13.1 Timer 1 Selects the clock source

$$\Rightarrow TH1 = 256 - \frac{2^{SMOD0} \times SYSCLK}{32 \times X \times BoudRate}$$

SBUF: Serial Port 0 Data Buffer

Bit	7	6	5	4	3	2	1	0
Name	SBUF.7	SBUF.6	SBUF.5	SBUF.4	SBUF.3	SBUF.2	SBUF.1	SBUF.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

SCON: Serial Port 0 Control

Bit	7	6	5	4	3	2	1	0
Name	SM00	SM01	SM02	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	Mode	Description	Band Rate
0	0	0	Synchronous	SYSCLK/12
0	1	1	8-bit UART	Variable
1	0	2	9-bit UART	SYSCLK /32 or /64
1	1	3	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: UART reception disabled.

1: UART 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 SM02 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	UARTEN	-	-	DPS	-	PD	IDLE
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

Bit 7: UART double baud rate bit.

Bit 6: UART Enable bit.

Bits 5, 4, 2~1: Reserved.

Bit 3: Data Pointer Select.

This bit is used to switch between DPTR0 and DPTR1.

0: Select DPTR0.

1: Select DPTR1.

Bit 0: Idle mode control bit.

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

7.14 Timer

The EM85F684A/EM85F682A has two Timers and a set of TIMER 3. Timer 0 and Timer 1 is the Standard 8051 architecture with four modes. TIMER 3 is a simple 8-bit Timer. For Register description, refer to the subsequent sections.

7.14.1 Timer 0 and Timer 1

Timer 0 and Timer 1 has four kinds of operating modes which can be selected through TMOD (TxM1 and TxM0), as shown in Table 7.14. Each mode has two Data Register (THx and TLx), which can be used to access a 16-bit register. The CKCON0 register determines the Clock source and frequency. Interrupts is enabled by setting the IE register's ET0 and ET1 to "1" which is activated and displayed through TCON. The following are detailed description of each operating mode.

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 Counters/Timers (Timer 0 only)

Table 7.14 Timer 0/1 Modes

7.14.1.1 *Timer 0/1 Mode 0: 13-bit Counter/Timer*

Timer 0 is used as 13-bit counter/timers in Mode 0. The THx register holds the eight MSBs of the 13-bit counter/timer. TLx holds the five LSBs in bit positions (TLx.4–TLx.0). The three upper bits of TLx (TLx.7–TLx.5) are indeterminate and should be masked out or ignored when reading. As the 13-bit timer register increments and overflows from 0x1FFF (all ones) to 0x0000, the timer overflow flag TFx in TCON is set and an interrupt will occur if Timer 0/1 interrupts are enabled.

If C/Tx is set to high, high-to-low transitions at the Timer input pin (TCx) will increase the Timer/Counter Data register. When C/Tx is clear to low, selects the system clock to increase the timer/Counter Data register.

Setting the TRx bit enables the timer when either GATEx = 0, or GATEx = 1 and the input signal INTx is high. Setting GATEx to '1' allows the timer to be controlled by the external input signal INTx, facilitating positive pulse width in INTx measurements.

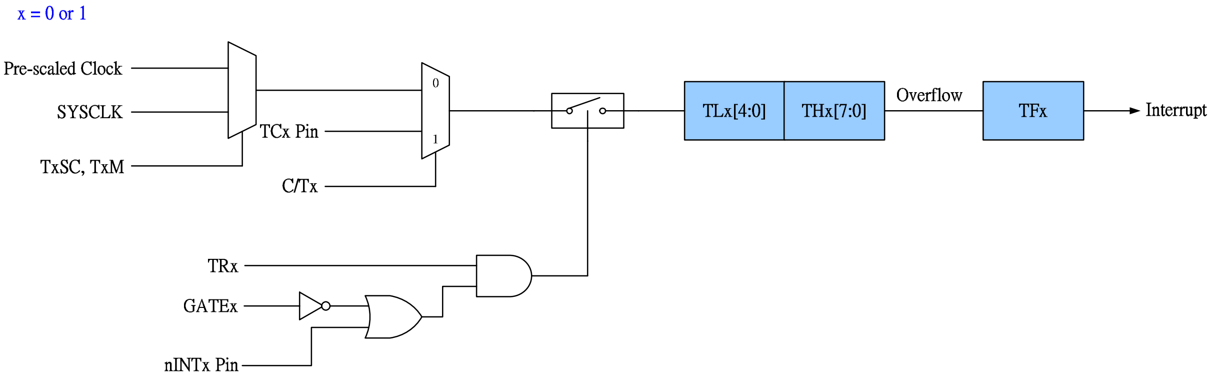


Figure 7.14.1 Timer 0/1 Mode 0 Block Diagram

7.14.1.2 *Timer 0/1 Mode 1: 16-bit Counter/Timer*

Mode 1 operation is the same as Mode 0, except that the counter/timer registers use all 16 bits. The Counter/Timers are enabled and configured in Mode 1 in the same manner as for Mode 0.

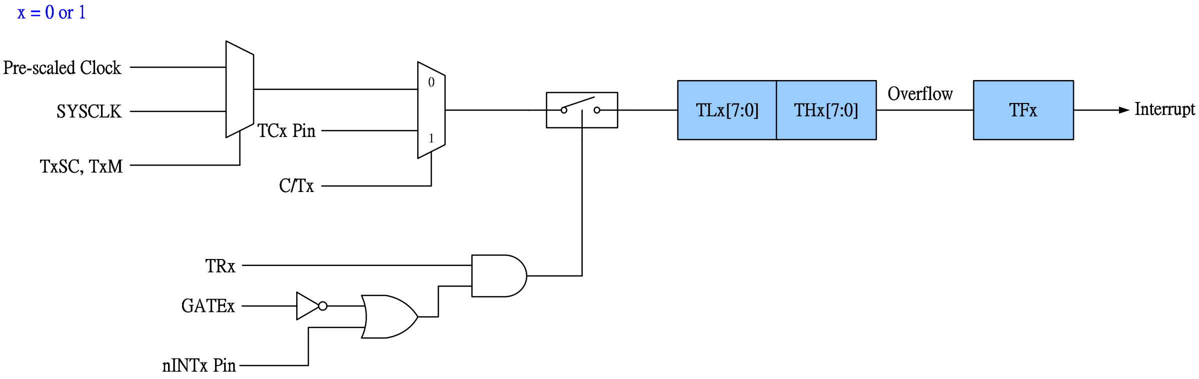


Figure 7.14.2 Timer 0/1 Mode 1 Block Diagram

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

Mode 2 configures Timer 0 and Timer 1 to operate as 8-bit counter/timers with automatic reload of the start value. TLx holds the count and THx holds the reload value. When the counter in TLx overflows from all ones to 0x00, 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 is not changed. Both counter/timers are enabled and configured in Mode 2 in the same manner as Mode 0.

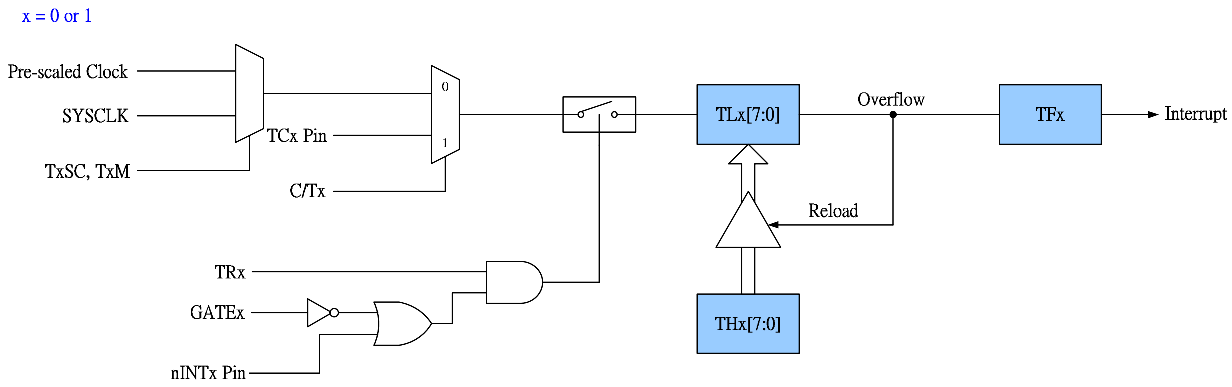


Figure 7.14.3 Timer 0/1 Mode 2 Block Diagram

7.14.1.4 *Timer 0 Mode 3 Two 8-bit Counter/Timers*

In Mode 3, Timer 0 is configured as two separate 8-bit counter/timers held in TL0 and TH0. TL0 can configure as timer or counter. TL0 is controlled using the Timer 0 control/status bits in TCON and TMOD: TR0, C/T0, GATE0, INT0 and TF0. TH0 only can configure as timer. TH0 is enabled using the Timer 1 run control bit TR1. TH0 sets the Timer 1 overflow flag TF1 on overflow and thus controls the Timer 1 interrupt.

Timer 1 is inactive in Mode 3. When Timer 0 is operating in Mode 3, Timer 1 can be operated in Modes 0, 1 or 2, but it cannot generate an interrupt and set the TF1 flag. The overflow of Timer1 can be used to generate Baud-Rate of serial ports for UART. TH1 and TL1 can be used as Timer function only. GATE1 bit is invalid. Timer1 run control is handled through its mode settings, because TR1 is used by Timer 0. When the Timer1 is in Modes 0, 1, or 2, Timer 1 is enabled. When the Timer 1 is in Mode 3, Timer 1 is disabled.

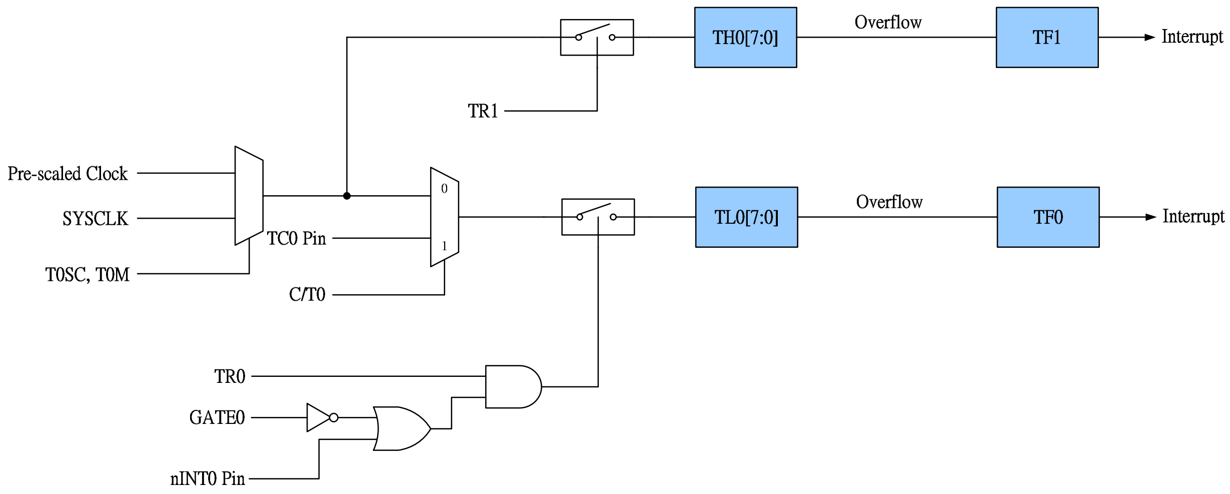


Figure 7.14.4 Timer 0 Mode 3 Block Diagram

CKCON0: Clock Control 0

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

SFR Address = 0x8E; SFR Page = All Pages

Bits 6, 4: Timer 1 Clock Select.

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

T1SC	T1M	Description
0	0	System clock divided by 12 (8051)
0	1	System clock divided by 4 (8051)
1	x	System clock

Bits 5, 3: Timer 0 Clock Select.

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

T0SC	T0M	Description
0	0	System clock divided by 12 (8051)
0	1	System clock divided by 4 (8051)
1	x	System clock

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

CKCON1: Clock Control 1

Bit	7	6	5	4	3	2	1	0
Name	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	LowFreqB
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	1

SFR Address = 0x8F; SFR Page = All Pages

Bit 7~0: Reserved, don’t use.
Bit 0: System clock selection.
Setting this bit activates CPU to enter low-speed operation.
0: CPU frequency using Low Speed Frequency (FLS). (Green mode)
1: CPU frequency using High Speed Frequency (FHS). (Normal Mode)

TCON: Timer Control

Bit	7	6	5	4	3	2	1	0
Name	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
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 = 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: External Interrupt 1 Status Flag.

This flag is set by hardware when an edge/level of type defined by IT1 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 1 service routine.

Bit 2: External Interrupt 1 Type Select.

This bit selects whether the configured INT1 interrupt will be edge or level sensitive. INT1 is configured falling edge triggered or rising edge triggered by the EIES1 bit in register EIESC1.

0: INT1 is low level triggered.

1: INT1 is edge triggered.

Bit 1: External Interrupt 0 Status Flag.

This flag is set by hardware when an edge/level of type defined by IT0 is detected. It can be cleared by software but is automatically cleared when the CPU vectors to the External Interrupt 0 service routine.

Bit 0: External Interrupt 0 Type Select.

This bit selects whether the configured INT0 interrupt will be edge or level sensitive. INT0 is configured falling edge triggered or rising edge triggered by the EIES0 bit in register EIESC1.

0: INT0 is low level triggered.

1: INT0 is edge triggered.

TMOD: Timer Mode

Bit	7	6	5	4	3	2	1	0
Name	GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0
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 = 0x89; SFR Page = All Pages

Bit 7: Timer 1 Gate Control.

0: Timer 1 enabled while TR1 control bit is set.

1: Timer 1 enabled while INT1 pin is high and TR1 control bit is set.

Bit 6: Counter/Timer 1 Select.

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

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

Bits 5~4: Timer 1 Mode Select.

T1M1	T0M0	Mode	Function Description
0	0	0	13-bit Counter/Timer
0	1	1	16-bit Counter/Timer
1	0	2	8-bit Counter/Timer with Auto-Reload
1	1	3	Timer 1 Inactive

Bit 3: Timer 0 Gate Control.

0: Timer 0 enabled while TR0 control bit is set.

1: Timer 0 enabled while INT0 pin is active high and TR0 control bit is set.

Bit 2: Counter/Timer 0 Se lect.

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

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

Bits 1~0: Timer 0 Mode Select.

T1M1	T0M0	Mode	Function Description
0	0	0	13-bit Counter/Timer
0	1	1	16-bit Counter/Timer
1	0	2	8-bit Counter/Timer with Auto-Reload
1	1	3	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

7.13.2 Timer 3 Register

In Timer mode, counting up is performed using internal clock. When the contents of up-timer are matched the TCxDA, then interrupt is generated and timer is cleared. Counting up resumes after the timer is cleared. The current contents of up-counter are loaded into TCxDB.

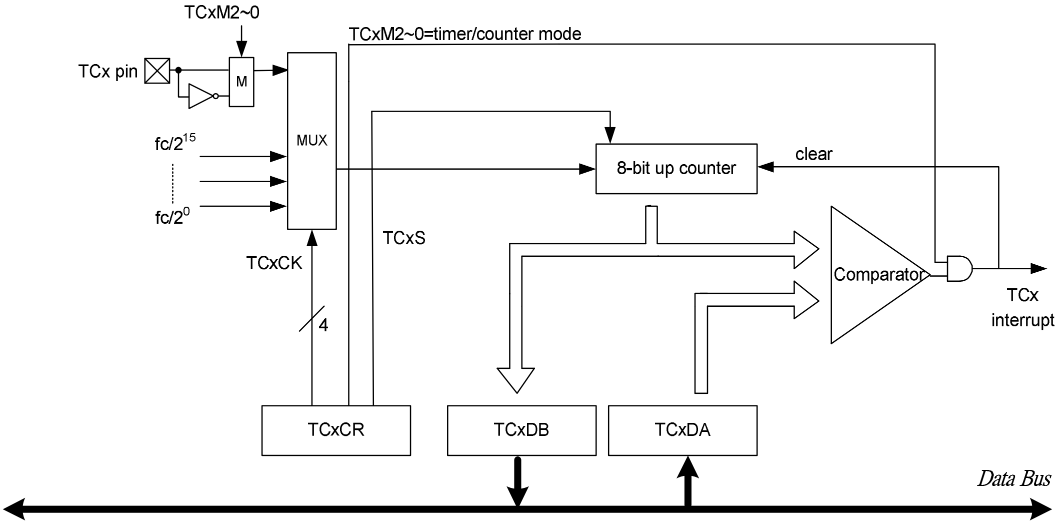


Figure 7-14.5 Timer 3 Mode

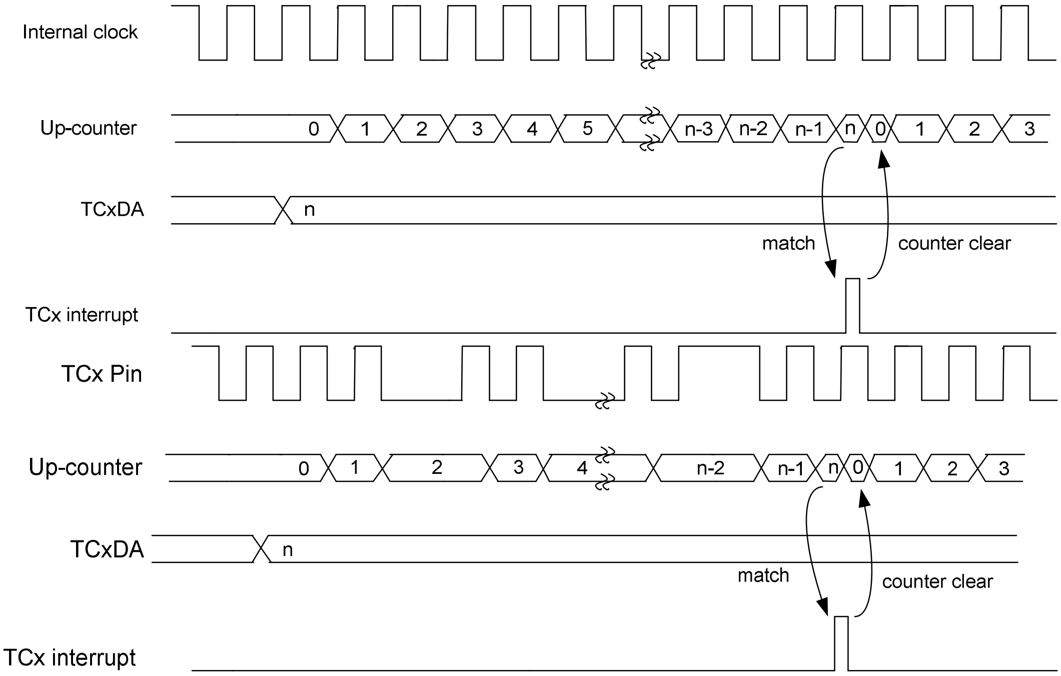


Figure 7.14.6 Timer Mode Waveform

TC3CR1: Timer 3 Control Register 1

Bit	7	6	5	4	3	2	1	0
Name	TC3S	--	--	TC3SF	--	TC3MOS	--	--
Type	R/W	R	R	R/W	R	R/W	R	R
Reset	0	1	0	0	0	0	0	0

SFR Address = 0xA9; SFR Page = 0

Bit 7: Timer 3 start control

0: Stop and clear timer.

1: Start.

Bit 4: Timer 3 Interrupt flag

Bits 6, 5, 3: Reserved

Data read from TC3DB is a number of counting.

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: reserved

TC3CR2: Timer 3 Control Register 2

Bit	7	6	5	4	3	2	1	0
Name	--	--	--	TC3SS0	TC3CK3	TC3CK2	TC3CK1	TC3CK0
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 = 0xAA; SFR Page = 0

Bits 7~5: Reserved. Timer 3 Rising Edge

Bit 4: Time 3 clock source selection bit.

0: The F_{512kHz} is used as count source.

1: The F_{HS} is used as count source.

Bits 3~0: Timer 3 clock source prescaler select.

TC3CK3	TC3CK2	TC3CK1	TC3CK0	Clock Source	Resolution 8 MHz	Max Time 8 MHz
				Normal	F_{HS} =8M	F_{HS} =8M
0	0	0	0	F_{HS}	125ns	32 μs
0	0	0	1	$F_{HS}/2$	250ns	64 μs
0	0	1	0	$F_{HS}/2^2$	500ns	128 μs
0	0	1	1	$F_{HS}/2^3$	1 μs	256 μs
0	1	0	0	$F_{HS}/2^4$	2 μs	512 μs
0	1	0	1	$F_{HS}/2^5$	4 μs	1024 μs
0	1	1	0	$F_{HS}/2^6$	8 μs	2048 μs
0	1	1	1	$F_{HS}/2^7$	16 μs	4096 μs
1	0	0	0	$F_{HS}/2^8$	32 μs	8192 μs
1	0	0	1	$F_{HS}/2^9$	64 μs	16384 μs
1	0	1	0	$F_{HS}/2^{10}$	128 μs	32768 μs
1	0	1	1	$F_{HS}/2^{11}$	256 μs	65536 μs
1	1	0	0	$F_{HS}/2^{12}$	512 μs	131072 μs
1	1	0	1	$F_{HS}/2^{13}$	1.024ms	262144 μs
1	1	1	0	$F_{HS}/2^{14}$	2.048ms	524.288ms
1	1	1	1	$F_{HS}/2^{15}$	4.096ms	1.048s

TC3DA: Timer 3 DATA Buffer A

Bit	7	6	5	4	3	2	1	0
Name	TC3DA7	TC3DA6	TC3DA5	TC3DA4	TC3DA3	TC3DA2	TC3DA1	TC3DA0
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 = 0

Bits 7~0: Data Buffer A of 8 bit Timer 3.

TC3DB: Timer 3 Data Buffer B

Bit	7	6	5	4	3	2	1	0
Name	TC3DB7	TC3DB6	TC3DB5	TC3DB4	TC3DB3	TC3DB2	TC3DB1	TC3DB0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xAC; SFR Page = 0

Bits 7~0: Data Buffer B of 8 bit timer (time value)

7.15 PWM

In PWM mode, up to 16-bit resolution PWM output is produced. 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.

For example, set period and duty cycle (Period > Duty), PWMXE=1/0 (X=A/B/C/D/E/F), PWMXA=1/0, then finally set TXEN=1. Figures 7.14.1 to 7.14.2 show the PWM output timing according to different PWMXA settings.

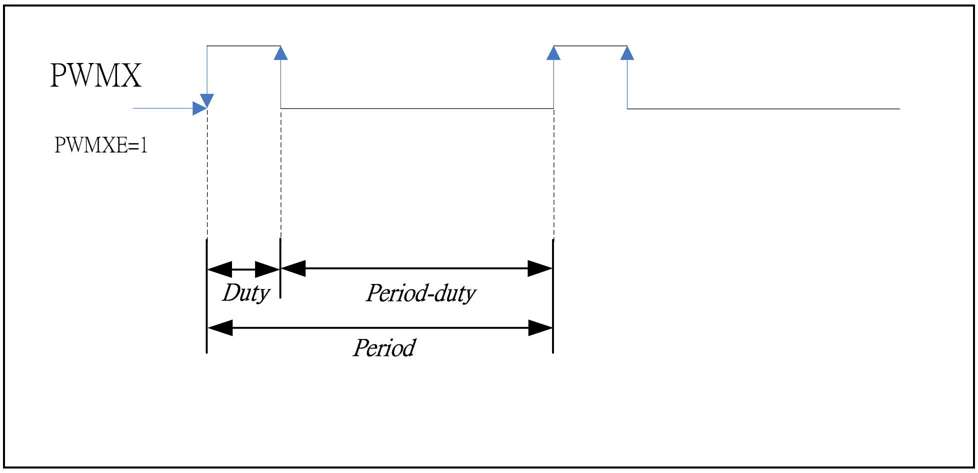


Figure 7.15.1 PWM Output Timing (PWMXA=0)

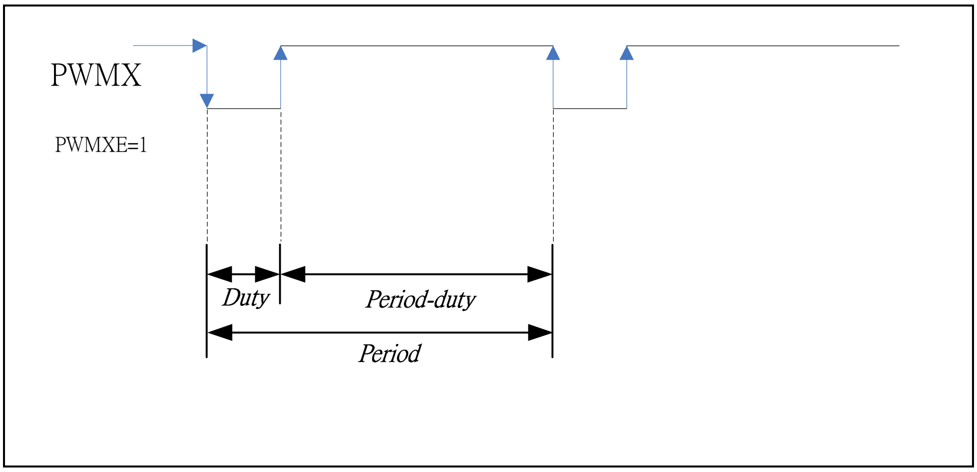


Figure 7.15.2 PWM Output Timing (PWMXA=1)

7.15.1 Increment Timer Counter (TMRX)

TMRX is an 8-bit clock counter with programmable prescalers. It is designed for the PWM module as baud rate clock generators. TMRX can be read only. If employed, it can be turned off for power saving by setting the TXEN bit. TMRX is internally designed and cannot be written to.

7.15.2 PWM Time Period (PRDX)

The PWM period is 16-bit resolution. The PWM time period is defined by writing to the PRDX register. When TMRX is equal to PRDX, the following events occur on the next increment cycle:

- TMRX is cleared
- The PWM pin is set to “1”

NOTE
The PWM output will not be set if the duty cycle is 0

- The PWMIF(PWM Interrupt Flag) pin is set to “1”

The following formula describes how to calculate the PWM time period:

$$Period = (PRDX + 1) \times \left(\frac{1}{Fpwmclk} \right) \times (TMRX\ \ prescale\ \ value)$$

Example:

PRDX = 49; Fpwmclk = 4 MHz; TMRX prescale (0, 0, 0) = 1 : 1,

Then

$$Period = (49 + 1) \times \left(\frac{1}{4M} \right) \times 1 = 12.5\mu s$$

7.15.3 PWM Duty Cycle (DTX)

The PWM duty cycle is defined by writing to the DTX register, and is latched DTX while TMRX is cleared. When DTX is equal to TMRX, the PWM pin is cleared. DTX can be loaded anytime. However, it cannot be latched into DL until the low byte of PWM period cycle is rewritten into PRDXL register.

The following formula describes how to calculate the PWM duty cycle:

Example:

**DTX = 10; F$_{\text{pwmclk}}$ = 4 MHz; TMRX prescale (0, 0, 0) = 1 : 1,
Then**

$$Duty\ cycle = (10) \times \left(\frac{1}{4M}\right) \times 1 = 2.5\mu s$$

Dual PWM Function

It consists of a complementary PWM (i.e. PWMX and /PWMX) X=A/B/C, one output PWM signal and another output inverted PWM signal. It can output any pulse width signal you want by programming the relative control registers.

Set the period and duty cycle (Period > Duty). Set PWMXE, PWMXA = 0/1, and finally set TXEN = 1.

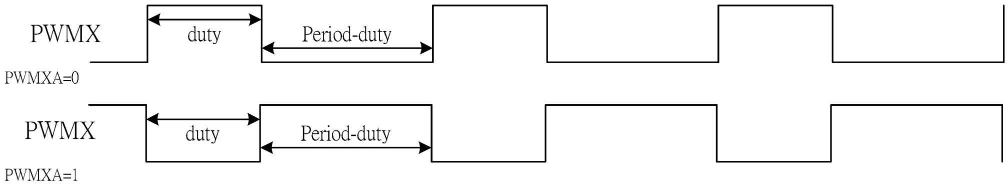


Figure 7.15.3 Dual PWMX Output Waveform

7.15.4 PWM Programming Process/Steps

1. Load the PWM duty cycle to DTX.
2. Load the PWM time period to PRDX.
3. Enable the interrupt function by writing to PWMIE (PWM interrupt enable) if required.
4. Load a desired value for the timer prescaler.
5. Set active level of duty of PWM.
6. Enable PWMX function, i.e., enable PWMXE control bit. (If using dual PWM function, enable IPWMXE control bit, too)
7. Finally, enable TMRX function, i.e., enable TXEN control bit.

***If the application needs to change PWM duty, period cycle at run time, refer to the following** programming steps:

1. Load the new duty cycle anytime.

2. Load the 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 the circuit**.

3. The circuit would automatically update the new duty, period cycle to generate new PWM waveform at the next PWM cycle.

7.15.5 PWM Register

PWMSF

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	-	-	PWMCPSF	PWMCDS	PWMBPSF	PWMBDSF	PWMAPSF	PWMADSF
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDF; SFR Page = 1

Bits 7~6: Reserved

Bit 5 (PWMCPSF)**:** PWMC period match status flag. Reset by software

Bit 4 (PWMCDSF)**:** PWMC duty match status flag. Reset by software.

Bit 3 (PWMBPSF)**:** PWMB period match status flag. Reset by software.

Bit 2 (PWMBDSF)**:** PWMB duty match status flag. Reset by software.

Bit 1 (PWMAPSF)**:** PWMA period match status flag. Reset by software.

Bit 0 (PWMADSF)**:** PWMA duty match status flag. Reset by software.

PWMTER (PWM Timer Enable Register)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	-	-	-	-	-	TCEN	TBEN	TAEN
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	

SFR Address = 0xDE; SFR Page = 1

Bits 7~3: Reserved. Don't use.

Bit 2 (TCEN): TMRC enable bit. All PWM functions are valid only as this bit is set

0: TMRC is off (default value)

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 value)

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 value)

1: TMRA is on

PWMXE/IPWMXE/TXEN: X=A, B, C

PWMXE	IPWMXE	TXEN	Function description
0	0	0	Not used as PWM function; I/O pin or other functional pin.
0	0	1	Timer function; I/O pin or other function pin.
0	1	0	PWMX: Not used as PWM function; I/O pin or other functional pin. IPWMX: PWM function, the waveform keeps at inactive level.
1	0	0	PWMX: PWM function, the waveform keeps at inactive level. IPWMX: Not used as PWM function; I/O pin or other functional pin.
1	1	0	PWMX and IPWMX: PWM function, the waveform keeps at inactive level.
0	1	1	PWMX: Not used as PWM function; I/O pin or other functional pin. IPWMX: PWM function, the normal PWM output waveform.
1	0	1	PWMX: PWM function, the normal PWM output waveform. IPWMX: Not used as PWM function; I/O pin or other functional pin.
1	1	1	PWMX and IPWMX: PWM function, the normal PWM output waveform.

PWMACR (PWMA Control Register)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PWMAE	-	PWMAA	-	PWMAS	TAP2	TAP1	TAP0
Type	R/W	-	R/W	-	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XE1; SFR Page = 1

Bit 7 (PWMAE): PWMA enable bit

0: Disable

1: Enable

Bit 6 : Reserved. Don’t use.

Bit 5 (PWMAA): Active level of PWMA

0: duty time is Logic 1

1: duty time is Logic 0

Bit 4 : Reserved. Don’t use.

Bit 3 (PWMAS): Clock selection for PWMA timer

0: F_{512kHz}

1: F_{HS}

Bits 2 ~ 0 (TAP2 ~ TAP0): TMR1 clock prescale 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

PRDAL (PWMA Period Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDA7	PRDA6	PRDA5	PRDA4	PRDA3	PRDA2	PRDA1	PRDA0
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 = 1

Bits 7~0: The content of the register is the PWMA period Low Byte.

- **PWMA duty/period reloads for PRDAL register update**

PRDAH (PWMA Period High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDA15	PRDA14	PRDA13	PRDA12	PRDA11	PRDA10	PRDA9	PRDA8
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 = 1

Bits 7~0: The content of the register is the PWMA period High Byte.

DTAL (PWMA Duty Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTA7	DTA6	DTA5	DTA4	DTA3	DTA2	DTA1	DTA0
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 = 1

Bits 7~0 (DTAL[7~0]): The content of the register is PWMA duty Low Byte.

DTAH (PWMA Duty High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTA15	DTA14	DTA13	DTA12	DTA11	DTA10	DTA9	DTA8
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 = 1

Bits 7~0 (DTAH[7~0]): The content of the register is PWMA duty High byte.

TMRAL (PWMA Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRA7	TMRA6	TMRA5	TMRA4	TMRA3	TMRA2	TMRA1	TMRA0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XE6; SFR Page = 1

Bits 7~0 (TMRAL[7~0]): The content of the register is PWMA timer Low Byte which is counting. This is read-only.

TMRAH (PWMA Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRA15	TMRA14	TMRA13	TMRA12	TMRA11	TMRA10	TMRA9	TMRA8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XE7; SFR Page = 1

Bits 7~0 (TMRAH[7~0]): The content of the register is PWMA timer High Byte which is counting. This is read-only.

PWMBCR (PWMB Control Register)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PWMBE	-	PWMBA	-	PWMBS	TBP2	TBP1	TBP0
Type	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 = 1

Bit 7 (PWMBE): PWMB enable bit

0: Disable

1: Enable

Bit 6 : Reserved. Don’t use.

Bit 5 (PWMBA): Active level of PWMB

0: duty time is Logic 1

1: duty time is Logic 0

Bit 4 : Reserved. Don’t use.

Bit 3 (PWMBS): Clock selection for PWMB timer

0: F 512kHz

1: FHS

Bits 2 ~ 0 (TBP2 ~ TBP0): TMR1 clock prescale 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

PRDBL (PWMB Period Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDB7	PRDB6	PRDB5	PRDB4	PRDB3	PRDB2	PRDB1	PRDB0
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 = 1

Bits 7~0: The content of the register is the PWMB period Low Byte.

- **PWMB duty/period reloads for PRDBL register update**

● PRDBH (PWMB Period High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDB15	PRDB14	PRDB13	PRDB12	PRDB11	PRDB10	PRDB9	PRDB8
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 = 0XEB; SFR Page = 1**
- **Bits 7~0:** The content of the register is the PWMB period High Byte.

DTBL (PWMB Duty Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTB7	DTB6	DTB5	DTB4	DTB3	DTB2	DTB1	DTB0
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 = 0XEC; SFR Page = 1

Bits 7~0: The content of the register is PWMB duty Low Byte.

DTBH (PWMB Duty High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTB15	DTB14	DTB13	DTB12	DTB11	DTB10	DTB9	DTB8
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 = 0XED; SFR Page = 1

Bits 7~0: The content of the register is PWMB duty High Byte.

TMRBL (PWMB Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRB7	TMRB6	TMRB5	TMRB4	TMRB3	TMRB2	TMRB1	TMRB0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XEE; SFR Page = 1

Bits 7~0 (TMRBL[7~0]): The content of the register is the PWMB timer Low Byte which is counting. This is read-only.

TMRBH (PWMB Timer High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRB15	TMRB14	TMRB13	TMRB12	TMRB11	TMRB10	TMRB9	TMRB8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XEF; SFR Page = 1

Bits 7~0 (TMRBH[7~0]): The content of the register is the PWMB timer High Byte which is counting. This is read-only.

PWMCCR (PWMC Control Register)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PWMCE	-	PWMCA	-	PWMCS	TCP2	TCP1	TCP0
Type	R/W	-	R/W	-	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XF1; SFR Page = 1

Bit 7 (PWMCE): PWMC enable bit

0: Disable

1: Enable

Bit 6 : Reserved. Don't use.

Bit 5 (PWMCA): Active level of PWMC

0: Duty Time is Logic 1

1: Duty Time is Logic 0

Bit 4 : Reserved. Don't use.

Bit 3 (PWMCS): Clock selection for PWMC timer

0: F_{512kHz}

1: F_{HS}

Bits 2 ~ 0 (TCP2 ~ TCP0): TMR1 clock prescale 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

PRDCL (PWMC Period Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDC7	PRDC6	PRDC5	PRDC4	PRDC3	PRDC2	PRDC1	PRDC0
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 = 0XF2; SFR Page = 1

Bits 7~0 (PRDCL[7~0]): The content of the register is the PWMC period Low Byte.

●**PWMC duty/period reloads for PRDCL register update**

- ## PRDCH (PWMC Period High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDC15	PRDC14	PRDC13	PRDC12	PRDC11	PRDC10	PRDC9	PRDC8
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 = 0XF3; SFR Page = 1**
- **Bits 7~0 (PRDCH[7~0]):** The content of the register is the PWMC period High Byte.

DTCL (PWMC Duty Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTC7	DTC6	DTC5	DTC4	DTC3	DTC2	DTC1	DTC0
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 = 0XF4; SFR Page = 1

Bits 7~0 (DTCL[7~0]): The content of the register is PWMC duty Low Byte.

DTCH(PWMC Duty High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTC15	DTC14	DTC13	DTC12	DTC11	DTC10	DTC9	DTC8
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 = 0XF5; SFR Page = 1

Bits 7~0 (DTCH[7~0]): The content of the register is PWMC duty High Byte.

TMRCL (PWMC Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRC7	TMRC6	TMRC5	TMRC4	TMRC3	TMRC2	TMRC1	TMRC0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XF6; SFR Page = 1

Bits 7~0 (TMRCL[7~0]): The content of the register is the PWMC timer Low Byte which is counting. This is read-only.

TMRCH (PWMC Timer High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRC15	TMRC14	TMRC13	TMRC12	TMRC11	TMRC10	TMRC9	TMRC8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XF7; SFR Page = 1

Bits 7~0 (TMRCH[7~0]): The content of the register is the PWMC timer High Byte which is counting. This is read-only.

PWMSF1

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	-	-	PWMFPSF	PWMFDSF	PWMEPSF	PWMEDSF	PWMDPSF	PWMDDSF
Type	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0xDF; SFR Page = 2

Bits 7~6: Reserved

Bit 5 (PWMFPSF**):**PWMF period match status flag. Reset by software

Bit 4 (PWMFDSF**):**PWMF duty match status flag. Reset by software.

Bit 3 (PWMEPSF**):**PWME period match status flag. Reset by software.

Bit 2 (PWMEDSF**):**PWME duty match status flag. Reset by software.

Bit 1 (PWMDPSF**):**PWMD period match status flag. Reset by software.

Bit 0 (PWMDDSF**):**PWMD duty match status flag. Reset by software.

PWMTER1 (PWM Timer Enable Register 1)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	-	-	-	-	-	TFEN	TEEN	TDEN
Type	-	-	-	-	-	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	

SFR Address = 0xDE; SFR Page = 2

Bits 7~3: Reserved. Don't use.

Bit 2 (TFEN): TMRC enable bit. All PWM function are valid only as this bit is set

0: TMRC is off (default value)

1: TMRC is on

Bit 1 (TEEN): TMRB enable bit. All PWM function are valid only as this bit is set

0: TMRB is off (default value)

1: TMRB is on

Bit 0 (TDEN): TMRA enable bit. All PWM function are valid only as this bit is set

0: TMRA is off (default value)

1: TMRA is on

PWMXE/IPWMXE/TXEN: X=D, E, F

PWMXE	IPWMXE	TXEN	Function description
0	0	0	Not used as PWM function; I/O pin or other functional pin.
0	0	1	Timer function; I/O pin or other function pin.
0	1	0	PWMX: Not used as PWM function; I/O pin or other functional pin. IPWMX: PWM function, the waveform keeps at inactive level.
1	0	0	PWMX: PWM function, the waveform keeps at inactive level. IPWMX: Not used as PWM function; I/O pin or other functional pin.
1	1	0	PWMX and IPWMX: PWM function, the waveform keeps at inactive level.
0	1	1	PWMX: Not used as PWM function; I/O pin or other functional pin. IPWMX: PWM function, the normal PWM output waveform.

1	0	1	PWMX: PWM function, the normal PWM output waveform. IPWMX: Not used as PWM function; I/O pin or other functional pin.
1	1	1	PWMX and IPWMX: PWM function, the normal PWM output waveform.

PWMDCR (PWMD Control Register)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PWMDE		PWMDA		PWMDS	TDP2	TDP1	TDP0
Type	R/W	-	R/W	-	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XE1; SFR Page = 2

Bit 7 (PWMDE): PWMD enable bit

0: Disable

1: Enable

Bit 5 (PWMDA): Active level of PWMD

0: duty time is Logic 1

1: duty time is Logic 0

Bit 4 : Reserved. Don't use.

Bit 3 (PWMDS): Clock selection for PWMD timer

0: Fs

1: Fm

Bits 2 ~ 0 (TDP2 ~ TDP0): TMR1 clock prescale option bits

TDP2	TDP1	TDP0	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

PRDDL (PWMD Period Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDD7	PRDD6	PRDD5	PRDD4	PRDD3	PRDD2	PRDD1	PRDD0
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 = 2

Bits 7~0: The content of the register is the PWMD period Low Byte.

- **PWMD duty/period reloads for PRDDL register update**

PRDDH (PWMD Period High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDD15	PRDD14	PRDD13	PRDD12	PRDD11	PRDD10	PRDD9	PRDD8
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 = 2

Bits 7~0: The content of the register is the PWMD period High Byte.

DTDL (PWMD Duty Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTD7	DTD6	DTD5	DTD4	DTD3	DTD2	DTD1	DTD0
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 = 2

Bits 7~0 (DTDL[7~0]): The content of the register is PWMD duty Low Byte.

DTDH (PWMD Duty High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTD15	DTD14	DTD13	DTD12	DTD11	DTD10	DTD9	DTD8
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 = 2

Bits 7~0 (DTDH[7~0]): The content of the register is PWMD duty High byte.

TMRDL (PWMD Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRD7	TMRD6	TMRD5	TMRD4	TMRD3	TMRD2	TMRD1	TMRD0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XE6; SFR Page = 2

Bits 7~0 (TMRDL[7~0]): The content of the register is PWMD timer Low Byte which is counting. This is read-only.

TMRDH (PWMD Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRD15	TMRD14	TMRD13	TMRD12	TMRD11	TMRD10	TMRD9	TMRD8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XE7; SFR Page = 2

Bits 7~0 (TMRDH[7~0]): The content of the register is PWMD timer High Byte which is counting. This is read-only.

PWMECR (PWME Control Register)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PWMEE		PWMEA		PWMES	TEP2	TEP1	TEP0
Type	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 = 2

Bit 7 (PWMEE): PWME enable bit

0: Disable

1: Enable

Bit 5 (PWMEA): Active level of PWME

0: duty time is Logic 1

1: duty time is Logic 0

Bit 4 : Reserved. Don’t use.

Bit 3 (PWMES): Clock selection for PWME timer

0: Fs

1: Fm

Bits 2 ~ 0 (TEP2 ~ TEP0): TMR1 clock prescale option bits

TEP2	TEP1	TEP0	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

PRDEL (PWME Period Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDE7	PRDE6	PRDE5	PRDE4	PRDE3	PRDE2	PRDE1	PRDE0
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 = 2

Bits 7~0: The content of the register is the PWME period Low Byte.

- **PWME duty/period reloads for PRDeL register update**

● PRDEH (PWME Period High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDE15	PRDE14	PRDE13	PRDE12	PRDE11	PRDE10	PRDE9	PRDE8
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 = 0XEB; SFR Page = 2**
- **Bits 7~0:** The content of the register is the PWME period High Byte.

DTEL (PWME Duty Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTE7	DTE6	DTE5	DTE4	DTE3	DTE2	DTE1	DTE0
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 = 0XEC; SFR Page = 2

Bits 7~0: The content of the register is PWME duty Low Byte.

DTEH (PWME Duty High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTE15	DTE14	DTE13	DTE12	DTE11	DTE10	DTE9	DTE8
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 = 0XED; SFR Page = 2

Bits 7~0: The content of the register is PWME duty High Byte.

TMREL (PWME Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRE7	TMRE6	TMRE5	TMRE4	TMRE3	TMRE2	TMRE1	TMRE0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XEE; SFR Page = 2

Bits 7~0 (TMRBL[7~0]): The content of the register is the PWME timer Low Byte which is counting. This is read-only.

TMREH (PWME Timer High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRE15	TMRE14	TMRE13	TMRE12	TMRE11	TMRE10	TMRE9	TMRE8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XEF; SFR Page = 2

Bits 7~0 (TMRBH[7~0]): The content of the register is the PWME timer High Byte which is counting. This is read-only.

PWMFCR (PWMF Control Register)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PWMFE	-	PWMFA	-	PWMFS	TFP2	TFP1	TFP0
Type	R/W	-	R/W	-	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XF1; SFR Page = 2

Bit 7 (PWMFE): PWMF enable bit

0: Disable

1: Enable

Bit 5 (PWMFA): Active level of PWMF

0: duty time is Logic 1

1: duty time is Logic 0

Bit 4 : Reserved. Don’t use.

Bit 3 (PWMFS): Clock selection for PWMF timer

0: Fs

1: Fm

Bits 2 ~ 0 (TFP2 ~ TFP0): TMR1 clock prescale option bits

TFP2	TFP1	TFP0	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

PRDFL (PWMF Period Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDF7	PRDF6	PRDF5	PRDF4	PRDF3	PRDF2	PRDF1	PRDF0
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 = 0XF2; SFR Page = 2

Bits 7~0 (PRDFL[7~0]): The content of the register is the PWMF period Low Byte.

●**PWMF duty/period reloads for PRDFL register update**

- **PRDFH (PWMF Period High Byte)**

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PRDF15	PRDF14	PRDF13	PRDF12	PRDF11	PRDF10	PRDF9	PRDF8
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 = 0XF3; SFR Page = 2**
- **Bits 7~0 (PRDFH[7~0]):** The content of the register is the PWMF period High Byte.

DTFL (PWMF Duty Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTF7	DTF6	DTF5	DTF4	DTF3	DTF2	DTF1	DTF0
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 = 0XF4; SFR Page = 2

Bits 7~0 (DTFL[7~0]): The content of the register is PWMF duty Low Byte.

DTFH(PWMF Duty High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DTF15	DTF14	DTF13	DTF12	DTF11	DTF10	DTF9	DTF8
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 = 0XF5; SFR Page = 2

Bits 7~0 (DTFH[7~0]): The content of the register is PWMF duty High Byte.

TMRFL (PWMF Timer Low Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRF7	TMRF6	TMRF5	TMRF4	TMRF3	TMRF2	TMRF1	TMRF0
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XF6; SFR Page = 2

Bits 7~0 (TMRFL[7~0]): The content of the register is the PWMF timer Low Byte which is counting. This is read-only.

TMRCH (PWMC Timer High Byte)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TMRF15	TMRF14	TMRF13	TMRF12	TMRF11	TMRF10	TMRF9	TMRF8
Type	R	R	R	R	R	R	R	R
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XF7; SFR Page = 2

Bits 7~0 (TMRFH[7~0]): The content of the register is the PWMF timer High Byte which is counting. This is read-only.

7.16 SPI

7.16.1 SPI Signal

7.16.1.1 *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 1.

The clock rate is selected by the SPI clock rate selects Bit SPR[2:0] in the SPICON of the master Device 2.

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.

7.16.1.2 *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.

7.16.1.3 *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.

7.16.1.4 *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.

7.16.2 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, SPIF 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.

7.16.3 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.

7.16.4 Operations

7.16.4.1 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 8 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.

7.16.4.2 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.

7.16.5 Timing Diagram

7.16.5.1 SPI Master Mode

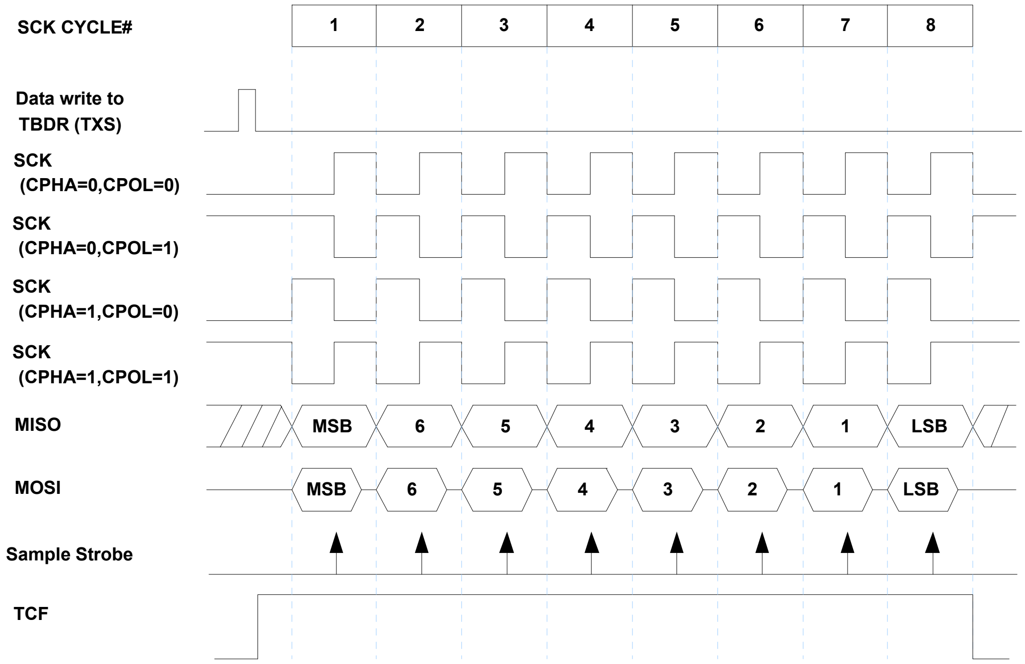


Figure 7.16.1 SPI Master Mode Timing Diagram

7.16.5.2 SPI Slave Mode Timing (CPHA=1 / CHPA=0)

When CPHA=0 condition, MISO MSB is ready before the first toggling of SCK.

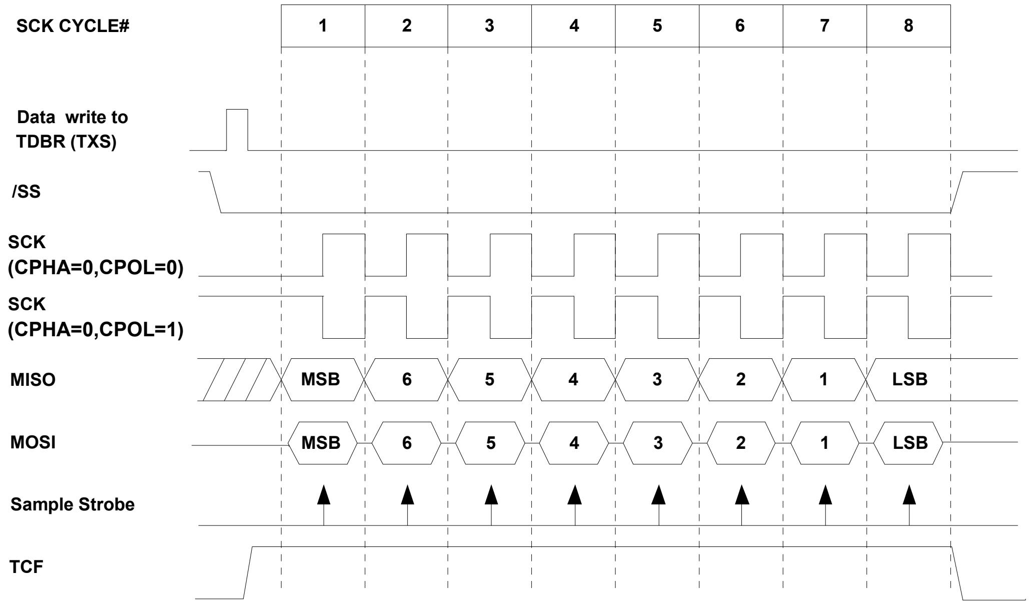


Figure 7.16.2 SPI Slave Mode Timing (CPHA=0) Diagram

When CPHA=1 condition, MISO MSB is not valid until the first toggling of SCK

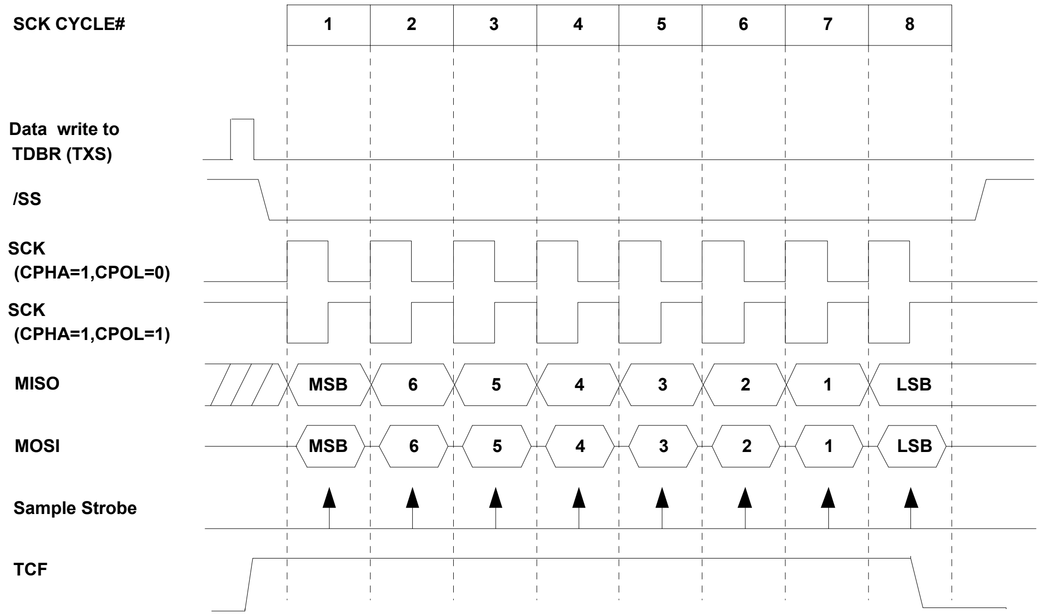


Figure 7.16.3 SPI Slave Mode Timing (CPHA=1) Diagram

7.16.5.3 Consecutively Receiving Bytes (CPHA=1 / CHPA=0)

Figure 7.16.4 and 7.16.5 are consecutive transmission charts, not real timing diagrams. TXS=0 means that the buffer is empty, so after 8 cycles TXS will be 0. “Write Byte 3” early before the ninth cycle.

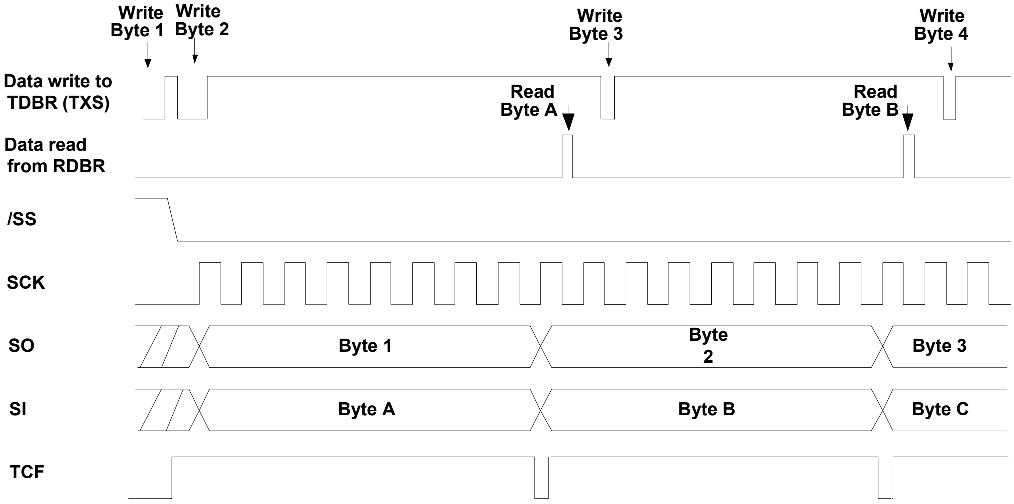


Figure 7.16.4 Consecutively Receiving Bytes Timing (Master or Slave Mode CPOL=0; CPHA=1)

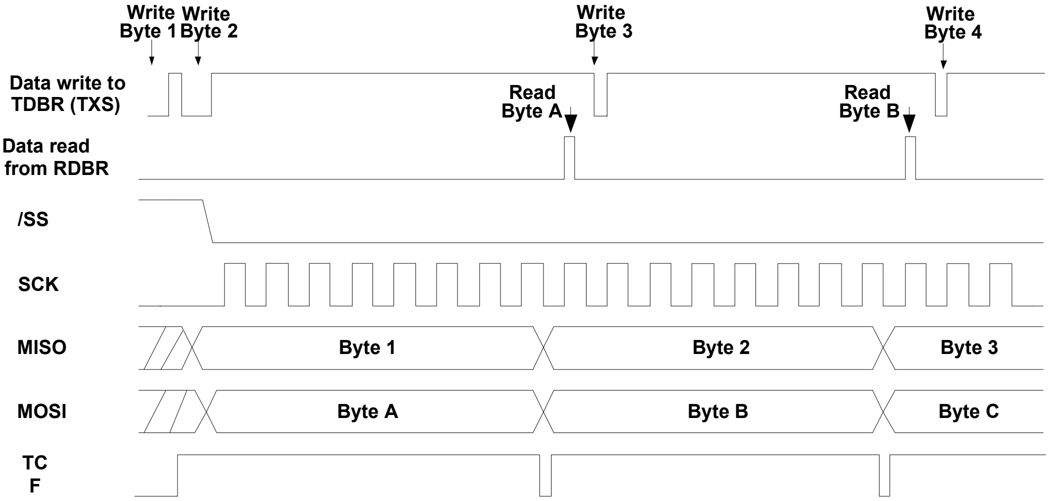


Figure 7.16.5 Consecutively Receiving Bytes Timing (Master or Slave Mode CPOL=0; CPHA=0)

7.16.5.4 SPI Slave Timing Requirement

Figure 7.16.7 shows SPI slave timing requirement. It's necessary to be 10 or 0 system clock cycles between consecutive SCK transfers for single mode or buffer mode, respectively. 2 system cycles is 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 to enter ISR (Interrupt Service Routine) after last SCK toggling. It needs 4 system cycles to be loaded into Tx shift registers after executing writing TDBR instruction.

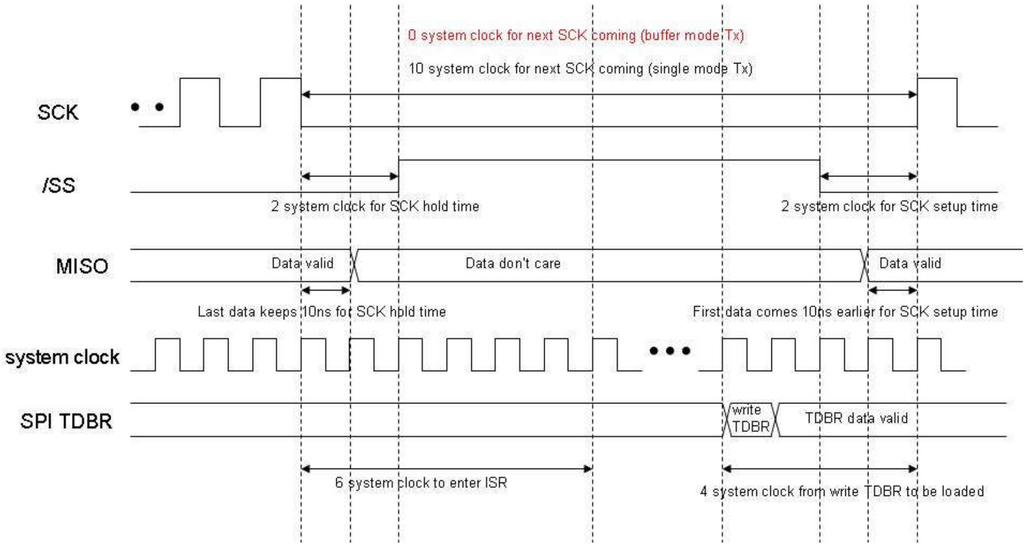
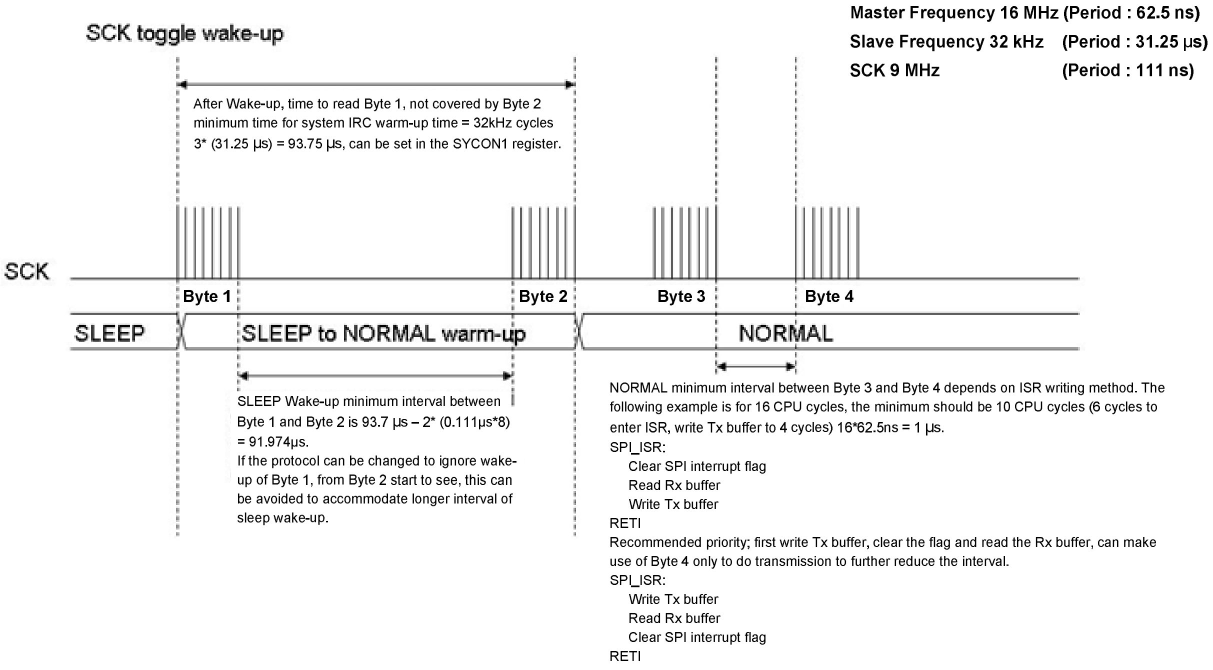

0 system clock for next SCK coming (buffer mode Tx)
10 system clock for next SCK coming (single mode Tx)
SCK
/SS
2 system clock for SCK hold time
2 system clock for SCK setup time
MISO
Data valid
Data don't care
Data valid
Last data keeps 10ns for SCK hold time
First data comes 10ns earlier for SCK setup time
system clock
SPI TDBR
write TDBR
TDBR data valid
6 system clock to enter ISR
4 system clock from write TDBR to be loaded

Figure 7.16.7 SPI Slave Timing Requirements

In buffer mode, the minimum interval between consecutive transfers is 0 system clock cycles. It can be more than 0 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 , the interval can be minimal as 10 system cycles if writing TDBR first in ISR. The interval will be larger as 16 system cycles if writing TDBR later in ISR. Therefore, TDBR writing should be done first to improve the efficiency of SPI slave single mode.

As a slave in buffer mode with 0 word (byte) interval, the maximum SCK provided by the master is about 8 MHz It depends on the IO pad and PCB delays between master and slave. The maximum SCK may increase to 12 MHz or 15 MHz if word (byte) interval is more than 0.



Master Frequency 16 MHz (Period : 62.5 ns)
Slave Frequency 32 kHz (Period : 31.25 μs)
SCK 9 MHz (Period : 111 ns)

After Wake-up, time to read Byte 1, not covered by Byte 2 minimum time for system IRC warm-up time = 32kHz cycles 3* (31.25 μs) = 93.75 μs, can be set in the SYCON1 register.

SLEEP Wake-up minimum interval between Byte 1 and Byte 2 is 93.7 μs – 2* (0.111μs*8) = 91.974μs.
If the protocol can be changed to ignore wake-up of Byte 1, from Byte 2 start to see, this can be avoided to accommodate longer interval of sleep wake-up.

NORMAL minimum interval between Byte 3 and Byte 4 depends on ISR writing method. The following example is for 16 CPU cycles, the minimum should be 10 CPU cycles (6 cycles to enter ISR, write Tx buffer to 4 cycles) 16*62.5ns = 1 μs.

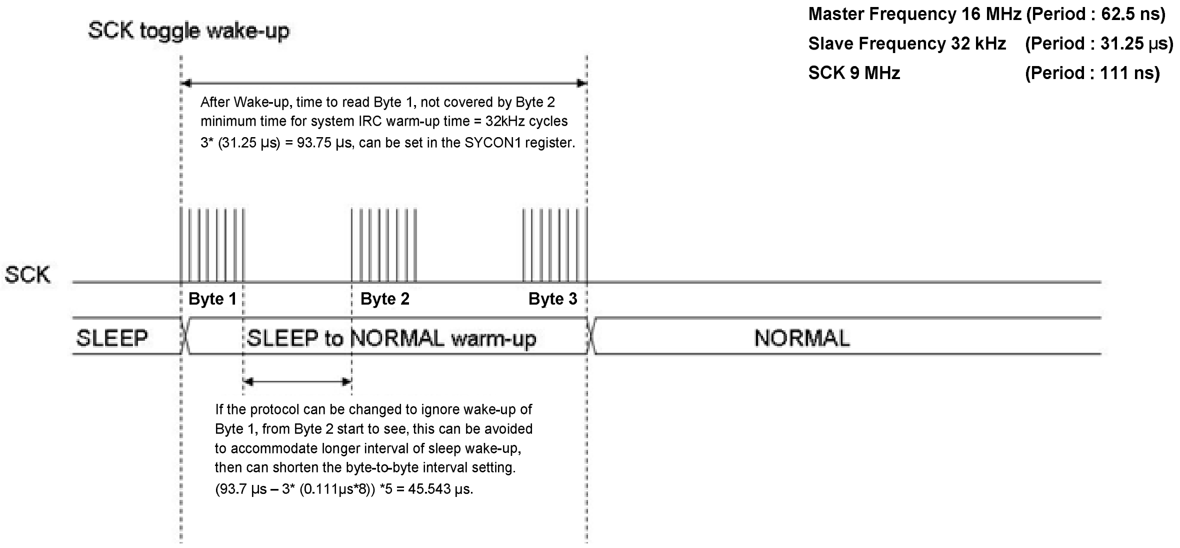
```
SPI_ISR:
    Clear SPI interrupt flag
    Read Rx buffer
    Write Tx buffer
RETI
```

Recommended priority; first write Tx buffer, clear the flag and read the Rx buffer, can make use of Byte 4 only to do transmission to further reduce the interval.

```
SPI_ISR:
    Write Tx buffer
    Read Rx buffer
    Clear SPI interrupt flag
RETI
```

Figure 7.16.8 SPI Single Mode Interval between Consecutive Transfers Timing Diagram 1

In Figure 7.16.8, the interval of SPI slave sleep wake-up transfers is also described. If the system needs 3 warm-up cycles (31.25 μs period), the interval cannot be less than 91.974 μs to avoid Byte 1 overwritten by Byte 2. The interval becomes so large because of 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 Byte 1 data in Figure 7.16.9.



Master Frequency 16 MHz (Period : 62.5 ns)
Slave Frequency 32 kHz (Period : 31.25 μs)
SCK 9 MHz (Period : 111 ns)

After Wake-up, time to read Byte 1, not covered by Byte 2 minimum time for system IRC warm-up time = 32kHz cycles 3* (31.25 μs) = 93.75 μs, can be set in the SYCON1 register.

If the protocol can be changed to ignore wake-up of Byte 1, from Byte 2 start to see, this can be avoided to accommodate longer interval of sleep wake-up, then can shorten the byte-to-byte interval setting. (93.7 μs – 3* (0.111μs*8)) *5 = 45.543 μs.

Figure 7.16.9 SPI Single Mode Interval between Consecutive Transfers Timing Diagram 2

7.16.6 Buffer mode function

The buffer mode is used to transmit and receive multiple data bytes (or words) in one interrupt to save repetitive ISR (Interrupt Sub-Routine) cycles of single mode. The setting flow is shown in 7.15.10. Different lengths can be set for Tx and Rx FIFO. When the interrupt happens, the application will enter ISR and first check “RX_INT/TX_INT” bits of status register to know that it’s Rx or Tx interrupt. Then the application can decide to read Rx FIFO data or fill new data into Tx FIFO.

For slave transmission condition, 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 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, the application must read all Rx FIFO data after entering Rx ISR. The error will happen if new Rx data is coming and there’s still previous data not read in Rx FIFO.

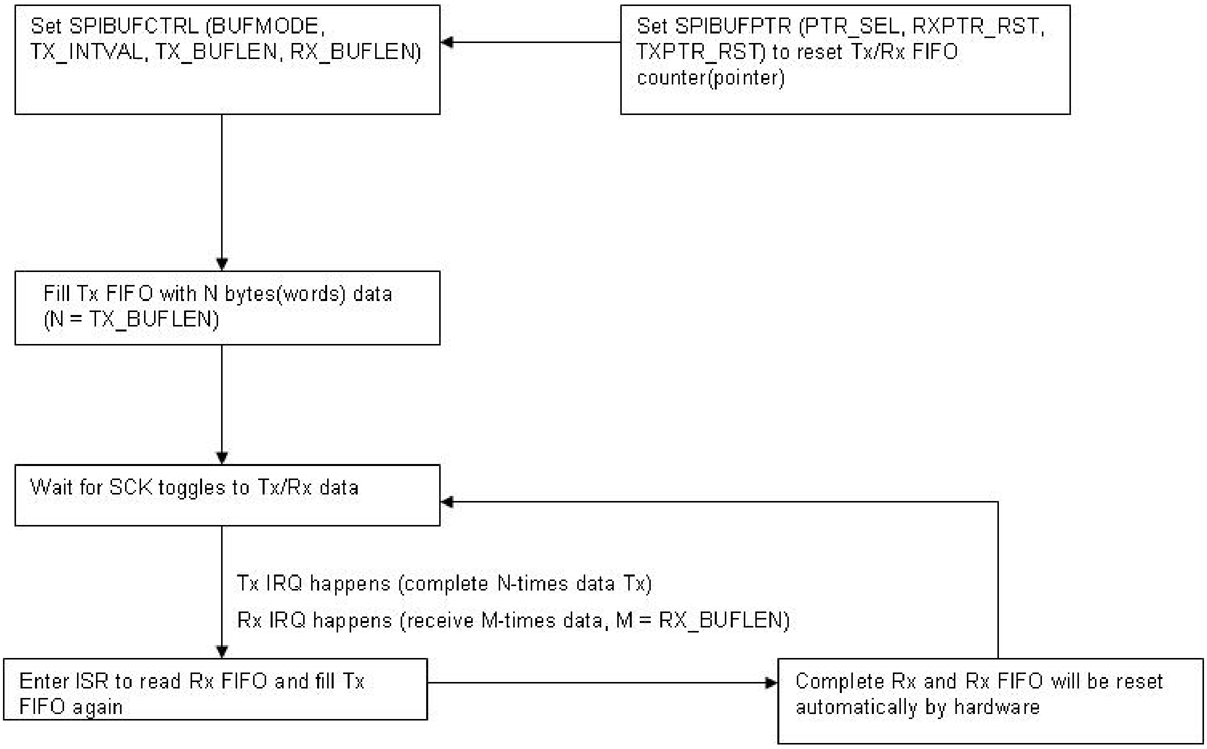

Set SPIBUFCTRL (BUFMODE, TX_INTVAL, TX_BUFLEN, RX_BUFLEN)
Set SPIBUFPTR (PTR_SEL, RXPTR_RST, TXPTR_RST) to reset Tx/Rx FIFO counter(pointer)
Fill Tx FIFO with N bytes(words) data (N = TX_BUFLEN)
Wait for SCK toggles to Tx/Rx data
Tx IRQ happens (complete N-times data Tx)
Rx IRQ happens (receive M-times data, M = RX_BUFLEN)
Enter ISR to read Rx FIFO and fill Tx FIFO again
Complete Rx and Rx FIFO will be reset automatically by hardware

Figure 7.15.10 SPI Buffer Mode Flowchart

In 8-bit mode, the data format in Tx/Rx FIFO is shown. 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.

7.16.7 SPI Register

SPICON1: SPI Control Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 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 = 0XA1; SFR Page = 0

Register SPICON1 is a general control register for SPI functionality.

Bit 7 (SPIEN): SPI Enable bit

0 : Disable SPI function

1 : Enable SPI function

SPIEN Bit =1 : Initialize state machine, release pin state, (register unchanged).

Bit 6 (CPHA): Clock Phase

0 : SCK toggle starts at the middle of first data bit.

1 : SCK toggle starts at the beginning of first data bit.

Bit 5 (CPOL): Clock polarity, the clock polarity and the clock phase should be identical for the Master and Slave devices involved in the communication link.

0 : SCK default Low

1 : SCK default High

(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 (MSTR): Master/Slave mode

0 : SPI is in master mode.

1 : SPI is in slave mode.

Bits 3 ~ 1 (SPR): SPI clock rate selection = system clock(Fsys) / Divisor. 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 greater than eight times (>2X) the Master Serial Clock Frequency

000 : system clock /2

001 : system clock /4

010 : system clock /8

011 : system clock /16

100 : system clock /32

101 : system clock /64

110 : system clock /128

111 : system clock /2

SPICON2: SPI Control Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	-	-	-	-	DORD	SPIHDEN	SDOOD	SCKOD
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0XA2; SFR Page = 0

Register SPICON2 is a general control register for SPI functionality.

Bits 7~4 Reserved

RCF_SEL: Active RCF (slave Rx busy) cannot disable Sleep instruction.

Bit 3 (DORD): Data shift-out direction at transmission

0 : MSB first

1 : LSB first

Bit 2 (SPIHDEN): The signal controlled by this bit is also routed to the boundary of eDSPSoCTP platform as output. Chip integrator could use this output to increase the driving capability of the I/O pads.

0 : Disable high current output.

1 : Enable high current output.

Bit 1 (SDOOD): The signal controlled by this bit is also routed to the boundary of eDSPSoCTP platform as output. Chip integrator could use this output to enable or disable the open-drain attribute of the SDO pad (either MOSI or MISO).

Bit 0 (SCKOD): The signal controlled by this bit is also routed to the boundary of eDSPSoCTP platform as output. Chip integrator could use this output to enable or disable the open-drain attribute of the SCK pad.

SPITDBR: SPI Transmit Data Buffer Register

SFR Address = 0xA3; SFR Page = Page 0

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.

SPIRDBR: SPI Receive Data Buffer Register

SFR Address = 0XA4; SFR Page = Page 0

The Receive Data Buffer Register (SPIRDBR) is an 8-bit read-only (RO) register. At the end of a data transfer, the data in the shift register is loaded into SPIRDBR.

SPISR1: SPI Status Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	MB_START	RXF	TXE	-	SCK_SEL	DEGL_SEL	SDOC	-
Type	R/W	R	R	R	R/W	R/W	R/W	R
Reset	1	0	0	0	0	0	0	0

SFR Address = 0XA5; SFR Page = Page 0

Register SPISR1 is the status register for SPI functionality.

Bit 7 (MB_START): During buffer mode, this bit needs to be set at “1” in order to send out the TX buffer value and avoid slow access of the TX buffer into FIFO which could result in the interruption of the second batch of sent data as the first batch of sent data is still incompletely in the process.

Bit 6 (RXF): Rx FIFO Full status bit for buffer mode only (HW auto clear after SW read)

Bit 5 (TXE): Tx FIFO Empty status bit for buffer mode only (HW auto clear after SW read)

Bit 4: Reserved

Bit 3 (SCK_SEL): 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. Slave data input will also pass through de-glitch delay cell when this bit is set.

0 : Slave SCK/SDI doesn't pass through de-glitch delay cell.

1 : Slave SCK/SDI passes through de-glitch delay cell.

Bit 2 (DEGL_SEL): 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 (SDOC): 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

SPISR2: SPI Status Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	SPISF	WKEN	--	RCF	SRO	TXS	RBF	TCF
Type	R/W	R/W	R	R	R/W	R	R/W	R
Reset	0	1	0	0	0	0	0	0

SFR Address = 0XA6; SFR Page = Page 0

Register SPISR2 is the status register for SPI functionality.

Bit 7 (SPISF): SPI interrupt flag. Set by hardware, cleared by software

Bit 6 (WKEN): SPI Wake-up enable. If this bit is set and SPI is enabled, SPI can be waken-up from IDLE/ Power-Down modes by SCK toggling.To enable SPI wakeup function, user must set WKEN bit **and** SPIE bit (EIE1.4).(wake-up signal: SCK first.)

Bit 4 (RCF): Receiving progress flag (Slave mode only)

0 : The SPI slave is not busy for receiving.

1 : The SPI slave is busy for receiving the data, the MCU should not go to sleep when the flag is high.

Bit 3 (SRO): 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 and not read out by FW. Note that the previous byte of data will be destroyed while this condition occurs.

Bit 2 (TXS): SPITDBR status flag.

0 : SPITDBR is empty.

1 : SPITDBR is full.

Bit 1 (RBF): Read Buffer Full Flag. This bit is set by HW and cleared by SW or HW (cleared automatically after reading). So it will be cleared automatically after reading Rx buffer (SPIRDBR) by CPU or ICE.

0 : Receiving is not completed, and SPIRDBR has not fully changed.

1 : Receiving is completed, and SPIRDBR has fully changed.

Bit 0 (TCF): Transmitting progress flag. This bit is both set and cleared by HW. 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.

0 : Tx transfer is not proceeding.

1 : Tx transfer is proceeding.

SPITX: SPI Transmit Data Buffer Control Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	TX_INTVAL	TXBLEN6	TXBLEN5	TXBLEN4	TXBLEN3	TXBLEN2	TXBLEN1	TXBLEN0
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 = 0xA7; SFR Page = Page 0

Register SPITX is the Transmit Data Buffer control register for buffer mode.

Bit 7 (TX_INTVAL): The system cycle interval between two master Tx transaction. Its system cycle count will be the value multiplied by 4. The interval is defined from the last serial SCK toggling of current transfer to the first SCK toggling of the next transfer.

0 : No system cycles between two Tx transfer (continuous)

1 : 4 system cycles between two Tx transfer.

Bits 6 ~ 0 (TXBLEN): Tx buffer length in buffer mode. The length is at most 64 bytes for the buffer mode FIFO. Can be set from 1 to 64 in decimal.

SPIRX: SPI Receive Data Buffer Control Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	BUFMODE	RXBLEN5	RXBLEN4	RXBLEN1	RXBLEN3	RXBLEN2	RXBLEN1	RXBLEN0
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 = 0x9A; SFR Page = Page 0

Register SPIRX is the Receive Data Buffer control register for buffer mode.

Bit 7 (BUFMODE): SPI Buffer Mode Enable

0 : Non-buffer mode(single mode)

1 : Buffer mode

Bits 6 ~ 0 (RXBLEN): Rx buffer length in buffer mode. The length is at most 8 bytes for the buffer mode FIFO. Can be set from 1 to 8 in decimal.

SPIBUFPTR1: SPI Read FIFO Pointer Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PTR_SEL	RP6	RP5	RP4	RP3	RP2	RP1	RP0
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 = Page 0

Register SPIBUFPTR1 is the FIFO counter (pointer) register for buffer mode.

Bit 7 (PTR_SEL): FIFO Counter (Pointer) selection for counter (pointer) register.

0 : Tx Counter (Pointer) can be read in counter(pointer) register.

1 : Rx Counter (Pointer) can be read in counter(pointer) register.

Bits 6~ 0 (RP): 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: SPI Write FIFO Pointer Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	PTR_RST	-	-	-	WP3	WP2	WP1	WP0
Type	R/W	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x9C; SFR Page = Page 0

Register SPIBUFPTR2 is the FIFO counter (pointer) register for buffer mode.

Bit 7 (PTR_RST): Tx/Rx FIFO counter (pointer) reset.

0 : Not reset

1 : Reset (Hardware auto clear)

Bits 3 ~ 0 (WP): 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.

7.17 EEPROM

2K bytes of EEPROM are located in the last flash 2K byte space. The stored data of EEPROM are not erased when the power is off and can be read and erase by firmware. A control register, EEPCTRL controls the EEPROM, that is, to read, write, erase the data from EEPROM. Writing a command into this register will execute an action to the EEPROM.

The command value is defined in the following tale. Note that there is an execution time laps for each command. Before writing the next command into the control register, allow enough time for the EEPROM to finish the previous command.

EEPCTRL: EEPROM Control Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	-	-	-	-	EEPROM_ FINISH	EEPROM_ ER	EEPROM_ RD	EEPROM_ WR
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x95; SFR Page = Page 1

Register EEPCTRL is control the EEPROM write/read/erase.

Bit 7 ~ 4 : Reserved.

Bits 3 (EEPROM_FINISH): EERPOM Finish, Set by H/W, Clear by F/W.

0 : Busy

1 : Finish

Bits 2 (EEPROM_ER): EEPROM Erase (1 Area : 128 Bytes).

1 : Erase

Bits 1 (EEPROM_RD): EEPROM Read.

1 : Read

Bits 0 (EEPROM_WR): EEPROM Write.

1 : Write

EEPAREASEL: EEPROM Data Area Select Register

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	-	-	-	-	EEPAREA SEL3	EEPAREA SEL2	EEPAREA SEL1	EEPAREA SEL0
Type	R	R	R	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR Address = 0x96; SFR Page = Page 1

Register EEPAREASEL is select the EEPROM data area.

Bit 7 ~ 4 : Reserved.

Bits 3 ~ 0 (EEPAREASEL): Select EEPROM data area (0 ~ 15).

EEPROM Data Buffer :

External RAM last 128 bytes match to area (byte0 ~ 127).

****Note : EEPROM use flash last 2K bytes space.***

EEPROM Write/Read/Erase Time :

System Clock	Read TIME(ms)
256KHz	**0.53**
32KHz	**4.25**
4KHz	**34.00**
500Hz	**272.00**

7.18 On-Chip Debug Support (OCDS)

EM85F684A/EM85F682A devices include an on-chip 2-Wire debug interface to allow Flash programming and simple debugging functions when used with ELAN IDE. The OCDS just need total of four pins and the pin definition is shown in Table 7.18.

OCD Debug Note: *To enter debug mode P81 (SCL_2W) and P82 (SDA_2W) while the program is in progress, the dummy should be set to an input floating state.*

OCDS Pins	Function
POWER	Power Supply
2W_SCL	Serial Clock
2W_SDA	Serial Data
GND	Ground

Table 7-18   OCDS Pin Definition

7.18.1 Limitations of On-Chip Debug

During the debugging process, the user must take care of the 2W_CLK and 2W_SDA pins for data and clock communications purposes to ensure that no other components are connected to these two pins. A circuit components restriction is shown in Figure 7.18.1.

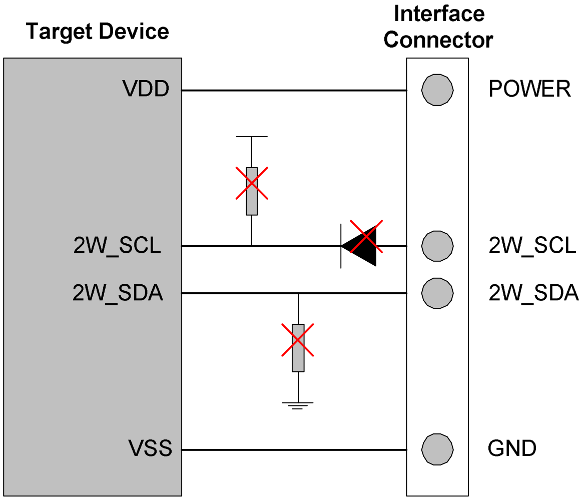


Figure 7.18.1 Diagram showing Circuit Components Restriction

The following are some guidelines to follow to ensure efficient debugging:

1. Do not use pull-high and pull-down on the 2W_SCL/2W_SDA pin.

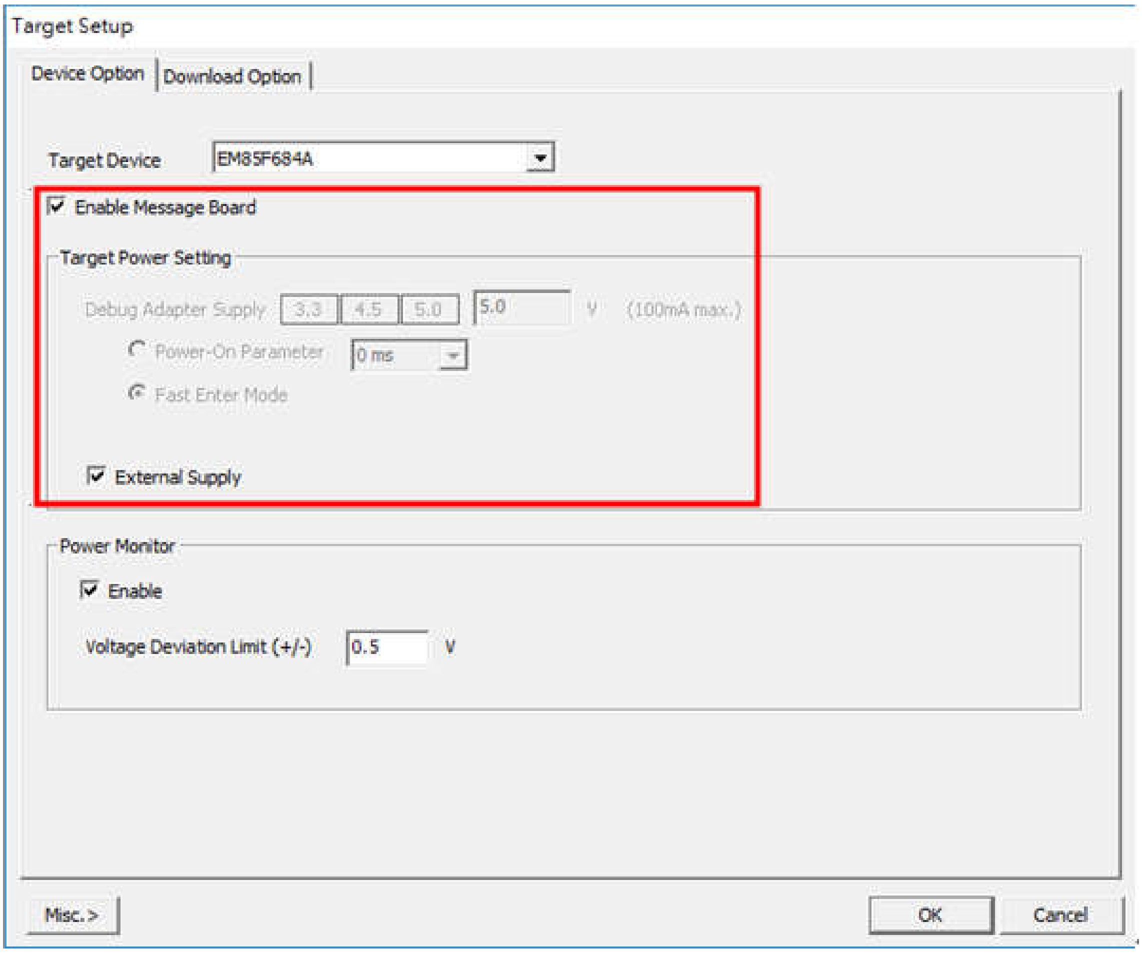
2. Do not use capacitors on the 2W_SCL/2W_SDA pin.

3. Do not use diodes on the 2W_SCL/2W_SDA pin.

During entering into OCD mode process, user must ensure a signal clear on the 2W_CLK and 2W_SDA pins. If into OCD mode is completed, the pin-shared function of 2W_CLK/2W_SDA is invalid. For more detailed OCDS information, refer to the 8051 OCDS User Guide.

UBRG (USB Bridge) Application Note:

1. **Using UBRG power as the device power source: UBRG Pin # 4 (power) must be connected to the device power pin (VDD50) and only 5V I/O can be used at this time. Meanwhile, Pin 81/82 must be in input state.**
2. **Using an external power (ex: USB Bus): UBRG Pin # 4 (power) must be connected to VDDIOH application circuits and only 5V/3.3V I/O can be used at this time. Meanwhile, Pin 81/82 must be in input state.**
3. **If only using UBRG to Download, It is recommended to pull out UBRG connect pin after loading the Code to MCU.**
4. **When changing the power source, user must also change the device power configuration.**



Target Device: EM85F684A

☑ Enable Message Board

Target Power Setting

Debug Adapter Supply 3.3 4.5 5.0 5.0 V (100mA max.)

○ Power-On Parameter 0 ms

● Fast Enter Mode

☑ External Supply

Power Monitor

☑ Enable

Voltage Deviation Limit (+/-) 0.5 V

7.19 Code Option

Information Page 0:

Code Option 0								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	-	PLL24M	ENWDTB	LOWFR1	LOWFR0	FREQ1	FREQ0	HSOSC
Default	1	1	0	1	1	1	1	1

Bit 6: PLL24M: PLL 24MHz enable bit(HSOSC must select 0 & FREQ1/0 must select 00/01/10)

1: Disable

0: Enable

*PS : PLL24M = Enable, clock source from XTAL 8/16MHz, so must select FREQ1,0 option.

Bit 5: ENWDTB: WDT enable bit

1: Disable

0: Enable

Bits 4~3: LOWFR1 LOWFR0 : LOW FREQ. switch Low frequency switch.

00: Frequency : 500 Hz

01: Frequency : 4 kHz

10: Frequency : 32 kHz

11: Frequency : 256 kHz

Bits 2~1: High Speed Clock Select

HSOSC	PLL24M	FREQ1~0	System clock
1 (IRC)	X	11	24MHz
		10	16MHz
		01	8MHz
		00	8MHz
0 (XTAL)	1	11	24MHz
		10	16MHz
		01	8MHz
		00	8MHz
	0	11	24MHz
		10	24MHz
		01	24MHz
		00	24MHz

Bit 0 (HSOSC): High Speed Clock Source Select

1: IRC mode

0: XTAL mode

Code Option 3								
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	-	-	-	EEPRST SEL	EEPROM EN	HWRSTS EL1	HWRST SEL0	HWRST EN
Default	1	1	1	1	1	1	1	1

Bit 4: EEPRSTSEL: EEPROM H/W Reset Select.

1: Latch Code Option

0: No Latch Code Option

Bit 3: EEPROMEN: EEPROM Enable

1: Disable

0: Enable

*PS : EEPROM use Flash Last 2K Bytes, When EEPROM Enable, Flash ROM size become to 30KB.

Bit 2~1: HWRST : Hardware Reset Pin Select (Low Active).

11: P96 (default).

10: P90.

01: P66.

00: P65.

Bit 0: HWRSTEN : Hardware Reset Pin Enable

1: Disable

0: Enable

*PS : Hardware reset pin must connect pull-high resistor.

7.20 CPU Register Descriptions

The arithmetic section of the processor performs extensive data manipulation and is comprised of the 8-bit arithmetic logic unit (ALU), an ACC register, B register (The B register is used during multiply and divide operations. In other cases may be used as normal SFR.) and PSW0 register (The PSW0 contains several bits that reflect the current state of the CPU). In addition, the EM85F684A/EM85F682A has Dual data pointer registers are implemented to speed up data block copying.

The EM85F684A/EM85F682A provides two 16-bit data pointers: DPTR0 and DPTR1, and a single bit called DPS (PCON.3) that allows the program code to switch between them. The data pointers are used by several 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

PCON: Power Control

Bit	7	6	5	4	3	2	1	0
Name	SMOD0	UARTEN	-	-	DPS	-	PD	IDLE
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 = 0x87; SFR Page = All Pages

Bit 7: UART double baud rate bit.

Bit 6: UART enable bit.

Bits 5, 4, 2~1: Reserved

Bit 3: Data Pointer Select

This bit is used to switch between DPTR0 and DPTR1.

0: Select DPTR0.

1: Select DPTR1.

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 into Power-Down Mode).

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

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

PSW0: 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.

Bits 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 use 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

Bits 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.

7.21 Instruction Set

The EM85F684A/EM85F682A is fully compatible to the standard 8051 microcontroller, maintains all instruction mnemonics and binary compatibility. The EM85F684A/EM85F682A incorporates some great architectural enhancements; allow the EM85F684A/EM85F682A CPU execution of instructions with high performance.

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	2	3
DEC @Ri	Decrement indirect RAM	1	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

Mnemonic	Description	Bytes	Clock Cycles
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
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

Mnemonic	Description	Bytes	Clock Cycles
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
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 @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

Mnemonic	Description	Bytes	Clock Cycles
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	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 7.21 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 program 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 64kB program memory space.

8 Absolute Maximum Ratings

Items	Rating		
Temperature under bias	-0°C	to	70°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	3V	to	5.5V

9 DC Electrical Characteristics

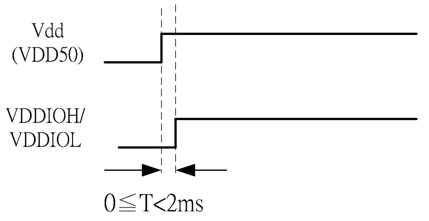
(Ta=25°C, VDD=5.0V±5%, VSS=0V)

Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
VDD	Power Supply Voltage		3.0	--	5.5	V
VDDIOH	Power Supply Voltage on IO (P5/6/P70~73,76,77/P8,P94~97 I/O power input)		3.0	--	5.5	V
VDDIOL	Power supply input pins for the I/O power of P92, P93		1.8	--	3.6	V
IRCE1 (ILRC512kHz)	Internal RC oscillator error per stage		-	±4	-	%
IRCE2 (IHRCkHz)	Internal RC oscillator error per stage		-	±1	-	%
IRC1	IRC:VDD to 5V	FREQ1 FREQ0=11	-	24	-	MHz
IRC2	IRC:VDD to 5V	FREQ1 FREQ0=10	-	16	-	MHz
IRC3	IRC:VDD to 5V	FREQ1 FREQ0=01	-	8	-	MHz
IIL	Input Leakage Current for input pins	VIN = VDD, VSS	-1	0	1	µA
VIH1	Input High Voltage (Schmitt trigger)	Ports 5, 6, 7, 8,9	0.7Vdd			V
VIL1	Input Low Voltage (Schmitt trigger)	Ports 5, 6, 7, 8,9			0.3Vdd	V
IOH1	Output High Voltage (Ports 5, 6, 7, 8,9)	VOH = 0.9VDD_IO VDD_IO=5V VDD_IO=3.3V	-	7 5	-	mA
IOH2	Output High Voltage (high drive) (Ports 5, 6, 7, 8,9)	VOH = 0.9VDD_IO VDD_IO=5V VDD_IO=3.3V	-	16 10	-	mA
IOL1	Output Low Voltage (Ports 5, 6, 7, 8,9)	VOL=0.1VDD_IO VDD_IO=5V VDD_IO=3.3V	-	22 10	-	mA
IOL2	Output Low Voltage (High sink) (Ports 5, 6, 7, 8,9)	VOL=0.1VDD_IO VDD_IO=5V VDD_IO=3.3V	-	30 19	-	mA
IPH	Pull-high current	Pull-high active, input pin at VSS (62.5k) VDD_IO=5V VDD_IO=3.3V	-	80 36	-	µA
ISB1	Power down current	WDT disabled	-	100	-	µA
ISB2	Power down current	WDT enabled	-	105	-	µA

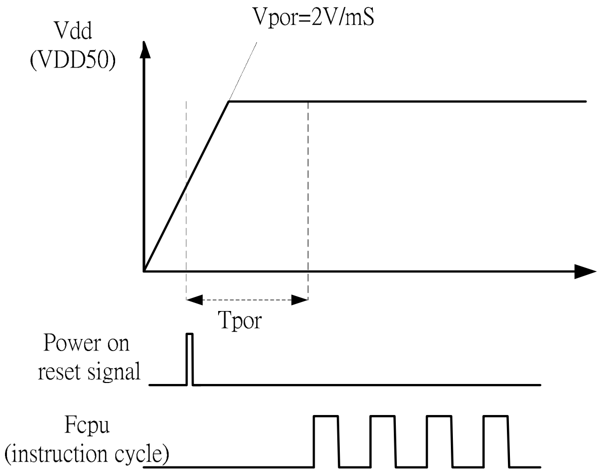
Symbol	Parameter	Condition	Min.	Typ.	Max.	Unit
ICC1	Operating supply current (Green mode)	Fs=256kHz WDT enable Output pin floating	—	500	—	µA
ICC2	Operating supply current (Green mode)	Fs=256kHz WDT disabled Output pin floating	—	500	—	µA
ICC3	Operating supply current (Normal mode)	Fosc=24 MHz (Crystal type), output pin floating, WDT enabled	—	10	—	mA
ICC4	Operating supply current (Normal mode)	Fosc=24 MHz (IRC), Output pin floating, WDT enabled	—	10	—	mA

10 Power Sequence

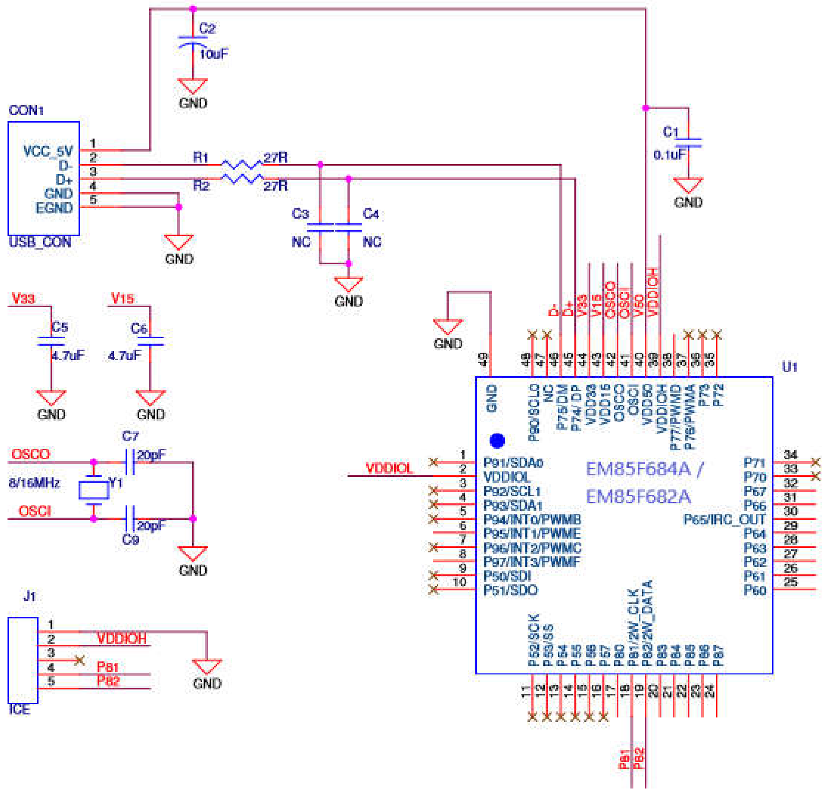
Symbol	Description	Typ.	Unit
Vpor	Vdd (VDD50) rise rate to ensure power-on reset	2	V/ms
Tpor	Power on warm up time	2~16	ms



PS. VDDIOH / VDDIOL power up time can’ t be earlier than VDD50 power up time

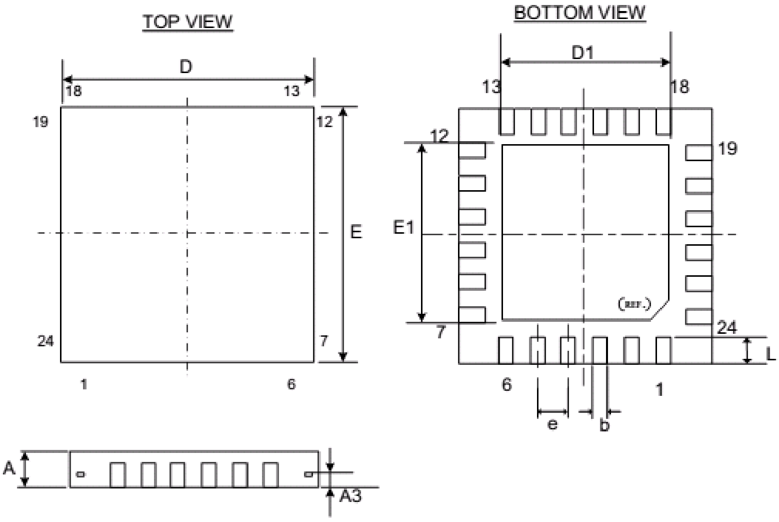


11 Application Circuit



註：
* VDDIOH 與 VDDIOL 請自行繪製電壓源輸入
* 使用Elane UBRIDGE做模擬或downlaod code 時， Device電源需選擇由外部電源供應
* When using Elan USB IAP,P80 is boot pin. Low level voltage after power on, entry boot code mode.

12 Package Information



	EMC		
Symbol	Min.	Nor.	Max.
A	0.70	0.75	0.80
A3	0.203 REF		
b	0.18	—	0.3
D	4.00 BSC		
D1	2.40	—	2.8
E	4.00 BSC		
E1	2.40	—	2.8
e	0.5BSC		
L	0.30	—	0.50

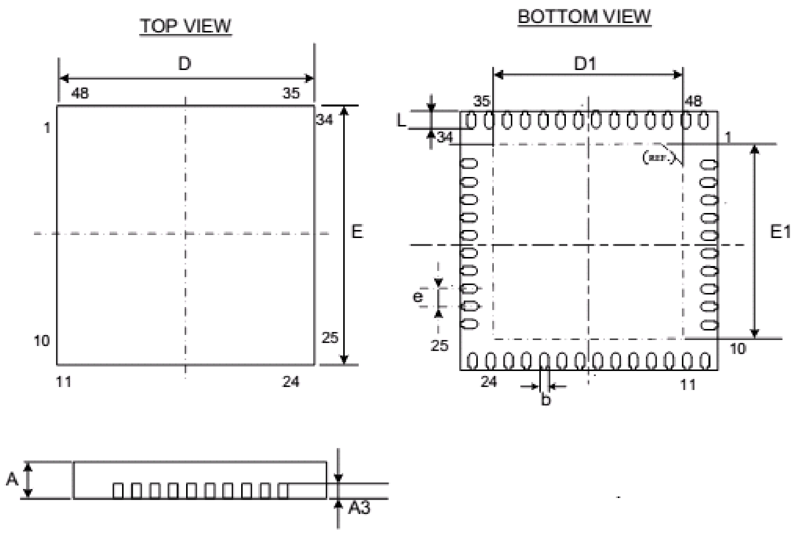
TITLE
QFN 24L (4 *4 0.8 MM)
PACKAGE OUTLINE DIMENSION

File :
QFN 24L

Edtion: B

Unit : mm
Scale: Free
Material:
Sheet:1 of 1

*Figure 8-1 QFN24 (4*4*0.8mm) Package outline Dimension*



Symbol	Min.	Nor.	Max.
A	0.50	0.55	0.60
A3	0.152 REF		
b	0.13	———	0.23
D	5.00 BSC		
D1	3.60	———	3.80
E	5.00 BSC		
E1	3.60	———	3.80
e	0.35 BSC		
L	0.35	0.40	0.45

TITLE: QFN 48L (5*5*0.6 MM) PACKAGE OUTLINE DIMENSION	
File : QFN 48 L	Edtion: A
	Unit : mm
	Scale: Free
	Material:
	Sheet:1 of 1

Figure 8-4 QFN48 Package outline Dimension