Super High-speed 1T 8051-based Flash MCU, 1 Kbytes SRAM, 16 Kbytes Flash, 128 bytes independent EEPROM, 12-bit ADC, Analog Comparator, LCD/LED Driver, 12-bit PWM, 3 Timer/Counters, MDU, UART, SSI, CheckSum module

1 General Description

The SC92F7447B/7446B/7445B (hereinafter referred to as the SC92F744XB) is a kind of enhanced super high-speed 1T 8051-based industrial Flash Microcontroller unit (MCU), in which the instruction system is completely compatible with standard 8051 product series.

The SC92F744XB contains 16K bytes Flash ROM, 1K bytes SRAM, 128 bytes EEPROM. The SC92F744XB has up to 46 General-purpose I/Os (GPIO), 16 IO external interrupters, three 16-bit timers, 17-channel 12-bit high-precision ADC, 1 analog comparator, 8-channel independent 12-bit PWM, IO ports driven hierarchical control, 1 16*16-bit hardware Multiplier-Divide Unit (MDU). The system clock source is internal ±1% high-precision 16/8/4/1.33MHz high-frequency oscillator. The low-frequency clock source is selectable between the ±4% precision 128 kHz oscillator and external 32.768 kHz crystal oscillator, two SSI communication interface and other resources. To improve the reliability and simplify the circuit design, the SC92F744XB also built in with 4-level optional LVR voltage, 2.4V ADC reference voltage, low-power WDT and other high-reliability circuits.

The SC92F744XB features excellent anti-interference performance, which make it possible to be widely applied to industrial control system, such as Internet of Things, intelligent home appliances, home automation, wireless communication, gaming peripherals and consumer applications.

2 features

Operating Voltage: 2.4V ~ 5.5V

Operating Temperature: -40 ~ 85℃

Package:

SC92F7447B (LQFP48)

SC92F7446B (LQFP44)

SC92F7445B (LQFP32)

Core: Ultra-speed 1T 8051

Flash ROM: 16 Kbytes Flash ROM (MOVC prohibited addressing 0000H~00FFH 256 bytes)

IAP: Code option into 0K, 0.5K, 1K or 16K

EEPROM: 128 bytes EEPROM can be rewritten 100,000 times. The data written-in has more than 10-year preservation life.

SRAM: Internal 256 bytes + external 768 bytes + PWM&LCD RAM 44 bytes

System Clock (f_{SYS}):

- Built-in 16 MHz high-speed RC oscillator (f_{HRC}):
- IC system clock can be set by the programmer as
  - 16 MHz @2.9~5.5V
  - 8/4/1.33 MHz@2.4~5.5V
- Frequency Error: Suitable for 3.0V ~ 5.5V and -20 ~ 85℃ application environment, no more than ±1%

Built-in Low-speed Crystal Oscillator Circuit:

- Available to externally connect 32K oscillator as Base Timer clock source, able to wake up STOP

Built-in 128 kHz LRC Low-speed Oscillator:

- Available to act as clock source of Base Timer (BTM) and WDT, which can wake up STOP
- Frequency Error: 4.0V ~ 5.5V and -20 ~ 85℃ application environment, no more than ±4% of frequency error after register correction

Low-voltage Reset (LVR):

- 4-level LVR voltage options: 4.3V, 3.7V, 2.9V, 2.3V
- The default is the Code Option value selected by the user

Flash Programming and Emulation:

- 2-wire JTAG programming and emulation interface
- JTAG mode and Normal mode can be set through code option

Interruption (INT):

- Totally 12 interrupt sources, including Timer0, Timer1, Timer2, INT0~2, ADC, PWM, UART, SSI, Base Timer, and CMP
- Three external interrupt vectors shared by 16 external interrupt I/Os, which can be defined in rising-edge, falling-edge or double-edge trigger mode.
- Two-level interrupt priority capability

Digital Peripheral:

- Up to 46 two-way independently controllable I/O interfaces, able to configure pull-high resistor independently
- P0~P3 (P3.0/1/2/3) 4-level control drive capability
- All IOs equipped with sink current drive capability (50mA)
- 11-bit WDT, optional clock division ratio
- 3 standard 80C51 timers: Timer0, Timer1 and Timer 2
- Eight 12-bit PWM output channels with variable period and individual duty cycle
- 1 independent UART communication port
- 1 UART/SPI/IIC communication interface (SSI)
- 16 * 16-bit hardware Multiplier-Divide Unit (MDU)

LCD/LED Driver:

- LCD/LED driver, select one from two, shared with common registers and IO interface
- 8 X 24, 6 X 26, 5 X 27 or 4X 28 segment LED drive
- LED segment interface with 4-level control drive capability
- 8 X 24, 6 X 26, 5 X 27 or 4 X 28 segment LCD drive
- **SC92F7445B without LCD/LED Driver**

Analog Peripheral:

- 17-channel 12-bit ±2LSB ADC
  - 17-channel 12-bit ±2LSB ADC
- Built-in 2.4V reference voltage
  - 2 options for ADC reference voltage: VDD and internal 2.4V
  - Internal one-channel ADC, where VDD can be measured directly
  - ADC conversion completion interrupt
- 1 analog comparator
  - 4-channel input and 1-channel reference voltage input
  - 16-level optional comparison voltage
- **SC92F7445B without LCD/LED Driver**

Power Saving Mode:

- IDLE Mode: can be woken up by any interrupt.
- STOP Mode: can be woken up by INT0 ~ 2、and Base Timer.

Naming Rules for 92 Series Products

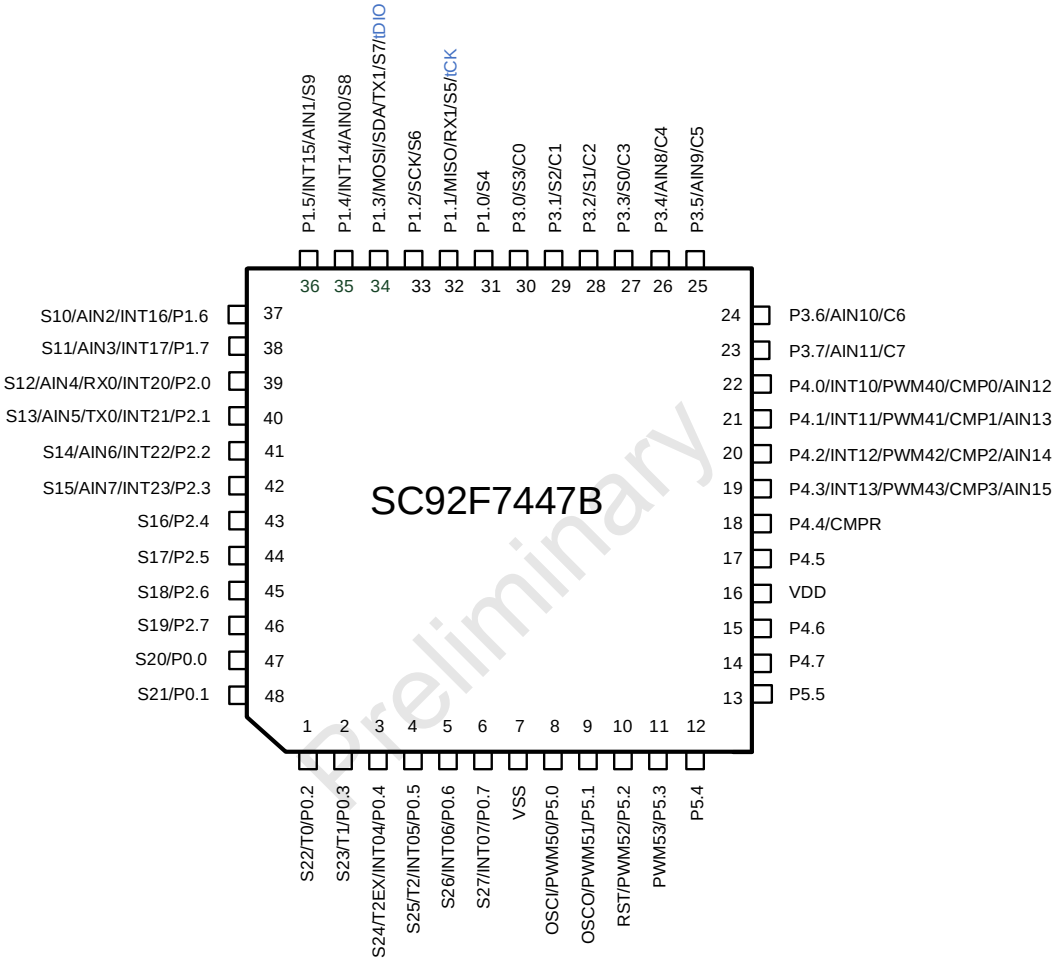
Name	SC	92	F	7	4	4	7	B	P	48	R
S/R	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪

S/R	**Meaning**
①	SinOne Chip abbreviation
②	Name of product series
③	Product Type (F: Flash MCU)
④	Serial Number: 7: GP Series, 8: TK series
⑤	ROM Size: 1 for 2K, 2 for 4K, 3 for 8K, 4 for 16K and 5 for 32K…
⑥	Subseries Number.: 0 ~ 9, A ~ Z
⑦	Number of Pins: 0: 8pin, 1: 16pin, 2: 20pin, 3: 28pin, 5: 32pin, 6: 44pin, 7: 48pin, 8: 64pin, 9: 100pin
⑧	Version Number: (default, B, C, D)
⑨	Package Type: (D: DIP; M: SOP; X: TSSOP; F: QFP; P: LQFP; Q: QFN; K: SKDIP)
⑩	Number of Pins.
⑪	Packaging Mode: (U: Tube; R: Tray; T: Reel)

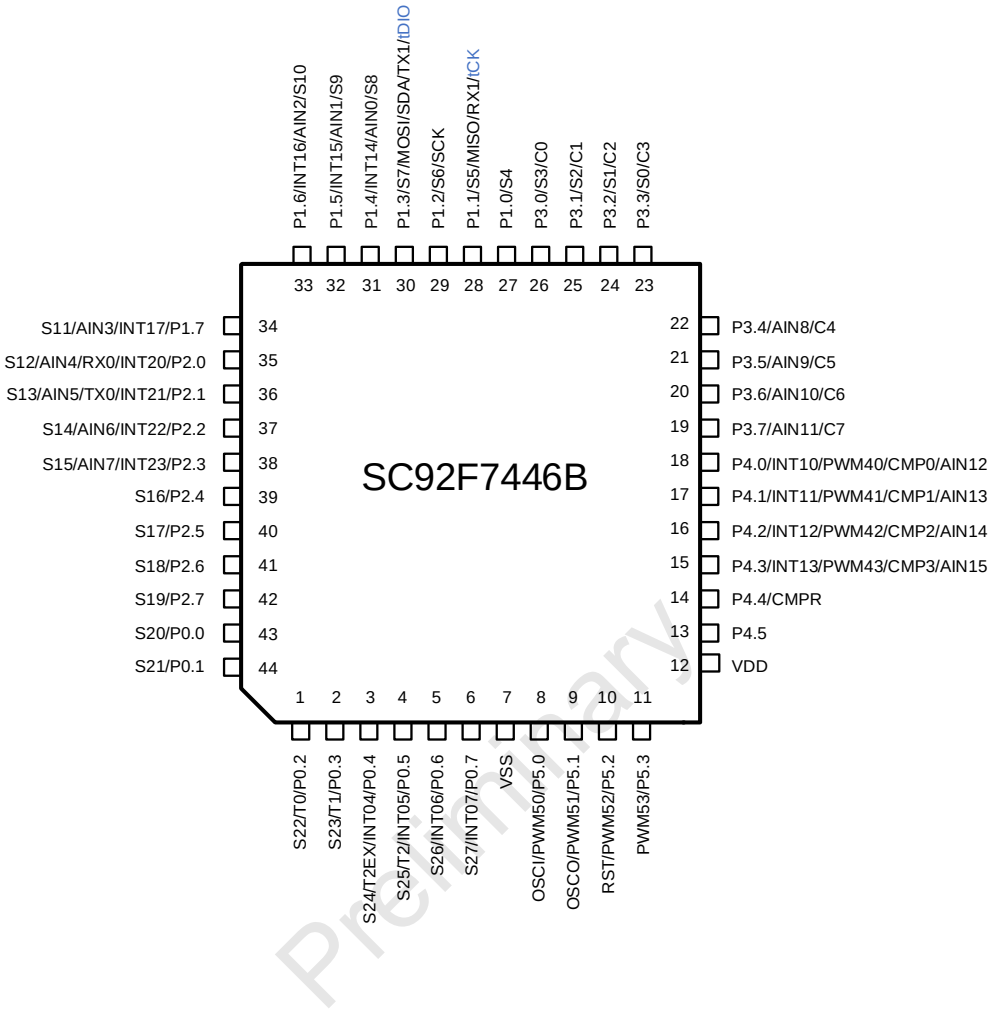
Contents

3 Pin Description

3.1 LQFP48/LQFP44 Pin Configuration



The SC92F7447B Pin Diagram



The SC92F7446B Pin Diagram

3.1.1 LQFP48/LQFP44 Pin Definition

Pin number		Pin Name	Type	Description
48pin	44pin			
1	1	**P0.2/T0/S22**	I/O	P0.2: GPIO P0.2 T0: Timer/Counter 0 External Input S22: LCD/LED SEG22
2	2	**P0.3/T1/S23**	I/O	P0.3: GPIO P0.3

				T1: Timer/Counter 1 External Input S23: LCD/LED SEG23
3	**3**	**P0.4/INT04/T2EX/S24**	I/O	P0.4: GPIO P0.4 INT04: Input 4 of external interrupt 0 T2EX: External Signal Input Capture for Timer2 S24: LCD/LED SEG24
4	**4**	**P0.5/INT05/T2/S25**	I/O	P0.5: GPIO P0.5 INT05: Input 5 of external interrupt 0 T2: Timer/Counter 2 External Input S25: LCD/LED SEG25
5	**5**	**P0.6/INT06/S26**	I/O	P0.6: GPIO P0.6 INT06: Input 6 of external interrupt 0 S26: LCD/LED SEG26
6	**6**	**P0.7/INT07/S27**	I/O	P0.7: GPIO P0.7 INT07: Input 7 of external interrupt 0 S27: LCD/LED SEG27
7	**7**	**VSS**	Power	Ground
8	**8**	**P5.0/PWM50/OSCI**	I/O	P5.0: GPIO P5.0 PWM50: PWM50 output OSCI: External 32KHz crystal oscillator input
9	**9**	**P5.1/PWM51/OSCO**	I/O	P5.1: GPIO P5.1 PWM51: PWM51 output OSCO: External 32KHz crystal oscillator

				output
10	**10**	**P5.2/PWM52/RST**	I/O	P5.2: GPIO P5.2 PWM52: PWM52 output RST: Reset Pin
11	**11**	**P5.3/PWM53**	I/O	P5.3: GPIO P5.3 PWM53: PWM53 output
12	**-**	**P5.4**	I/O	P5.4: GPIO P5.4
13	**-**	**P5.5**	I/O	P5.5: GPIO P5.5
14	**-**	**P4.7**	I/O	P4.7: GPIO P4.7
15	**-**	**P4.6**	I/O	P4.6: GPIO P4.6
16	**12**	**VDD**	Power	Power
17	**13**	**P4.5**	I/O	P4.5: GPIO P4.5
18	**14**	**P4.4/CMPR**	I/O	P4.4: GPIO P4.4 CMPR: Comparator Reference Voltage Input
19	**15**	**P4.3/INT13/PWM43/CMP3/AIN15**	I/O	P4.3: GPIO P4.3 INT13: Input 3 of external interrupt 1 PWM43: PWM43 output CMP3: Analog Comparator Input Channel 3 AIN15: ADC Input Channel 15
20	**16**	**P4.2/INT12/PWM42/CMP2/AIN14**	I/O	P4.2: GPIO P4.2

				INT12: Input 2 of external interrupt 1 PWM42: PWM42 output CMP2: Analog Comparator Input Channel 2 AIN14: ADC Input Channel 14
21	**17**	**P4.1/INT11/PWM41/CMP1/AIN13**	I/O	P4.1: GPIO P4.1 INT11: Input 1 of external interrupt 1 PWM41: PWM41 output CMP1: Analog Comparator Input Channel 1 AIN13: ADC Input Channel 13
22	**18**	**P4.0/INT10/PWM40/CMP0/AIN12**	I/O	P4.0: GPIO P4.0 INT10: Input 0 of external interrupt 1 PWM40: PWM40 output CMP0: Analog Comparator Input Channel 0 AIN12: ADC Input Channel 12
23	**19**	**P3.7/AIN11/C7**	I/O	P3.7: GPIO P3.7 AIN11: ADC Input Channel 11 C7: LCD/LED common drive output 7
24	**20**	**P3.6/AIN10/C6**	I/O	P3.6: GPIO P3.6 AIN10: ADC Input Channel 10 C6: LCD/LED common drive output 6
25	**21**	**P3.5/AIN9/C5**	I/O	P3.5: GPIO P3.5 AIN9: ADC Input Channel 9 C5: LCD/LED common drive output 5

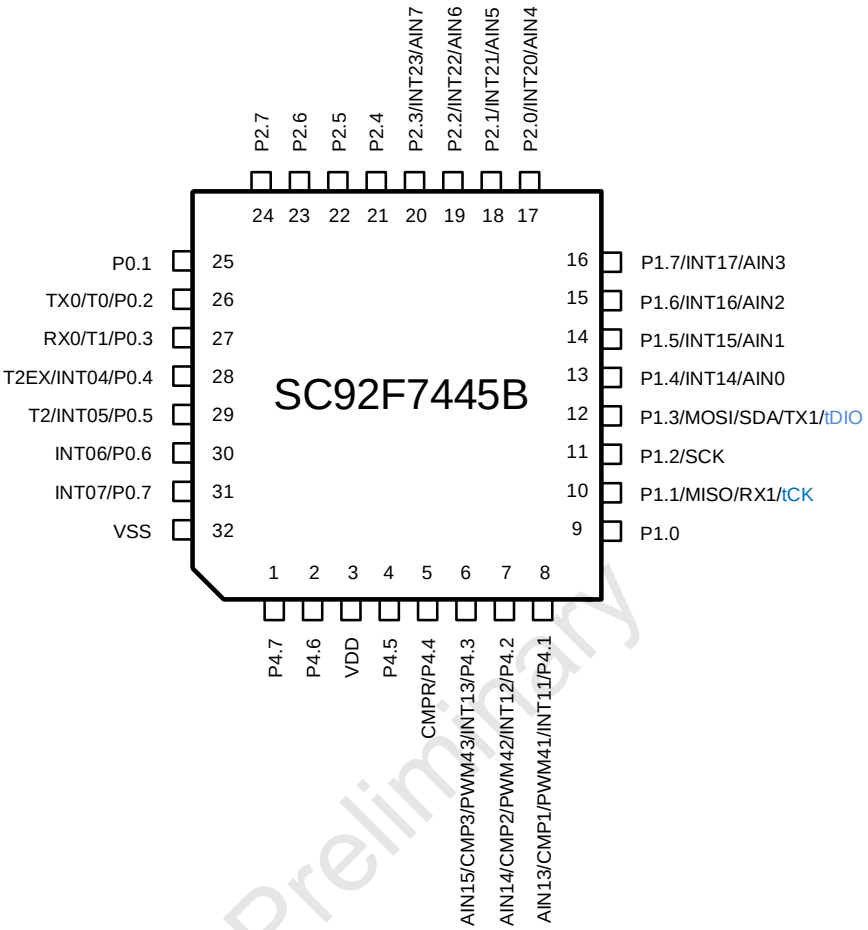
26	**22**	**P3.4/AIN8/C4**	I/O	P3.4: GPIO P3.4 AIN8: ADC Input Channel 8 C4: LCD/LED common drive output 4
27	**23**	**P3.3/S0/C3**	I/O	P3.3: GPIO P3.3 S0: LCD/LED SEG0 C3: LCD/LED common drive output 3
28	**24**	**P3.2/S1/C2**	I/O	P3.2: GPIO P3.2 S1: LCD/LED SEG1 C2: LCD/LED common drive output 2
29	**25**	**P3.1/S2/C1**	I/O	P3.1: GPIO P3.1 S2: LCD/LED SEG2 C1: LCD/LED common drive output 1
30	**26**	**P3.0/S3/C0**	I/O	P3.0: GPIO P3.0 S3: LCD/LED SEG3 C0: LCD/LED common drive output 0
31	**27**	**P1.0/S4**	I/O	P1.0: GPIO P1.0 S4: LCD/LED SEG4
32	**28**	**P1.1/MISO/RX1/S5/tCK**	I/O	P1.1: GPIO P1.1 MISO: SPI master-in/slave-out RX1: UART1 Receiver S5: LCD/LED SEG5 tCK: Programming and Emulation Clock Pin
33	**29**	**P1.2/SCK/S6**	I/O	P1.2: GPIO P1.2

				SCK: SCK for SPI and TWI S6: LCD/LED SEG6
34	**30**	**P1.3/MOSI/SDA/TX1/S7/tDIO**	I/O	P1.3: GPIO P1.3 MOSI: SPI master-out/slave-in SDA: SDA for TWI TX1: UART1 Transmitter S7: LCD/LED SEG7 tDIO: Programming and Emulation Data Pin
35	**31**	**P1.4/INT14/AIN0/S8**	I/O	P1.4: GPIO P1.4 INT14: Input 4 of external interrupt 1 AIN0: ADC Input Channel 0 S8: LCD/LED SEG8
36	**32**	**P1.5/INT15/AIN1/S9**	I/O	P1.5: GPIO P1.5 INT15: Input 5 of external interrupt 1 AIN1: ADC Input Channel 1 S9: LCD/LED SEG9
37	**33**	**P1.6/INT16/AIN2/S10**	I/O	P1.6: GPIO P1.6 INT16: Input 6 of external interrupt 1 AIN2: ADC Input Channel 2 S10: LCD/LED SEG10
38	**34**	**P1.7/INT17/AIN3/S11**	I/O	P1.7: GPIO P1.7 INT17: Input 7 of external interrupt 1 AIN3: ADC Input Channel 3 S11: LCD/LED SEG11

39	**35**	**P2.0/INT20/RX0/AIN4/S12**	I/O	P2.0: GPIO P2.0 INT20: Input 0 of external interrupt 2 RX0: UART0 Receiver AIN4: ADC Input Channel 4 S12: LCD/LED SEG12
40	**36**	**P2.1/INT21/TX0/AIN5/S13**	I/O	P2.1: GPIO P2.1 INT21: Input 1 of external interrupt 2 TX0: UART 0 Transmitter AIN5: ADC Input Channel 5 S13: LCD/LED SEG13
41	**37**	**P2.2/INT22/AIN6/S14**	I/O	P2.2: GPIO P2.2 INT22: Input 2 of external interrupt 2 AIN6: ADC Input Channel 6 S14: LCD/LED SEG14
42	**38**	**P2.3/INT23/AIN7/S15**	I/O	P2.3: GPIO P2.3 INT23: Input 3 of external interrupt 2 AIN7: ADC Input Channel 7 S15: LCD/LED SEG15
43	**39**	**P2.4/S16**	I/O	P2.4: GPIO P2.4 S16: LCD/LED SEG16
44	**40**	**P2.5/S17**	I/O	P2.5: GPIO P2.5 S17: LCD/LED SEG17
45	**41**	**P2.6/S18**	I/O	P2.6: GPIO P2.6 S18: LCD/LED SEG18

46	**42**	**P2.7/S19**	I/O	P2.7: GPIO P2.7 S19: LCD/LED SEG19
47	**43**	**P0.0/S20**	I/O	P0.0: GPIO P0.0 S20: LCD/LED SEG20
48	**44**	**P0.1/S21**	I/O	P0.1: GPIO P0.1 S21: LCD/LED SEG21

3.2 LQFP32 Pin Configuration (Without LCD/LED driver)



The SC92F7445B Pin Diagram

3.2.1 LQFP32 Pin Definition

Pin number	Pin Name	Type	Description
1	**P4.7**	I/O	P4.7: GPIO P4.7
2	**P4.6**	I/O	P4.6: GPIO P4.6
3	**VDD**	Power	Power

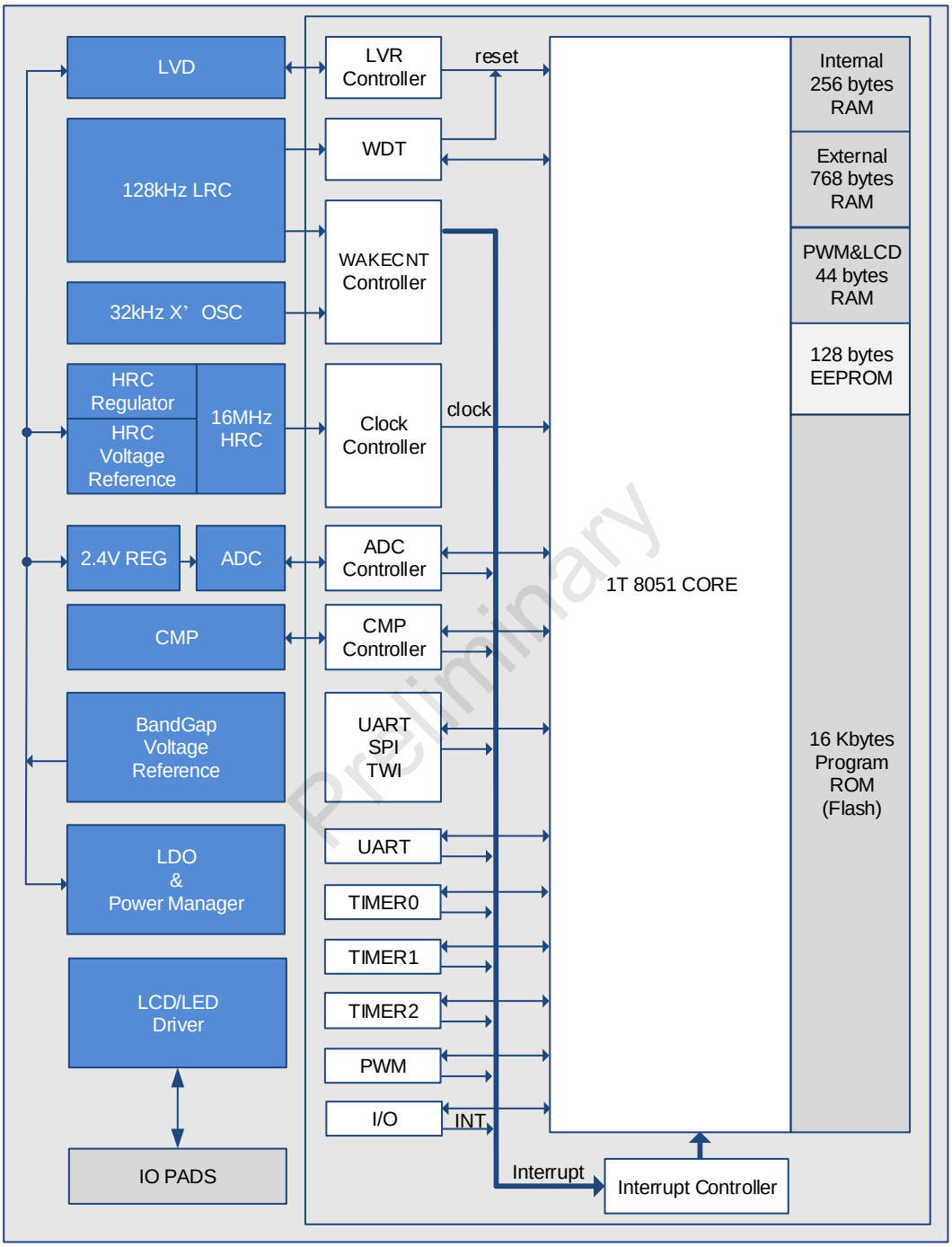
4	**P4.5**	I/O	P4.5: GPIO P4.5
5	**P4.4/CMPR**	I/O	P4.4: GPIO P4.4 CMPR: Comparator Reference Voltage Input
6	**P4.3/INT13/PWM43/CMP3/AIN15**	I/O	P4.3: GPIO P4.3 INT13: Input 3 of external interrupt 1 PWM43: PWM43 output CMP3: Analog Comparator Input Channel 3 AIN15: ADC Input Channel 15
7	**P4.2/INT12/PWM42/CMP2/AIN14**	I/O	P4.2: GPIO P4.2 INT12: Input 2 of external interrupt 1 PWM42: PWM42 output CMP2: Analog Comparator Input Channel 2 AIN14: ADC Input Channel 14
8	**P4.1/INT11/PWM41/CMP1/AIN13**	I/O	P4.1: GPIO P4.1 INT11: Input 1 of external interrupt 1 PWM41: PWM41 output CMP1: Analog Comparator Input Channel 1 AIN13: ADC Input Channel 13
9	**P1.0**	I/O	P1.0: GPIO P1.0
10	**P1.1/MISO/RX1/tCK**	I/O	P1.1: GPIO P1.1 MISO: SPI master-in/slave-out RX1: UART1 Receiver tCK: Programming and Emulation Clock Pin

11	**P1.2/SCK**	I/O	P1.2: GPIO P1.2 SCK: SCK for SPI and TWI
12	**P1.3/MOSI/SDA/TX1/tDIO**	I/O	P1.3: GPIO P1.3 MOSI: SPI master-out/slave-in SDA: SDA for TWI TX1: UART1 Transmitter tDIO: Programming and Emulation Data Pin
13	**P1.4/INT14/AIN0**	I/O	P1.4: GPIO P1.4 INT14: Input 4 of external interrupt 1 AIN0: ADC Input Channel 0
14	**P1.5/INT15/AIN1**	I/O	P1.5: GPIO P1.5 INT15: Input 5 of external interrupt 1 AIN1: ADC Input Channel 1
15	**P1.6/INT16/AIN2**	I/O	P1.6: GPIO P1.6 INT16: Input 6 of external interrupt 1 AIN2: ADC Input Channel 2
16	**P1.7/INT17/AIN3**	I/O	P1.7: GPIO P1.7 INT17: Input 7 of external interrupt 1 AIN3: ADC Input Channel 3
17	**P2.0/INT20/RX0/AIN4**	I/O	P2.0: GPIO P2.0 INT20: Input 0 of external interrupt 2 RX0: UART0 Receiver AIN4: ADC Input Channel 4
18	**P2.1/INT21/TX0/AIN5**	I/O	P2.1: GPIO P2.1

			INT21: Input 1 of external interrupt 2 TX0: UART 0 Transmitter AIN5: ADC Input Channel 5
19	**P2.2/INT22/AIN6**	I/O	P2.2: GPIO P2.2 INT22: Input 2 of external interrupt 2 AIN6: ADC Input Channel 6
20	**P2.3/INT23/AIN7**	I/O	P2.3: GPIO P2.3 INT23: Input 3 of external interrupt 2 AIN7: ADC Input Channel 7
21	**P2.4**	I/O	P2.4: GPIO P2.4
22	**P2.5**	I/O	P2.5: GPIO P2.5
23	**P2.6**	I/O	P2.6: GPIO P2.6
24	**P2.7**	I/O	P2.7: GPIO P2.7
25	**P0.1**	I/O	P0.1: GPIO P0.1
26	**P0.2/T0**	I/O	P0.2: GPIO P0.2 T0: Timer/Counter 0 External Input
27	**P0.3/T1**	I/O	P0.3: GPIO P0.3 T1: Timer/Counter 1 External Input
28	**P0.4/INT04/T2EX**	I/O	P0.4: GPIO P0.4 INT04: Input 4 of external interrupt 0 T2EX: External Signal Input Capture for Timer2
29	**P0.5/INT05/T2**	I/O	P0.5: GPIO P0.5

			INT05: Input 5 of external interrupt 0 T2: Timer/Counter 2 External Input
30	**P0.6/INT06**	I/O	P0.6: GPIO P0.6 INT06: Input 6 of external interrupt 0
31	**P0.7/INT07**	I/O	P0.7: GPIO P0.7 INT07: Input 7 of external interrupt 0
32	**VSS**	Power	Ground

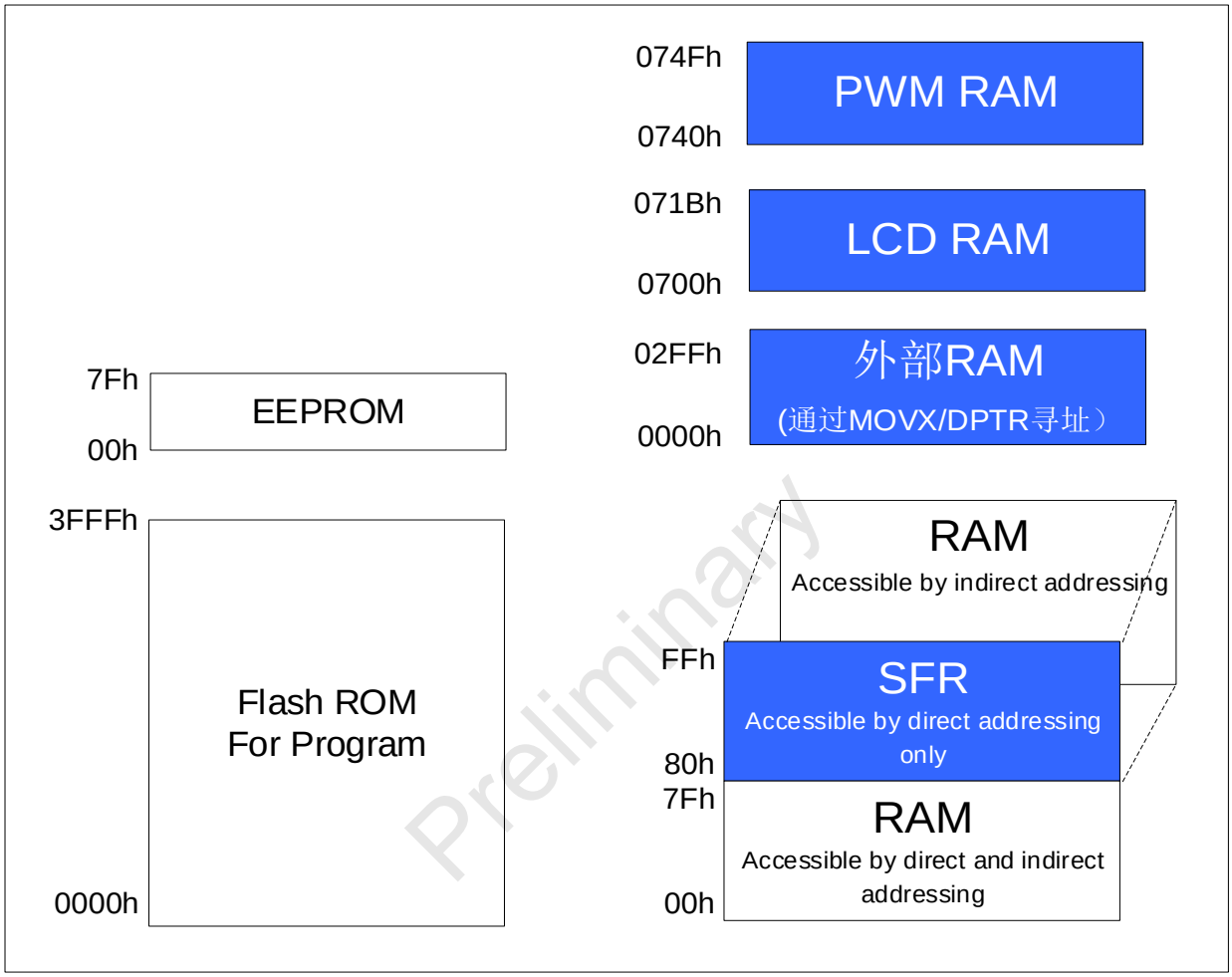
4 Inner BLOCK Diagram



SC92F744XB BLOCK DIAGRAM

5 Flash ROM and SRAM Structure

The structures of the SC92F744XB's Flash ROM and SRAM are shown as follows:



Flash ROM and SRAM Structure Diagram

5.1 Flash Rom

The SC92F744XB provides 16 Kbytes of Flash ROM with the ROM address of 0000H ~ 3FFFH. These 16 Kbytes of Flash ROM can be rewritten 10,000 times, which is able to program and erase by specialized ICP programming device (SOC PRO52/DPT52/SC LINK) provided by SinOne.

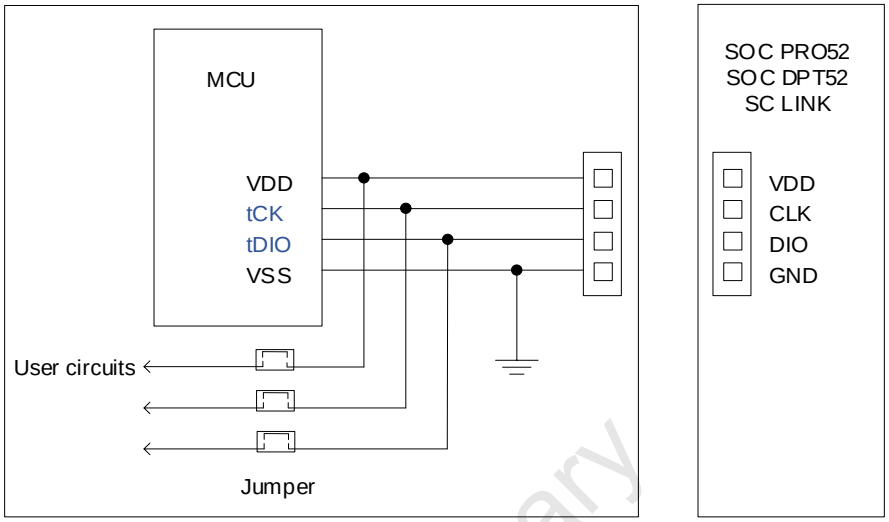
The MOVC instruction is non-addressable within 256 bytes (address of 0000H ~ 00FFH). That is to say, the user code unable to read the contents of the 256 bytes region that ensures that the encryption function of chip program. For more details, refer to "SinOne SC92F Series MCU Application Guide".

EEPROM is an data memory separated from 16K bytes ROM with the address of 00H ~ 7FH, which can be accessed by single-byte reading and writing operations in the program; for more details, refer to 21 EEPROM and IAP Operations.

Note: The EEPROM can be rewritten 100,000 times. Users cannot exceed the limit value, otherwise there will be an exception!

The SC92F744XB 16 Kbytes Flash ROM provide Empty Check, Program, Verify and Erase function other than Read function. This Flash ROM and EEPROM usually needs no Erase operation before writing. Directly writing data can realize coverage of new data.

The SC92F744XB Flash ROM can be programmed by tDIO, tCK, VDD and VSS, with its specific connection shown as follows:



ICP Mode Flash Writer Programming Connection Diagram

tDIO, tCK are 2-wire JTAG programming and emulation interface, JTAG mode and Normal mode can be set through code option. The specific operation is as follows:

OP_CTM1 (C2H@FFH) Customer Option Register1 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Symbol	VREFS	-	-	DISJTG	IAPS[1:0]		-	-
R/W	R/W	-	-	R/W	R/W	R/W	-	-
POR	n	x	x	n	n	n	x	x

Bit Number	Bit Symbol	Description
4	**DISJTG**	JTAG mode switch 0 : JTAG mode eanble 1 : Normal mode, JTAG function is invalid

5.2 Customer Option Memory (User Programming Setting)

A separate Flash data memory is embedded inside the SC92F744XB, called Code Option area, to save the user's presets. These presets will be written into IC when programming and loaded into SFR as default values during reset.

Option-related SFR Operating Instructions:

Reading and writing operations to option-related SFR are controlled by both register OPINX and register OPREG, with its respective address of Option SFR depending on register OPINX, as shown below:

Symbol	Address	Description	7	6	5	4	3	2	1	0
OP_HRCR	83H@FFH	System Clock Change Register	OP_HRCR[7: 0]							
OP_CTM0	C1H@FFH	Customer Option Register 0	ENWDT	ENXTL	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
OP_CTM1	C2H@FFH	Customer Option Register 1	VREFS	-	-	DISJTG	IAPS[1: 0]		-	-

OP_HRCR (83H@FFH) System Clock Change Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	OP_HRCR[7:0]							
R/W	R/W							
POR	n	n	n	n	n	n	n	n

Bit Number	Bit Mnemonic	Description
7 ~ 0	**OP_HRCR[7: 0]**	Internal high-frequency RC frequency adjustment Central value 10000000b corresponds to HRC central frequency, the larger the value is, the faster the frequency will be, vice versa.

OP_CTM0 (C1H@FFH) Customer Option Register0 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	ENWDT	ENXTL	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
R/W	R/W	R/W	R/W		R/W	R/W	R/W	
POR	n	n	n		n	n	n	

Bit Number	Bit Mnemonic	Description
7	**ENWDT**	Watchdog (WDT) control bit (This bit is transferred by the system to the value set by the user Code Option) 0: WDT invalid 1: WDT valid (WDT stops counting during IAP execution)
6	**ENXTL**	External High-Frequency crystal oscillator selection bit 0: External 32K crystal Interface disable, P5.0 and P5.1 valid 1: External 32K crystal Interface enable, P5.0 and P5.1 invalid
5 ~ 4	**SCLKS[1: 0]**	System clock frequency selection bits 00: System clock frequency is HRC frequency divided by 1; 01: System clock frequency is HRC frequency divided by 2; 10: System clock frequency is HRC frequency divided by 4; 11: System clock frequency is HRC frequency divided by 12;
3	**DISRST**	IO/RST selection bit 0: configure P5.2 as External Reset input pin 1: configure P5.2 as GPIO
2	**DISLVR**	LVR control bit 0: LVR valid 1: LVR invalid
1 ~ 0	**LVRS [1: 0]**	LVR voltage selection bits 11: 4.3V reset 10: 3.7 V reset 01: 2.9V reset 00: 2.3 V reset

OP_CTM1 (C2H@FFH) Customer Option Register1 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	VREFS	-	-	DISJTG	IAPS[1:0]		-	-
R/W	R/W	-	-	R/W	R/W	R/W	-	-
POR	n	x	x	n	n	n	x	x

Bit Number	Bit Mnemonic	Description
7	**VREFS**	Reference voltage selection bit (Initial values are configured by the user and loaded from Code Options) 0: Configure ADC VREF as V_{DD} 1: Configure ADC VREF as internally correct 2.4V
4	**DISJTG**	JTAG mode switch 0 : JTAG mode eanble 1 : Normal mode, JTAG function is invalid
3 ~ 2	**IAPS[1: 0]**	EEPROM and IAP Area Selection Bits 00: Code memory prohibits IAP operations, only EEPROM data memory is used for data storage 01: last 0.5k code memory allows IAP operation (7E00H ~ 3FFFH) 10: Last 1k code memory allows IAP operation (7C00H ~ 3FFFH) 11: All code memory allows IAP operation (0000H ~ 3FFFH)

5.2.1 Customer-Option-related Registers Operation Instructions

Option-related SFRs reading and writing operations are controlled by both OPINX and OPREG registers, with their respective position of Option SFR depending on OPINX and its value written to option-related SFR depending on register OPREG:

Symbol	Address	Description		POR
OPINX	FEH	Option Pointer	OPINX[7: 0]	00000000b
OPREG	FFH	Option Register	OPREG[7: 0]	nnnnnnnnb

When operating Option-related SFRs, register OPINX stores the address of option-related registers and register OPREG stores corresponding value.

For Example: To configure OP_HRCR as 0x01, specific operation method is shown below:

C program example:

```
OPINX = 0x83;          //Write OP_HRCR address into OPINX register

OPREG = 0x01;          //Write 0x01 into OPREG register (the value to be written into OP_HRCR register)
```

Assembler program example:

```
MOV OPINX, #83H;       //Write OP_HRCR address into OPINX register

MOV OPREG, #01H;       //Write 0x01 into OPREG register (the value to be written into
OP_HRCR register)
```

Note: It is forbidden to write any value beyond SFR address of Customer Option region into OPINX register! Or else, it may cause abnormal system operation.

5.3 SRAM

The SRAM of the SC92F744XB is divided into internal 256 bytes RAM, external 768 bytes RAM and 44 bytes PWM&LCD RAM. The address of Internal RAM range from 00H to FFH, including high 128 bytes (address of from80H to FFH) only addressed indirectly and low 128 bytes (address of from 00H to 7FH) addressed both directly and indirectly).

The address of SFRs are also from 80H to FFH. The difference between SFR and internal high 128 bytes SRAM is that the former is addressed directly but the latter addressed indirectly only.

The address of External RAM from 0000H to 02FFH, which can be accessible by MOVX instruction.

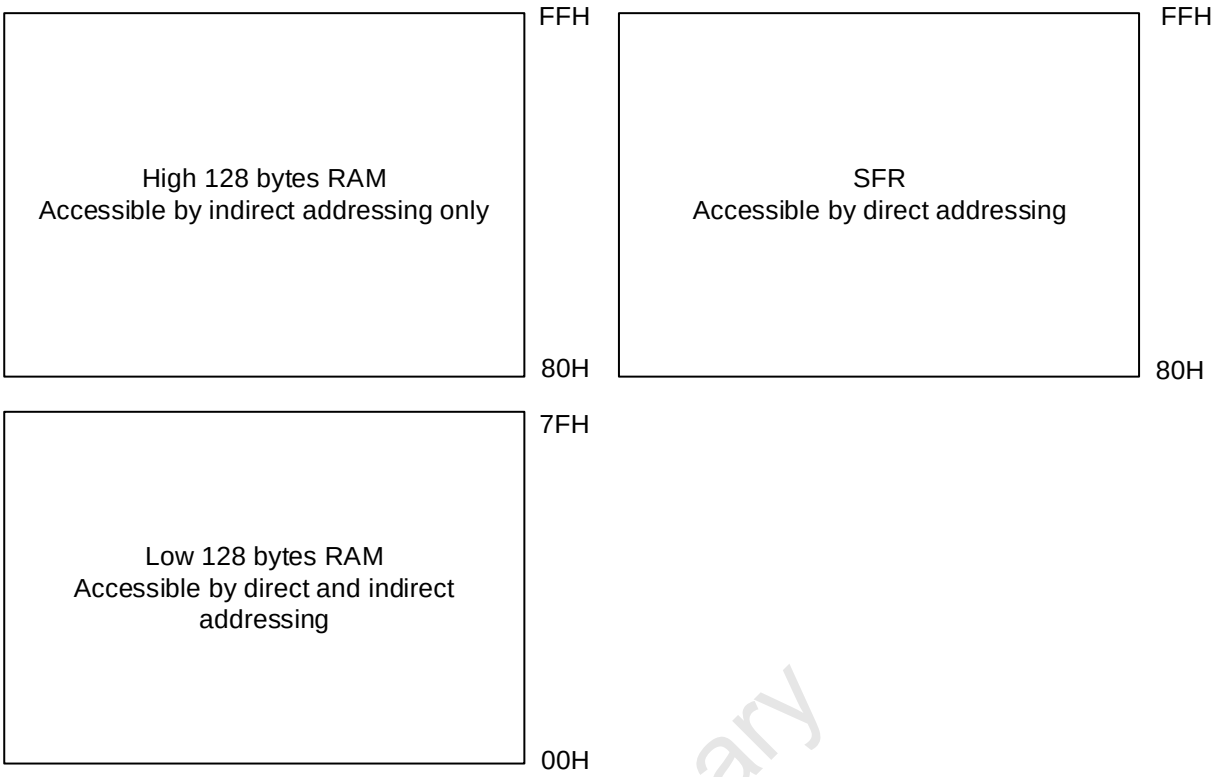
5.3.1 Internal 256 bytes SRAM

Internal low 128 bytes SRAM area is divided into three parts:

1. Register bank 0 ~ 3, address from 00H to 1FH. The RS1 and RS0 of PSW register can select the currently active SFR. Using Register bank 0 ~ 3 can accelerate arithmetic speed;

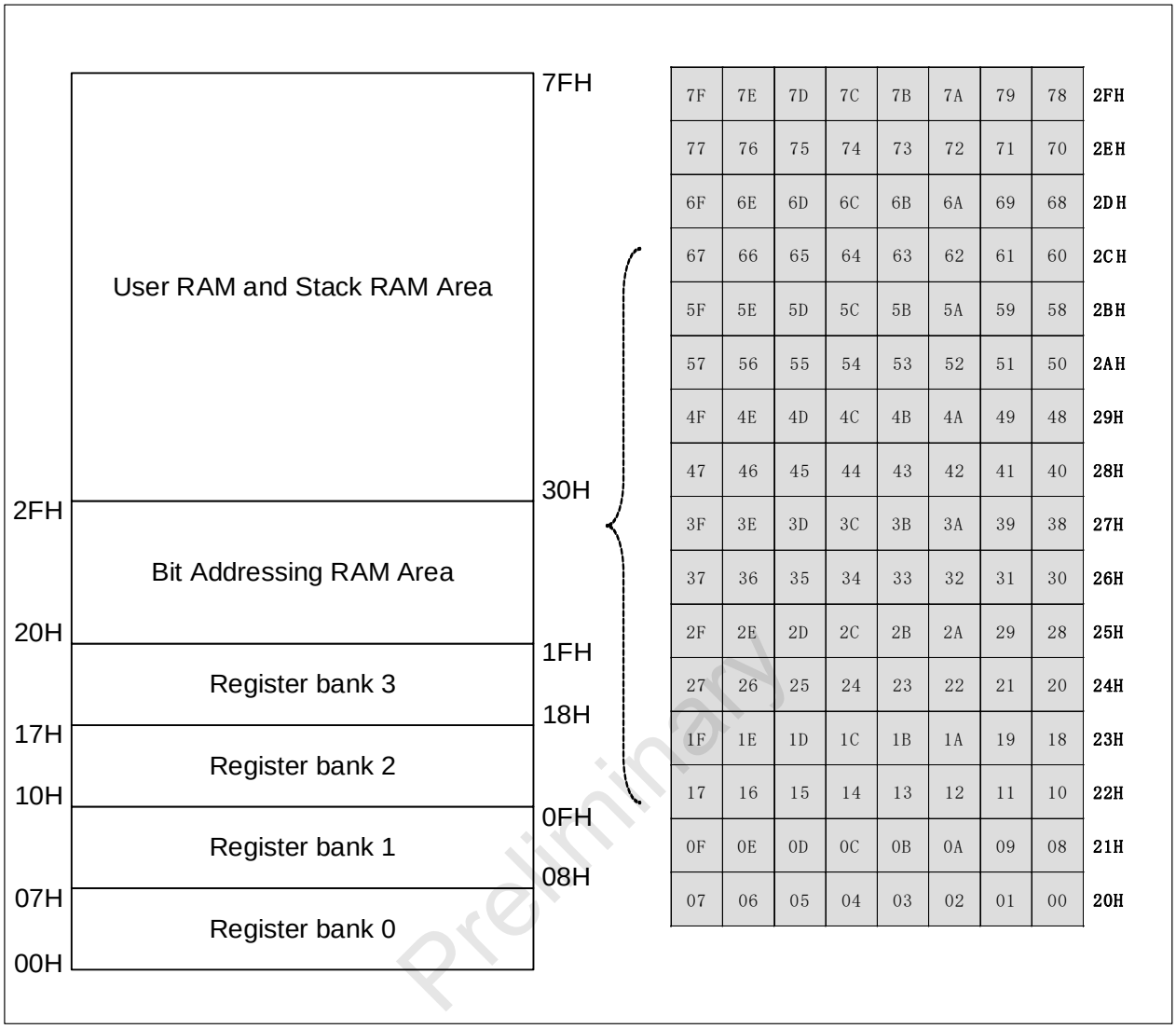
2. Bit addressing area , 20H ~2FH; user can use it as normal RAM or bitwise addressing RAM; for the latter, the bit address is from 00H to 7FH (bitwise addressing is different from normal SRAM byte-oriented addressing), which can be distinguished by instructions in program;

3. User RAM and stack area, the 8-bit stack pointer will point to stack area after the SC92F744XB reset; in general, users can set initial value in initializer, it is recommended to configure in the unit interval from E0H to FFH.



Internal 256 bytes RAM Structure Diagram

Internal low 128 bytes RAM structure is shown below:



SRAM Structure Diagram

5.3.2 External 768 bytes SRAM

The external 768 bytes RAM (SRAM) can be accessed by instruction ‘’MOVX @DPTR’’ or instruction MOVX A ,@Ri or MOVX @Ri,A together whit EXADH register. EXADH register stores high address of external SRAM; Ri register stores low 8-bit address of external SRAM.

EXADH (F7H) External SRAM Operating Address High (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	-	-	EXADH [2:0]		
POR	x	x	x	x	x	0	0	0

Bit Number	Bit Mnemonic	Description
2 ~ 0	**EXADH [2: 0]**	External SRAM Address High position
7 ~ 2	-	Reserved

5.3.3 PWM&LCD 44 bytes SRAM

PWM duty cycle adjustment register occupies 0740H~074FH and is only writable and unreadable. Refer to the 13.2 PWM related SFR register for specific operation methods.

LCD/LED display RAM occupies 0700H~071BH. Refer to 15.2 LCD/LED display RAM configuration for specific operation methods.

6 Special Function Register (SFR)

6.1 SFR Mapping

The SC92F744XB provides some registers equipped with special functions, called SFR. The address of such SFRs is from 80H to FFH, some are bit-addressable, and others are not. It is very convenient for these bit addressable registers to change the value of single bit, of which the address is end up with figure “0” or “8”. All SFR shall use direct addressing for addressing.

The name and address of the SC92F744XB special function registers are shown in the table below:

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
F8h	-	-	-	-	CHKSUML	CHKSUMH	OPINX	OPREG
F0h	B	IAPKEY	IAPADL	IAPADH	IAPADE	IAPDAT	IAPCTL	EXADH
E8h	-	EXA0	EXA1	EXA2	EXA3	EXBL	EXBH	OPERCON
E0h	ACC	-	-	-	-	-	-	-
D8h	P5	P5CON	P5PH	-	-	-	-	-
D0h	PSW	-	-	PWMCON	PWMCFG	-	-	-
C8h	T2CON	T2MOD	RCAP2L	RCAP2H	TL2	TH2	BTMCON	WDTCON

C0h	P4	P4CON	P4PH	-	-	-	INT2F	INT2R
B8h	IP	IP1	INT0F	INT0R	INT1F	INT1R	-	-
B0h	P3	P3CON	P3PH	P3VO	-	-	CMPCFG	CMPCON
A8h	IE	IE1	ADCCFG2	ADCCFG0	ADCCFG1	ADCCON	ADCVL	ADCVH
A0h	P2	P2CON	P2PH	P2VO	-	-	-	-
98h	SCON	SBUF	P0CON	P0PH	P0VO	SSCON0	SSCON1	SSDAT
90h	P1	P1CON	P1PH	DDRCON	P1VO	SSCON2	IOHCON0	IOHCON1
88h	TCON	TMOD	TL0	TL1	TH0	TH1	TMCON	OTCON
80h	P0	SP	DPL	DPH	-	-	-	PCON
	Bit Addressable	Not Bit Addressable						

Notes:

1. **The hollow spaces in the table above means that there is no such register RAM, which is not recommended for users.**

2. **The address of SFR for system configuration is F1H ~ FFH, user use it may result in system exceptions. Users are not allowed to conduct clearing or other operations to these registers during the system initialization process.**

6.2 SFR Instructions

For details on each SFR, see the following table:

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
P0	80H	P0 Data Register	P07	P06	P05	P04	P03	P02	P01	P00	00000000b
SP	81H	Stack Pointer	SP[7:0]								00000111b
DPL	82H	Data Pointer Low byte	DPL[7:0]								00000000b
DPH	83H	Data Pointer High byte	DPH[7:0]								00000000b

PCON	87H	Power Management Control Register	SMOD	-	-	-	-	-	STOP	IDL	0xxxxx00b
TCON	88H	Timer Control Register	TF1	TR1	TF0	TR0	IE1	-	IE0	-	00000x0xb
TMOD	89H	Timer Operating Mode Register	-	C/T1	M11	M01	-	C/T0	M10	M00	x000x000b
TL0	8AH	Timer0 Low 8 bits	TL0[7:0]								00000000b
TL1	8BH	Timer1 Low 8 bits	TL1[7:0]								00000000b
TH0	8CH	Timer0 High 8 bits	TH0[7:0]								00000000b
TH1	8DH	Timer1 High 8 bits	TH1[7:0]								00000000b
TMCON	8EH	Timer Frequency Control Register	-	-	-	-	-	T2FD	T1FD	T0FD	xxxxx000b
OTCON	8FH	Output Control Register	SSMOD[1:0]		-	-	VOIRS[1:0]		SCS	BIAS	00xx0000b
P1	90H	P1 Data Register	P17	P16	P15	P14	P13	P12	P11	P10	00000000b
P1CON	91H	P1 I/O Control Register	P1C7	P1C6	P1C5	P1C4	P1C3	P1C2	P1C1	P1C0	00000000b
P1PH	92H	P1 Pull-up Resistor Control Register	P1H7	P1H6	P1H5	P1H4	P1H3	P1H2	P1H1	P1H0	00000000b
DDRCON	93H	Display drive control register	DDRON	DMOD	DUTY[1:0]		VLCD[3:0]				00000000b
P1VO	94H	P1 Display Drive Output Register	P17VO	P16VO	P15VO	P14VO	P13VO	P12VO	P11VO	P10VO	00000000b
SSCON2	95H	SSI Control Register 2	SSCON2[7:0]								00000000b
IOHCON0	96H	IOH Setup Register 0	P1H[1:0]		P1L[1:0]		P0H[1:0]		P0L[1:0]		00000000b
IOHCON1	97H	IOH Setup Register 1	-	-	P3L[1:0]		P2H[1:0]		P2L[1:0]		xx000000b
SCON	98H	Serial Port Control Register	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00000000b

SBUF	99H	Serial Port Data Cache Register	SBUF[7:0]							00000000b
P0CON	9AH	P0 I/O Control Register	P0C7	P0C6	P0C5	P0C4	P0C3	P0C2	P0C1	P0C0	00000000b
P0PH	9BH	P0 Pull-up Resistor Control Register	P0H7	P0H6	P0H5	P0H4	P0H3	P0H2	P0H1	P0H0	00000000b
P0VO	9CH	P0 Port LCD Voltage Output Register	P07VO	P06VO	P05VO	P04VO	P03VO	P02VO	P01VO	P00VO	00000000b
SSCON0	9DH	SSI Control Register 0	SSCON0[7:0]								00000000b
SSCON1	9EH	SSI Control Register 1	SSCON1[7:0]								00000000b
SSDAT	9FH	SSI Data Register	SSD[7:0]								00000000b
P2	A0H	P2 Data Register	P27	P26	P25	P24	P23	P22	P21	P20	00000000b
P2CON	A1H	P2 I/O Control Register	P2C7	P2C6	P2C5	P2C4	P2C3	P2C2	P2C1	P2C0	00000000b
P2PH	A2H	P2 Pull-up Resistor Control Register	P2H7	P2H6	P2H5	P2H4	P2H3	P2H2	P2H1	P2H0	00000000b
P2VO	A3H	P2 Display Drive Output Register	P27VO	P26VO	P25VO	P24VO	P23VO	P22VO	P21VO	P20VO	00000000b
IE	A8H	Interrupt Enable Register	EA	EADC	ET2	EUART	ET1	EINT1	ET0	EINT0	00000000b
IE1	A9H	Interrupt Enable Register 1	-	-	ECMP	-	EINT2	EBTM	EPWM	ESSI	xx0x0000b
ADCCFG2	AAH	ADC Configuration Register 2	-	-	-	-	-	LOWSP	ADCCK[1:0]		xxxxx000b
ADCCFG0	ABH	ADC Configuration Register 0	EAIN7	EAIN6	EAIN5	EAIN4	EAIN3	EAIN2	EAIN1	EAIN0	00000000b
ADCCFG1	ACH	ADC Configuration Register 1	EAIN15	EAIN14	EAIN13	EAIN12	EAIN11	EAIN10	EAIN9	EAIN8	00000000b
ADCCON	ADH	ADC Control Register	ADCEN	ADCS	EOC/ ADCIF	ADCIS[4:0]					00000000b

ADCVL	AEH	ADC Result Register	ADCV[3:0]				-	-	-	-	0000xxxxb
ADCVH	AFH	ADC Result Register	ADCV[11:4]								00000000b
P3	B0H	P3 data register	P37	P36	P35	P34	P33	P32	P31	P30	00000000b
P3CON	B1H	P3 Input/Output Control Register	P3C7	P3C6	P3C5	P3C4	P3C3	P3C2	P3C1	P3C0	00000000b
P3PH	B2H	P3 Pull Resistance Control Register	P3H7	P3H6	P3H5	P3H4	P3H3	P3H2	P3H1	P3H0	00000000b
P3VO	B3H	P3 Display Drive Output Register	P37VO	P36VO	P35VO	P34VO	P33VO	P32VO	P31VO	P30VO	00000000b
CMPCFG	B6H	Analog comparator setup register	-	-	-	-	CMPIM[1:0]		CMPIS[1:0]		xxxx0000b
CMPCON	B7H	Analog comparator control register	CMPEN	CMPIF	CMPSTA	-	CMPRF[3:0]				000x0000b
IP	B8H	Interrupt Priority Control Register	-	IPADC	IPT2	IPUART	IPT1	IPINT1	IPT0	IPINT0	x0000000b
IP1	B9H	Interrupt Priority Control Register 1	-	-	IPCMP	-	IPINT2	IPBTM	IPPWM	IPSSI	xx0x0000b
INT0F	BAH	INT0 Falling Edge Interrupt Control Register	INT0F7	INT0F6	INT0F5	INT0F4	-	-	-	-	0000xxxxb
INT0R	BBH	INT0 Rising Edge Interrupt Control Register	INT0R7	INT0R6	INT0R5	INT0R4	-	-	-	-	0000xxxxb
INT1F	BCH	INT1 Falling Edge Interrupt Control Register	INT1F7	INT1F6	INT1F5	INT1F4	INT1F3	INT1F2	INT1F1	INT1F0	00000000b
INT1R	BDH	INT1 Rising Edge Interrupt Control Register	INT1R7	INT1R6	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0	00000000b
P4	C0H	P4 data register	P47	P46	P45	P44	P43	P42	P41	P40	00000000b
P4CON	C1H	P4 Input/Output Control Register	P4C7	P4C6	P4C5	P4C4	P4C3	P4C2	P4C1	P4C0	00000000b

P4PH	C2H	P4 Pull Resistance Control Register	P4H7	P4H6	P4H5	P4H4	P4H3	P4H2	P4H1	P4H0	00000000b
INT2F	C6H	INT2 Falling Edge Interrupt Control Register	-	-	-	-	INT2F3	INT2F2	INT2F1	INT2F0	xxxx0000b
INT2R	C7H	INT2 Rising Edge Interrupt Control Register	-	-	-	-	INT2R3	INT2R2	INT2R1	INT2R0	xxxx0000b
T2CON	C8H	Timer2 Control Register	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	00000000b
T2MOD	C9H	Timer2 Operating Mode Register	-	-	-	-	-	-	T2OE	DCEN	xxxxxx00b
RCAP2L	CAH	Timer2 Reload Low 8 bits	RCAP2L[7:0]								00000000b
RCAP2H	CBH	Timer2 Reload High 8 bits	RCAP2H[7:0]								00000000b
TL2	CCH	Timer2 Low 8 bits	TL2[7:0]								00000000b
TH2	CDH	Timer2 High 8 bits	TH2[7:0]								00000000b
BTMCON	CEH	Low-Frequency Timer Control Register	ENBTM	BTMIF	-	-	BTMFS[3:0]				00xx0000b
WDTCON	CFH	WDT Control Register	-	-	-	CLRWDT	-	WDTCKS[2:0]			xxx0x000b
PSW	D0H	Program Status Word Register	CY	AC	F0	RS1	RS0	OV	F1	P	00000000b
PWMCON	D3H	PWM Period Setting Register	PWMPD[7:0]								00000000b
PWMCFG	D4H	PWM Setup Register	ENPWM	PWMIF	PWMCK[1:0]		PWMPD[11:8]				00000000b
P5	D8H	P5 Data Register	-	-	P55	P54	P53	P52	P51	P50	xx000000b
P5CON	D9H	P5 I/O Control Register	-	-	P5C5	P5C4	P5C3	P5C2	P5C1	P5C0	xx000000b
P5PH	DAH	P5 Pull-up Resistor Control Register	-	-	P5H5	P5H4	P5H3	P5H2	P5H1	P5H0	xx000000b

ACC	E0H	Accumulator	ACC[7:0]								00000000b
EXA0	E9H	Extended Accumulator 0	EXA[7:0]								00000000b
EXA1	EAH	Extended Accumulator 1	EXA[15:8]								00000000b
EXA2	EBH	Extended Accumulator 2	EXA[23:16]								00000000b
EXA3	ECH	Extended Accumulator 3	EXA[31:24]								00000000b
EXBL	EDH	Extended B Register L	EXB [7:0]								00000000b
EXBH	EEH	Extended B Register H	EXB [15:8]								00000000b
OPERCON	EFH	Arithmetic Control Register	OPERS	MD	-	-	-	-	-	CHKSUMS	00xxxxx0b
B	F0H	B Register	B[7:0]								00000000b
IAPKEY	F1H	IAP Protection Register	IAPKEY[7:0]								00000000b
IAPADL	F2H	IAP Address Low byte Register	IAPADR[7:0]								00000000b
IAPADH	F3H	IAP Address High byte Register	-	IAPADR[14:8]							x0000000b
IAPADE	F4H	IAP Extended Address Register	IAPADER[7:0]								00000000b
IAPDAT	F5H	IAP Data Register	IAPDAT[7:0]								00000000b
IAPCTL	F6H	IAP Control Register	-	-	-	-	PAYTIMES[1:0]		CMD[1:0]		xxxx0000b
EXADH	F7H	External SRAM Operating Address High	-	-	-	-	-	EXADH [2:0]			xxxxx000b
CHKSUML	FCH	Check Sum Result Register Low	CHKSUML[7:0]								00000000b
CHKSUMH	FDH	Check Sum Result Register High	CHKSUMH[7:0]								00000000b

OPINX	FEH	Option Pointer	OPINX[7:0]	00000000b
OPREG	FFH	Option Register	OPREG[7:0]	nnnnnnnnb

6.2.1 C51 Core SFRs

Program Counter (PC)

PC does not belong to SFR .16-bit PC is the register used to control instruction execution sequence. After power-on or reset of microcontroller unit, PC value is 0000H, that is to say, the microcontroller unit is to execute program from 0000H.

Accumulator ACC (E0H)

Accumulator ACC is one of the commonly-used registers in 8051-based microcontroller unit, using A as mnemonic symbol in the instruction system. It is usually used to store operand and results for calculation or logical operations.

B Register (F0H)

B Register shall be used together with Accumulator A in multiplication and division operations. For example, instruction “MUL A, B” is used to multiply 8-bit unsigned numbers of Accumulator A and Register B. As for the acquired 16-bit product, low byte is placed in A and High byte in B. As for “DIV A, B” is used to divide A by B, place integer quotient in A and remainder in B. Register B can also be used as common temporary register.

Stack Pointer SP (81H)

Stack pointer is an 8-bit specialized register, it indicates the address of top stack in common RAM. After resetting of microcontroller unit, the initial value of SP is 07H, and the stack will increase from 08H. 08H ~ 1FH is address of register banks 1 ~ 3.

PSW (D0H) Program Status Word Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

<table>
<tr><th>Bit Number</th><th>Bit Mnemonic</th><th>Description</th></tr>
<tr><td>7</td><td>CY</td><td>Carry Flag bit

1: The top digit of add operation has carry bit or the top digit of subtraction operation has the borrow digit

0: The top digit of add operation has no carry bit or the top digit of subtraction operation has no borrow digit</td></tr>
<tr><td>6</td><td>AC</td><td>Carry-bit auxiliary flag bit (adjustable upon BCD code add and subtraction operations)

1: There is carry bit in bit 3 upon add operation and borrow bit in bit 3 upon subtraction operation

0: No borrow bit and carry bit</td></tr>
<tr><td>5</td><td>F0</td><td>User flag bit</td></tr>
<tr><td>4 ~ 3</td><td>RS1、RS0</td><td>Register banks selection bits

<table>
<tr><td>RS1</td><td>RS0</td><td>Current Selected Register banks 0 ~ 3</td></tr>
<tr><td>0</td><td>0</td><td>Group 0 (00H ~ 07H)</td></tr>
<tr><td>0</td><td>1</td><td>Group 1 (08H ~ 0FH)</td></tr>
<tr><td>1</td><td>0</td><td>Group 2 (10H ~ 17H)</td></tr>
<tr><td>1</td><td>1</td><td>Group 3 (18H ~ 1FH)</td></tr>
</table>
</td></tr>
<tr><td>2</td><td>OV</td><td>Overflow flag bit</td></tr>
<tr><td>1</td><td>F1</td><td>F1 flag bit

User customized flag</td></tr>
<tr><td>0</td><td>P</td><td>Parity flag bit. This flag bit is the parity value of the number of 1 in accumulator ACC.

1: Odd number of number of 1 in ACC

0: Even number of number of 1 in ACC (including 0)</td></tr>
</table>

Data Pointer DPTR (82H, 83H)

The Data pointer DPTR of SC92F744XB is a 16-bit dedicated register, which is composed of Low byte DPL (82H) and High byte DPH (83H). DPTR is a register that can directly conduct 16-bit operation, which can also conduct operations on DPL and DPH by byte.

7 Power, Reset and System Clock

7.1 Power Circuit

The SC92F744XB Power includes circuits such as BG, LDO, POR and LVR, which are able to reliably work within the scope of 2.4V ~ 5.5V. Besides, a calibrated 2.4V reference is build in the IC, which is used as ADC internal reference voltage. The user can search for specific configuration contents in 18 Analog-to-digital converter (ADC).

7.2 Power-on Reset

After the SC92F744XB power-on, the processes carried out before execution of client software are as follows:

- Reset stage
- Loading information stage
- Normal operation stage

7.2.1 Reset Stage

The SC92F744XB will always be in reset mode. There will not be a valid clock until the voltage supplied to the SC92F744XB is higher than certain voltage. The duration of reset stage is related to rising speed of external power. Once the external supply voltage is up to built-in POR voltage, the reset stage would be completed.

7.2.2 Loading Information Stage

There is a preheating counter inside the SC92F744XB. During the reset stage, this preheating counter is always reset as zero. After the voltage is higher than POR voltage, internal RC oscillator starts to oscillate and this preheating counter starts to count. When internal preheating counter counts up to certain number, one byte data will be read from IFB of Flash ROM (including Code Option) for every certain number of HRC clock, which is saved to internal system registers. After the preheating is completed, such reset signal will end.

7.2.3 Normal Operating Stage

After the loading information stage has been completed, the SC92F744XB starts to read instruction code from Flash and enters normal operating stage. At this time, LVR voltage is the set value of Code Option written by user.

7.3 Reset Modes

The SC92F744XB has 4 kinds of reset modes: ① External RST reset ②Low-voltage reset (LVR) ③Power-on reset (POR) ④Watchdog (WDT) reset.

7.3.1 External Reset

External reset is to supply a certain width reset pulse signal to the SC92F744XB from the RST pin to realize the SC92F744XB reset.

User can configure P5.2 pin as RST (reset pin) in Customer Option via PC program software before programming.

7.3.2 Low-voltage Reset (LVR)

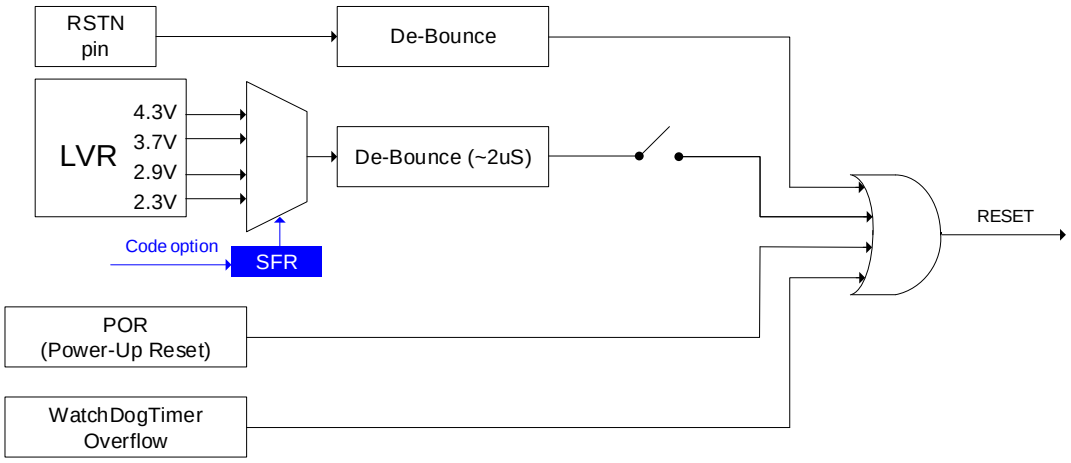
The SC92F744XB provides a low-voltage reset circuit. There are 4-level LVR voltage options: 4.3V, 3.7V, 2.9V and 2.3V. The default is the Option value written by user.

OP_CTM0(C1H@FFH) Customer Option Register 0 (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	ENWDT	ENXTL	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
R/W	R/W	R/W	R/W		R/W	R/W	R/W	
POR	n	n	n		n	n	n	

Bit Number	Bit Mnemonic	Description
2	**DISLVR**	LVR control bit 0: LVR valid 1: LVR invalid
1 ~ 0	**LVRS [1: 0]**	LVR voltage selection bits 11: 4.3 V reset 10: 3.7 V reset 01: 2.9 V reset 00: 2.3 V reset

The Circuit Diagram of the SC92F744XB Resetting Part is shown below:



The SC92F744XB Reset Diagram

7.3.3 Power-on Reset (POR)

The SC92F744XB provides a power-on reset circuit. When power voltage V_{DD} is up to POR reset voltage, the system will be reset automatically.

7.3.4 Watchdog Reset (WDT)

The SC92F744XB has a WDT, the clock source of which is the internal 128 kHz oscillator. User can select whether to enable Watchdog Reset function by programmer Code Option.

OP_CTM0 (C1H@FFH) Customer Option Register 0 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	ENWDT	ENXTL	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
R/W	R/W	R/W	R/W		R/W	R/W	R/W	
POR	n	n	n		n	n	n	

Bit Number	Bit Mnemonic	Description
7	**ENWDT**	WDT control bit (This bit is transferred by the system to the value set by the user Code Option) 1: WDT valid

		0: WDT invalid

WDTCON (CFH) WDT Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	CLRWDT	-	WDTCKS[2: 0]		
R/W	-	-	-	R/W	-	R/W		
POR	x	x	x	0	x	0	0	0

Bit Number	Bit Mnemonic	Description
4	**CLRWDT**	Clear WDT (Only valid when set to 1) 1: WDT counter restart, cleared by system hardware
2 ~ 0	**WDTCKS [2: 0]**	WDT clock selection bits

WDTCKS[2: 0]	WDT overflow time
000	500ms
001	250ms
010	125ms
011	62.5ms
100	31.5ms
101	15.75ms
110	7.88ms

		111	3.94ms	
7 ~ 5, 3	-	Reserved		

7.3.5 Register Reset Value

During reset, most registers are set to their initial values and the WDT remains disable. The initial value of program counter (PC) is 0000h, and the initial value of stack pointer SP is 07h. Reset of “Hot Start” (such as WDT, LVR, etc.) will not influence SRAM which always keep the value before resetting. The SRAM contents will be retained until the power voltage is too low to keep RAM alive.

The initial value of power-on reset in SFRs is shown in the table below:

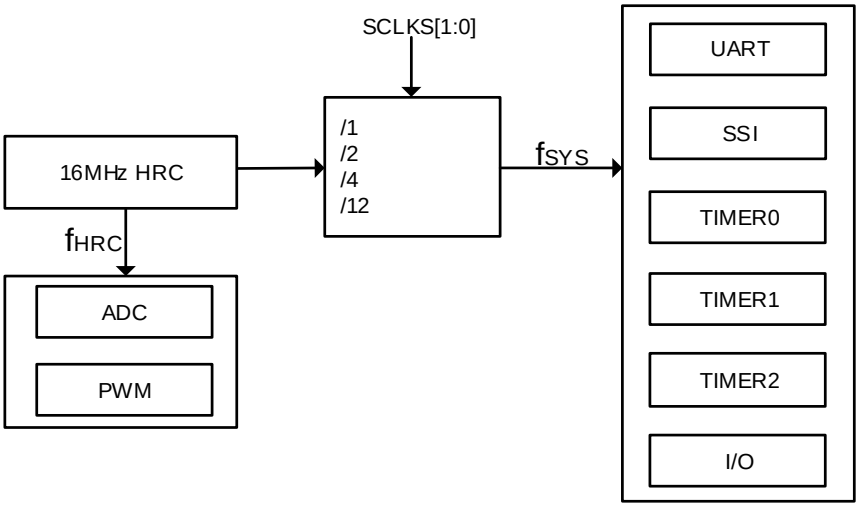
Mnemonic	Reset value	Mnemonic	Reset value
ACC	00000000b	P1VO	00000000b
B	00000000b	P2	00000000b
PSW	00000000b	P2CON	00000000b
SP	00000111b	P2PH	00000000b
DPL	00000000b	P2VO	00000000b
DPH	00000000b	P3	00000000b
PCON	0xxxxx00b	P3CON	00000000b
ADCCFG0	00000000b	P3PH	00000000b
ADCCFG1	00000000b	P3VO	00000000b
ADCCFG2	xxxxx000b	P4	00000000b
ADCCON	00000000b	P4CON	00000000b
ADCVH	00000000b	P4PH	00000000b
ADCVL	0000xxxxb	P5	xx000000b

BTMCON	00xx0000b	P5CON	xx000000b
IAPADE	00000000b	P5PH	xx000000b
IAPADH	x0000000b	PWMCFG	00000000b
IAPADL	00000000b	PWMCON	00000000b
IAPCTL	xxxx0000b	RCAP2H	00000000b
IAPDAT	00000000b	RCAP2L	00000000b
IAPKEY	00000000b	SBUF	00000000b
IE	00000000b	SCON	00000000b
IE1	xx000000b	SSCON0	00000000b
INT0R	0000xxxxb	SSCON1	00000000b
INT1R	00000000b	SSCON2	00000000b
INT2R	xxxx0000b	SSDAT	00000000b
INT0F	0000xxxxb	TCON	00000x0xb
INT1F	00000000b	TMCON	xxxxx000b
INT2F	xxxx0000b	TMOD	x000x000b
IP	x0000000b	TH0	00000000b
IP1	xxx00000b	TL0	00000000b
OPINX	00000000b	TH1	00000000b
OPREG	nnnnnnnnb	TL1	00000000b
EXADH	xxxxx000b	T2CON	00000000b
OTCON	00xx0000b	TH2	00000000b
IOHCON0	00000000b	TL2	00000000b

IOHCON1	xx000000b	T2MOD	xxxxxx00b
P0	00000000b	WDTCON	xxx0x000b
P0CON	00000000b	CMPCFG	xxxx0000b
P0PH	00000000b	CMPCON	000x0000b
P0VO	00000000b	DDRCON	00000000b
P1	00000000b	CHKSUMH	00000000b
P1CON	00000000b	CHKSUML	00000000b
P1PH	00000000b	-	-

7.4 High-speed RC Oscillator

The SC92F744XB has a built-in adjustable high-precision HRC. HRC is precisely calibrated to 16 MHz@5V/25℃ when delivery. The user can set system clock as 16/8/4/1.33MHz by programmer Code Option. The calibration process is to filter the influence of processing deviation on precision. There will be certain drifting of this HRC depending on operating temperature and voltage. As for voltage drifting (3.0V ~ 5.5V) and temperature drifting (-20℃ ~ 85℃), the deviation is within ±1%.



The SC92F744XB Internal Clock Relationship

OP_CTM0 (C1H@FFH) Customer Option Register 0 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	ENWDT	ENXTL	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
R/W	R/W	R/W	R/W		R/W	R/W	R/W	
POR	n	n	n		n	n	n	

Bit Number	Bit Mnemonic	Description
5 ~ 4	SCLKS[1: 0]	System clock frequency selection bits: 00: reserved; System clock frequency is HRC frequency divided by1; 01: system clock frequency is HRC frequency divided by 2; 10: system clock frequency is HRC frequency divided by 4; 11: system clock frequency is HRC frequency divided by 12;

The SC92F744XB has a special function: the user can modify SFR value to adjust frequency of HRC within certain scope. User can realize this operation by configuring OP_HRCR register. For configuration method of this register, refer to 5.2.1 Customer-Option-related Registers Operation Instructions.

OP_HRCR (83h@FFH) System Clock Change Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	OP_HRCR[7: 0]							
R/W	R/W							
POR	n	n	n	n	n	n	n	n

<table>
<tr><th>Bit Number</th><th>Bit Mnemonic</th><th>Description</th></tr>
<tr><td>7~0</td><td>OP_HRCR[7:0]</td><td>
HRC frequency change register

User can change high-frequency oscillator frequency f_{HRC} by modifying the value of this register, and then change the IC system clock frequency f_{SYS}:

1. Initial value OP_HRCR[s] after OP_HRCR[7: 0] power-on is a fixed value, which guarantee f_{HRC} is 16MHz, there may be difference in OP_HRCR[s] of each IC

2. When initial value is OP_HRCR[s], IC system clock frequency f_{SYS} can set specifically as 16/8/4/1.33MHz by Option. For each change of 1 for OP_HRCR [7: 0], the change of f_{SYS} frequency is about 0.23%.

The relationship between OP_HRCR [7: 0] and output frequency f_{SYS} is shown as follows:

<table>
<tr><th>OP_HRCR [7:0]</th><th>f_{SYS} actual output frequency
(taking 16M as an example)</th></tr>
<tr><td>OP_HRCR [s]-n</td><td>16000*(1-0.23%*n)kHz</td></tr>
<tr><td>...</td><td>....</td></tr>
<tr><td>OP_HRCR [s]-2</td><td>16000*(1-0.23%*2) = 15926.4kHz</td></tr>
<tr><td>OP_HRCR [s]-1</td><td>16000*(1-0.23%*1) = 15963.2kHz</td></tr>
<tr><td>OP_HRCR [s]</td><td>16000kHz</td></tr>
<tr><td>OP_HRCR [s]+1</td><td>16000*(1+0.23%*1) = 16036.8kHz</td></tr>
<tr><td>OP_HRCR [s]+2</td><td>16000*(1+0.23%*2) = 16073.6kHz</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>OP_HRCR [s]+n</td><td>16000*(1+0.23%*n)kHz</td></tr>
</table>
</td></tr>
</table>

		Notes: **1. The value of OP_HRCR[7:0] after each power-on of the IC is the value of high-frequency oscillator frequency f_{HRC} closest to 16MHz; the user can modify the value of HRC after each power-on by means of EEPROM to make IC system clock frequency f_{SYS} work at the frequency the user needs.** **2. To guarantee IC operating reliably, the maximum operating frequency of IC shall not exceed 10% of 16MHz, which is 17.6MHz;** **3. The user shall confirm the change of HRC frequency will not influence other functions.**

7.5 Low-speed RC Oscillator and Low-speed Clock Timer

The SC92F744XB is equipped with a built-in 128 kHz RC oscillation circuit and a 32.768k Hz crystal oscillation circuit, which can be set as clock source of low-frequency clock timer Base Timer. This oscillator is directly connected to Base Timer , which can wake up CPU from STOP mode and generate interrupt.

BTMCON (CEH) Low-Frequency Timer Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Symbol	ENBTM	BTMIF	-	-	BTMFS[3:0]			
R/W	R/W	R/W	-	-	R/W			
POR	0	0	x	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**ENBTM**	Low-frequency Base Timer start control bit 0: Base Timer not start 1: Base Timer start
6	**BTMIF**	Base Timer interrupt application flag bit When CPU receives Base Timer interrupt, this flag will be cleared

		automatically by hardware.
3 ~ 0	**BTMFS [3: 0]**	Low-frequency clock interrupt frequency selection bits 0000: an interrupt is generated for every 15.625ms 0001: an interrupt is generated for every 31.25ms 0010: an interrupt is generated for every 62.5ms 0011: an interrupt is generated for every 125ms 0100: an interrupt is generated for every 0.25s 0101: an interrupt is generated for every 0.5s 0110: an interrupt is generated for every 1.0s 0111: an interrupt is generated for every 2.0s 1000: an interrupt is generated for every 4.0s 1001~1111: reserve
5 ~ 4	-	Reserved

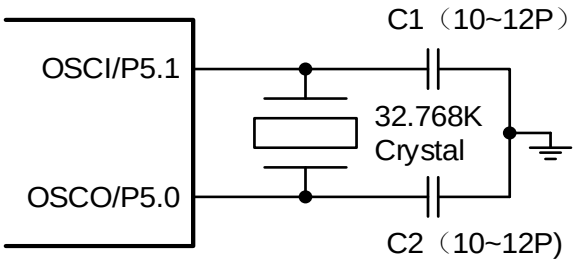
OP_CTM0 (C1h@FFH) Customer Option Register0 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Symbol	ENWDT	ENXTL	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
R/W	R/W	R/W	R/W		R/W	R/W	R/W	
POR	n	n	n		n	n	n	

Bit Number	Bit Symbol	Description
6	ENXTL	External 32KHz crystal option 0: External 32KHz crystal unenabled,P5.0, P5.1 valid and internal LRC invalid. 1: External 32KHz crystal enabled,P5.0, P5.1 invalid and internal LRC

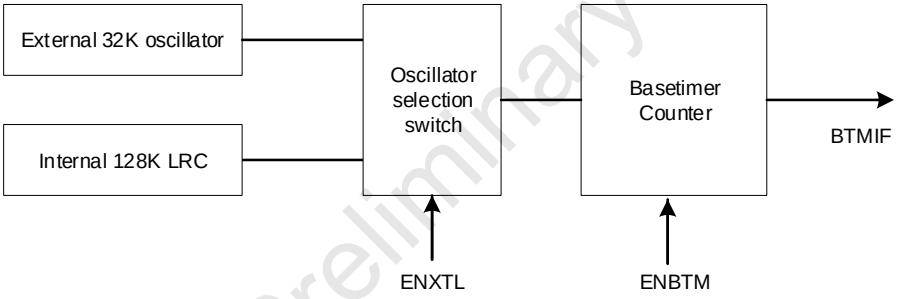
		invalid.

The circuit diagram of the P5.0/P5.1 connect external 32KHz crystal as the Base Timer as follows:



32KHz external crystal connecting circuit diagram

The selection relation diagram of internal and external oscillations of the Base Timer is as follows:



Base Timer block

7.6 STOP Mode and IDLE Mode

The SC92F744XB provides a SFR PCON, the user can configure bit 0 and bit 1 of this register to control MCU to enter different operating modes.

When PCON.1 = 1, internal high-frequency system clock would stop and system enter STOP mode, to save power. The system can be woken up from STOP by external interrupt INT0 ~ INT2, low-frequency clock interrupt, WDT, and external reset input.

When PCON.0 = 1, the program would stop running and System enter IDLE mode. But the external equipment and clock will continue running, CPU will keep all states before entering IDLE mode. The system can be woken up from IDLE by any interrupt.

PCON (87H) Power Management Control Register (only for write, *unreadable*)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit	SMOD	-	-	-	-	-	STOP	IDL

Mnemonic								
R/W	W	-	-	-	-	-	W	W
POR	0	x	x	x	x	x	0	0

Bit Number	Bit Mnemonic	Description
1	**STOP**	STOP mode control bit 0: normal operating mode 1: stop mode, high-frequency oscillator stops operating, low-frequency oscillator and WDT can select to work based on configuration
0	**IDL**	IDLE mode control bit 0: normal operating mode 1: IDLE mode, the program stops operating, but external equipment and clock continue to operate and all CPU states are saved before entering IDLE mode

Notes: When Configure MCU to enter STOP or IDLE mode, the instruction of configuring PCON register should be followed by 8 "NOP" instructions rather than other instructions. Or else, it will be unable to execute following instructions normally after wake-up!

For example, configure MCU to enter STOP mode:

C program example:

```
#include"intrins.h"

    PCON |= 0x02;          //Set to 1 for PCON bit1 STOP bit, configure MCU to enter STOP mode
    _nop_ ();              //At least 8 _nop_ () required
    _nop_ ();
    _nop_ ();
    _nop_ ();
    _nop_ ();
    _nop_ ();
    _nop_ ();
    ……
```

Assembly program example:

```
ORL PCON, #02H   ; Set to 1 for PCON bits1 STOP bit, configure MCU to enter STOP mode
    NOP                    ; At least 8 NOP required
    NOP
    NOP
    NOP
    NOP
    NOP
    NOP
    NOP
    ……
```

8 CPU and Function System

8.1 CPU

CPU used by the SC92F744XB is the high-speed 1T standard 8051 core, whose instructions are completely compatible with traditional 8051 core microcontroller unit.

8.2 Addressing Mode

The addressing mode of the SC92F744XB 1T 8051 CPU instructions includes: ①Immediate Addressing ② Direct Addressing ③ Indirect Address ④ Register Addressing ⑤ Relative Addressing ⑥ Indexed Addressing ⑦ Bit Addressing

8.2.1 Immediate Addressing

Immediate addressing is also called immediate operand addressing, which is the operand given to participate in operation in instruction, the instruction is illustrated as follows:

MOV A, #50H (This instruction is to move immediate operand 50H to Accumulator A)

8.2.2 Direct Addressing

In direct addressing mode, the instruction operand field indicates the address to participate in operation operand. Direct addressing can only be used to address SFRs, internal data registers and bit address space. The SFRs and bit address space can only be accessed by direct addressing. For example:

ANL 50H, #91H (The instruction indicates the data in 50H unit AND immediate operand 91H, and the results are stored in 50H unit. 50H refers to direct address, indicating one unit in internal data register RAM.)

8.2.3 Indirect Addressing

Indirect addressing is expressed as adding “@” before R0 or R1. Suppose the data in R1 is 40H and the data of internal data register 40H unit is 55H, then the instruction will be

MOV A, @R1 (Move the data 55h to Accumulator A).

8.2.4 Register Addressing

Register addressing is to operate the data in the selected registers R7 ~ R0, Accumulator A, general-purpose register B, address registers and carry bit C. The registers R7-R0 is indicated by lower 3 bits of instruction code. ACC, B, DPTR and carry bit C are implied in the instruction code. Therefore, register addressing can also include an implied addressing mode. The selection of register operating area depends on RS1 and RS0 of PSW. The registers indicated by instruction operand refers to the registers in current operating area.

INC R0 refers to (R0) +1→R0

8.2.5 Relative Addressing

The data in the second byte of the instruction, whose result shall be taken as the jump address of jump instruction. The Jump address is the target jump address, the current value in PC is the base address and the data in the

second byte of the instruction is the offset address. Because the target jump address is relative to base address in PC, such addressing mode is called relative addressing. The offset is signed number, which ranges from +127 to -128, such addressing mode is mainly applied to jump instruction.

JC $+50H

It indicates that if the carry bit C is 0, the contents in program counter PC remain the same, meaning no jump. On the contrary, if the carry bit C is 1, take the sum of the current value in PC and base address as well as offset 50H as the target jump address of this jump instruction.

8.2.6 Indexed Addressing

In indexed addressing mode, the instruction operand is to develop an indexed register to store indexed base address. Upon indexed addressing, the result by adding offset and indexed base address is taken as the address of operation operand. The indexed registers include PC and address register DPTR.

MOVC A, @A+DPTR

It indicates Accumulator A is used as offset register. Take the sum of the value in A and that in the address register DPTR as the address of operand. Then take the figure in the address out and transmit it to Accumulator A.

8.2.7 Bits Addressing

Bit addressing is a kind of addressing mode when conducting bit operation on internal data storage RAM and SFRs which are able to carry out bit operations. Upon bit operations, by taking carry bit C as bit operation accumulator, the instruction operand will give the address of this bit directly, then execute bit operation based on the nature of operation code.

MOV C, 20H (Transmit the bit operation register with address of 20H into carry bit C)

9 Interrupt

The SC92F744XB provides 12 interrupt sources: Timer0, Timer1, Timer2, INT0 ~ 2, ADC, PWM, UART, SSI, Base Timer and CMP. These 12 interrupt sources are equipped with 2-level interrupt priority-capability and each interrupt source can be individually configured in high priority or low priority. As for three external interrupts, the triggering condition of each interrupt source can be set as rising edge, falling edge or dual-edge trigger. Each interrupt is equipped with independent priority setting bit, interrupt flag, interrupt vector and enable bit. Global interrupt enable bit EA can enable or disable all interrupts.

9.1 Interrupt Source and Vector

Lists for the SC92F744XB interrupt source, interrupt vector and related control bit are shown below:

Interrupt Source	Interrupt condition	Interrupt Flag	Interrupt Enable Control	Interrupt Priority Control	Interrupt Vector	Query Priority	Interrupt Number (C51)	Flag Clear Mode	Capability of Waking up STOP
INT0	Compliant with External interrupt 0 conditions	IE0	EINT0	IPINT0	0003H	1 (high)	0	H/W Auto	Yes
Timer0	Timer0 overflow	TF0	ET0	IPT0	000BH	2	1	H/W Auto	No
INT1	Compliant with External interrupt 1 conditions	IE1	EINT1	IPINT1	0013H	3	2	H/W Auto	Yes
Timer1	Timer1 overflow	TF1	ET1	IPT1	001BH	4	3	H/W Auto	No
UART	Receiving or transmitting completed	RI/TI	EUART	IPUART	0023H	5	4	Must be cleared by user	No
Timer2	Timer2 overflow	TF2	ET2	IPT2	002BH	6	5	Must be cleared by user	No
ADC	ADC conversion completed	ADCIF	EADC	IPADC	0033H	7	6	Must be cleared by user	No
SSI	Receiving or transmitting completed	SPIF/TWIF	ESSI	IPSPI	003BH	8	7	Must be cleared by user	No
PWM	PWM overflow	PWMIF	EPWM	IPPWM	0043H	9	8	Must be cleared by user	No
BTM	Base timer overflow	BTMIF	EBTM	IPBTM	004BH	10	9	H/W Auto	Yes
INT2	External interrupt 2 conditions	-	EINT2	IPINT2	0053H	11	10	-	Yes

	compliant								
CMP	Comparator interrupt conditions compliant	CMPIF	ECMP	IPCMP	0063H	12	12	Must be cleared by user	Yes

Under the circumstance where the master interrupt control bit EA and the respective interrupt control bit have been enable, the interrupt occurrence is shown below:

Timer Interrupt: Interrupt generates when Timer0 or Timer1 overflows and the interrupt flag TF0 or TF1 is set to "1". When the microcontroller unit responds to the timer interrupt, the interrupt flag TF0 or TF1 is reset automatically by hardware. Interrupt generates when Timer2 overflows and the interrupt flag TF2 is set to "1". Once Timer2 interrupt generates, the hardware would not automatically clear TF2 bit, which must be cleared by the user's software.

UART Interrupt: When UART0 completes receiving or transmitting a frame of data, bit RI or TI will be set to "1" automatically by hardware, and UART interrupt occurs. Once UART interrupt occurs, the hardware would not automatically clear up RI/TI bit, which shall be cleared by user's software.

ADC Interrupt: After ADC conversion is completed, ADC interrupt generates, whose interrupt flag is the ADC conversion completion flag EOC/ADCIF (ADCCON.5). When user starts ADCS conversion, EOC will be reset automatically by hardware. Once conversion completes, EOC would be set to "1" automatically by hardware. User should clear the ADC interrupt flag by software when the interrupt service routine is executed after ADC interrupt generates.

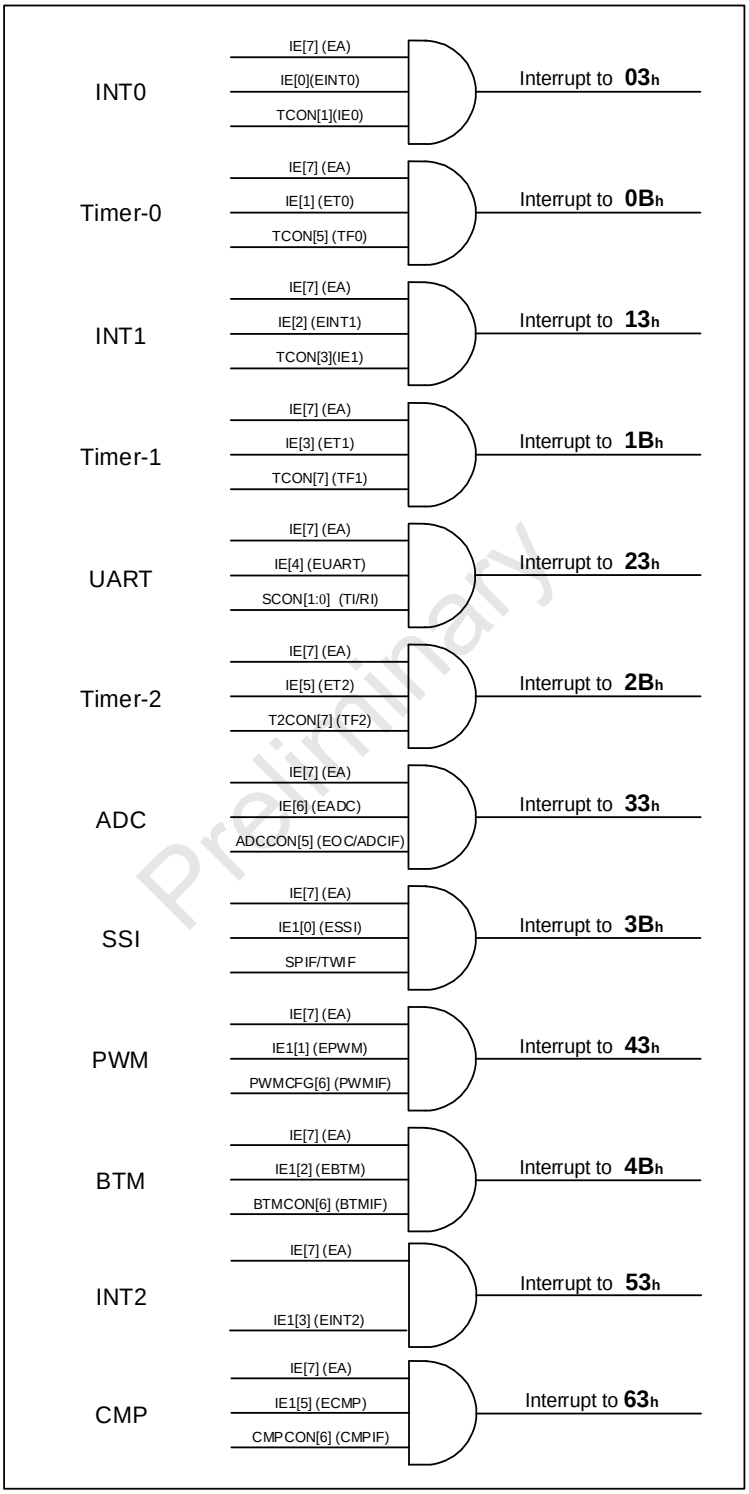
SSI Interrupt: When SSI completes receiving or transmitting a frame of data, SPIF/TWIF bit will be set to "1" automatically by hardware, and SSI interrupt generates. When the microcontroller unit serves SSI interrupt, the interrupt flag SPIF/TWIF must be cleared by software.

PWM Interrupt: When PWM counter overflows (beyond PWMPD), the flag will be set as 1 automatically by hardware. Meanwhile, if the PWM interrupt control bit IE1[1] (EPWM) is set as 1, PWM interrupt will occurs. Once PWM interrupt occurs, the hardware would not clear the interrupt flag automatically, which shall be cleared by user's software.

External Interrupt INT0 ~ 2: When any external interrupt pin meets the interrupt conditions, external interrupt generates. The external interrupt INT0 and INT1 would set up interrupt flag IE0 and IE1 respectively, which will be automatically cleared by hardware rather than user. There are 4 external interrupt sources for INT0, 8 external interrupt sources for INT1 and 4 external interrupt sources for INT2, which can be set in rising edge, falling edge or dual edge interrupt trigger mode by setting SFRs (INTxF and INTxR). User can set the priority level of each interrupt through IP register. Besides, external interrupt INT0 ~ 2 can also wake up STOP mode of microcontroller unit.

9.2 Interrupt Structure Diagram

The SC92F744XB interrupt structure is shown in the figure below:



The SC92F744XB Interrupt Structure and Vector

9.3 Interrupt Priority

The SC92F744XB microcontroller unit has two-level interrupt priority capability. The interrupt requests of these interrupt sources can be programmed as high-priority interrupt or low-priority interrupt, which is to realize the nesting of two levels of interrupt service programs. One interrupt can be interrupted by a higher priority interrupt request when being responded to, which can not be interrupted by another interrupt request at the same priority level, until such response to the first-come interrupt ends up with the instruction “RETI”. Exist the interrupt service routine and return to main program, the system would execute one more instruction before responding to new interrupt request.

That is to say:

① A lower priority interrupt can be interrupted by a higher priority interrupt request, but not vice verse;

② Any kind of interrupt being responded to can not be interrupted by another interrupt request at the same priority level.

Interrupt query sequence: As for the sequence of that the SC92F744XB microcontroller unit responds to the same priority interrupts which occur in the meantime, the priority sequence of interrupt response shall be the same as the interrupt query number in C51, which is to preferentially respond to the interrupt with smaller query number then the interrupt with bigger query number.

9.4 Interrupt Processing Flow

When any interrupt generates and is responded by CPU, the operation of main program will be interrupted to carry out the following operations:

① Complete execution of instruction being currently executed;

② Push the PC value into stack for site protection;

③ Load Interrupt vector address into program counter (PC);

④ Carry out corresponding interrupt service program;

⑤ End Interrupt service program ends and execute RETI;

⑥ Pop PC value from stack and return to the program before responding to the interrupt.

During this process, the system will not immediately respond to other interrupts at the same priority level, but it will keep all interrupt requests having occurred and respond to new interrupt requests upon completing handling of the current interrupt.

9.5 Interrupt-related Registers

IE (A8H) Interrupt Enable Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	EA	EADC	ET2	EUART	ET1	EINT1	ET0	EINT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**EA**	Global interrupt enable control bit 0: Disable all interrupts 1: Enable all interrupts
6	**EADC**	ADC interrupt enable control bit 0: Disable ADC interrupts 1: Interrupt is allowed upon completing ADC conversion
5	**ET2**	Timer2 interrupt enable control bit 0: Disable Timer2 interrupt 1: Enable Timer2 interrupt
TONG4	**EUART**	UART interrupt enable control bit 0: Disable UART interrupt 1: Enable UART interrupt
3	**ET1**	Timer1 interrupt enable control bit 0: Disable Timer1 interrupt 1: Enable Timer1 interrupt

2	**EINT1**	External interrupt 1 enable control 0: Disable INT1 interrupt 1: Enable INT1 interrupt
1	**ET0**	Timer0 interrupt enable control bit 0: Disable Timer0 interrupt 1: Enable Timer0 interrupt
0	**EINT0**	External interrupt 0 enable control bit 0: Disable INT0 interrupt 1: Enable INT0 interrupt

IP (B8H) Interrupt Priority Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	IPADC	IPT2	IPUART	IPT1	IPINT1	IPT0	IPINT0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
6	**IPADC**	ADC interrupt priority selection bit 0: ADC interrupt priority is low 1: ADC interrupt priority is high
5	**IPT2**	Timer2 interrupt priority selection bit 0: Timer2 interrupt priority is low 1: Timer2 interrupt priority is high

4	**IPUART**	UART interrupt priority selection bit 0: UART interrupt priority is low 1: UART interrupt priority is high
3	**IPT1**	Timer1 interrupt priority selection bit 0: Timer1 interrupt priority is low 1: Timer1 interrupt priority is high
2	**IPINT1**	INT1 interrupt priority selection bit 0: INT1 interrupt priority is low 1: INT1 interrupt priority is high
1	**IPT0**	Timer 0 interrupt priority selection bit 0: Timer0 interrupt priority is low 1: Timer0 interrupt priority is high
0	**IPINT0**	INT0 interrupt priority selection bit 0: INT0 interrupt priority is low 1: INT0 interrupt priority is high
7	**-**	Reserved

IE1 (A9H) Interrupt Enable Register 1 (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	ECMP	-	EINT2	EBTM	EPWM	ESSI
R/W	-	-	R/W	-	R/W	R/W	R/W	R/W
POR	x	x	0	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
5	**ECMP**	Analog comparator interrupt enabling control bit 0: Disable analog comparator interrupt 1: Enable analog comparator interrupt
3	**EINT2**	External interrupt 2 enabling control bit 0: Disable External interrupt 2 1: Enable External interrupt 2
2	**EBTM**	Base Timer interrupt enabling control bit 0: Disable Base Timer interrupt 1: Enable Base Timer interrupt
1	**EPWM**	PWM interrupt enabling control bit 0: Disable PWM interrupt 1: Enable interrupt upon PWM counting overflows
0	**ESSI**	Three-in-on serial interrupt enabling control 0: Disable serial port interrupt 1: Enable serial port interrupt
7 ~ 6, 4	**-**	Reserved

IP1 (B9H) Interrupt Priority Control Register 1 (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	IPCMP	-	IPINT2	IPBTM	IPPWM	IPSSI
R/W	-	-	R/W	-	R/W	R/W	R/W	R/W
POR	x	x	0	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
5	**IPCMP**	Analog comparator interrupt priority selection bit 0: Analog comparator interrupt priority is low 1: Analog comparator interrupt priority is high
3	**IPINT2**	INT2 interrupt priority selection bit 0: INT2 interrupt priority is low 1: INT2 interrupt priority is high
2	**IPBTM**	Base Timer interrupt priority selection bit 0: Base Timer interrupt priority is low 1: Base Timer interrupt priority is high
1	**IPPWM**	PWM interrupt priority selection bit 0: PWM interrupt priority is low 1: PWM interrupt priority is high
0	**IPSSI**	Three-in-on serial interrupt priority selection bit 0: SSI interrupt priority is low 1: SSI interrupt priority is high
7 ~ 6, 4	**-**	Reserved

TCON (88H) Timer Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	TF1	TR1	TF0	TR0	IE1	-	IE0	-
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	-
POR	0	0	0	0	0	x	0	x

Bit Number	Bit Mnemonic	Description
3	**IE1**	INT1 overflow interrupt request flag bit When INT1 overflow occurs, interrupt generates, hardware set IE1 to ”1”; when the application is interrupted, upon CPU responds, the hardware resets it to ”0”
1	**IE0**	INT0 overflow interrupt request flag bit When INT1 overflow occurs, interrupt generates, hardware set IE0 to ”1”; when the application is interrupted, upon CPU responds, the hardware resets it to ”0”
2, 0	**-**	Reserved

INT0F (BAH) INT0 Falling Edge Interrupt Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	INT0F7	INT0F6	INT0F5	INT0F4	-	-	-	-
R/W	R/W	R/W	R/W	R/W	-	-	-	-
POR	0	0	0	0	x	x	x	x

Bit Number	Bit Mnemonic	Description
7 ~ 4	**INT0Fn** **(n=7 ~ 4)**	INT0 falling edge interrupt control bit 0: INT0n falling edge interrupt off 1: INT0n falling edge interrupt enabling
3 ~ 0	**-**	Reserved

INT0R (BBH) INT0 Rising Edge Interrupt Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	INT0R7	INT0R6	INT0R5	INT0R4	-	-	-	-
R/W	R/W	R/W	R/W	R/W	-	-	-	-
POR	0	0	0	0	x	x	x	x

Bit Number	Bit Mnemonic	Description
7 ~ 4	**INT0Rn** **(n=7 ~ 4)**	INT0 rising edge interrupt control bit 0: INT0n rising edge interrupt off 1: INT0n rising edge interrupt enabling
3 ~ 0	**-**	Reserved

INT1F (BCH) INT1 Falling Edge Interrupt Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	INT1F7	INT1F6	INT1F5	INT1F4	INT1F3	INT1F2	INT1F1	INT1F0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**INT1Fn** **(n=7 ~ 0)**	INT1 falling edge interrupt control bit 0: INT1n falling edge interrupt off 1: INT1n falling edge interrupt enabling

INT1R (BDH) INT1 Rising Edge Interrupt Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	INT1R7	INT1R6	INT1R5	INT1R4	INT1R3	INT1R2	INT1R1	INT1R0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**INT1Rn** **(n=7 ~ 0)**	INT1 rising edge interrupt control bit 0: INT1n rising edge interrupt off 1: INT1n rising edge interrupt enabling

INT2F (C6H) INT2 Falling Edge Interrupt Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	-	INT2F3	INT2F2	INT2F1	INT2F0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
POR	x	x	x	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
3 ~ 0	**INT2Fn** **(n=3 ~ 0)**	INT2 falling edge interrupt control bit 0: INT2n falling edge interrupt off 1: INT2n falling edge interrupt enabling
7 ~ 4	**-**	Reserved

INT2R (C7H) INT2 Rising Edge Interrupt Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	-	INT2R3	INT2R2	INT2R1	INT2R0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
POR	x	x	x	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
3 ~ 0	**INT2Rn**	INT2 rising edge interrupt control bit

	(n=3 ~ 0)	0: INT2n rising edge interrupt off 1: INT2n rising edge interrupt enabling
7 ~ 4	**-**	Reserved

10 Timer/Counter T0 and T1

The SC92F744XB has two 16-bit Timer/Counters, Timer0 (T0) and Time1 (T1), with two operating modes: counter mode and timer mode. The operating modes selected by bit C/Tx in the SFR TMOD. T0 and T1 are essentially adding counters with different counting source. The source of timer generated from system clock or frequency division clock, but the source of counters is the input pulse to external pin. Only when TRx = 1, will T0 and T1 be enabled on for counting.

In counter mode, each input pulse on P0.2/T0 and P0.3/T1 pin will make the count value of T0 and T1 increase by 1 respectively.

In timer mode, users can select $f_{SYS}/12$ or f_{SYS} (f_{SYS} is the system clock after frequency division) as counting source of T0 and T1 by configuring SFR TMCON.

Timer/Counter T0 has 4 operating modes, and Timer/Counter T1 has 3 operating modes (Mode 3 does not exist):

① Mode 0: 13-bit Timer/Counter mode

② Mode 1: 16-bit Timer/Counter mode

③ Mode 2: 8-bit automatic reload mode

④ Mode 3: Two 8-bit timers/counters mode

In above modes, modes 0, 1 and 2 of T0 and T1 are the same, and mode 3 is different.

10.1 T0 and T1-related Registers

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
TCON	88H	Timer Control Register	TF1	TR1	TF0	TR0	IE1	-	IE0	-	00000x0xb
TMOD	89H	Timer Operating Mode Register	-	C/T1	M11	M01	-	C/T0	M10	M00	x000x000b
TL0	8AH	Timer0 Low byte	TL0[7: 0]								00000000b
TL1	8BH	Timer1 Low byte	TL1[7: 0]								00000000b
TH0	8CH	Timer0 High byte	TH0[7: 0]								00000000b

TH1	8DH	Timer1 High byte	TH1[7: 0]								00000000b
TMCON	8EH	Timer Frequency Control Register	-	-	-	-	-	T2FD	T1FD	T0FD	xxxxx000b

Register instructions are shown below:

TCON (88H) Timer Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	TF1	TR1	TF0	TR0	IE1	-	IE0	-
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	-
POR	0	0	0	0	0	x	0	x

Bit Number	Bit Mnemonic	Description
7	**TF1**	Timer1 overflow flag bit Set by hardware on Timer/Counter overflow. Cleared by hardware when the processor vectors to interrupt routine.
6	**TR1**	Timer1 run control bit Set/cleared by software to turn Timer/Counter on/off.
5	**TF0**	Timer0 overflow flag bit Set by hardware on Timer/Counter overflow. Cleared by hardware when the processor vectors to interrupt routine.
4	**TR0**	Timer0 run control bit Set/cleared by software to turn Timer/Counter on/off.
2, 0	**-**	Reserved

TMOD (89H) Timer Operating Mode Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	C/T1	M11	M01	-	C/T0	M10	M00
R/W	-	R/W	R/W	R/W	-	R/W	R/W	R/W
POR	x	0	0	0	x	0	0	0
	T1				T0			

Bit Number	Bit Mnemonic	Description
6	**C/T1**	Timer or Counter selector 1 0: Cleared for Timer operation (input from internal system clock fsys). 1: Set for Counter operation (input from external pin T1/P0.3).
5 ~ 4	**M11, M01**	Timer1 operating mode

Mode	M11	M01	Operation
0	0	0	13-bit TIMER/Counter, TL1 high 3 bits invalid
1	0	1	16-bit Timer/Counter
2	1	0	8-bit Auto-Reload Mode. TH1 holds a value which is reloaded into 8-bit Timer/CounterTL1 each time it overflows.
3	1	1	Timer/Counter 1 is stopped

2	**C/T0**	Timer or Counter selector 0 0: Cleared for Timer operation (input from internal system clock fsys).

<table>
<tr><td></td><td></td><td>1: Set for Counter operation (input from external pin T1/P0.2).</td></tr>
<tr><td>1 ~ 0</td><td>M10, M00</td><td>Timer0 operating mode
<table>
<tr><td>Mode</td><td>M10</td><td>M00</td><td>Operation</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>13-bit TIMER/Counter, TL0 high 3 bits invalid</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>16-bit Timer/Counter</td></tr>
<tr><td>2</td><td>1</td><td>0</td><td>8-bit Auto-Reload Mode.

TH0 holds a value which is reloaded into 8-bit Timer/Counter TL0 each time it overflows.</td></tr>
<tr><td>3</td><td>1</td><td>1</td><td>Split Timer Mode.

TL0 is an 8-bit Timer/Counter controlled by the standard Timer0 control bits. TH0 is only an 8-bit timer controlled by Timer1 control bits</td></tr>
</table></td></tr>
<tr><td>7, 3</td><td>-</td><td>Reserved</td></tr>
</table>

TMOD[0] ~ TMOD[2] of TMOD register is to set operating mode of T0; TMOD[4] ~ TMOD[6] is to set the operating mode of T1.

The function of timer and counter Tx is selected by the control bit C/Tx of SFR TMOD, and it's operating mode selected by M0x and M1x. Only when TRx, the switch of T0 and T1, is set to 1, will T0 and T1 be enabled

TMCON (8EH) Timer Frequency Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	-	-	-	T2FD	T1FD	T0FD
R/W	-	-	-	-	-	R/W	R/W	R/W
POR	x	x	x	x	x	0	0	0

Bit Number	Bit Mnemonic	Description
1	**T1FD**	T1 input frequency selection control bit 0: T1 clock source is f_{SYS}/12 1: T1 clock source is f_{SYS}
0	**T0FD**	T0 input frequency selection control bit 0: T0 clock source is f_{SYS}/12 1: T0 clock source is f_{SYS}

IE (A8H) Interrupt Enable Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	EA	EADC	ET2	-	ET1	EINT1	ET0	EINT0
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
POR	0	0	0	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
3	**ET1**	Timer1 interrupt enable control bit 0: Disable Timer1 interrupt 1: Enable Timer1 interrupt
1	**ET0**	Timer0 interrupt enable control bit 0: Disable Timer0 interrupt 1: Enable Timer0 interrupt

IP (B8H) Interrupt Priority Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	IPADC	IPT2	IPUART	IPT1	IPINT1	IPT0	IPINT0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
3	**IPT1**	Timer1 interrupt priority selection bit 0: Configure Timer1 interrupt priority as “low” 1: Configure Timer1 interrupt priority as ”high”
1	**IPT0**	Timer0 interrupt priority selection bit 0: Configure Timer0 interrupt priority as “low” 1: Configure Timer0 interrupt priority as “high”

10.2 T0 Operating Modes

Timer0 can be configured in one of four operating modes by setting the bit pairs (M10, M00) in the TMOD register.

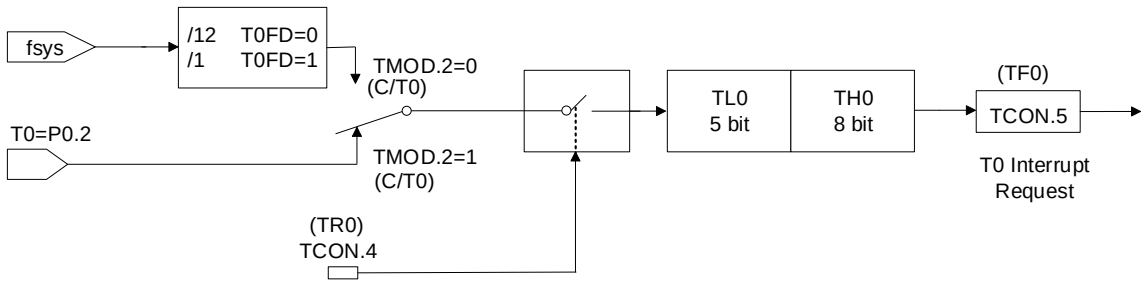
Operating Mode 0: 13-bit Timer/Counter

TH0 register is to store the high 8 bits (TH0.7 ~ TH0.0) of 13-bit Timer/Counter and TL0 is to store the low 5 bits (TL0.4 ~ TL0.0). The high three bits of TL0 (TL0.7 ~ TL0.5) are filled with uncertain values, they shall be omitted upon reading. When 13-bit Timer/Counter overflows with count increment, the system will set timer overflow flag TF0 to 1. An interrupt will be generated if the timer0 interrupt is enabled.

C/T0 bit selects the clock input source of Timer/Counter. If C/T0=1, the level fluctuation from high to low of Counter 0 input pin T0 (P0.2) will make Counter 0 data register add 1. If C/T0=0, the frequency division of system clock is the clock source of Timer0.

When TR0 = 1, Timer 0 is enabled. Setting TR0 would not reset the timer forcibly. It means that the timer register will start to count from the value of last clearing of TR0. Therefore, before enable the timer, it is required to configure the initial value of timer register.

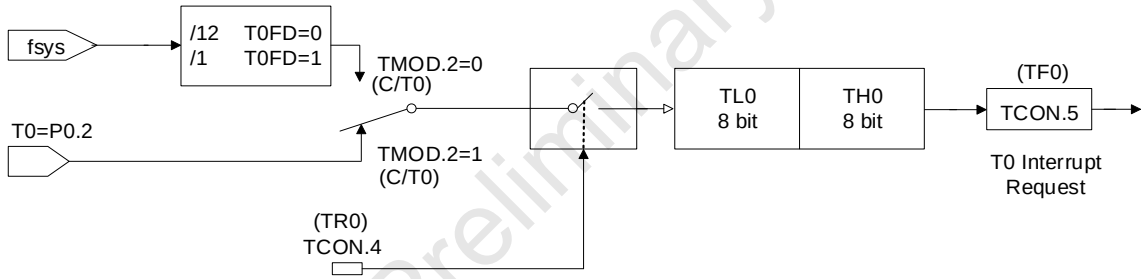
When configured as a timer, the SFR T0FD is used to select fractional frequency ratio of clock source.



Operating mode 0: 13-bit Timer/Counter

Operating Mode 1: 16 Counter/Timer

Except for using 16 bits of (valid for all 8 bits of TL0) Timer/Counter, in mode 1 and mode 0, the operating mode, opening and configuration method are the same.



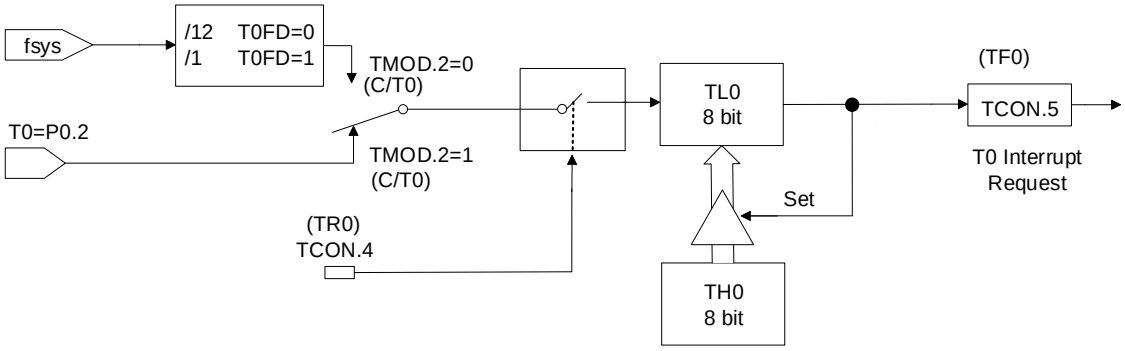
Operating mode 0: 16-bit Timer/Counter

Operating Mode 2: 8 Automatic Reload Counter/Timer

In operating mode 2, Timer0 is 8-bit automatic reload Timer/Counter. TL0 is to store counting value and TH0 is to store the reload value. When the counter in TL0 overflows and turn to 0x00, the overflow flag of Timer TF0 will be set to 1, and the data in register TH0 will be reloaded into register TL0. If the timer interrupt enabled, setting TF0 to 1 will generate an interrupt, but the reloaded value in TH0 will remain the same. Before starting the Timer to count correctly, TL0 shall be initialized to the required value.

Except for automatic reloaded function, the enabling and configuration mode of Timer/Counter in operating mode 2 shall be the same as that in mode 0 and mode 1.

When configured as a timer, the SFR TMCON bit 0 (T0FD) is used to select fractional frequency ratio of system clock f_{SYS}.



Operating Mode 2: 8 Automatic Reload Counter/Timer

Operating Mode 3: Two 8-bit Counter/Timer (only for Timer0)

In operating mode 3, Timer0 is used as two independent 8-bit Timer/Counters, respectively controlled by TL0 and TH0. TL0 is controlled by control bit (in TCON) and status bit (in TMOD) of Timer0 (TR0), C/T0, TF0. Timer0 is selected as Timer or Counter by TMOD bit 2 (C/T0).

TH0 is only limited to in Timer Mode, which is unable to configure as a Counter by TMOD.2 (C/T0). TH0 is enabled by set the timer control bit TR1 to 1. When overflow occurs and interrupt is discovered, set TF1 to 1 and proceed the interrupt as T1 interrupt.

When T0 is configured in Operating Mode 3, TH0 Timer occupies T1 interrupt resources and TCON register and the 16-bit counter of T1 will stop counting, equivalently “TR1=0”. When adopting TH0 timer, it is required to configure TR1=1.

10.3 T1 Operating Modes

Timer1 can be configured in one of three operating modes by setting the bit pairs (M11, M01) in the TMOD register.

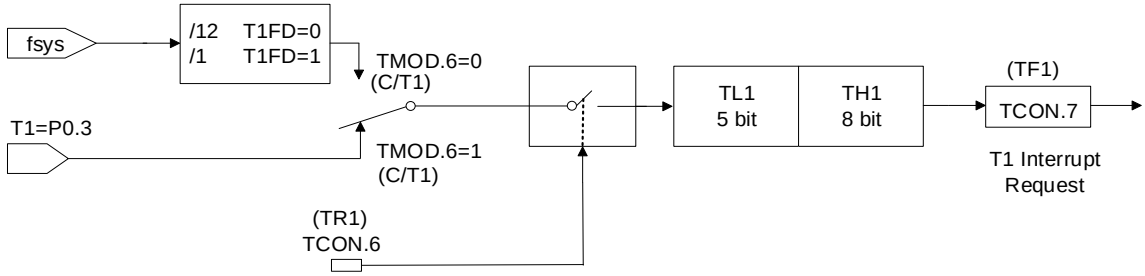
Operating Mode 0: 13-bit Timer/Counter

TH1 register is to store high 8-bit (TH1.7 ~ TH1.0) of 13-bit Timer/Counter and TL1 is to store low 5-bit (TL1.4 ~ TL1.0). The high 3-bit of TL1 (TL1.7 ~ TL1.5) are uncertain values, they shall be omitted upon reading. When 13-bit Timer/Counter overflow with count increment, the system will set timer overflow flag TF1 as1. An interrupt will be generated if the timer1 interrupt is enabled. C/T1 bit selects the clock input source of Timer/Counter.

If C/T1=1, the level fluctuation from high to low of timer1 input pin T1 (P0.3) will make timer1 data register add 1. If C/T1=0, the frequency division of system clock is the clock source of timer1.

When TR1 is set to 1 and the timer is enabled. Setting TR1 does not force to reset timer counters, it means, if set TR1 to 1, the timer register will start to count from the value of last clearing of TR1. Therefore, before allowing timer, it is required to configure the initial value of timer register.

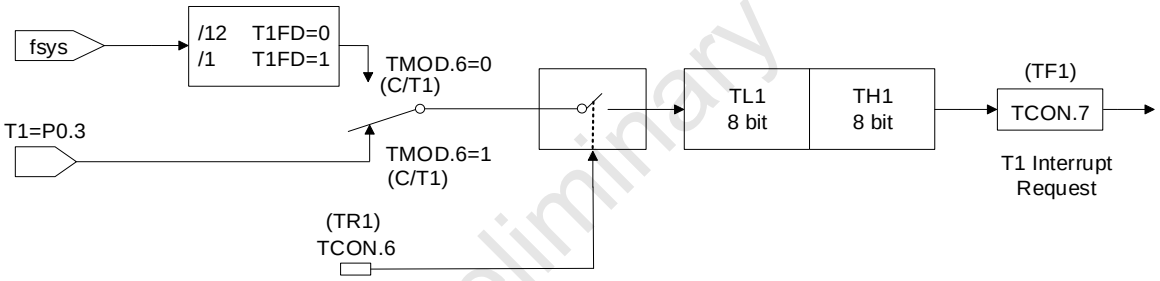
When configured as timer, the SFR T1FD is used to select fractional frequency ratio of clock source.



Operating mode 0: 13-bit Timer/Counter

Operating Mode 1: 16 Counter/Timer

Except for using 16-bit (valid for 8-bit data of TL1) Timer/Counter, the operating mode of mode 1 and mode 0 is the same. And the opening and configuration mode of both are also the same.



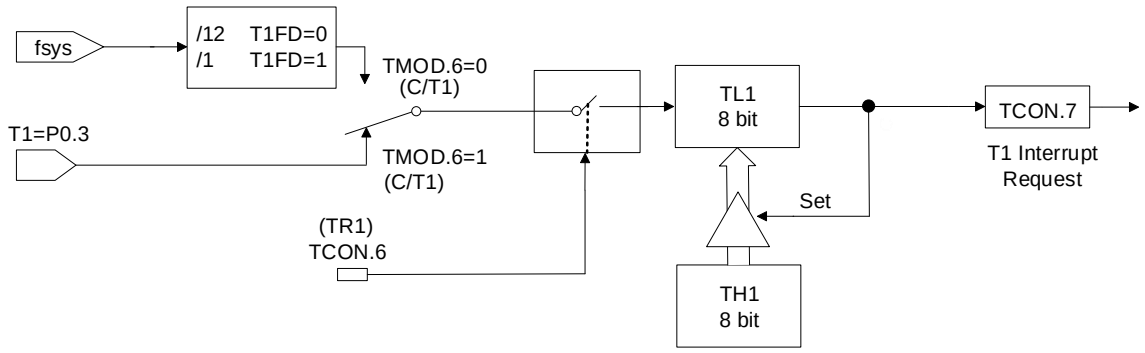
Operating mode 0: 16-bit Timer/Counter

Operating Mode 2: 8 Automatic Reload Counter/Timer

In operating mode 2, Timer1 is 8-bit automatic reload Timer/Counter. TL1 is to store counting value and TH1 is to store the reload value. When the counter in TL1 overflows 0x00, the overflow flag of Timer TF1 will be set to 1, and the value of register TH1 will be reloaded into register TL1. If enable the timer interrupt, setting TF1 to 1 will generate an interrupt, but the reloaded value in TH1 will remain unchanged. Before allowing Timer to correctly count, TL1 shall be initialized to the required value.

Except for automatic reloaded function, the enabling and configuration mode of Timer/Counter in operating mode 2 shall be the same as that of mode 0 and mode 1.

When configured as timer, the SFR TMCON bit 4 (T1FD) is used to select the ratio of clock source of timer to fractional frequency of system clock f_{SYS}.



Operating Mode 2: 8 Automatic Reload Counter/Timer

11 Timer/Counter T2

Timer2 inside the SC92F744XB microcontroller unit has two operating modes, namely counter mode and timer mode. There is a control bit C/T2 in SFR T2CON to select Timer or Counter for T2. They are adding counters in nature, differing in counting source. The clock source of T2 comes from system clock or frequency division clock, but the source of counters is the input pulse to external pin. TR2 is the counting switch of Timer/Counter T2. Only when TR2 = 1, will T2 be enabled for counting.

In counter mode, each input pulse on T2 pin will make the counting value of T2 increase by 1.

In timer mode, users can select $f_{SYS}/12$ or f_{SYS} as counting source of T2 by configuring SFR TMCON.

Timer/Counter T2 has 4 operating modes:

① Mode 0: 16-bit capture mode

② Mode 1: 16-bit automatic reload timer mode

③ Mode 2: Baud rate generator mode

④ Mode 3: Programmable clock output mode

11.1 T2-related Registers

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
T2CON	C8H	Timer2 Control Register	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2	00000000b
T2MOD	C9H	Timer2 Operating Mode Register	-	-	-	-	-	-	T2OE	DCEN	xxxxxx00b
RCAP2L	CAH	Timer2 Reload Low Byte	RCAP2L[7: 0]								00000000b
RCAP2H	CBH	Timer2 Reload High Byte	RCAP2H[7: 0]								00000000b
TL2	CCH	Timer2 Low Byte	TL2[7: 0]								00000000b
TH2	CDH	Timer2 High Byte	TH2[7: 0]								00000000b
TMCON	8EH	Timer Frequency Control Register	-	-	-	-	-	T2FD	T1FD	T0FD	xxxxx000b

Register instructions are shown below:

T2CON (C8H) Timer2 Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**TF2**	Timer2 overflow flag bit 0: No overflow (must be cleared by software) 1: Overflow (if RCLK=0 and TCLK=0, set to 1 by hardware)
6	**EXF2**	T2EX pin external event input (falling edge) detected flag bit 0: No external event input (must be cleared by software) 1:When detecting external input (if EXEN2=1, SET to 1 by hardware)
5	**RCLK**	UART0 receiving clock control bit 0: Timer1 generates receiving baud rate 1: Timer2 generates receiving baud rate
4	**TCLK**	UART0 transmitting clock control bit 0: Timer1 generates transmitting baud rate 1: Timer2 generates transmitting baud rate
3	**EXEN2**	External event input (falling edge) on T2EX pin used as reload/capture trigger allowed/prohibited control bit 0: Omit event on T2EX pin 1: When the timer2 is not used as UART0 clock, a falling edge is

		detected on T2EX pin and a capture or reload will be generated.
2	**TR2**	Timer2 start/stop control bit 0: Stop Timer2 1: Start Timer2
1	**C/T2**	Timer2 Timer/Counter mode selection bit 2 0: Timer mode, used as I/O interface on T2 pin 1: Counter mode
0	**CP/RL2**	Capture/reload mode selection bit 0: 16-bit Timer/Counter with reload function 1: 16-bit Timer/Counter with capture function, T2EX as timer2 external capture signal input port

T2MOD (C9H) Timer2 Operating Mode Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	-	-	-	-	T2OE	DCEN
R/W	-	-	-	-	-	-	R/W	R/W
POR	x	X	x	x	x	x	0	0

Bit Number	Bit Mnemonic	Description
1	**T2OE**	Timer2 output allow bit 0: Set T2 as clock input or I/O port 1: Set T2 as clock output

0	**DCEN**	Decreasing counting allow bit 0: Prohibits Timer2 as incremental/decreasing counter, Timer2 only used as incremental counter 1: Allow Timer2 as incremental/decreasing timer
7 ~ 2	**-**	Reserved

TMCON (8EH) Timer Frequency Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	-	-	-	T2FD	T1FD	T0FD
R/W	-	-	-	-	-	R/W	R/W	R/W
POR	x	x	x	x	x	0	0	0

Bit Number	**Bit Mnemonic**	Description
2	**T2FD**	T2 input frequency selection control bit 0: T2 clock source is f_{SYS}/12 1: T2 clock source is f_{SYS}

IE (A8H) Interrupt Enable Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	EA	EADC	ET2	EUART	ET1	EINT1	ET0	EINT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
5	**ET2**	Timer2 interrupt enable control bit 0: Disable TIMER2 interrupt 1: Enable TIMER2 interrupt

IP (B8H) Interrupt Priority Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	IPADC	IPT2	IPUART	IPT1	IPINT1	IPT0	IPINT0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
5	**IPT2**	Timer2 interrupt priority selection bit 0: Configure Timer2 interrupt priority as “low” 1: Configure Timer2 interrupt priority as “high”

11.2 T2 Operating Modes

The operating mode and configuration mode of Timer2 are shown in the table below:

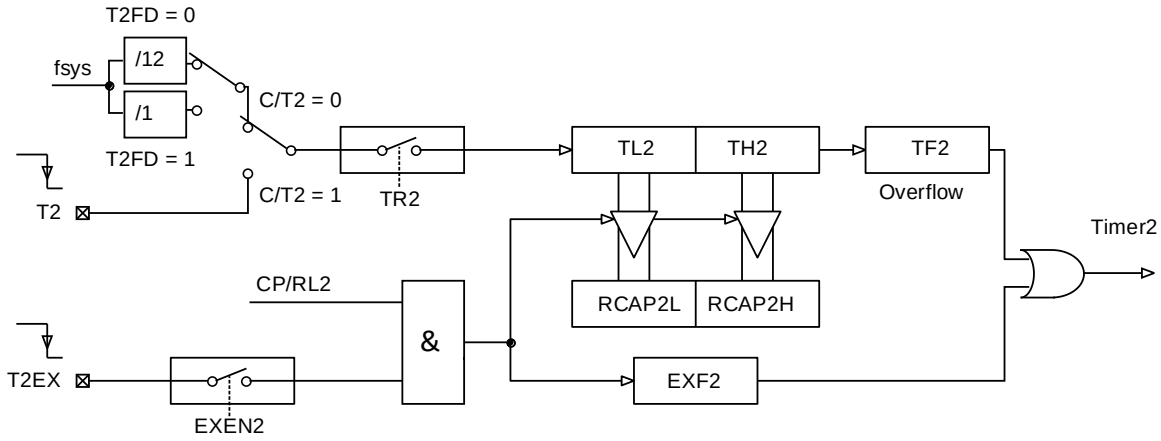
$\overline{C/T2}$	T2OE	DCEN	TR2	$\overline{CP/RL2}$	RCLK	TCLK	Mode	
X	0	X	1	1	0	0	0	16-bit capture 16-bit capture 16-bit capture
X	0	0	1	0	0	0	1	16-bit automatic reload timer
X	0	1	1	0	0	0		
X	0	X	1	X	1	X	2	Baud Rate Generator
					X	1		
0	1	X	1	X	0	0	3	Only used for programmable clock
					1	X	3	Programmable clock output with baud rate generator
					X	1		
X	X	X	0	X	X	X	X	Timer2 stops, but T2EX channel is also available
1	1	X	1	X	X	X		Not recommended

Operating Mode 0: 16-bit capture

In capture mode, there are two options for EXEN2 bit in T2CON.

If EXEN2 = 0, Timer2 is taken as 16-bit timer or counter; if ET2 is set to 1, Timer2 will set up TF2 and generate an interrupt when Timer2 overflows.

If EXEN2=1, conduct the same operations as above on Timer2, the falling edge signal on external input T2EX can make current value in TH2 and TL2 captured into RCAP2H and RCAP2L. Besides, the falling edge signal on T2EX can also cause EXF2 in T2CON to be set to 1. If ET2 is set to 1, bit EXTF2, like TF2, will also trigger an interrupt.



Operating Mode 0: 16-bit capture

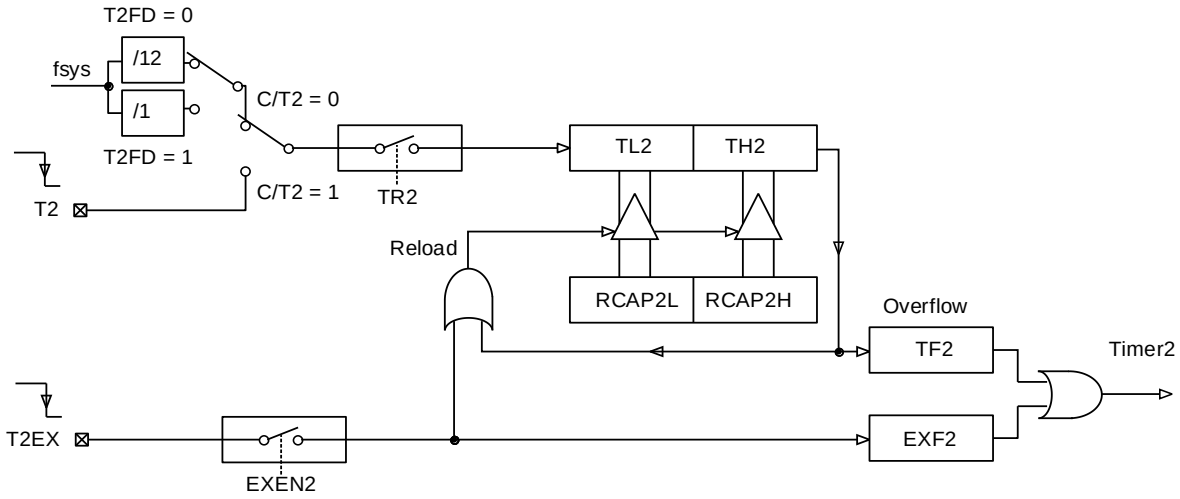
Operating Mode 1: 16-bit Automatic Reload Timer

In the 16-bit automatic reload mode, Timer2 can be selected to work in incrementing or decreasing counting mode. This function can be selected by DCEN bit in T2MOD (decreasing counting allowed). After system reset, the reset value of DCEN bit is 0 and Timer2 is defaulted as decreasing counting. When setting DCEN to 1, the incrementing or decreasing counting depends on the level of T2EX pin.

When DCEN = 0, There are two options for EXEN2 bit in T2CON:

1. EXEN=0, Timer2 will increase to 0xFFFFH and set TF2 bit after overflow. Meanwhile, the timer will load 16-bit value in registers RCAP2H and RCAP2L written by user software into registers TH2 and TL2 automatically.

2. EXEN2=1, both the overflow and the falling edge signal on external input T2EX can trigger a 16-bit count value reloading and set EXF2 bit. If T2 interrupt is enabled (ET2=1), both TF2 and EXF2 bit can generate an interrupt.



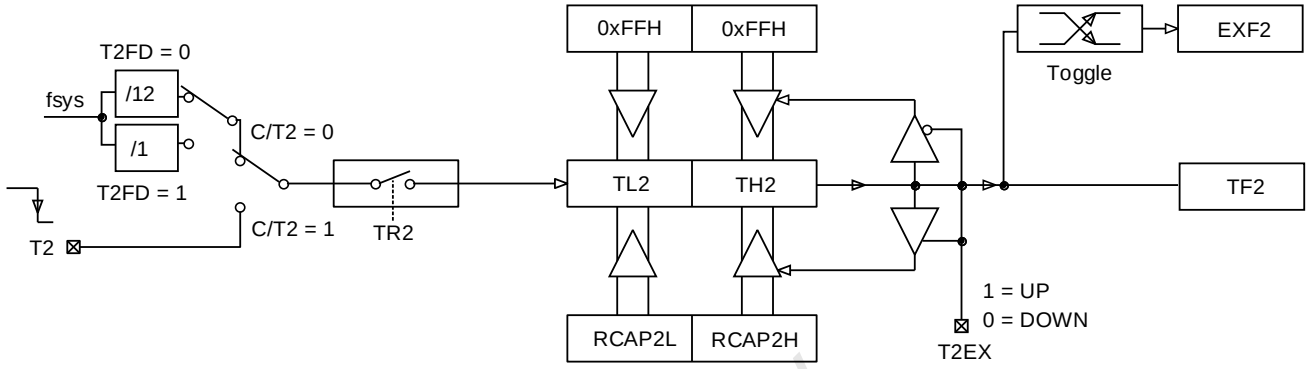
Operating Mode 1: 16-bit Automatic Reload Timer DCEN = 0

Configure CEN bit to allow Timer2 for incremental or decreasing counting. When DCEN=1, T2EX pin controls the count direction, and the control of EXEN2 becomes invalid.

Setting T2EX to 1 can conduct incremental count on Timer2. The Timer overflows when it increases to 0xFFFFH, then it sets TF2 bit. Besides, the overflow can also respectively cause 16-bit value in RCAP2H and RCAP2L to be reloaded into timer registers.

Setting T2EX to 0 can conduct decreasing count on Timer2. When the value in TH2 and TL2 is equal to that of RCAP2H and RCAP2L, the timer overflows. TF2 bit will be set up and 0xFFFFH reloaded into timer register.

No matter whether timer2 overflows or not, bit EXF2 will be used as the 17th bit of the results. Under such operating mode, EXF2 is no longer taken as interrupt flag.



Operating Mode 1: 16-bit Automatic Reload Timer DCEN = 1

Operating Mode 2: Baud Rate Generator

Configure TCLK and RCLK in T2CON register to select Timer2 as baud rate generator. The baud rate of receiver and transmitter can be different. If Timer2 is taken as either one between receiver and transmitter, Timer1 will be taken as another Baud-rate register.

Configure TCLK and RCLK in T2CON register to make Timer2 in baud rate generator mode. Such mode is similar to automatic reload mode

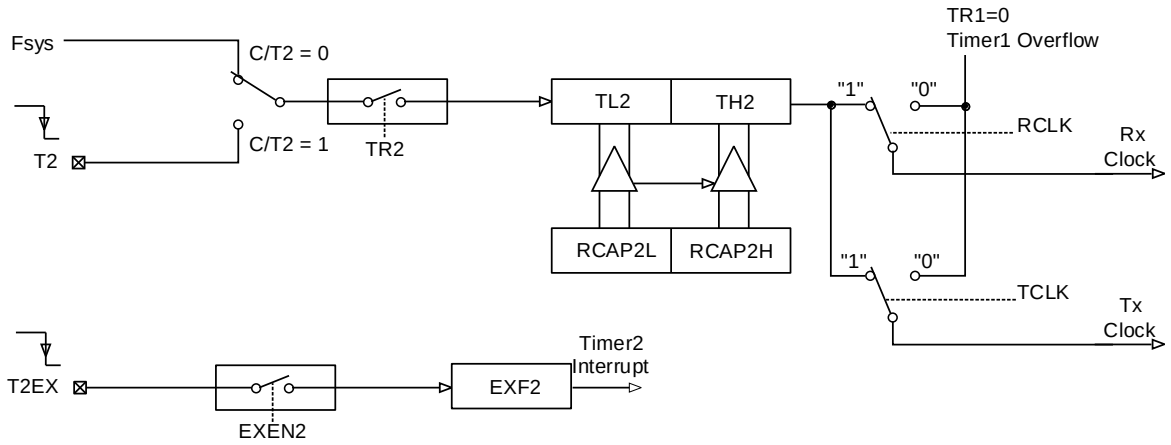
Overflow of Timer2 can make the value in registers RCAP2H and RCAP2L reloaded into the Timer2 and counting, but no interrupt will occur.

If EXEN2 is set to 1, the falling edge on T2EX pin will be set up EXF2 without a reloading. Therefore, when Timer2 is taken as baud rate generator, T2EX can be taken as an additional external interrupt

The baud rate of UART0 mode 1 and mode 3 depends on overflow rate of Timer2 and the following formula:

$$\text{BaudRate} = \frac{\text{fsys}}{[\text{RCAP2H,RCAP2L}]}$$; (note: [RCAP2H, RCAP2L] must be larger than 0x0010)

The schematic diagram of Timer2 as baud rate generator is shown as follows:



Mode 2: Baud Rate Generator

Operating Mode 3: Programmable Clock Output

In this mode, T2(P0.5) can be programmed to output a 50% duty cycle clock: when C/$\overline{T2}$ = 0 and T2OE = 1, Timer2 is taken as clock generator

In this mode, duty cycle of T2 output clock is 50%

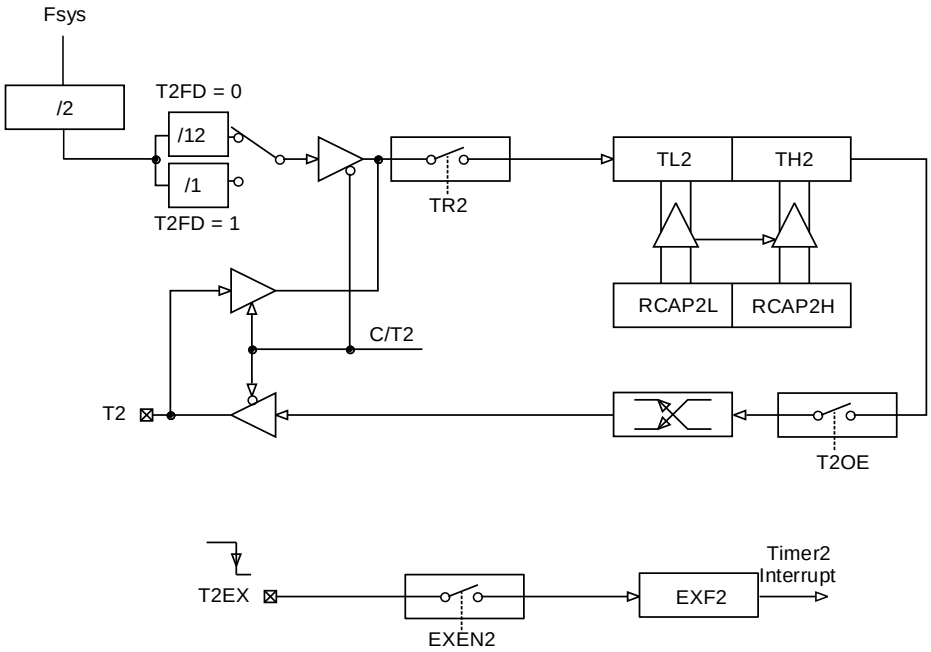
$$\text{Colck Out Frequency} = \frac{\text{fn2}}{(65536-[\text{RCAP2H,RCAP2L}])\times 4};$$

Including, fn2 is the clock frequency of Timer2

$$\text{fn2} = \frac{\text{fsys}}{12}; \quad \text{T2FD = 0}$$

$$\text{fn2} = \text{fsys}; \quad \text{T2FD = 1}$$

Overflow of Timer2 does not generate an interrupt, T2 pin is clock output.



Operating Mode 3: Programmable Clock Output

Note:

1. Both TF2 and EXF2 can generate interrupt request of Timer2, both of which has the same interrupt vector;

2. TF2 and EXF2 can be set by software, only software and hardware reset can clear TF2 and EXF2;

3. When EA = 1 and ET2 = 1, setting up TF2 or EXF2 to 1 can arouse interrupt of Timer2;

4. When Timer2 is taken as baud rate generator, the value written in TH2/TL2 or RCAP2H/RCAP2L may influence the accuracy of baud rate and thus result in error of communication.

12 Multiplier-Divider Unit (MDU)

The SC92F744XB provides a 16-bit multiplier-divider, which is composed of extended accumulator EXA0 ~ EXA3, extended B register EXB and operation control register OPERCON. It can replace the software 16-bit*16-bit multiply operation and 32-bit /16-bit division operation.

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
EXA0	E9H	Extended Accumulator 0	EXA [7: 0]								00000000b
EXA1	EAH	Extended Accumulator 1	EXA [15: 8]								00000000b
EXA2	EBH	Extended Accumulator 2	EXA [23: 16]								00000000b
EXA3	ECH	Extended Accumulator 3	EXA [31: 24]								00000000b
EXBL	EDH	Extended B Register L	EXB [7: 0]								00000000b
EXBH	EEH	Extended B Register H	EXB [15: 8]								00000000b

OPERCON (EFH) Arithmetic Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	OPERS	MD	-	-	-	-	-	CHKSUMS
R/W	R/W	R/W	-	-	-	-	-	R/W
POR	0	0	x	x	X	x	x	0

Bit Number	Bit Mnemonic	Description
7	**OPERS**	Multiplier and divider operation trigger control bit (Operator Start) Set to start a new multiply-divide operation, this bit is only the trigger signal calculated with multiplier, when this bit is zero, the calculation is

<table>
<tr><td></td><td></td><td>completed. This bit is only valid for writing 1.</td></tr>
<tr><td>6</td><td>MD</td><td>Multiplier and divider selection bit

0: Multiply operation, writing of multiplicand and multiplier and reading of product are shown below:
<table>
<tr><th>Byte
Operations</th><th>Byte 3</th><th>Byte 2</th><th>Byte 1</th><th>Byte 0</th></tr>
<tr><td>multiplicand 16bits</td><td>-</td><td>-</td><td>EXA1</td><td>EXA0</td></tr>
<tr><td>multiplier 16bits</td><td>-</td><td>-</td><td>EXBH</td><td>EXBL</td></tr>
<tr><td>product 32bits</td><td>EXA3</td><td>EXA2</td><td>EXA1</td><td>EXA0</td></tr>
</table>
1: Division operation: writing of dividend and divisor and reading of quotient and remainder are shown below:
<table>
<tr><th>Byte
Operations</th><th>Byte 3</th><th>Byte 2</th><th>Byte 1</th><th>Byte 0</th></tr>
<tr><td>dividend 32bits</td><td>EXA3</td><td>EXA2</td><td>EXA1</td><td>EXA0</td></tr>
<tr><td>divisor 16bits</td><td>-</td><td>-</td><td>EXBH</td><td>EXBL</td></tr>
<tr><td>quotient 32bits</td><td>EXA3</td><td>EXA2</td><td>EXA1</td><td>EXA0</td></tr>
<tr><td>remainder 16bits</td><td>-</td><td>-</td><td>EXBH</td><td>EXBL</td></tr>
</table></td></tr>
</table>

Note:

1. **During the operation process, it is forbidden to read or write EXA and EXB data registers.**
2. **The time for operation conversion of multiplier is 16/f_{SYS}.**

13 PWM

The SC92F744XB provides an independent counter, which is able to support 8-channel PWM output: PWM0 ~ 7. The SC92F744XB PWM has the following functions:

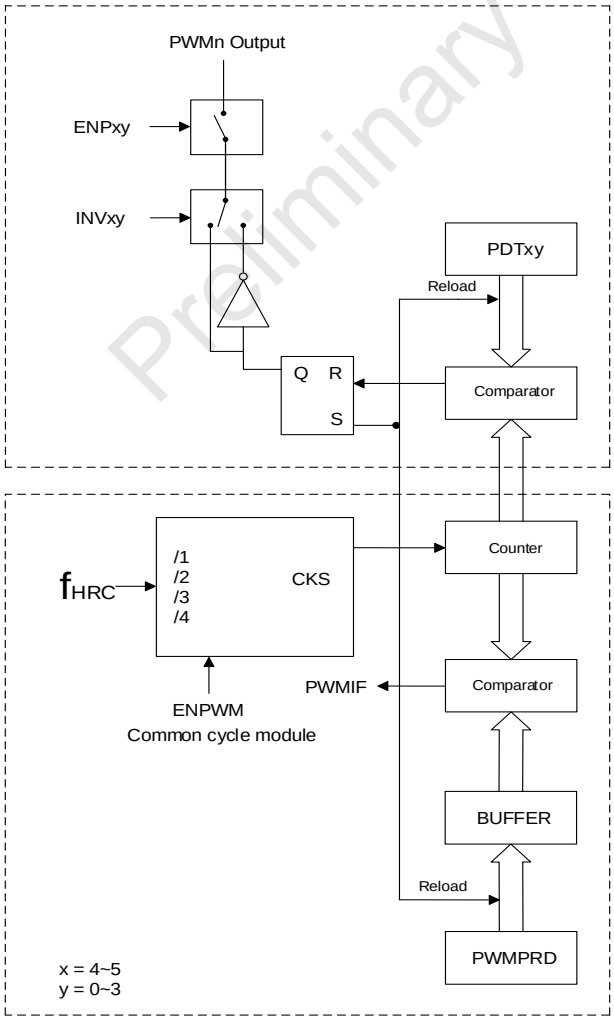
① 12-bit PWM precision;

② 8-channels have the same period, but the duty ratio can be set separately;

③ Output can be configured in forward or reverse direction.

The cycle and duty cycle of the SC92F744XB PWM is adjustable. Registers PWMCFG, PWMCON controll PWM status and cycle as well as opening of each channel of PWM and duty cycle of output waveform can be adjusted separately.

13.1 PWM block Diagram



The SC92F744XB PWM block Diagram

13.2 PWM-related Registers

PWMCFG (D4H) PWM Configuration Register(Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	ENPWM	PWMIF	PWMCK[1:0]		PWMPD[11:8]			
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**ENPWM**	PWM module switch control bit (Enable PWM) 1: Clock is allowed to enter the PWM unit, PWM is in working state, and the state of PWM output port is controlled by register ENPxy (x=4~5, y=0,3) 0: PWM unit stops working, PWM counter is cleared, all PWM output ports are set to GPIO state
6	**PWMIF**	PWM interrupt request flag bit (PWM Interrupt Flag) When the PWM counter overflows (that is to say, when the count exceeds PWMPD), this bit is automatically set to 1 by the hardware. If IE1[1] (EPWM) is also set to 1 at this time, PWM interrupt is generated. **Note: After the PWM interrupt occurs, the hardware will not automatically clear this bit, which must be cleared by the user's software.**
5~4	**PWMCK[1:0]**	PWM Clock Source Selector 00: f_{HRC} 01: $f_{HRC}/2$ 10: $f_{HRC}/4$ 11: $f_{HRC}/8$
3~0	**PWMPD[11:8]**	The period of PWM setting high 4-bits.

		This value represents the (period - 1) of the PWM output waveform;That is to say, the period value of PWM output is (PWMPRD[11:0] + 1)* PWM clock;

PWMCON (D3H) PWM Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Symbol	**PWMPD[7:0]**							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7~0	**PWMPD[7:0]**	The period shared by PWM seting lower 8-bits; This value represents the (period - 1) of the PWM output waveform ;That is to say, the period value of PWM output is (PWMPRD[11:0] + 1)* PWM clock;

IE1 (A9H) Interrupt Enable Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	ECMP	-	EINT2	EBTM	EPWM	ESSI
R/W	-	-	R/W	-	R/W	R/W	R/W	R/W
POR	x	x	0	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
1	**EPWM**	PWM Interrupt Control Bit 0: Clear to disable the PWM interrupt 1: Set to enable the interrupt when PWM counter overflows

IP1 (B9H) Interrupt Priority Register 1 (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	IPCMP	-	IPINT2	IPBTM	IPPWM	IPSSI
R/W	-	-	R/W	-	R/W	R/W	R/W	R/W
POR	x	x	0	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
1	**IPPWM**	PWM interrupt priority selection bit 0: Clear to configure PWM interrupt priority as “low” 1: Set to configure PWM interrupt priority as “ high”

PWM duty cycle adjustment register (write)

740H	ENP140	INV140	-	-	PDT140[11:8]
741H	PDT140[7:0]				
742H	ENP141	INV141	-	-	PDT141[11:8]
743H	PDT141[7:0]				

744H	ENP142	INV142	-	-	PDT142[11:8]
745H	PDT142[7:0]				
746H	ENP143	INV143	-	-	PDT143[11:8]
747H	PDT143[7:0]				
748H	ENP150	INV150	-	-	PDT150[11:8]
749H	PDT150[7:0]				
74AH	ENP151	INV151	-	-	PDT151[11:8]
74BH	PDT151[7:0]				
74CH	ENP152	INV152	-	-	PDT152[11:8]
74DH	PDT152[7:0]				
74EH	ENP153	INV153	-	-	PDT153[11:8]
74FH	PDT153[7:0]				

Note: PWM duty cycle adjustment register is only writable and unreadable!

Bit Number	Bit Mnemonic	Description
7	**ENPxy** **（x=4,5, y=0~3）**	Pxy Port PWM Waveform Output Selection 0: PXY port PWM output is turned off and acts as GPIO port 1 1: when ENPWM=1, Pxy acts as the PWM waveform output port.
6	**INVxy** **（x=4,5, y=0~3）**	Pxy Port PWM Waveform Output Reverse Control 1: PWM waveform output of PXY port is reverse 0: PWM waveform output of PXY port is not reverse

3~0	**PDTxy [11:8]** **（x=4,5, y=0~3）**	Pxy port PWM waveform duty cycle length setting; The high-level width of the PWM waveform on the Pxy pin is (PDTxy [11:0]) PWM clocks

Bit Number	Bit Mnemonic	Description
7~0	**PDTxy [7:0]** **（x=4,5, y=0~3）**	Pxy port PWM waveform duty cycle length setting; The high-level width of the PWM waveform on the Pxy pin is (PDTxy [11:0]) PWM clocks

Note: if ENPWM set to 1, the PWM module is turned on, but when ENPxy=0, the PWM output is turned off and acts as GPIO. At this time, the PWM module can be used as a 12-bit Timer, and when EPWM(IE1.1) is set to 1, PWM still generates interrupts.

13.3 PWM Waveforms and Directions

The influence of changing various SFR parameters on PWM waveform is shown as follows:

① Diagram for Duty Cycle Change features

When PWMn outputs waveform, if it is required to change the duty cycle, users can change the value of high level configuration registers (PDTxy). Note that changing the value of PWMDTYn will change the duty cycle immediately.

② Period Change features

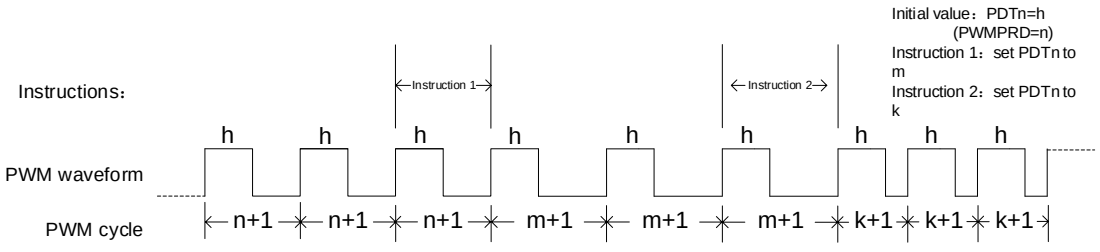


Diagram for Period Change Features

When PWMn outputs waveform, if it is required to change the period, the user can change the value of period configuration registers PWMPRD. Same as changing the duty cycle, change the value of PWMPRD will change the period immediately..

③ Relationship between Period and Duty cycle

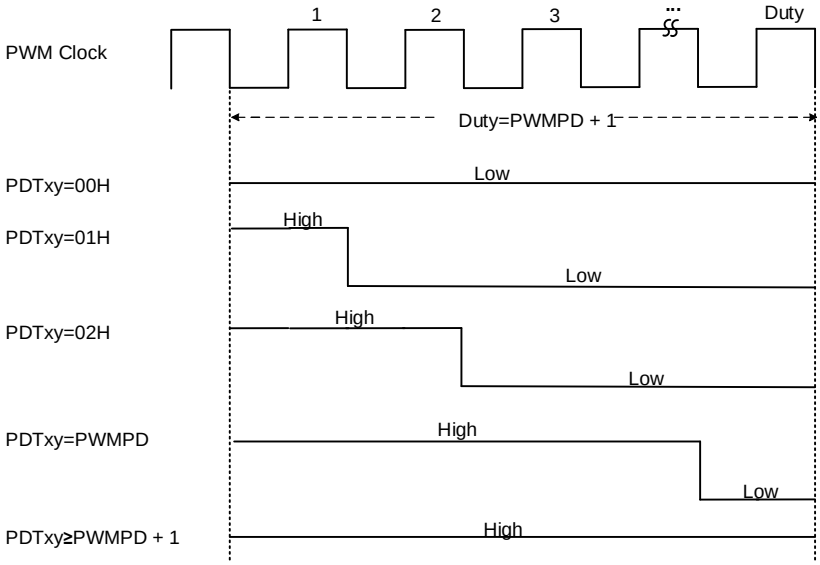


Diagram for Relationship between Period and Duty cycle

The relationship between period and duty cycle is shown in the figure above. The precondition of this result is the PWM output reverse control (INVxy) is initialized to 0; if it is required to get the contrary result, set INVxy to 1.

14 General-purpose I/O (GPIO)

The SC92F744XB offers up to 46 bidirectional controllable GPIOs, input and output control registers are used to control the input and output state of various ports, when the port is used as input, each I/O port is equipped with internal pull-up resistor controlled by PxPHy. Such 46 IOs are shared with other functions, including P3 can be used as LCD COM driver by configuring output voltage as 1/4 V_{DD} or 1/3V_{DD}. Under output state, I/O port reads the value in the port data register.

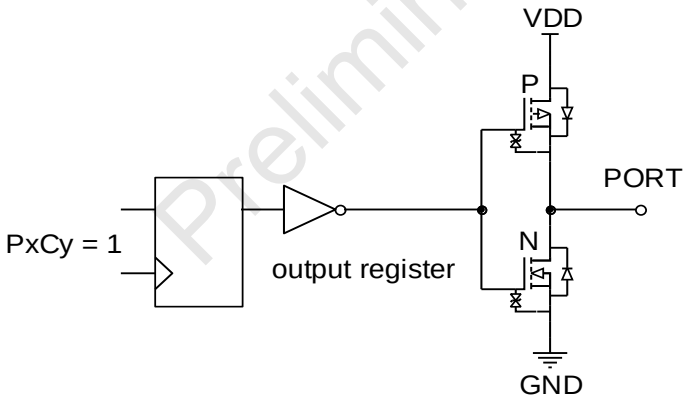
Note: Unused IO port or IO port with no package pin shall be configured as strong push-pull output mode.

14.1 GPIO Structure Diagram

Strong Push-pull Output Mode

In strong push-pull output mode, it is able to provide continuous high current drive: high output for the current larger than 18mA and low output for the current larger than 65mA

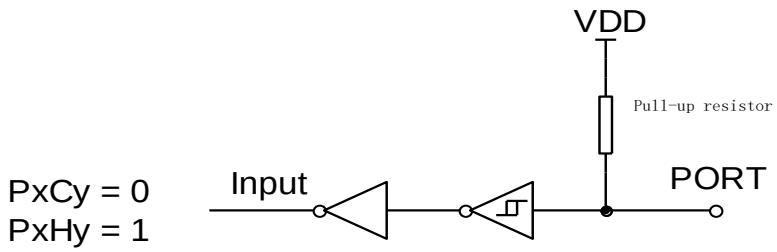
The port structure diagram for strong push-pull output mode is shown below:



Strong Push-pull Output Mode

Pull-up Input Mode

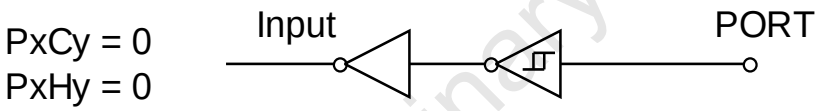
In pull-up input mode, a pull-up resistor is connected on the input port, only when the level on the input port is pulled down, low level signal can be detected.



Pull-up Input Mode

High Impedance Input Mode. (Input only)

The port structure diagram for input only mode is shown below:



High- impedance Input Mode

14.2 I/O Port-related Registers

P0CON (9AH) P0 I/O Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P0C7	P0C6	P0C5	P0C4	P0C3	P0C2	P0C1	P0C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P0PH (9BH) P0 Pull-up Resistor Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0

Bit Mnemonic	P0H7	P0H6	P0H5	P0H4	P0H3	P0H2	P0H1	P0H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P1CON (91H) P1 I/O Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P1C7	P1C6	P1C5	P1C4	P1C3	P1C2	P1C1	P1C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P1PH (92H) P1 Pull-up Resistor Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P1H7	P1H6	P1H5	P1H4	P1H3	P1H2	P1H1	P1H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P2CON (A1H) P2 I/O Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P2C7	P2C6	P2C5	P2C4	P2C3	P2C2	P2C1	P2C0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P2PH (A2H) P2 Pull-up Resistor Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P2H7	P2H6	P2H5	P2H4	P2H3	P2H2	P2H1	P2H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P3CON (B1H) P3 I/O Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P3C7	P3C6	P3C5	P3C4	P3C3	P3C2	P3C1	P3C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P3PH (B2H) P3 Pull-up Resistor Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P3H7	P3H6	P3H5	P3H4	P3H3	P3H2	P3H1	P3H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

POR	0	0	0	0	0	0	0	0

P4CON (C1H) P4 I/O Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P4C7	P4C6	P4C5	P4C4	P4C3	P4C2	P4C1	P4C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P4PH (C2H) P4 Pull-up Resistor Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P4H7	P4H6	P4H5	P4H4	P4H3	P4H2	P4H1	P4H0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P5CON (D9H) P5 I/O Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	P5C5	P5C4	P5C3	P5C2	P5C1	P5C0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	0	0	0	0	0	0

P5PH (DAH) P5 Pull-up Resistor Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	P5H5	P5H4	P5H3	P5H2	P5H1	P5H0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**PxCy** **(x=0 ~ 5, y=0 ~ 7)**	Px port input and output control bit 0: Pxy as input mode (initial value) 1: Pxy as strong push-pull output mode
7 ~ 0	**PxHy** **(x=0 ~ 5, y=0 ~ 7)**	Px port pull-up resistance configuration, only valid when PxCy=0: 0: Pxy as high-impedance input mode (initial value), the pull-up resistor is turned off. 1: Pxy pull-up resistance is turned on.

P0 (80H) P0 Data Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P1 (90H) P1 Data Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P2 (A0H) P2 Data Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P3 (B0H) P3 Data Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P4 (C0H) P4 Data Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

P5 (D8H) P5 Data Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7~0	**P0.x** **(x=0~7)**	P0 Latches Data Register
7~0	**P1.x** **(x=0~7)**	P1 Latches Data Register
7~0	**P2.x** **(x=0~7)**	P2 Latches Data Register
7~0	**P3.x** **(x=0~7)**	P3 Latches Data Register

7~0	**P4.x** **(x=0~7)**	P4 Latches Data Register
5~0	**P5.x** **(x=0~5)**	P5 Latches Data Register

IOHCON0 (96H) IOH Configuration Register 0 (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P1H[1:0]		P1L[1:0]		P0H[1:0]		P0L[1:0]	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 6	**P1H[1: 0]**	P1 high 4-bit IOH configuration bits 00: Set P1 high 4-bit IOH level 0 (Maximum value); 01: Set P1 high 4-bit IOH level 1; 10: Set P1 high 4-bit IOH level 2; 11: Set P1 high 4-bit IOH level 3 (Minimum value);
5 ~ 4	**P1L[1: 0]**	P1 low 4-bit IOH configuration bits 00: Set P1 low 4-bit IOH level 0 (Maximum value); 01: Set P1 low 4-bit IOH level 1; 10: Set P1 low 4-bit IOH level 2; 11: Set P1 low 4-bit IOH level 3 (Minimum value);
3 ~ 2	**P0H[1: 0]**	P0 high 4-bit IOH configuration bits

		00: Set P0 high 4-bit IOH level 0 (Maximum value); 01: Set P0 high 4-bit IOH level 1; 10: Set P0 high 4-bit IOH level 2; 11: Set P0 high 4-bit IOH level 3 (Minimum value);
1 ~ 0	**P0L[1: 0]**	P0 low 4-bit IOH configuration bits 00: Set P0 low 4-bit IOH level 0 (Maximum value); 01: Set P0 low 4-bit IOH level 1; 10: Set P0 low 4-bit IOH level 2; 11: Set P0 low 4-bit IOH level 3 (Minimum value);

IOHCON1 (97H) IOH Configuration Register 1 (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	P3L[1:0]		P2H[1:0]		P2L[1:0]	
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
5 ~ 4	**P3L[1: 0]**	P3 low 4-bit IOH configuration bits 00: Set P3 low 4-bit IOH level 0 (Maximum value); 01: Set P3 low 4-bit IOH level 1; 10: Set P3 low 4-bit IOH level 2; 11: Set P3 low 4-bit IOH level 3 (Minimum value);
3 ~ 2	**P2H[1: 0]**	P2 high 4-bit IOH configuration bits 00: Set P2 high 4-bit IOH level 0 (Maximum value);

		01: Set P2 high 4-bit IOH level 1; 10: Set P2 high 4-bit IOH level 2; 11: Set P2 high 4-bit IOH level 3 (Minimum value);
1 ~ 0	**P2L[1: 0]**	P2 low 4-bit IOH configuration bits 00: Set P2 low 4-bit IOH level 0 (Maximum value); 01: Set P2 low 4-bit IOH level 1; 10: Set P2 low 4-bit IOH level 2; 11: Set P2 low 4-bit IOH level 3 (Minimum value);
7~6	**-**	Reserve

15 LCD/LED Display Driver

SC92F7447B/ 7446B integrates LCD/LED hardware display driver circuit, facilitating user to realize LCD and LED display driver. The major features of LCD/LED driver are shown below:

1. Supports LCD/LED display driver functions, and users can only select one at the same time;
2. LCD and LED display driver share the same IO pins and registers.

LCD display driver functions are shown below:

1. 4 display driver modes: 8 X 24, 6 X 26, 5 X 27 or 4X 28;
2. 2 bias modes: 1/4 Bias or 1/3 Bias;
3. 4 level com port drive capabilities;
4. Display driver circuit can select either built-in 128K LRC or external 32K oscillator as clock source with frame frequency of about 64 Hz.

LED display driver functions are shown below:

1. 4 display driver modes: 8 X 24, 6X 26, 5 X 27 or 4X 28;
2. 4 level segment port drive capabilities;
3. Display driver circuit can select either built-in 128K LRC or external 32K oscillator as clock source with frame frequency of about 64 Hz.

15.1 LCD/LED Display driver-related registers

DDRCON (93H) Display Driver Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	DDRON	DMOD	DUTY[1:0]		VLCD[3:0]			
R/W	R/W	R/W	R/W		R/W			
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**DDRON**	**LCD/LED display drive enable control** 0: Display Drive Scan Enable 1: Display Drive Scan Disable
6	**DMOD**	**LCD/LED display drive Mode** 0: LCD Mode; 1: LED Mode;
5~4	**DUTY[1:0]**	**LCD/LED display duty cycle control** 00: 1/8 duty cycle, S4 to S27 are segments, C0 to C7 are common； 01: 1/6 duty cycle, S2 to S27 are segments, C2 to C7 are common； 10: 1/5 duty cycle, S1 to S27 are segments, C3 to C7 are common； 11: 1/4 duty cycle, S0 to S27 are segments, C4 to C7 are common； or S4 to S27 are segments, C0 to C3 are common
3~0	**VLCD[3:0]**	**LCD Voltage Regulation** $VLCD=V_{DD}*(17+VLCD[3:0])/32$

P0VO (9CH) P0 Display Driver Output Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P07VO	P06VO	P05VO	P04VO	P03VO	P02VO	P01VO	P00VO
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7~0	**P0nVO**	**Enable P0n Display Driver Output** **0: Disable P0n Display Driver Output function** **1: Enable P0n Display Driver Output function**

P1VO (94H) P1 Display Driver Output Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P17VO	P16VO	P15VO	P14VO	P13VO	P12VO	P11VO	P10VO
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7~0	**P1nVO**	**Enable P1n Display Driver Output** **0: Disable P1n Display Driver Output function** **1: Enable P1n Display Driver Output function**

P2VO (A3H) P2 Display Driver Output Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P27VO	P26VO	P25VO	P24VO	P23VO	P22VO	P21VO	P20VO

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7~0	**P2nVO**	**Enable P2n Display Driver Output** **0: Disable P2n Display Driver Output function** **1: Enable P2n Display Driver Output function**

P3VO (B3H) P3 Display Driver Output Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	P37VO	P36VO	P35VO	P34VO	P33VO	P32VO	P31VO	P30VO
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7~0	**P3nVO**	**Enable P3n Display Driver Output** **0: Disable P3n Display Driver Output function** **1: Enable P3n Display Driver Output function**

OTCON (8FH) Output Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	SSMOD[1:0]		-	-	VOIRS[1:0]		SCS	BIAS
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
POR	0	0	x	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
3~2	**VOIRS[1:0]**	**Selection bits of voltage dividing resistance of LCD port** 00: Set the total resistance value of internal partial varistor to 100kΩ 01: Set the total resistance value of internal partial varistor to 200kΩ 10: Set the total resistance value of internal partial varistor to 400kΩ 11: Set the total resistance value of internal partial varistor to 800kΩ For each Common switch, the first 1/16 time is fixed to select 100k resistor, and the last 15/16 time is switched to the resistance value selected by VORIS.
1	**SCS**	**LCD/LED Segment/Common multiplex pin selection** 0: when set to 1/4 duty cycle, s0 to s27 are segments and C4 to C7 are common 1: when set to 1/4 duty cycle, s0 to s27 are segments and C0 to C3 are common
0	**BIAS**	**LCD Bias Voltage Setting:** 0: 1/4 bias 1: 1/3 bias

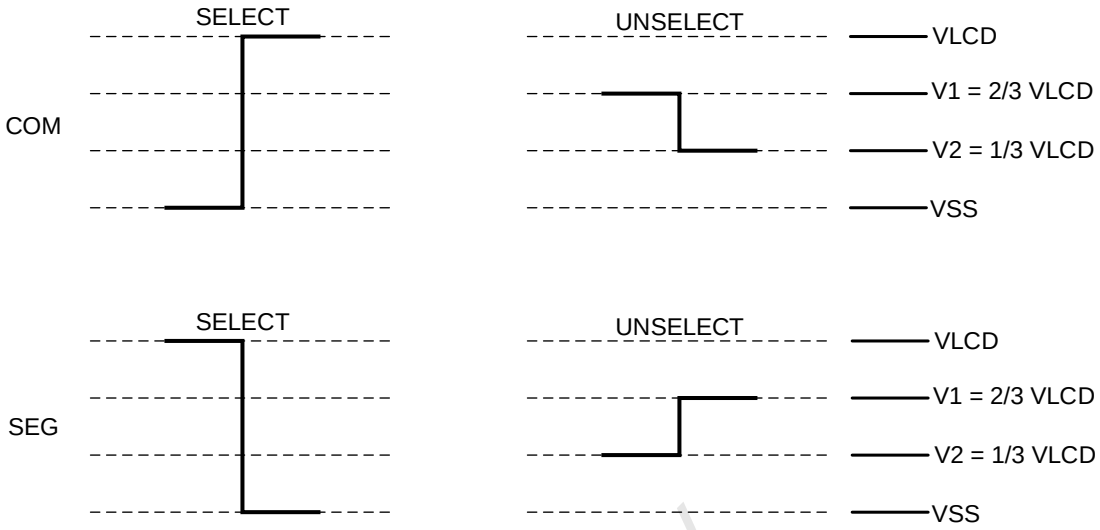
15.2 LCD/LED display RAM configuration

地址 Address	7	6	5	4	3	2	1	0
	COM7	COM6	COM5	COM4	COM3	COM2	COM1	COM0
700H	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0	SEG0
701H	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1	SEG1
702H	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2	SEG2
703H	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3
704H	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4
705H	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5
706H	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6
707H	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7
708H	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8
709H	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9
70AH	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10
70BH	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11
70CH	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12
70DH	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13
70EH	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14
70FH	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15

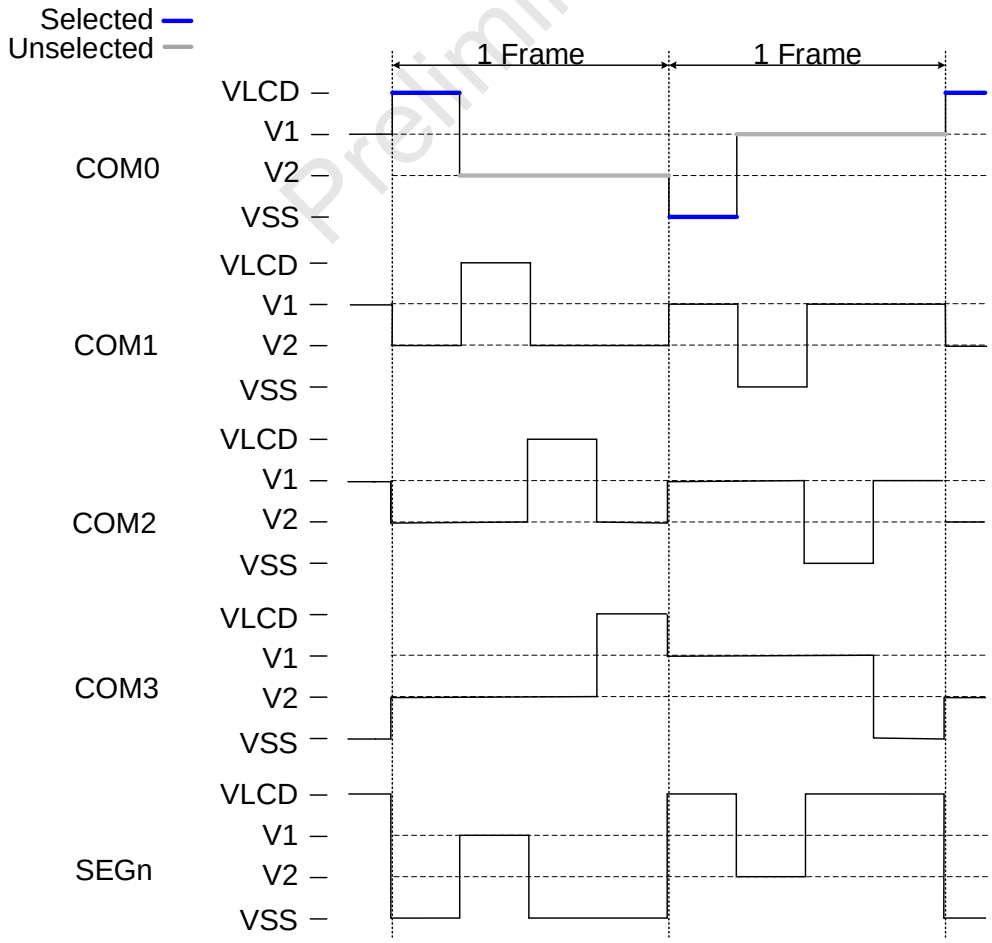
710H	SEG16	SEG16	SEG16	SEG16	SEG16	SEG16	SEG16	SEG16
711H	SEG17	SEG17	SEG17	SEG17	SEG17	SEG17	SEG17	SEG17
712H	SEG18	SEG18	SEG18	SEG18	SEG18	SEG18	SEG18	SEG18
713H	SEG19	SEG19	SEG19	SEG19	SEG19	SEG19	SEG19	SEG19
714H	SEG20	SEG20	SEG20	SEG20	SEG20	SEG20	SEG20	SEG20
715H	SEG21	SEG21	SEG21	SEG21	SEG21	SEG21	SEG21	SEG21
716H	SEG22	SEG22	SEG22	SEG22	SEG22	SEG22	SEG22	SEG22
717H	SEG23	SEG23	SEG23	SEG23	SEG23	SEG23	SEG23	SEG23
718H	SEG24	SEG24	SEG24	SEG24	SEG24	SEG24	SEG24	SEG24
719H	SEG25	SEG25	SEG25	SEG25	SEG25	SEG25	SEG25	SEG25
71AH	SEG26	SEG26	SEG26	SEG26	SEG26	SEG26	SEG26	SEG26
71BH	SEG27	SEG27	SEG27	SEG27	SEG27	SEG27	SEG27	SEG27

15.3 LCD waveform

15.3.1 1/3Bias LCD waveform

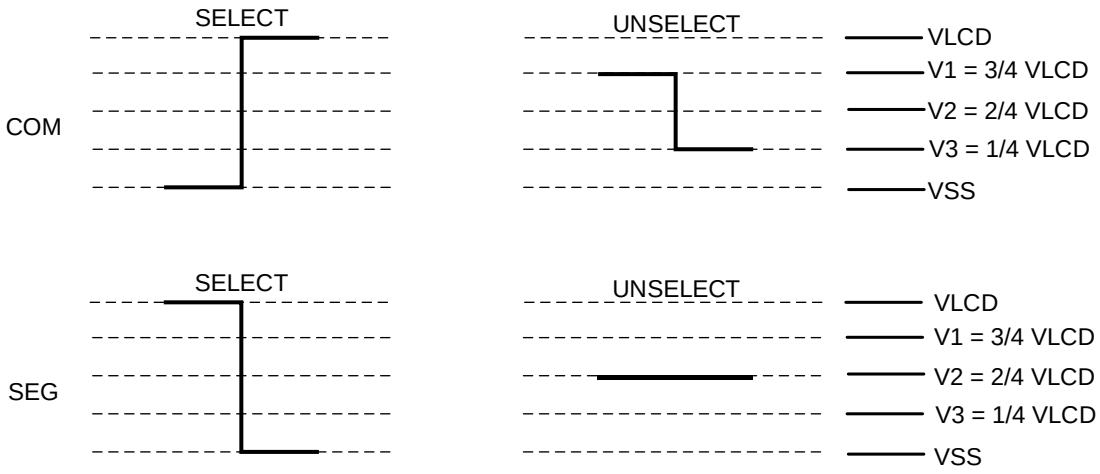


Waveform in 1/3 Bias LCD application

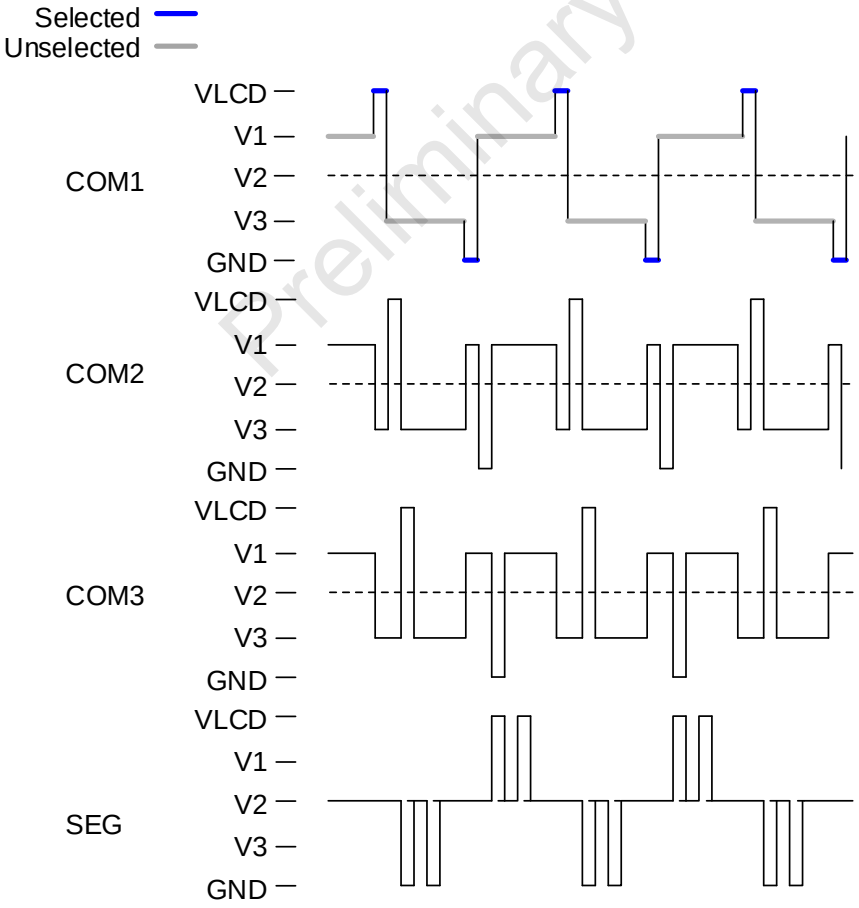


COM and SEG Waveform in 1/3 Bias LCD application

15.3.2 1/4Bias LCD waveform

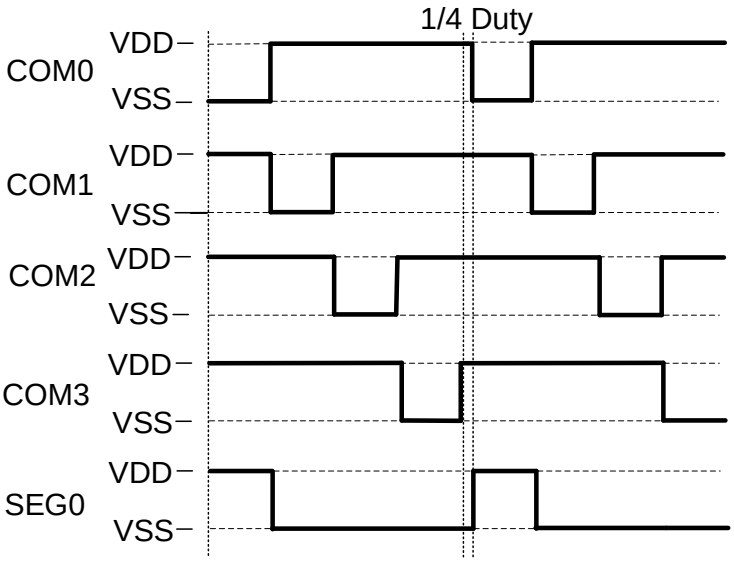


Waveform in 1/4 Bias LCD application



COM and SEG Waveform in 1/4 Bias LCD application

15.4 LED waveform



COM and SEG Waveform in LED Application

15.5 LCD/LED Demo programme

15.5.1 LCD Configuration Demo programme

```
unsigned char xdata LCDRAM[30] _at_ 0x700;
unsigned char lcd_addr;
unsigned char lcd_data;

DDRCON |= 0x00;        //0: LCD Mode ;1: LED Mode
DDRCON |= 0x30;        //1/4 duty cycle
DDRCON |= 0x07;        // VLCD=VDD*3/4
DDRCON |= 0x80;        // Display drive scan Enable

P0VO = 0xFF;           // Enable P0 Display Driver Output function
P1VO = 0xFF;           // Enable P1 Display Driver Output function
P2VO = 0xFF;           // Enable P2 Display Driver Output function
P3VO = 0xFF;           // Enable P3 Display Driver Output function
OTCON = 0x06;          // Set the total resistance value of the internal partial varistor to 200 kω
                       //1/4 bias voltage; S4 to S27 are segments, C0 to C3 are common
```

```
LCDRAM[lcd_addr] = lcd_data;   // Write the value to be displayed to LCD RAM.
```

15.5.2 LED Configuration Demo programme

```
unsigned char xdata LEDRAM[30] _at_ 0x700;

unsigned char led_addr;

unsigned char led_data;

 DDRCON |= 0x4F;        // 0: LCD Mode ;1: LED Mode
                        //1/8 duty cycle
                        // S4 to S27 are segments, C0 to C7 are common
DDRCON |= 0x80;         // Display drive scan Enable
IOHCON0 = 0xC0;         // Set P1 high 4-bit IOH level 3 (Minimum value);
                        // And other pins to IOH level 0 (maximum)
IOHCON1 = 0x00;
P0VO = 0xFF;            // Enable P0 Display Driver Output function
P1VO = 0xFF;            // Enable P1 Display Driver Output function
P2VO = 0xFF;            // Enable P2 Display Driver Output function
P3VO = 0xFF;            // Enable P3 Display Driver Output function


OTCON = 0x00;

LCDRAM[led_addr] = led_data;   // Write the value to be displayed to LED RAM.
```

16 Serial Interface 0 (UART0)

The SC92F744XB supports a full-duplex serial port. It is convenient for connecting other device or equipment, for example, WiFi module or other drive chips with UART communication interface. UART0 functions and features are shown below:

1. Three kinds of communication mode: Mode 0, Mode 1 and Mode 3;
2. Configure Timer1 or Timer2 as baud rate generator;
3. Completion of transmission and reception can generate interrupt RI/TI, and such interrupt flag needs to be cleared up by software.

SCON (98H) Serial Port Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 6	**SM0 ~ 1**	Serial communication mode control bits 00: Mode 0, 8-bit half-duplex synchronous communication mode. Serial data is received and transmitted on RX pin. TX pin is used to transmit shift clock. Receive and transmit 8 bits for each frame, and low bits will be received or transmitted firstly; 01: Mode 1, 10-bit full-duplex asynchronous communication composing of 1 starting bit, 8 data bits and 1 stopping bit, with communication baud rate changeable; 10: Reserved; 11: Mode 3, 11-bit full-duplex asynchronous communication, composing of 1 starting bit, 8 data bits and 1 programmable 9^{th} bit and 1 stopping bit, with communication baud rate changeable.
5	**SM2**	Serial communication mode control bit 2, this control bit is only valid for mode 3 0: RI is set upon receiving a complete data frame to generate interrupt

		request; 1: When receiving a complete data frame, only when RB8=1, will RI be set to generate interrupt request.
4	**REN**	Receive allowing control bit 0: Receiving data not allowed; 1: Receiving data allowed.
3	**TB8**	Only valid for mode 3, 9th bit of receiving data
2	**RB8**	Only valid for mode 3, 9th bit of receiving data
1	**TI**	Transmission interrupt flag bit
0	**RI**	Reception interrupt flag bit

SBUF (99H) Serial Data Cache Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	SBUF[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**SBUF[7: 0]**	Serial Port Data Cache Register SBUF contains two registers: one for transmitting shift register and one for receiving latch; data written into SBUF will be transmitted to shift register and initiate transmitting process; reading SBUF will return the contents of receiving latch.

PCON (87H) Power Management Control Register (only readable, * unreadable*)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	SMOD	-	-	-	-	-	STOP	IDL
R/W	W	-	-	-	-	-	W	W
POR	0	x	x	x	x	x	0	0

Bit Number	Bit Mnemonic	Description
7	**SMOD**	Baud rate multiplying power configuration bit, only valid for mode 0 (SM0 ~ 1 = 00): 0: Serial port operates under clock of 1/12 system clock 1: Serial port operates under clock of 1/4 system clock

16.1 Baud Rate of Serial Communication

In mode 0, baud rate can be programmed as 1/12 or 1/4 of system clock and determined by SMOD (PCON.7) bit. When SMOD is set to 0, the serial port operates in 1/12 of system clock. When SMOD is set to 1, serial port operates in 1/4 of system clock.

In mode 1 and mode 3, the user can select overflow rate of Timer1 or Timer2 as baud rate by configuration.

Set TCLK (T2CON.4) and RCLK (T2CON.5) bit to configure Timer2 as TX and RX clock source of baud rate (Refer to the timer section for details). No matter TCLK or RCLK is set to logic 1, Timer2 can be in the mode of baud rate generator. If TCLK and RCLK are set to logic 0, Timer1 can be baud clock source of Tx and Rx.

Mode 1 and Mode 3 baud rate formula is shown below, including that [TH1、TL1] are the 16-bit counter registers of Timer1, and [RCAP2H、RCAP2L] are the 16-bit reload registers of Timer2.

1. When Timer1 is used as baud rate generator, it must stop counting, meaning TR1=0:

$$\text{BaudRate} = \frac{\text{fsys}}{[\text{TH1,TL1}]}$$ **(Note: [TH1, TL1] must be larger than 0x0010)**

2. When Timer2 is used as baud rate generator:

$$\text{BaudRate} = \frac{\text{fsys}}{[\text{RCAP2H,RCAP2L}]}$$ **(Note: [RCAP2H, RCAP2L] must be larger than 0x0010)**

17 SPI/TWI/UART Serial Interface (SSI)

The SC92F744XB integrates SPI/TWI/UART serial interface circuits (SSI), which is convenient for connecting MCU to devices or equipment with different interfaces. The user can configure SSI in any communication mode among SPI, TWI and UART by configuring SSMOD[1: 0] bit of register OTCON. Its features are shown below:

1. SPI mode can be configured as master mode or slave mode
2. TWI mode can only be used as slave in communication
3. UART mode can work in Mode 1 (10-bit full-duplex asynchronous communication) and Mode 3 (11-bit full-duplex asynchronous communication)

Specific configuration modes are shown below:

OTCON (8FH) Output Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	SSMOD[1:0]		-	-	VOIRS[1:0]		SCS	BIAS
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
POR	0	0	x	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 6	**SSMOD[1: 0]**	SSI communication mode control bits 00: SSI OFF 01: SSI is set in SPI communication mode; 10: SSI is set in TWI communication mode; 11: SSI is set in UART communication mode;

17.1 Serial Peripheral Interface (SPI)

SSMOD[1: 0] = 01, SSI is configured as SPI interface. Serial Peripheral Interface (SPI) is a kind of high-speed serial communication interface, allowing MCU and peripheral equipment (including other MCUs) to conduct full-duplex synchronous serial communication.

17.1.1 SPI Operation-related Registers

SSCON0 (9DH) SPI Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	SPEN	-	MSTR	CPOL	CPHA	SPR2	SPR1	SPR0
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	x	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**SPEN**	SPI Enable Control Bit 0: Disable SPI 1: Enable SPI
5	**MSTR**	SPI Master/Slave Selection Bit 0: SPI as slave equipment 1: SPI as master equipment
4	**CPOL**	Clock Polarity Control Bit 0: SCK is at low level under idle state 1: SCK is at high level under idle state
3	**CPHA**	Clock Phase Control Bit 0: First edge collection data of SCK period 1: Second edge collection data of SCK period
2 ~ 0	**SPR[2: 0]**	SPI Clock Speed Selection Bits 000: f_{SYS} /4 001: f_{SYS} /8

		010: f_{SYS} /16 011: f_{SYS} /32 100: f_{SYS} /64 101: f_{SYS} /128 110: f_{SYS} /256 111: f_{SYS} /512
6	-	Reserved

SSCON1 (9EH) SPI Status Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	SPIF	WCOL	-	-	TXE	DORD	-	TBIE
R/W	R/W	R/W	-	-	R/W	R/W	-	R/W
POR	0	0	x	x	0	0	x	0

Bit Number	Bit Mnemonic	Description
7	**SPIF**	SPI Data Transmit Flag Bit 0: Must be cleared by software 1: Data transmission completed and flag is set to 1 by hardware
6	**WCOL**	Write-in Conflict Flag Bit 0: Cleared by software, indicating write-in conflict is processed 1: Set to 1 by hardware, indicating one conflict is detected
3	**TXE**	Transmit Buffer Empty Flag Bit 0: Transmitting buffer not empty

		1: Transmitting buffer empty, must be cleared by software
2	**DORD**	Transfer Direction Configuration Bit 0: Transmit MSB first 1: Transmit LSB first
0	**TBIE**	Transmitting Buffer Interrupt Enable Bit 0: Transmission interrupt not enable 1: Transmission interrupt enable, when SPIF=1, TBIE=1, it will generate SPI interrupt
5 ~ 4, 1	-	Reserved

SSDAT (9FH) SPI Data Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	SPD[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**SPD[7: 0]**	SPI Data Cache Register Data written to SSDAT will be sent to the transmitting shift register. Upon reading SSDAT, data from the receive shift register is received.

17.1.2 Signal Description

Master-Out/Slave-In (MOSI)

This signal connects master device with one slave device. Data is serially transmitted from master device to slave device via MOSI, featuring master device output and slave device input.

Master-In and Slave-Out (MISO):

This signal connects slave device with master device. Data is serially transmitted from slave device to master device via MISO, featuring slave device output and master device input. When SPI is configured as slave device and is not selected, the MISO pin of slave device is in high-impedance state.

SPI Serial Clock (SCK)

SCK signal is used to control synchronous movement of input and output data on MOSI and MISO. Transmit one byte for every 8 clock periods. If no slave device is selected, SCK signal will be ignored from slave device.

17.1.3 Operating Modes

SPI can be configured as master mode or slave mode. The configuration and initialization of SPI module can be completed via setting SSCON0 register (SPI Control Register) and SSCON1 (SPI State Register). After completing configuration, data is transmitted by setting SSCON0, SSCON1 and SSDAT (SPI Data Register).

During SPI communication period, data is synchronically and serially moved in or out. Serial clock line (SCK) makes data movement and sampling on two serial data lines (MOSI and MISO) keep synchronous. If any slave device is not selected, it is unable to participate in activities on SPI line.

When SPI master device transmits data to slave device via MOSI, slave device sends data to master device via MISO as response, which realizes synchronous full-duplex transmission of data transmitting and receiving at the same clock. The transmit shift register and the receive shift register use the same special function address. Conducting write operations to SPI data register(SSDAT) will write data to the transmit shift register, and conducting read operations to SSDAT will obtain the data from the receive shift register.

The SPI interface of some devices will lead to SS pin (Slave Select, active-low). When communicating with the SC92F744XB SPI, the SS pin from other devices on SPI bus shall be connected based on different communication modes. The following table lists the connection modes of the SS pin from other devices on SPI bus under different communication modes of the SC92F744XB SPI:

SC92F744XB SPI	Other Devices on SPI Bus	Mode	SS of Slave Device (Slave Device Select Pins)
Master Mode	Slave Mode	One Master One Slave	Pull low
		One Master Multiple Slaves	The SC92F744XB leads to multiple I/Os, which respectively connect to the SS pin of slave device. Before data transmission, the SS pin of slave

			device must be pulled low
Slave Mode	Master Mode	One Master One Slave	Pull high

Master Mode

- **Mode Startup:**

  Start of all data transmission on SPI bus is controlled by SPI master device. When MSTR bit in SSCON0 register is set to 1, SPI operates in master mode, and only one master device can start the transmission.

- **Transmitting:**

  In SPI master mode, write one byte of data to SPI data register SSDAT, the data will write to the transmit shift buffer. If any data already exists in the transmit shift register, one WCOL signal will be generated from master SPI to indicate writing is too fast. However, data in the transmit shift register will not be influenced and transmitting will not be interrupted as well. Besides, if the transmit shift register is empty, the master device will move the data in the transmit shift register to MOSI line serially according to SPI clock frequency on SCK. After transmission, SPIF bit in SSCON1 register will be set to 1. If SPI interrupt is allowed, when SPIF bit is set to 1, an interrupt will be generated as well.

- **Receiving:**

  When master device transmits data to slave device via MOSI line, corresponding slave device will also transmit the contents in the transmit shift register to the receive shift register of master device via MISO line so as to realize full-duplex operations. Therefore, setting SPIF flag bit to 1 indicates that transmission is completed and data has been received. Data received from slave device is stored in the receive shift register of master device in accordance with MSB first or LSB first transmission direction. When one byte of data is completely moved to the receive register, the processor can obtain such data by reading SSDAT register.

Slave Mode

- **Mode Startup:**

  When the MSTR bit in SSCON0 register is clear to 0, SPI operates in slave mode.

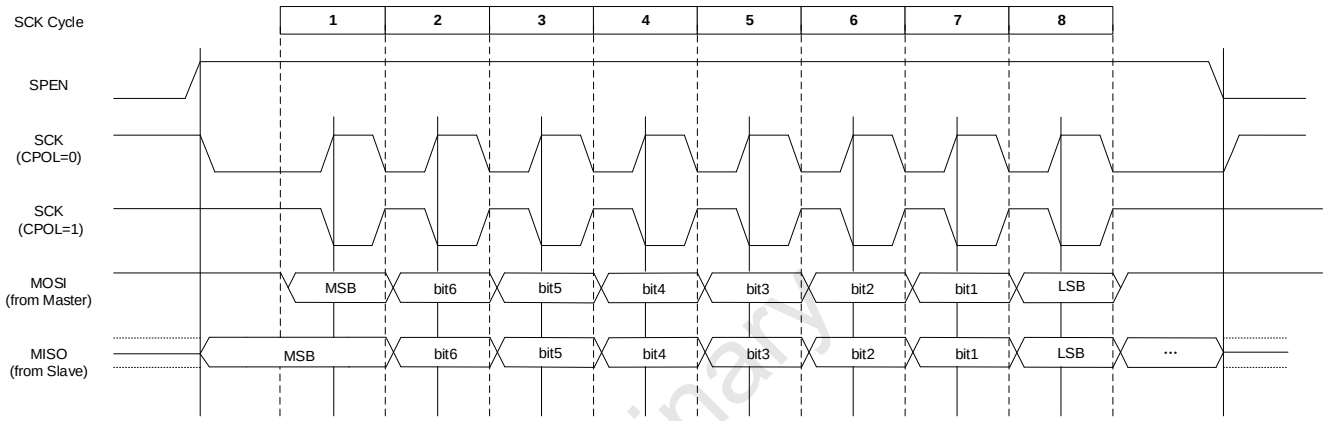
- **Transmitting and Receiving:**

  In slave mode, according to SCK signal controlled by master device, data is moved in via MOSI pin and out via MISO pin. A 1-bit counter records the number of SCK edge. When the receive shift register moves in 8-bit data (one byte) and the transmit shift register moves out 8-bit data (one byte), SPIF flag is set to 1. Data can be obtained by reading SSDAT register. If SPI interrupt is allowed, when setting SPIF to 1, an interrupt will be generated as well. At this time, the receive shift register keeps original data and set SPIF bit to 1, thus SPI slave device will not receive any data until SPIF is cleared to 0. SPI slave device must write the data to be transmitted before master device starts a new data transmission to the transmit shift register. If no data is written before transmitting, slave device will transmit “0x00” bytes to master device. If SSDAT writing operation occurs during the process of transmission, the WCOL flag bit of SPI slave device is set to 1. That is to say, if data is already included in the transmit shift register, WCOL bit of SPI slave device is set to 1, indicating conflict of SSDAT writing. But the data of shift register will not be influenced and transmission will not be interrupted.

17.1.4 Transfer Form

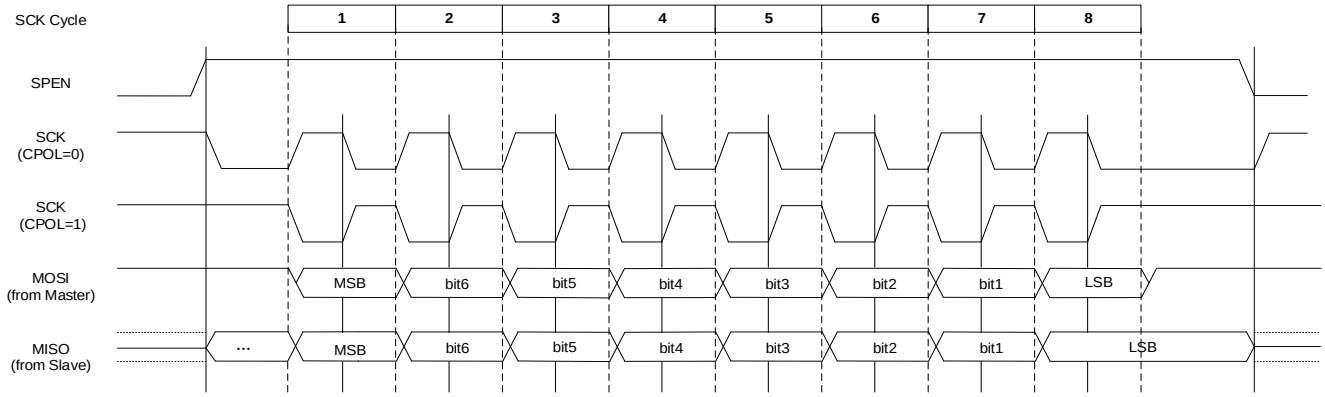
By setting CPOL bit and CPHA bit of SSCON0 register by software, the user can select four combinations of SPI clock polarity and clock phase. CPOL bit defines the polarity of clock, meaning the level status when idle, which has little influence on SPI transmission format. CPHA bit defines the phase of clock, meaning clock edge allowing data sampling shift. In two devices of master and slave communication, the configuration of clock polarity and phase shall be consistent.

When CPHA = 0, first edge of SCK captures data, and slave device must get the data ready before the first edge of SCK.



CPHA = 0 Data Transmission

When CPHA = 1, master device outputs data to MOSI line at the first edge of SCK, slave device takes the first edge of SCK as the signal of start transmitting and start capturing data at the second edge of SCK. Therefore, user must complete SSDAT writing operation in two edges of first SCK. Such data transmission form is the preferred form of communication between one master device and one slave device.



CPHA = 1 Data Transmission

17.1.5 Error Detection

Writing to SSDAT register may cause conflict during the period of transmitting data sequence, set WCOL bit in SSCON1 register to 1. Setting WCOL bit to 1 will not generate interrupt, and transmitting will not be interrupted.

WCOL bit shall be cleared by software.

17.2 TWI

17.2 Two-Wire Interface (TWI)

SSCON0 (9DH) TWI Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	TWEN	TWIF	-	GCA	AA	STATE[2: 0]		
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
POR	0	0	x	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**TWEN**	TWI Enable Control Bit 0: Disable TWI 1: Enable TWI
6	**TWIF**	TWI Interrupt Flag Bit 0: cleared by software 1: Under the following conditions, interrupt flag bit will be set by hardware ① First frame of address matched successfully ②Successfully receiving or transmitting 8-bit data ③Restart ④Slave device receives stopping signal
4	**GCA**	General Address Response Flag Bit 0: Non-response general address

		1: When GC = 1 and the general address matches, this bit will set to 1 by hardware and cleared to 0 automatically
3	**AA**	Receiving Enable Bit 0: Information sent by receiving master not allowed 1: Information sent by receiving master allowed
2 ~ 0	**STATE[2: 0]**	Device status flag Bits 000: slave device is in idle state, wait for TWEN to be set to 1, and detect TWI startup signal. When slave device receives stopping conditions, it will skip to this state 001: Slave device is receiving first frame of address and read and write bits (8^{th} bit for read and write bit, 1 for reading, 0 for writing). After receiving initial conditions, slave device will skip to this state. 010: State of slave device receiving data 011: State of slave device transmitting data 100: In the state of transmitting data of slave device, when the master device returns to UACK (high level for acknowledge bit), skip to this state, wait for restarting signal or stopping signal. 101: When the slave device is in transmitting state, setting AA to 0 and it will enter this state, waiting for restarting signal or stopping signal.
5	-	Reserved

SSCON1 (9EH) TWI Address Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	TWA[6: 0]							GC
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 1	**TWA[6: 0]**	TWI Address Register
0	**GC**	TWI General Address Enable Bit 0: Prohibits responding general address 1: Allow responding general address

SSDAT (9FH) TWI Data Cache Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	TWDAT[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**TWDAT[7: 0]**	TWI Data Cache Register

17.2.1 Signal Description

TWI Clock Signal Line (SCL)

This clock signal is sent from master device and connects all slave device. One byte of data is transmitted for every 9 clock periods. First 8 periods are used for data transmission and last one for receiver response clock.

TWI Data Signal Line (SDA)

SDA is a bidirectional signal line, and shall be in high level when idling, which is pulled up by pull-up resistance on SDA line.

17.2.2 Operating Modes

TWI communication of the SC92F744XB has only slave device mode:

- **Mode Startup:**

  When TWI enabling flag bit opens (TWEN = 1) and receives start-up signal sent from master device, this mode is initiated.

  The slave device enters first frame address (STATE[2: 0] = 001) state from idle mode (STATE[2: 0] = 000), and waits for first frame data from master device. First frame data is sent by master device, including 7-bit address bit and 1-bit read and write bit, all slave devices on TWI bus will receive first frame data of master device. After transmitting first frame data, master device will release SDA signal line. If the address sent by master device is the same as the value of address register of slave device, it indicates that the slave device has been selected and the selected slave device will judge to connect the 8th bit on the bus, which is the data read and write bit (=1, reading the command; =0, writing the command), then occupies SDA signal line, after transmitting a low-level response signal at the 9th clock period of SCL, release the bus. After the slave device is selected, enter into different status according to different read and write bits.

- **Non-general address response, slave device receiving mode:**

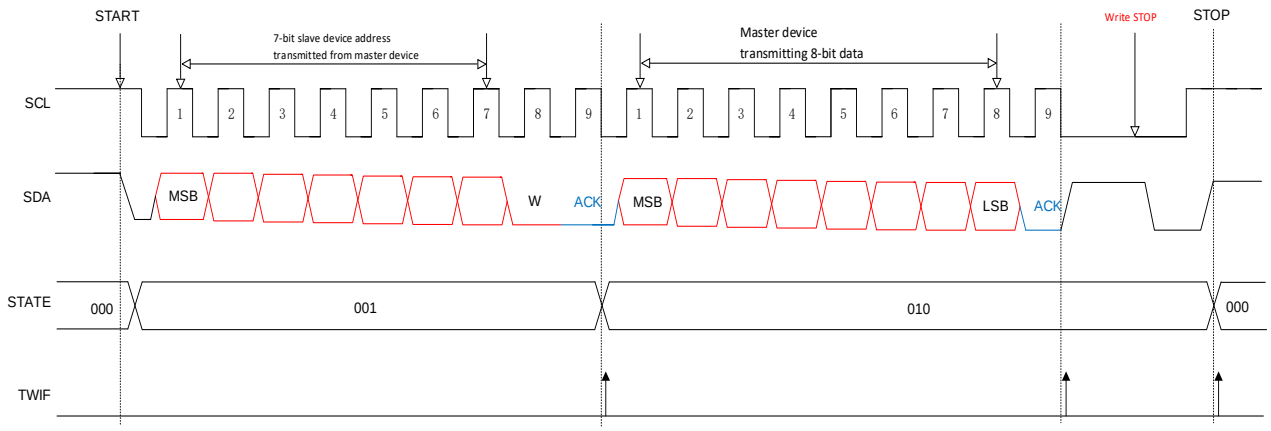
  If the read and write bit received from the first frame is writing (0), the slave device enters into the receiving state of slave device (STATE [2: 0] = 010), and wait for data sent from receiving master device. Master device will release the bus for transmitting every 8 bits and then wait for the response signal of 9th period of slave device.

  1. If the response signal from slave device is in low level, there are three modes of master communication:

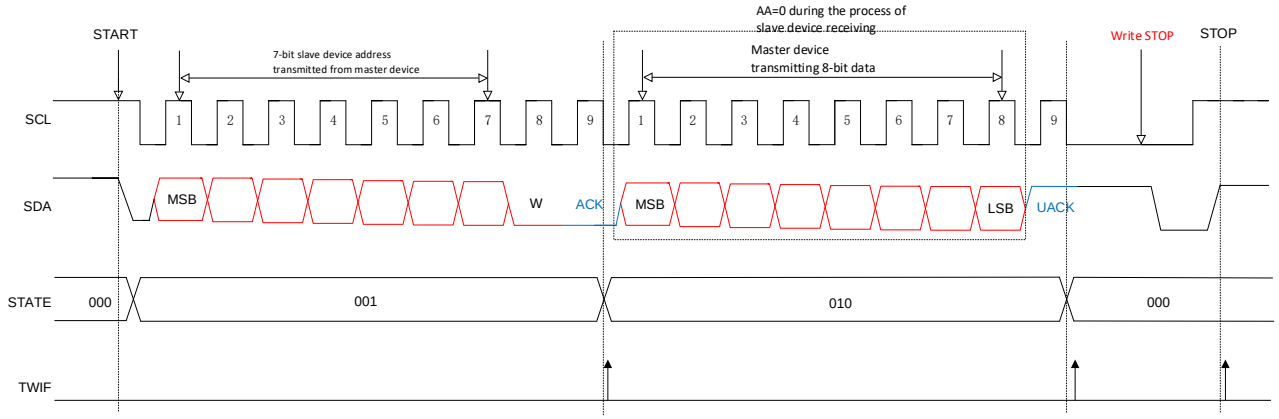
     1) Continue to send data;

     2) Resend start signal, then the slave device enters into the state of receiving first frame address (STATE[2: 0] = 001);

     3) Send stopping signal, indicating this transmission is ended, slave device returns to idle state and wait for next start signal from master device.



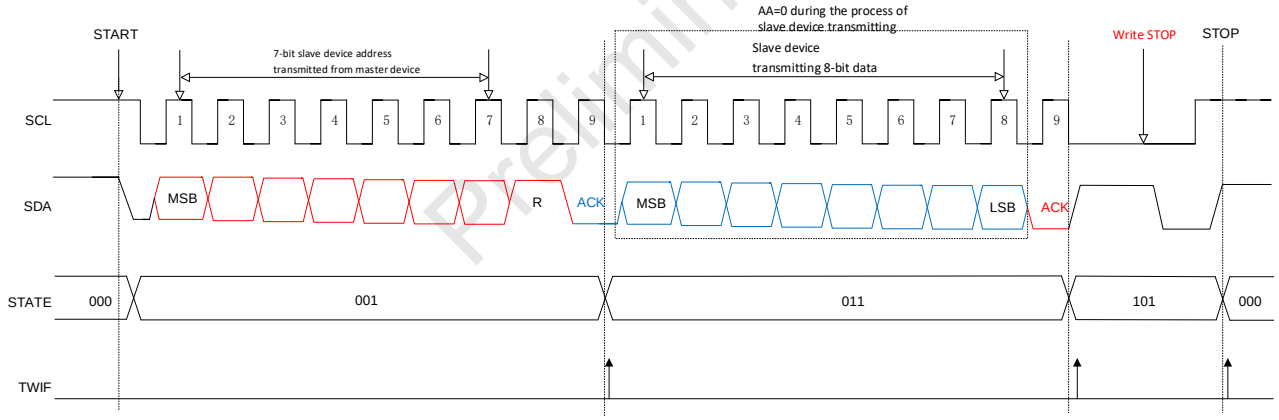
  2. If the response state of slave device is in high level (during the receiving process, the value of AA in slave device register is rewritten to 0), it indicates that after transmitting current bytes, the slave device will stop this transmission automatically and return to idle state (STATE[2: 0] = 000), without receiving data sent from master device any more.



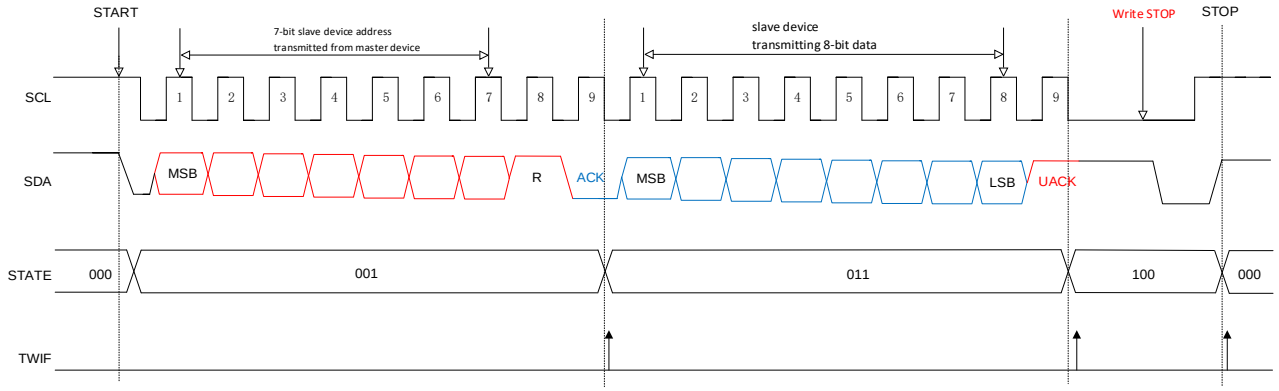
- **Non-general address response, master device transmitting mode:**

  If the read and write bit received from the first frame is reading (1), the slave device will occupy the bus and send data to master device. The slave device will release the bus for transmitting every 8-bit data and wait for the response from master device:

  1. If the response from master device is low level, the slave device continues to send data. During the transmitting process, if the value of AA in slave device register is rewritten to 0, the slave device will automatically end the transmission and release the bus after transmitting current bytes, and wait for stop signal or restart signal of the master device(STATE[2: 0] = 101).



  2. If the response from master device is high level, then the slave device state will wait for the stop signal or restart signal of the master device (STATE[2: 0] =100).



- **Response to General Address:**

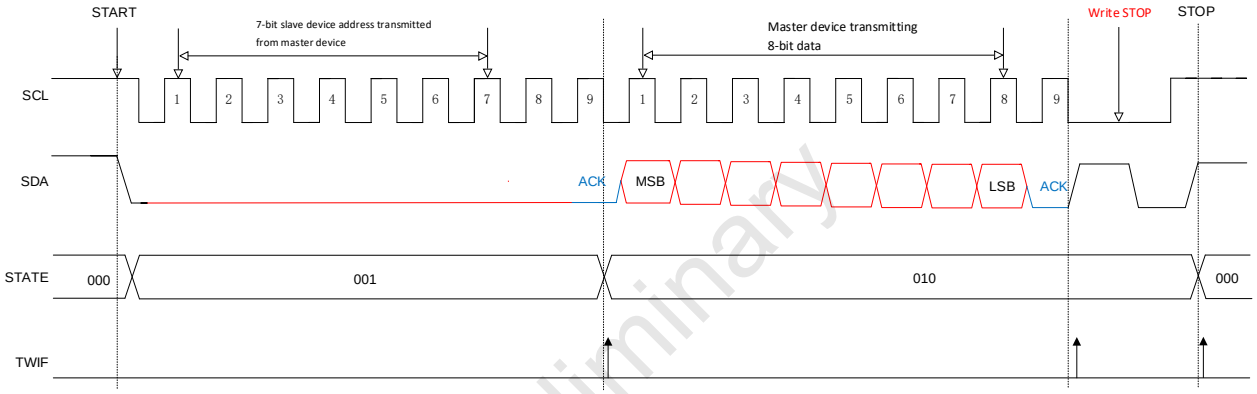
  When GC=1, general address is allowed to be used. When the slave device enters into the state of receiving first frame address (STATE[2: 0] = 001), the address bit data received in first frame data will be 0x00, at this time, all slave device will respond the master device. The read and write bit sent from master device must be write (0), all slave device will enter into the state of receiving data (STATE[2: 0] = 010). The master device will release SDA line for transmitting every 8-bit data and read the state on SDA line:

  1. If any response from slave device occurs, there are three modes of master device communication, as shown below:

     1) Continue to transmit data;

     2) Restart;

     3) Transmit the stop signal and end this communication.



  2. If there is no response from slave device, SDA will be in idle state.

Note: When using general address under the mode of one master and multiple slaves, the read and write bit sent by master device can not be read (1) status, or else, all the other devices on the bus will also transmit response except for equipment transmitting data.

17.2.3 Operating Steps

The operating steps of TWI in SSI are shown below:

① Configure SSMOD[1: 0] and select TWI mode;

② Configure SSCON0 TWI control register;

③ Configure SSCON1 TWI address register;

④ If the slave device receives data, wait for interrupt flag bit TWIF in SSCON0 to be set. The interrupt flag bit will be set to 1 when the slave device receives every 8-bit data. The interrupt flag bit shall be cleared by the user manually;

⑤ If the slave device transmits data, write the data to be transmit into TWDAT, TWI will transmit the data automatically. Interrupt flag bit TWIF will be set to 1 for transmitting every 8 bits.

17.3 Serial Interface 1 (UART1)

SSMOD[1: 0] = 11, SSI is configured as UART interface.

SSCON0 (9DH) Serial Port 1 Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	SM0	-	SM2	REN	TB8	RB8	TI	RI
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	x	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**SM0**	Serial Communication Mode Control Bit 0: Mode 1, 10-bit full-duplex asynchronous communication composing of 1 starting bit, 8 data bits and 1 stopping bit, with communication baud rate changeable; 1: Mode 3, 11-bit full-duplex asynchronous communication, composing of 1 starting bit, 8 data bits and 1 programmable 9th bit and 1 stopping bit, with communication baud rate changeable.
5	**SM2**	Serial Communication Mode Control Bit 2, this control bit is only valid for mode 3 0: Configure RI for receiving each complete data frame to generate interrupt request; 1: When receiving a complete data frame and only when RB8=1, will RI be configured to generate interrupt request.
4	**REN**	Receive Allowing Control Bit 0: Receiving data not allowed; 1: Receiving data allowed.
3	**TB8**	Only valid for mode 3, 9th bit of receiving data
2	**RB8**	Only valid for mode 3, 9th bit of receiving data

1	**TI**	Transmit Interrupt Flag Bit
0	**RI**	Receive Interrupt Flag Bit
6	-	Reserved

SSCON1 (9EH) Serial Port 1 Baud Rate Control Register Low (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	BAUD1L [7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

SSCON2 (95H) Serial Port 1 Baud Rate Control Register Low (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	BAUD1H [7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**BAUD1 [15: 0]**	Serial Port Baud Rate Control Bit $\text{BaudRate} = \frac{\text{fsys}}{\text{BAUD1H, BAUD1L}}$ **Note: [BAUD1H, BAUD1L] must be larger than 0x0010**

SSDAT (9FH) Serial Port Data Cache Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	SBUF[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**SBUF[7: 0]**	Serial Data Buffer SBUF contains two registers: one for transmit shift register and one for receiving latch, data writing to SBUF will be sent to shift register and initiate transmitting process, reading SBUF1 will return the contents of receiving latch.

18 Analog-to-Digital Converter (ADC)

The SC92F744XB has a 12-bit high-precision successive approximation ADC with 17-channel, the external 16 ADC channel is multiplexing with other IO ports. Cooperating with the internal 2.4V reference voltage, one internal channel connected to 1/4 V_{DD} can be used for measuring V_{DD} voltage.

There are 2 options for ADC reference voltage:

① VDD pin (internal V_{DD});

② Precise 2.4V reference output from internal Regulator (at this time, MCU supply voltage VDD can not be lower than 2.9V).

Note: f_{ADC} is directly obtained by internal f_{HRC} frequency division. When configuring, the user should pay attention that the clock frequency f_{ADC} of ADC cannot be greater than the frequency f_{SYS} of system clock, otherwise, ADC conversion results will be abnormal!

18.1 ADC-related Registers

ADCCON (ADH) ADC Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	ADCEN	ADCS	EOC/ADCIF	ADCIS[4: 0]				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	n

Bit Number	Bit Mnemonic	Description
7	**ADCEN**	ADC Power Control Bit 0: Disable ADC module power 1: Enable ADC module power
6	**ADCS**	ADC Start Trigger Control Bit (ADC Start) Write “1” for this bit, an ADC conversion started, this bit is the trigger signal only for ADC switch. This bit is valid only for writing “1”.

5	**EOC /ADCIF**	End Of Conversion / ADC Interrupt Flag 0: Conversion not completed 1: ADC conversion completed and need the user cleared up by software. ADC conversion completion flag EOC: when the user sets up ADCS for conversions, this bit will be cleared to 0 by hardware automatically; after completing conversion, this bit will be configured to 1 automatically by hardware; ADC interrupt request flag ADCIF: this bit is also used as interrupt request flag of ADC interrupt. If ADC interrupt is enabled, this bit must be cleared by the user with software after ADC interrupt generated.
4 ~ 0	**ADCIS[4: 0]**	ADC Input Selection Bits 00000: Select AIN0 as ADC input 00001: Select AIN1 as ADC input 00010: Select AIN2 as ADC input 00011: Select AIN3 as ADC input 00100: Select AIN4 as ADC input 00101: Select AIN5 as ADC input 00110: Select AIN6 as ADC input 00111: Select AIN7 as ADC input 01000: Select AIN8 as ADC input 01001: Select AIN9 as ADC input 01010: Select AIN10 as ADC input 01011: Select AIN11 as ADC input 01100: Select AIN12 as ADC input 01101: Select AIN13 as ADC input 01110: Select AIN14 as ADC input 01111: Select AIN15 as ADC input 10000 ~ 11110: Reserved 11111: ADC input is 1/4 V_{DD}, used for measuring power voltage

ADCCFG2 (AAH) ADC Configuration Register 2 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	-	-	LOWSP	ADCCK[1:0]	
R/W	-	-	-	-	-	R/W	R/W	R/W
POR	x	x	x	x	x	0	0	0

Bit Number	Bit Mnemonic	Description
2	**LOWSP**	ADC Sampling Clock Frequency Selector 0: Configure ADC sampling time as 6 ADC sampling clock periods 1: Configure ADC sampling time as 36 ADC sampling clock periods LOWSP controls ADC sampling clock frequency, conversion clock frequency of ADC is controlled by ADCCK[2: 0], independent of the influence of LOWSP bit The whole process from sampling to conversion of ADC needs 6 or 36 ADC sampling clocks plus 14 ADC conversion clocks, therefore, in practical application, the total time of ADC from sampling to conversion shall be calculated as follows: LOWSP=0: TADC1= (6+14)/fADC; LOWSP=1: T_{ADC2}= (36+14)/f_{ADC}
1 ~ 0	**ADCCK[1: 0]**	ADC Sampling Clock Frequency Selector 01: Configure ADC clock frequency f_{ADC} as f_{HRC}/12; 10: Configure ADC clock frequency f_{ADC} as f_{HRC}/6; Others: Reserved
7 ~ 3	-	Reserved

ADCCFG0 (ABH) ADC Configuration Register 0 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	EAIN7	EAIN6	EAIN5	EAIN4	EAIN3	EAIN2	EAIN1	EAIN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

ADCCFG1 (ACH) ADC Configuration Register 1 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	EAIN15	EAIN14	EAIN13	EAIN12	EAIN11	EAIN10	EAIN9	EAIN8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
0	**EAINx** **(x=0 ~ 15)**	ADC Port Configuration Register 0: Configure AINx as IO PORT 1: Configure ANIx as ADC input and remove pull-up resistance automatically.

OP_CTM1 (C2H@FFH) Customer Option Register 1 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	VREFS	-	-	DISJTG	IAPS[1:0]		-	-

R/W	R/W	-	-	R/W	R/W	R/W	-	-
POR	n	x	x	n	n	n	x	x

Bit Number	Bit Mnemonic	Description
7	**VREFS**	Reference Voltage Selection Bit (Default values are configured by the user and loaded from Code Option) 0: Configure ADC VREF as V_{DD} 1: Configure ADC VREF as internal correct 2.4 V

ADCVL (AEH) ADC Conversion Value Register (Low Bit) (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	ADCV[3: 0]				-	-	-	-
R/W	R/W	R/W	R/W	R/W	-	-	-	-
POR	0	0	0	0	x	x	x	x

ADCVH (AFH) ADC Conversion Value Register (High Bit) (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	ADCV[11: 4]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
11 ~ 4	**ADCV[11: 4]**	ADC conversion value high byte values
3 ~ 0	**ADCV[3: 0]**	ADC conversion value low 4-bit values

IE (A8H) Interrupt Enable Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	EA	EADC	ET2	EUART	ET1	EINT1	ET0	EINT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
6	**EADC**	ADC Interrupt Enable Control Bit 0: EOC/ADCIF interrupt not allowed 1: EOC/ADCIF interrupt allowed

IP (B8H) Interrupt Priority Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	IPADC	IPT2	IPUART	IPT1	IPINT1	IPT0	IPINT0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
6	**IPADC**	ADC Interruption Priority Selection Bit 0: Set the interrupt priority of ADC to be "low" 1: Set the interrupt priority of ADC to be "high"

18.2 ADC Conversion Steps

Operating steps for the user to practically conduct ADC conversion are shown below:

① Configure ADC input pin; (configure corresponding bit of AINx as ADC input, in general, ADC pin will be prefixed);

② Configure ADC reference voltage Vref and ADC conversion frequency;

③ Enable ADC;

④ Select ADC input channel; (Configure ADCIS bit and select ADC input channel);

⑤ Enable ADCS, and start conversion;

⑥ Wait for EOC/ADCIF=1, if ADC interrupt is enabled, ADC interrupt will be generated and the user shall clear EOC/ADCIF flag to 0 by software;

⑦ Obtain 12-bit data from ADCVH, ADCVL from high bit to low bit, and complete a conversion

⑧ If no change in input channel, repeat Step 5 to Step 7 for next conversion.

Note: Before setting up IE[6] (EADC), it is recommended for the user to use software to clear the EOC/ADCIF flag first. After completing ADC interrupt service process, user shall eliminate EOC/ADCIF to avoid generating ADC interrupt constantly.

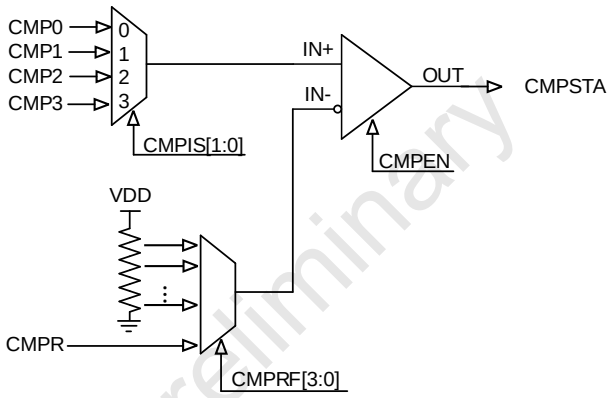
19 Analog comparator

SC92F744XB has a built-in analog comparator, which can be used in alarm circuit, power supply voltage monitoring circuit, zero crossing detection circuit, etc.

This comparator has four analog signal positive inputs: CMP0~3, which can be switched and selected by CMPIS [1:0]. The negative input terminal voltage can be switched to one of the external voltage on the CMPR pin or the internal 16-stage comparison voltage via CMPRF[3:0].

The interrupt mode of the comparator can be conveniently set through CMPIM[1:0]. When the interrupt condition set by CMPIM[1:0] occurs, the comparator interrupt flag CMPIF will be set to 1, which needs software to clear.

19.1 analog comparator structure block biagram



Analog Comparator Structure Block Diagram

CMPCON (B7H) analog comparator control register (read/write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	CMPEN	CMPIF	CMPSTA	-	CMPRF[3:0]			
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
POR	0	0	0	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
7	**CMPEN**	Analog comparator enable control bit 0: Disable analog comparator; 1: Enable analog comparators
6	**CMPIF**	Analog comparator interrupt flag bit 0: comparator interrupt not triggered; 1: When the comparator meets the interrupt trigger condition, this bit will be automatically set to 1 by the hardware. If IE1[5] (ECMP) is also set to 1 at this time, comparator interrupt is generated. After the comparator interrupt occurs, the hardware will not automatically clear this bit, which must be cleared by the user's software.
5	**CMPSTA**	Analog comparator output state 0: positive terminal voltage of comparator is less than negative terminal voltage 1: positive terminal voltage of comparator is greater than negative terminal voltage
3~0	**CMPRF[3:0]**	**Selection of Comparison Voltage at Negative Side of Analog Comparator;** 0000: CMPR is selected as the comparison voltage of analog comparator; 0001: 1/16VDD is selected as the comparison voltage of the analog comparator; 0010: 2/16VDD is selected as the comparison voltage of the analog comparator; 0011: 3/16VDD is selected as the comparison voltage of the analog comparator; 0100: 4/16VDD is selected as the comparison voltage of the analog comparator; 0101: 5/16VDD is selected as the comparison voltage of the analog

		comparator; 0110: 6/16VDD is selected as the comparison voltage of the analog comparator; 0111: 7/16VDD is selected as the comparison voltage of the analog comparator; 1000: 8/16VDD is selected as the comparison voltage of the analog comparator; 1001: 9/16VDD is selected as the comparison voltage of the analog comparator; 1010: 10/16VDD is selected as the comparison voltage of the analog comparator; 1011: 11/16VDD is selected as the comparison voltage of the analog comparator; 1100: 12/16VDD is selected as the comparison voltage of the analog comparator; 1101: 13/16VDD is selected as the comparison voltage of the analog comparator; 1110: 14/16VDD is selected as the comparison voltage of the analog comparator; 1111: 15/16VDD is selected as the comparison voltage of the analog comparator;
4	-	Reserve

CMPCFG (B6H) analog comparator setup register (read/write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	-	CMPIM[1:0]		CMPIS[1:0]	
R/W	-	-	-	-	R/W	R/W	R/W	R/W
POR	x	x	x	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
3~2	**CMPIM[1:0]**	**Analog Comparator Interrupt Mode Selection:** 00: no interruption 01: rising edge interrupt: when IN+ is less than IN- to greater than IN- an interrupt will be generated; 10: Falling Edge Interrupt: when IN+ is greater than IN- to less than IN- an interrupt will be generated; 11: Double Edge Interrupt: when IN+ is less than IN- to greater than IN-, or IN+ is greater than IN- to less than IN- an interrupt will be generated respectively; ;
1~0	**CMPIS[1:0]**	**Selection of Input Channel at Positive End of Analog Comparator:** 00: CMP0 is selected as the input of the positive terminal of the analog comparator; 01: CMP1 is selected as the input of the positive terminal of the analog comparator; 10: CMP2 is selected as the input of the positive terminal of the analog comparator; 11: select CMP3 as the input of the positive terminal of the analog comparator;
7~4	**-**	Reserve

20 EEPROM and IAP Operations

There are two options for the SC92F744XB IAP operating scope:

EEPROM and IAP operating modes are shown below:

1. Internal highest address 128 bytes EEPROM can be used as data storage;
2. The whole 16K bytes of IC ROM and 128 bytes of EEPROM can be used for IAP operations, which is mainly used for remote program updating.

As Code Option, the user shall select IAP operating space before it is written to IC by programmer:

OP_CTM1 (C2H@FFH) Customer Option Register 1 (Read/Write)

Bit number	7	6	5	4	3	2	1	0
Bit Mnemonic	VREFS	-	-	DISJTG	IAPS[1:0]		-	-
R/W	R/W	-	-	R/W	R/W	R/W	-	-
POR	n	x	x	n	n	n	x	x

Bits	Name	Description
3 ~ 2	**IAPS[1: 0]**	EEPROM and IAP Area Selection Bits 00: Code memory prohibits IAP operations, only EEPROM data memory is used for data storage 01: last 0.5k code memory allows IAP operation (7E00H ~ 3FFFH) 10: Last 1k code memory allows IAP operation (7C00H ~ 3FFFH) 11: All code memory allows IAP operation (0000H ~ 3FFFH)

20.1 EEPROM / IAP Operating-related Registers

Description for EEPROM / IAP operating-related registers:

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
IAPKEY	F1H	IAP Protection Register	IAPKEY[7:0]								00000000b
IAPADL	F2H	IAP Write Address Low Register	IAPADR[7:0]								00000000b
IAPADH	F3H	IAP Write Address High Register	-	IAPADR[14:8]							x0000000b
IAPADE	F4H	IAP Write Extended Address Register	IAPADER[7:0]								00000000b
IAPDAT	F5H	IAP Data Register	IAPDAT[7:0]								00000000b
IAPCTL	F6H	IAP Control Register	-	-	-	-	PAYTIMES [1:0]		CMD[1:0]		xxxx0000b

IAPKEY (F1H) IAP Protection Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	IAPKEY[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**IAPKEY[7: 0]**	Enable EEPROM/IAP function and operation time limit configuration, Written values must be non-zero: ① Enable IAP function; ② If no IAP writing command is received after n system clocks, IAP function will be reclosed.

IAPADL (F2H) IAP Write Address Low Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	IAPADR[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**IAPADR[7: 0]**	EEPROM/IAP writing address low byte

IAPADH (F3H) IA Write Address High Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	IAPADR[14:8]						
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
6 ~ 0	**IAPADR[14: 8]**	EEPROM/IAP writing address high 7-bit
7	-	Reserved

IAPADE (F4H) IAP Write Extended Address Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	IAPADER[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**IAPADER[7: 0]**	IAP Extended Address: 0x00: MOVC and IAP programming for Code 0x02: MOVC and IAP programming for EEPROM Other: Reserved

IAPDAT (F5H) IAP Data Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	IAPDAT[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**IAPDAT**	Data written by EEPROM / IAP

IAPCTL (F6H) IAP Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	-	-	PAYTIMES[1: 0]		CMD[1: 0]	
R/W	-	-	-	-	R/W	R/W	R/W	R/W
POR	x	x	x	x	0	0	0	0

Bit Number	Bit Mnemonic	Description
3 ~ 2	**PAYTIMES[1: 0]**	Upon EEPROM/IAP writing operation, CPU Hold Time length configuration 00: Configure CPU HOLD TIME 6mS@16/8/4/1.33MHz 01: Configure CPU HOLD TIME 3mS@16/8/4/1.33MHz 10: Configure CPU HOLD TIME 1mS@16/8/4/1.33MHz 11: Reserved **Note: The CPU Hold is for PC pointer, other functional module continues to work; interrupt flag is saved, and interrupt is generated after completing Hold, but several times of interrupt can only be saved once.** Recommended Selection: 2.7V ~ 5.5 V for $V_{DD,}$ 10 is available 2.4V ~ 5.5V for V_{DD}, 01 or 00 is available
1 ~ 0	**CMD[1: 0]**	EEPROM / IAP writing operating command 10: Write Others: Reserved **Note: The statement of EEPROM/IAP write operation shall be followed by at least 8 NOP instructions to guarantee subsequent instruction can be implemented normally after finishing IAP operation!**

20.2 EEPROM / IAP OPerating procedures:

Writing procedure of the SC92F744XB EEPROM/IAP are shown below:

① Write 0x00 into IAPADE[7: 0]: select Code memory and conduct IAP operation; write 0x02 into IAPADE[7: 0]: select EEPROM and conduct EEPROM reading and writing operations;

② Write data into IAPDAT[7: 0] (data for EEPROM / IAP writing ready);

③ Write address into {IAPADR[14: 8], IAPADR[7: 0]} (target address of EEPROM/IAP operation ready);

④ Write a nonzero value n into IAPKEY[7: 0] (switch on protection of EEPROM / IAP, and EEPROM / IAP function will be switched off when there is no writing command within n system clocks);

⑤ Write CPU Hold time into IAPCTL[3: 0] (configue CPU Hold time by setting CMD[1: 0] to 1 or 0, CPU is Hold up and start up EEPROM/IAP writing);

⑥ EEPROM/IAP writing ends, CPU proceeds to subsequent operations.

Notes:

1. **When programming IC, if "Code memory Prohibits IAP Operations" is selected by Code Option, IAP is unavailable upon IAPADE[7: 0]=0x00 (Select Code memory), meaning it is unable to write data, and such data can only be read by MOVC command.**

2. **When IAPADE=0x01 or 0x02, MOVC instruction and writing operation are conducted in EEPROM or IFB data memory. If any interrupt occurs and there are also MOVC operations in this interrupt, it may result in error of MOVC operations and thus abnormal operation of program. To avoid such situation, the user shall disable global interrupt control bit (EA=0) before conducting IAPADE=0x01 or 0x02 operations. After operation completed, configure IAPADE =0x00 and enable global interrupt control bit (EA=1).**

20.2.1 128 bytes Independent EEPROM Operating Demo program

```
#include ”intrins.h”

unsigned char EE_Add;

unsigned char EE_Data;

unsigned char code * POINT =0x0000;
```

C Demo Program of EEPROM Write Operation:

```
EA = 0;                 // Disable global Interrupt

IAPADE = 0x02;          //Select EEPROM data memory

IAPDAT = EE_Data;       //Transmit data to EEPROM data register

IAPADH = 0x00;          //High-bit address default write 0x00

IAPADL = EE_Add;        //Write EEPROM target address low bit

IAPKEY = 0xF0;          //This value can be adjusted as required, it shall guarantee that
```

```
                    // The time interval between this instruction implemented and writing IAPCTL value shall be less than 240 (0xf0) system clocks, or else, IAP function is closed;

                    //Pay special attention to enabling interrupt;

IAPCTL = 0x0A;      //Implement EEPROM write operation, 1.5ms@16/8/4/1.33MHz;

_nop_ ();           //Wait (at least 8 _nop_ ())

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

IAPADE = 0x00;      //Return to ROM data memory

EA = 1;             //Enable master interrupt
```

C Demo Program of EEPROM Read Operation

```
EA = 0;                         //Disable master interrupt

IAPADE = 0x02;                  //Select EEPROM data memory

EE_Data = * ( POINT +EE_Add);   //Read value in IAP_Add to IAP_Data

IAPADE = 0x00;                  //Return to ROM data memory, prevent MOVC operates to EEPROM

EA = 1;                         // Enable global interrupt
```

20.2.2 16K bytes Code memory IAP Operating Demo program

```
#include ”intrins.h”

unsigned int IAP_Add;

unsigned char IAP_Data;

unsigned char code * POINT =0x0000;
```

C Demo Program of IAP Write Operation:

```
IAPADE = 0x00;      //Select Code memory
```

```
IAPDAT = IAP_Data;              //Transmit data to IAP data register

IAPADH = (unsigned char) ( (IAP_Add >> 8));     //Write IAP target address high bit

IAPADL = (unsigned char)IAP_Add;                //Write IAP target address low bit

IAPKEY = 0xF0; //This value can be adjusted as required; it shall guarantee this //instruction is
                                                implemented to assigned IAPTL value;

                                // Time interval shall be less than 240 (0xf0) system clocks, or
                            else, IAP function is closed;

                                //Pay special attention upon starting interrupt

IAPCTL = 0x0A;                  //Implement EEPROM write operation, 1.5ms@16/8/4/1.33MHz;

_nop_ ();                       //Wait (at least 8 _nop_ ())

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();

_nop_ ();
```

C Demo Program of IAP Read Operation:

```
IAPADE = 0x00;                  //Select Code memory

IAP_Data = * ( POINT+IAP_Add);  //Read value in IAP_Add to IAP_Data
```

Note: IAP operation in 16K bytes Code memory has certain risks, the user shall implement corresponding safety measures in software. Incorrect operation may result in the user program to be rewritten! Unless such function is required by the user (such as used for remote program update, etc.), it is not recommended to used by the user.

21 CheckSum Module

The SC92F744XB is equipped with a check sum module, which is used for generating 16-bit check sum of code memory in real time. The user can compare such check sum with theoretical value to monitor whether the contents in code memory are correct.

Note: Check sum is the sum of data in the whole code memory, which is the data of 0000H ~ 3FFDH address unit. If there are residual values from the user's last operations in address unit, it may result in inconsistency of check sum with theoretical value. Therefore, it is recommended that the user shall erase the whole Code memory or write 0 before programming code so as to guarantee the consistency between check sum and theoretical value.

21.1 CheckSum-Related Registers

CHKSUML (FCH) Check Sum Result Register Low Bit (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	CHKSUML[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**CHKSUML [7: 0]**	CheckSum Result Register Low Bit

CHKSUMH (FDH) Check Sum Result Register High Bit (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	CHKSUMH[7: 0]							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**CHKSUMH [7: 0]**	CheckSum Result Register High Bit

OPERCON (EFH) Arithmetic Control Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	OPERS	MD	-	-	-	-	-	CHKSUMS
R/W	R/W	R/W	-	-	-	-	-	R/W
POR	0	0	x	x	x	x	x	0

Bit Number	Bit Mnemonic	Description
0	**CHKSUMS**	CheckSum Operation Starts Trigger **C**ontrol Bit (Start) Write “1” for this bit, start to conduct Check sum calculation. This bit is valid for only writing 1.

22 Electrical Characteristics

22.1 Absolute Maximum Ratings

Symbol	Parameter	Min Value	Max Value	Unit
VDD/VSS	DC supply voltage	-0.3	5.5	V
Voltage ON any Pin	Input/output voltage of any pin	-0.3	V_{DD}+0.3	V
T_A	Ambient temperature	-40	85	℃
T_{STG}	Storage temperature	-55	125	℃

22.2 Recommended Operating Conditions

Symbol	Parameter	Min Value	Max Value	Unit	System Clock Frequency
V_{DD1}	Operating voltage	2.9	5.5	V	16MHz
V_{DD2}	Operating voltage	2.4	5.5	V	8/4/1.33MHz
T_A	Ambient temperature	-40	85	℃	

22.3 DC Characteristics

22.3.1 VDD = 5V, TA = +25℃, unless otherwise specified

Symbol	Parameter	Min Value	Typical Value	Max Value	Unit	Testing Conditions
Current						

I_{op1}	Operating current	-	9	-	mA	f_{SYS}=16MHz
I_{op2}	Operating current	-	6	-	mA	f_{SYS}=8MHz
I_{op3}	Operating current	-	5	-	mA	f_{SYS}=4MHz
I_{op4}	Operating current	-	4	-	mA	f_{SYS}=1.33MHz
I_{pd1}	Standby Current (Power Down Mode)	-	0.7	1.0	μA	
I_{IDL1}	Standby Current (IDLE Mode)	-	7	-	mA	
I_{BTM}	Base Timer Operating Current	-	6	10	μA	BTMFS[3: 0]= 1000 One interrupt occurs for every 4.0 seconds
I_{WDT}	WDT Current	-	6	10	μA	WDTCKS[2: 0]= 000 WDT overflows every 500ms
IO Port Features						
V_{IH1}	Input high voltage	$0.7V_{DD}$	-	V_{DD}+0.3	V	
V_{IL1}	Input low voltage	-0.3	-	$0.3V_{DD}$	V	
V_{IH2}	Input high voltage	$0.8V_{DD}$	-	V_{DD}	V	Schmidt trigger input: RST/tCK/SCK
V_{IL2}	Input low voltage	-0.2	-	$0.2V_{DD}$	V	
I_{OL1}	Output low current	-	40	-	mA	V_{Pin}=0.4V
I_{OL2}	Output low current	-	65	-	mA	V_{Pin}=0.8V

I_{OH1}	Output high current P3H-P5	-	18	-	mA	V_{Pin}=4.3V
I_{OH2}	Output high current P3H-P5	-	9	-	mA	V_{Pin}=4.7V
I_{OH3}	Output high current P0-P3L	-	18	-	mA	V_{Pin}=4.3V Pxyz=0,I_{OH} level 0
	Output high current P0-P3L	-	9	-	mA	V_{Pin}=4.3V Pxyz=1,I_{OH} level 1
	Output high current P0-P3L	-	5	-	mA	V_{Pin}=4.3V Pxyz=2,I_{OH} level 2
	Output high current P0-P3L	-	2	-	mA	V_{Pin}=4.3V Pxyz=3,I_{OH} level 3
I_{OH4}	Output high current P0-P3L	-	9	-	mA	V_{Pin}=4.7V Pxyz=0,I_{OH} level 0
	Output high current P0-P3L	-	5	-	mA	V_{Pin}=4.7V Pxyz=1,I_{OH} level 1
	Output high current P0-P3L	-	2	-	mA	V_{Pin}=4.7V Pxyz=2,I_{OH} level 2
	Output high current P0-P3L	-	1	-	mA	V_{Pin}=4.7V Pxyz=3,I_{OH} level 3
R_{PH1}	Pull-up resistance	-	30	-	kΩ	
Internal calibrated 2.4V as ADC reference voltage						
V_{DD24}	Internal reference 2.4V voltage output	2.37	2.40	2.45	V	T_A=-40~85℃

22.3.2 VDD = 3.3V, TA = +25℃, unless otherwise specified

Symbol	Parameter	Min Value	Typical Value	Max Value	Unit	Testing Conditions
Current						
I_{op5}	Operating current	-	6	-	mA	f_{SYS}=16MHz
I_{op6}	Operating current	-	5	-	mA	f_{SYS}=8MHz
I_{op7}	Operating current	-	4	-	mA	f_{SYS}=4MHz
I_{op8}	Operating current		3	-	mA	f_{SYS}=1.33MHz
I_{pd2}	Standby Current (Power Down Mode)	-	0.6	1	uA	
I_{IDL2}	Standby Current (IDLE Mode)	-	5	-	mA	
I/O Port Features						
V_{IH3}	Input high voltage	0.7V_{DD}	-	V_{DD}+0.3	V	
V_{IL3}	Input low voltage	-0.3	-	0.3V_{DD}	V	
V_{IH4}	Input high voltage	0.8V_{DD}	-	V_{DD}	V	Schmidt trigger input: RST/tCK/SCK
V_{IL4}	Input low voltage	-0.2	-	0.2V_{DD}	V	
I_{OL3}	Output low current	-	30	-	mA	V_{Pin}=0.4V
I_{OL4}	Output low current	-	55	-	mA	V_{Pin}=0.8V
I_{OH5}	Output high current	-	6	-	mA	V_{Pin}=3.0V
R_{PH2}	Pull-up resistance	-	55	-	kΩ	

Internal calibrated 2.4V as ADC reference voltage						
V_{DD24}	Internal reference 2.4V voltage output	2.37	2.40	2.45	V	T_A=-40~85℃

22.4 AC Characteristics

(V_{DD} = 2.4V ~ 5.5V, TA = 25℃, unless otherwise specified)

Symbol	Parameter	Min	Typical	Max	Unit	Conditions
T_{OSC}	External 32K oscillator start-up time	-	-	1	s	External 32K crystal oscillator
T_{POR}	Power On Reset time	-	5	10	ms	
T_{PDW}	Power Down Mode waking-up time	-	1	1.5	ms	
T_{Reset}	Reset Pulse Width	18	-	-	μs	Valid for Low level
f_{HRC}	RC oscillation stability	15.84	16	16.16	MHz	V_{DD}=3.0~5.5V T_A=-20~85 ℃

22.5 ADC Characteristics

(T_A= 25℃, unless otherwise specified)

Symbol	Parameter	Min	Typical	Max	Unit	Conditions
V_{AD}	Supply Voltage	2.4	5.0	5.5	V	
N_R	Precision	-	12	-	bit	GND≤V_{AIN}≤V_{DD}
V_{AIN}	ADC Input Voltage	GND	-	V_{DD}	V	

R_{AIN}	ADC input resistance	1	-		MΩ	V_{IN}=5V
I_{ADC1}	ADC conversion current 1	-	-	2	mA	ADC Module on V_{DD}=5V
I_{ADC2}	ADC conversion current 2	-	-	1.8	mA	ADC module on V_{DD}=3.3V
DNL	Differential nonlinear error	-	-	±3	LSB	V_{DD}=5V V_{REF}=5V
INL	Integral nonlinear error	-	-	±4	LSB	
E_Z	Offset error	-	-	±7	LSB	
E_F	Full scale error	-	-	±8	LSB	
E_{AD}	Total absolute error	-	-	±8	LSB	
T_{ADC1}	ADC conversion time 1	-	7.5	-	μs	ADC Clock = 2.67MHz ADC sampling period = 6
T_{ADC2}	ADC conversion time 2	-	15	-	μs	ADC Clock = 1.33MHz ADC sampling period = 6

22.6 Analog Comparator Characteristics

(V_{DD} =5V, T_A = 25℃, unless otherwise specified)

Symbol	Parameter	Min	Typical	Max	Unit	Conditions
V_{CM}	Common Mode Input Voltage	0	-	V_{DD}	V	
V_{OS}	Input Offset voltage	-	10	30	mV	

V_{HYS}	Hysteresis voltage	-	25	-	mV	
I_{CMP}	Comparator conversion current	-	-	100	μA	V_{DD}=5V
T_{CMP}	Response time	-	-	2	μs	

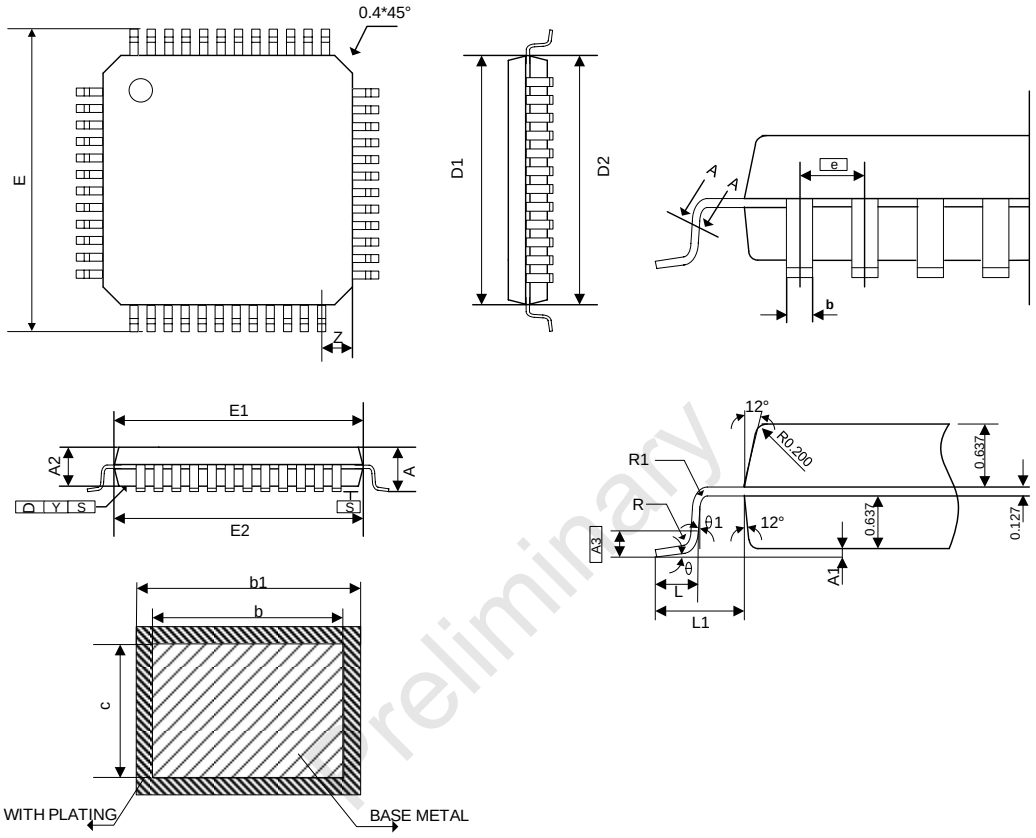
23 Ordering Information

PRODUCT NO	PKG	PACKING
SC92F7447BP48R	LQFP48	TRAY
SC92F7446BP44R	LQFP44	TRAY
SC92F7445BP32R	LQFP32	TUBE

24 Packageing Information

SC92F7447BP48R

LQFP48 Dimension Unit: mm

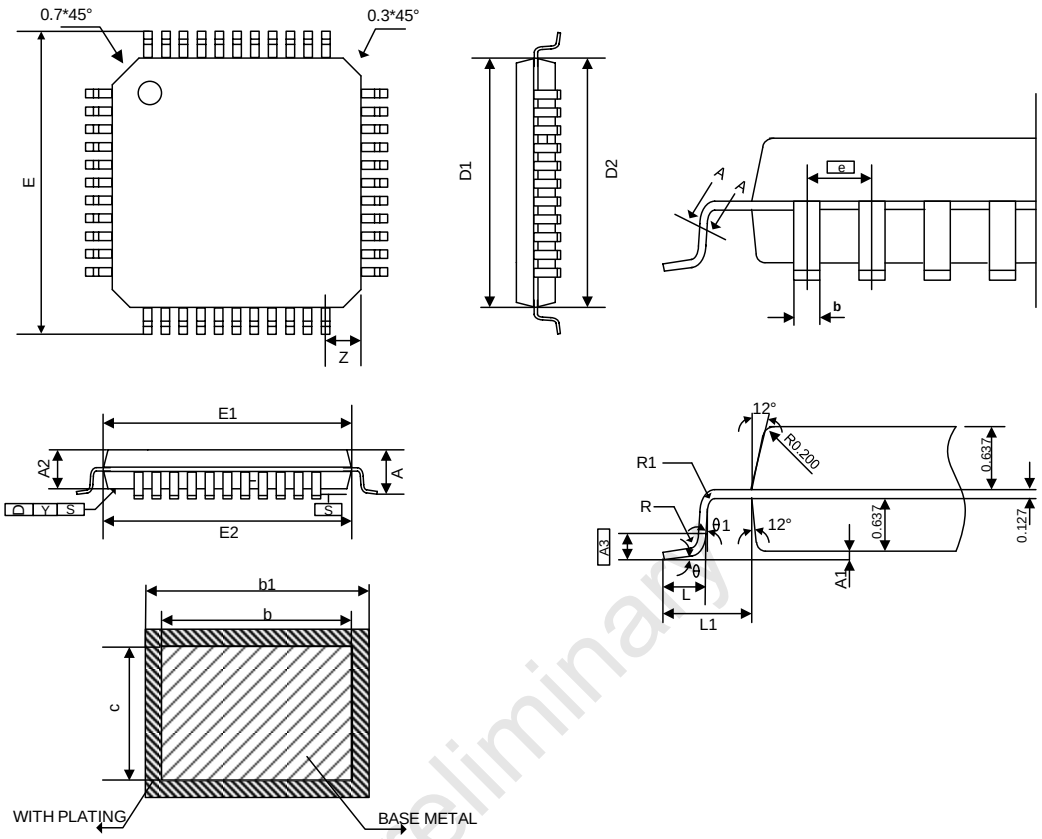


Symbol	mm (milimetre)		
	Min	Normal	Max
A	1.45	1.55	1.65
A1	0.01	--	0.21
A2	1.3	1.4	1.5
A3	--	0.254	--
b	0.15	0.20	0.25
b1	0.16	0.22	0.28

c	--	0.127	--
D1	6.85	6.95	7.05
D2	6.9	7.00	7.10
E	8.8	9.00	9.20
E1	6.85	6.95	7.05
E2	6.9	7.00	7.10
e	--	0.5	--
L	0.43	--	0.71
L1	0.90	1.0	1.10
R	0.1	--	0.25
R1	0.1	--	--
θ	0°	--	10°
θ1	0°	--	--
y	--	--	0.1
Z	--	0.75	--

SC92F7446BP44R

LQFP44 Dimension Unit: mm

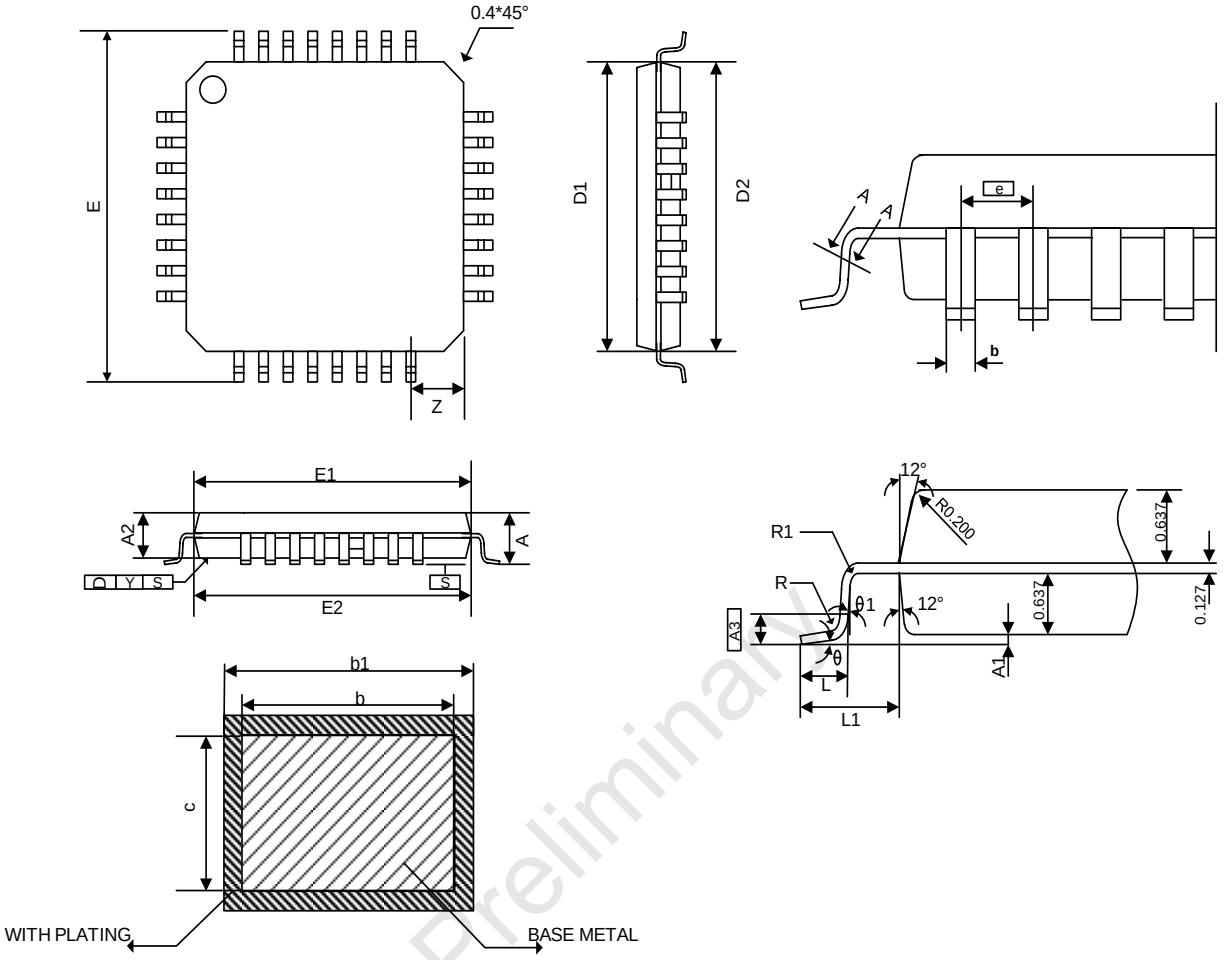


Symbol	mm (milimetre)		
	Min	Normal	Max
A	1.45	1.55	1.65
A1	0.015	--	0.21
A2	1.3	1.4	1.5
A3	--	0.254	--
b	0.25	0.30	0.35
b1	0.26	0.32	0.38
c	--	0.127	--

D1	9.85	9.95	10.05
D2	9.9	10.00	10.10
E	11.8	12.00	12.20
E1	9.85	9.95	10.05
E2	9.9	10.00	10.10
e	--	0.8	--
L	0.42	--	0.72
L1	0.95	1.0	1.15
R	0.1	--	0.25
R1	0.1	--	--
θ	0°	--	10°
θ1	0°	--	--
y	--	--	0.1
Z	--	1.0	--

SC92F7445BP32R

LQFP32 Dimension Unit: mm



Symbol	mm (milimetre)		
	Min	Normal	Max
A	1.45	1.55	1.65
A1	0.01	--	0.21
A2	1.3	1.4	1.5
A3	--	0.254	--
b	0.30	0.35	0.40
b1	0.31	0.37	0.43
c	--	0.127	--

D1	6.85	6.95	7.05
D2	6.9	7.00	7.10
E	8.8	9.00	9.20
E1	6.85	6.95	7.05
E2	6.9	7.00	7.10
e	--	0.8	--
L	0.43	--	0.71
L1	0.90	1.0	1.10
R	0.1	--	0.25
R1	0.1	--	--
θ	0°	--	10°
θ1	0°	--	--
y	--	--	0.1
Z	--	0.70	--

25 Revision History

Version	Notes	Date
V0.1	Initial Release.	May 2020