High-speed 1T 8051-based Flash MCU, 512 bytes SRAM, 8 Kbytes Flash, 128 bytes independent EEPROM, 12-bit ADC, 6-channel 8-bit PWM Outputs, 3 Timer/Counters, UART

1 General Description

The SC92F7302/7301/7300 (hereinafter referred to as the SC92F730X) is a series of enhanced 1T 8051-based industrial Flash Microcontroller unit (MCU), in which the instruction system is completely compatible with standard 8051 product series.

The SC92F730X integrates 8K bytes Flash ROM, 512 bytes SRAM, 128 bytes EEPROM, up to 18 General-purpose I/Os (GPIO), 8 IO external interrupters, three 16-bit timers, 9-channel 12-bit high-precision ADC, 6-channel independent 8-bit PWM,IO drive level control (P0 and P2 ports), internal ±1% high-precision 24/12/6/2MHz high-frequency oscillator and low-frequency 32KHz oscillator, UART, and other communication interfaces. To improve the reliability and simplify the circuit design, the SC92F730X also integrates 4-level optional LVR, 2.4V ADC reference voltage, WDT and other high-reliability circuits.

The SC92F730X features excellent anti-interference performance, which make it possible to be widely applied to IoT control, such as intelligent home appliances, home automation, charger, power supply aviation model, intercom, wireless communication, gaming machines.

2 Features

Operating Voltage: 2.4V ~ 5.5V

Operating Temperature: -40 ~ 85℃

EMS

- ESD
  - HBM：MIL-STD-883J Class 3B
  - MM：JEDEC EIA/JESD22-A115 Class C
  - CDM：ANSI/ESDA/JEDEC JS-002-2018 Class C3
- EFT
  - EN61000-4-4 Level 4

Package:

SOP20/TSSOP20/QFN20

SOP16

SOP8

Core: High speed 1T 8051

Flash ROM: 8K bytes Flash ROM (MOVC) prohibited addressing 0000H ~ 00FFH) can be rewritten for 10, 000 times

IAP:

- Can be Code option into 0K, 0.5K, 1K or 8K
- When the user performs IAP operations on the Flash ROM, the LVR should be set to 3.7V or higher, and the VDD voltage should be set between 3.7V and 5.5V
- When the user performs IAP operations on the EEPROM, the VDD voltage should be set between 2.4V and 5.5V

Attention: The programming voltage can only be 5V, when the chip is being programmed by users!

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

SRAM: Internal 256 bytes + external 256 bytes

System Clock (f_{SYS}):

- Built-in high-frequency 24MHz oscillator (f_{HRC}):
  - The system clock source of IC can be set to one of the four types of frequency division by programmer selection:24MHz@3.7~5.5V, 12/6/2MHz@2.4~5.5V
  - Frequency Error: no more than ±1% of frequency error in 2.4V ~ 5.5V and -40 ~ 85℃ application environment

Built-in Low-Frequency 32kHz Oscillator (LRC):

- Clock source of Base Timer (BTM), which can wake up the SC92F730X from stop mode
- Clock source of Watchdog (WDT)

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

Interruption (INT):

- 9 interrupt sources: Timer0, Timer1, Timer2, INT0, INT2, ADC, PWM, UART, Base Timer
- 2 external interrupt vectors shared by 8 interrupt ports, all of which can be defined in rising-edge, falling-edge or dual-edge trigger mode
- Two-level interrupt priority capability

Digital Peripheral:

- Up to 18 bidirectional independently controllable I/O interfaces, able to configure pull-high resistor independently
- P0/P2 ports with 4-level drive capability
- All I/Os equipped with sink current drive capability (70 mA)
- 11-bit WDT with optional clock division ratio
- 3 standard 80C51 Timer/Counters: Timer0, Timer1 and Timer2
- 6-channel 8-bit PWM output channels with variable period and individual duty cycle
- 5 I/Os as output of the 1/2-bias LCD COM
- 1 independent UART communication interface

Analog Peripheral:

- 9-channel 12-bit ADC
  - Built-in 2.4V reference voltage
  - 2 options for ADC reference voltage: V_{DD} and internal 2.4V
  - Internal one-channel ADC, where VDD can be measured directly
  - ADC conversion completion interrupt

Power Saving Mode:

- IDLE Mode: can be woken up by any interrupt.
- STOP Mode: can be woken up by INT0, INT2、Base Timer

Naming Rules for 92 Series Products

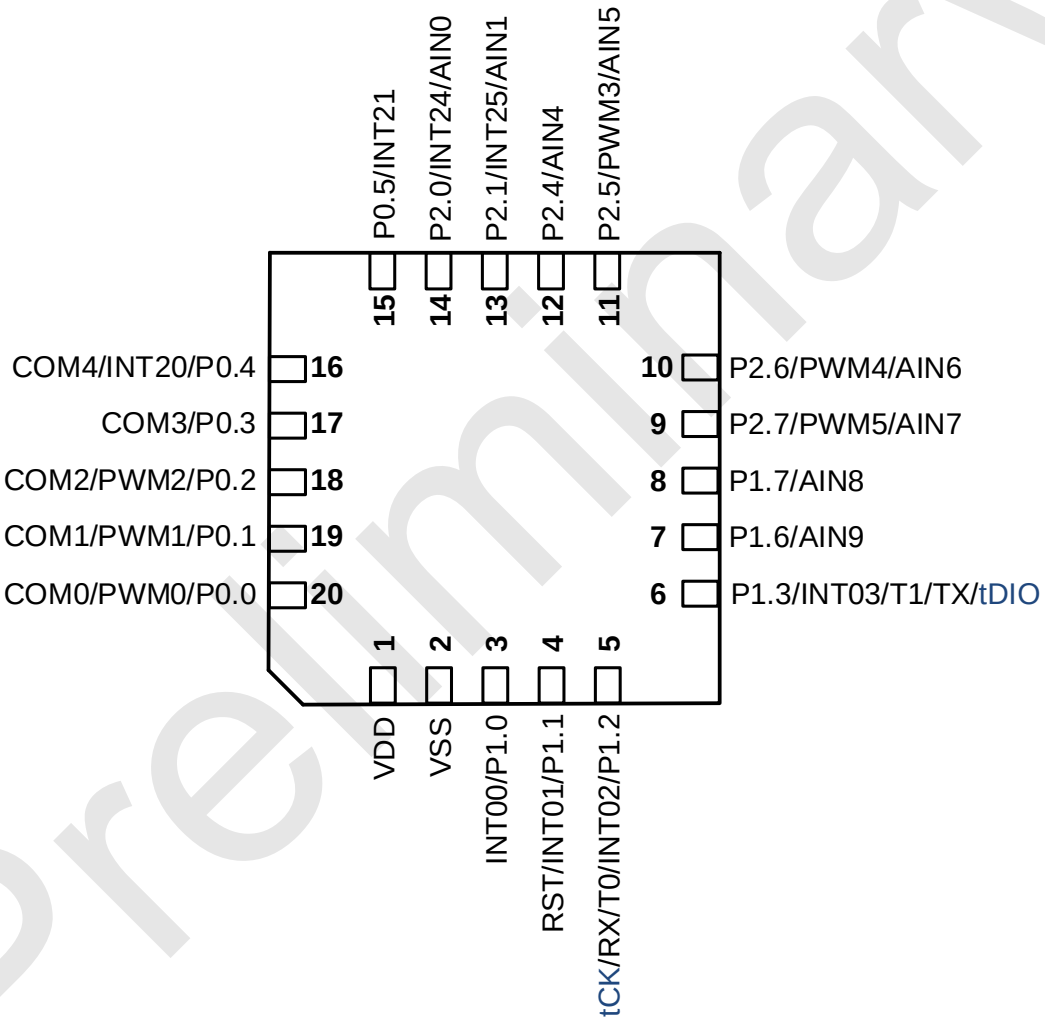
Name	SC	92	F	7	3	0	2	X	M	20	U
S/R	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪

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; N: NSOP, F: QFP; P: LQFP; Q: QFN; K: SKDIP)
⑩	Number of Pins.
⑪	Packaging Mode: (U: Tube; R: Tray; T: Reel)

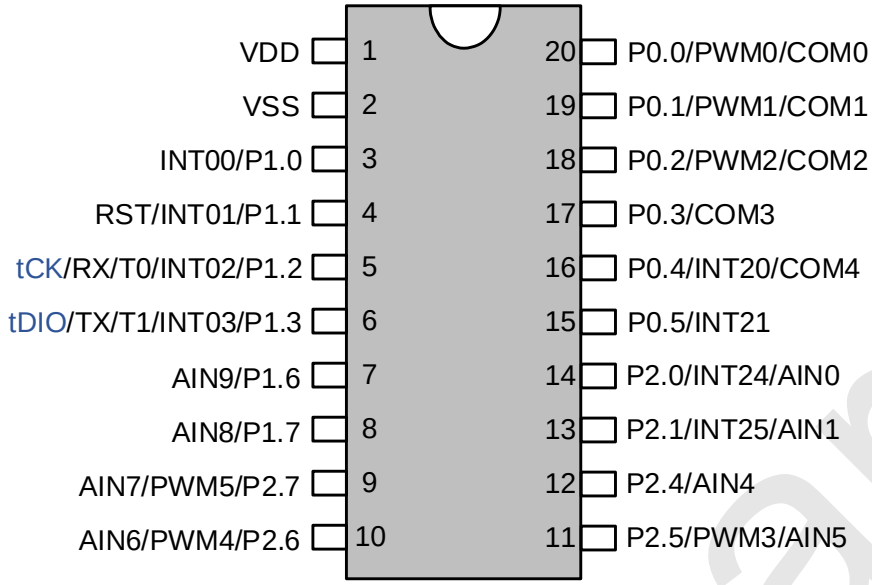
Contents

3 Pin Description

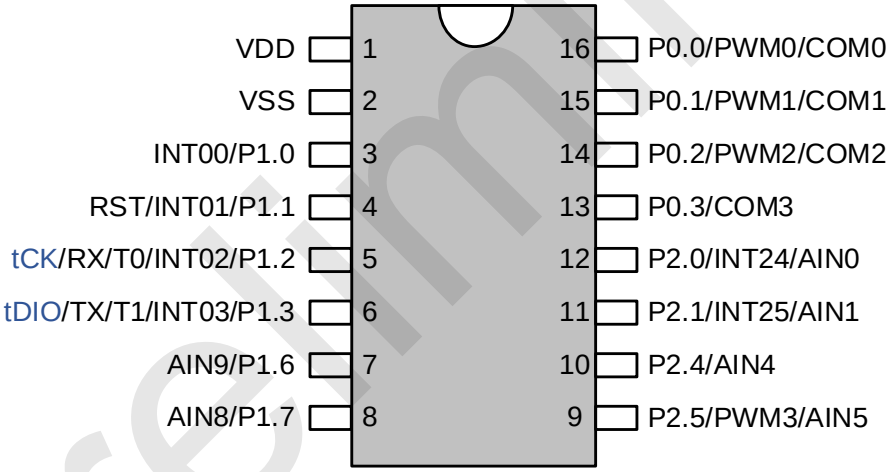
3.1 Pin Configuration



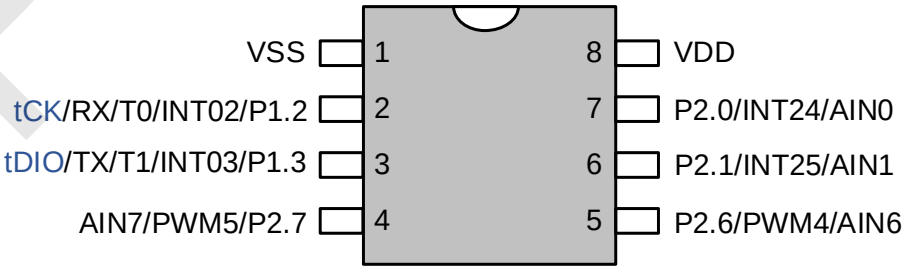
The SC92F7302 Pin Diagram
Suitable for QFN20 package



The SC92F7302 Pin Diagram
Suitable for SOP20/TSSOP20 package



The SC92F7301 Pin Diagram
Suitable for SOP16 package



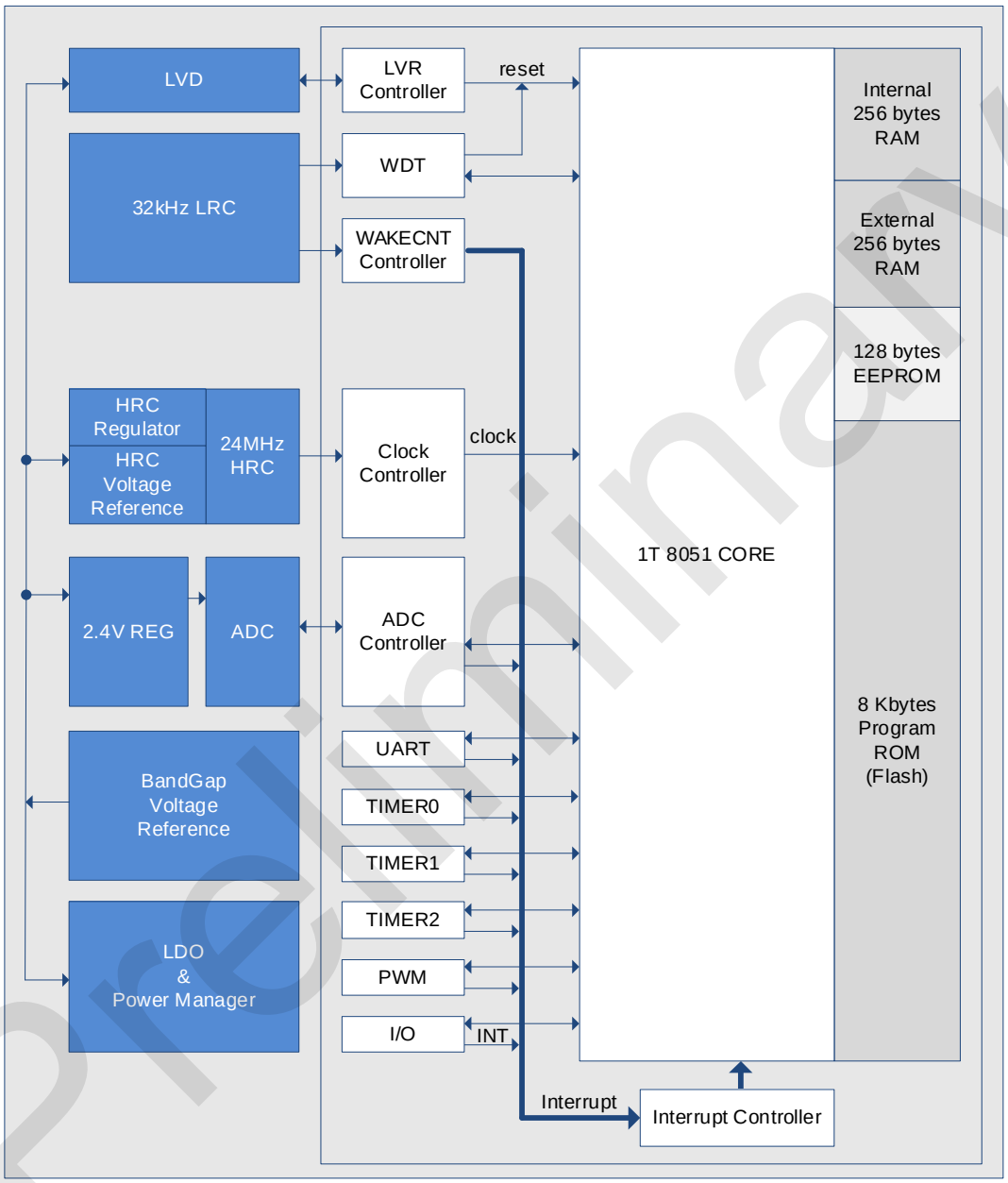
The SC92F7300 Pin Diagram
Suitable for SOP8 package

3.2 Pin Definition

Pin number			Pin Name	Type	Description
20PIN	16PIN	8PIN			
1	1	8	**VDD**	Power	Power
2	2	1	**VSS**	Power	Ground
3	3	-	**P1.0/INT00**	I/O	P1.0: GPIO P1.0 INT00: Input 0 of external interrupt 0
4	4	-	**P1.1/INT01/RST**	I/O	P1.1: GPIO P1.1 INT01: Input 1 of external interrupt 0 RST: Reset Pin
5	5	2	**P1.2/INT02/T0/RX/tCK**	I/O	P1.2: GPIO P1.2 INT02: Input 2 of external interrupt 0 T0: Timer/Counter 0 External Input RX0: UART0 Receiver tCK: Programming and Emulation Clock Pin
6	6	3	**P1.3/INT03/T1/TX/tDIO**	I/O	P1.3: GPIO P1.3 INT03: Input 3 of external interrupt 0 T1: Timer/Counter 1 External Input TX0: UART 0 Transmitter tDIO: Programming and Emulation Data Pin
7	7	-	**P1.6/AIN9**	I/O	P1.6: GPIO P1.6 AIN9: ADC Input Channel 9
8	8	-	**P1.7/AIN8**	I/O	P1.7: GPIO P1.7 AIN8: ADC Input Channel 8
9	-	4	**P2.7/PWM5/AIN7**	I/O	P2.7: GPIO P2.7 PWM5: PWM5 Output AIN7: ADC Input Channel 7
10	-	5	**P2.6/PWM4/AIN6**	I/O	P2.6: GPIO P2.6 PWM4: PWM4 Output AIN6: ADC Input Channel 6

11	**9**	**-**	**P2.5/PWM3/AIN5**	I/O	P2.5: GPIO P2.5 PWM3: PWM3 Output AIN5: ADC Input Channel 5
12	**10**	**-**	**P2.4/AIN4**	I/O	P2.4: GPIO P2.4 AIN4: ADC Input Channel 4
13	**11**	**6**	**P2.1/INT25/AIN1**	I/O	P2.1: GPIO P2.1 INT25: Input 5 of external interrupt 2 AIN1: ADC Input Channel 1
14	**12**	**7**	**P2.0/INT24/AIN0**	I/O	P2.0: GPIO P2.0 INT24: Input 4 of external interrupt 2 AIN0: ADC Input Channel 0
15	**-**	**-**	**P0.5/INT21**	I/O	P0.5: GPIO P0.5 INT21: Input 1 of external interrupt 2
16	**-**	**-**	**P0.4/INT20/COM4**	I/O	P0.5: GPIO P0.5 INT21: Input 1 of external interrupt 2 COM4: LCD common drive output 4
17	**13**	**-**	**P0.3/COM3**	I/O	P0.3: GPIO P0.3 COM3: LCD common drive output 3
18	**14**	**-**	**P0.2/PWM2/COM2**	I/O	P0.2: GPIO P0.2 PWM2: PWM2 Output COM2: LCD common drive output 2
19	**15**	**-**	**P0.1/PWM1/COM1**	I/O	P0.1: GPIO P0.1 PWM1: PWM1 Output COM1: LCD common drive output 1
20	**16**	**-**	**P0.0/PWM0/COM0**	I/O	P0.0: GPIO P0.0 PWM0: PWM0 Output COM0: LCD common drive output 0

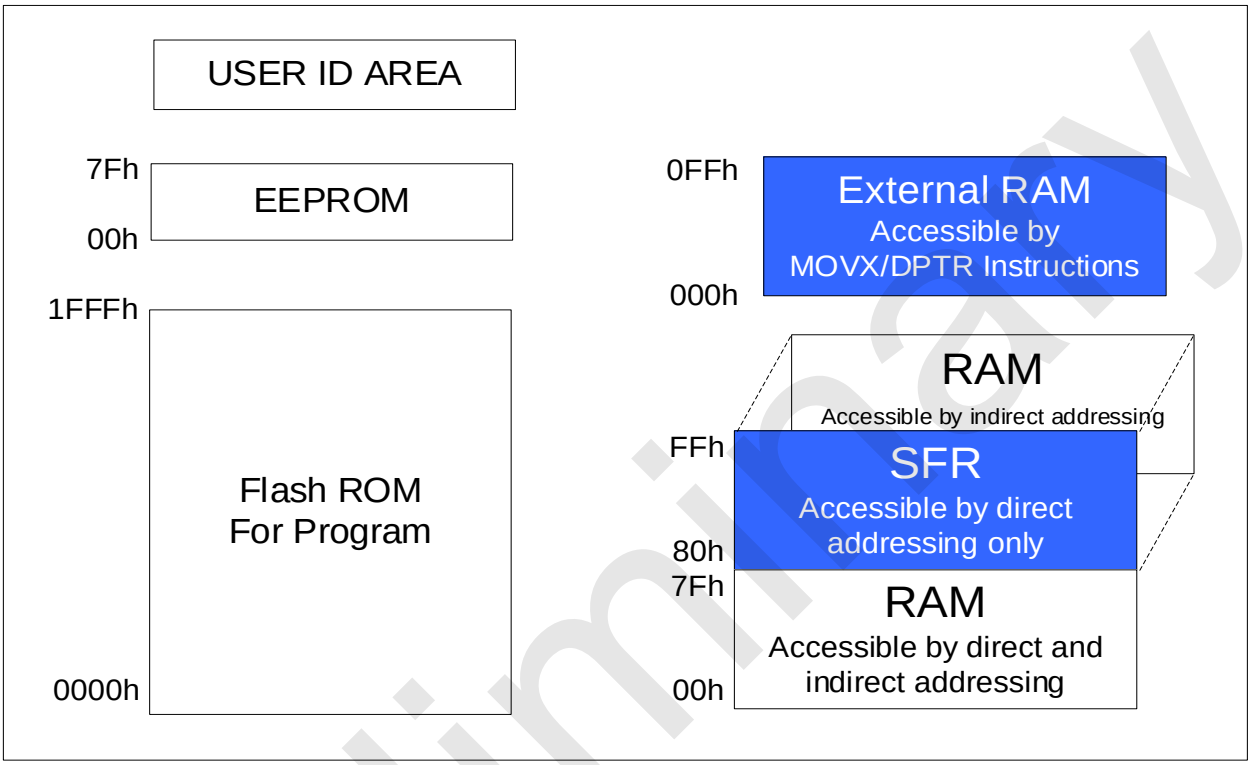
4 Inner Block Diagram



SC92F730X BLOCK DIAGRAM

5 Flash ROM and SRAM Structure

The structures of the SC92F730X Flash ROM and SRAM are shown as follows:



Flash ROM and SRAM Structure Diagram

5.1 Flash ROM

The SC92F730X provides 8 Kbytes of Flash ROM with the ROM address of 0000H ~ 1FFFH. These 8 Kbytes of Flash ROM can be rewritten 10,000 times, which is able to program and erase by specialized ICP programming device (SC LINK/SC LINK PRO) provided by SinOne.

The MOVC instruction is non-addressable within 256 bytes. That is to say, it is unable to read the contents of the 256 bytes region by program, so as to realize the encryption function of chip program. For more details, refer to "SinOne SC92F Series MCU Application Guide".

EEPROM is a data memory separated from 8K 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 17 EEPROM and IAP Operations. **Note：The number of erasure of EEPROM is 100,000 times. The user should not exceed the rated burn number of EEPROM, otherwise there will be an exception!**

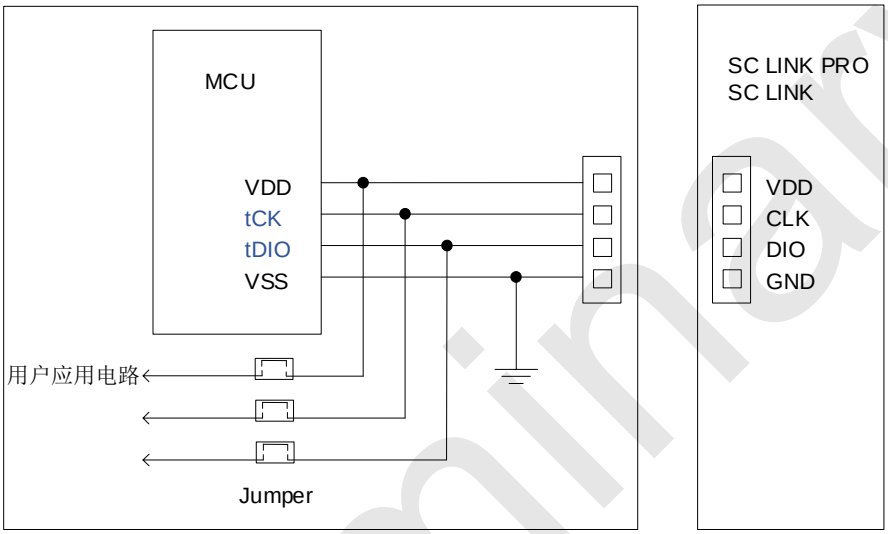
User ID area: The user ID is written when leave the factory, and the user can only perform reading operations in this area. For more details, refer to 17 EEPROM and IAP Operations.The SC92F730X 8 Kbytes Flash ROM provide Empty Check, Program, Verify and Erase function other than Read function. This Flash ROM and EEPROM usually needs no Erase operation before writing. Directly writing data can realize coverage of new data.

Attention:

1. **Users should set LVR to 3.7V or higher and ensure VDD is between 3.7V and 5.5V when performing IAP operations on Flash ROM**

2. **Users should ensure VDD is between 2.4V and 5.5V when performing IAP operations on EEPRON**

The SC92F730X 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

Attention: The programming voltage can only select 5V,when the user is programming the chip

5.2 Customer Option Memory (User Programming Setting)

A separate Flash data memory is embedded inside the SC92F730X, 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	-	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
OP_CTM1	C2H@FFH	Customer Option Register 1	VREFS	-	-	-	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	-	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
R/W	R/W	-	R/W		R/W	R/W	R/W	
POR	n	x	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)
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 P1.1 as External Reset input pin 1: configure P1.1 as GPIO
2	**DISLVR**	LVR enable bit 0: LVR enable 1:LVR disable

Bit Number	Bit Mnemonic	Description
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
6	**-**	Reserved

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

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	VREFS	-	-	-	IAPS[1: 0]		-	-
R/W	R/W	-	-	-	R/W	R/W	-	-
POR	n	x	x	x	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
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 (3E00H ~ 3FFFH) 10: Last 1k code memory allows IAP operation (3C00H ~ 3FFFH) 11: All code memory allows IAP operation (0000H ~ 3FFFH)
6 ~ 4, 1 ~ 0	**-**	Reserved

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	7	6	5	4	3	2	1	0	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 SC92F730X Microcontroller Unit, which integrates SRAM of 512 bytes, is divided into internal 256 bytes RAM and external 256 bytes 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 is also from 80H to FFH. But 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 000H to 0FFH, which is addressed by MOVX instruction.

5.3.1 Internal 256 bytes SRAM

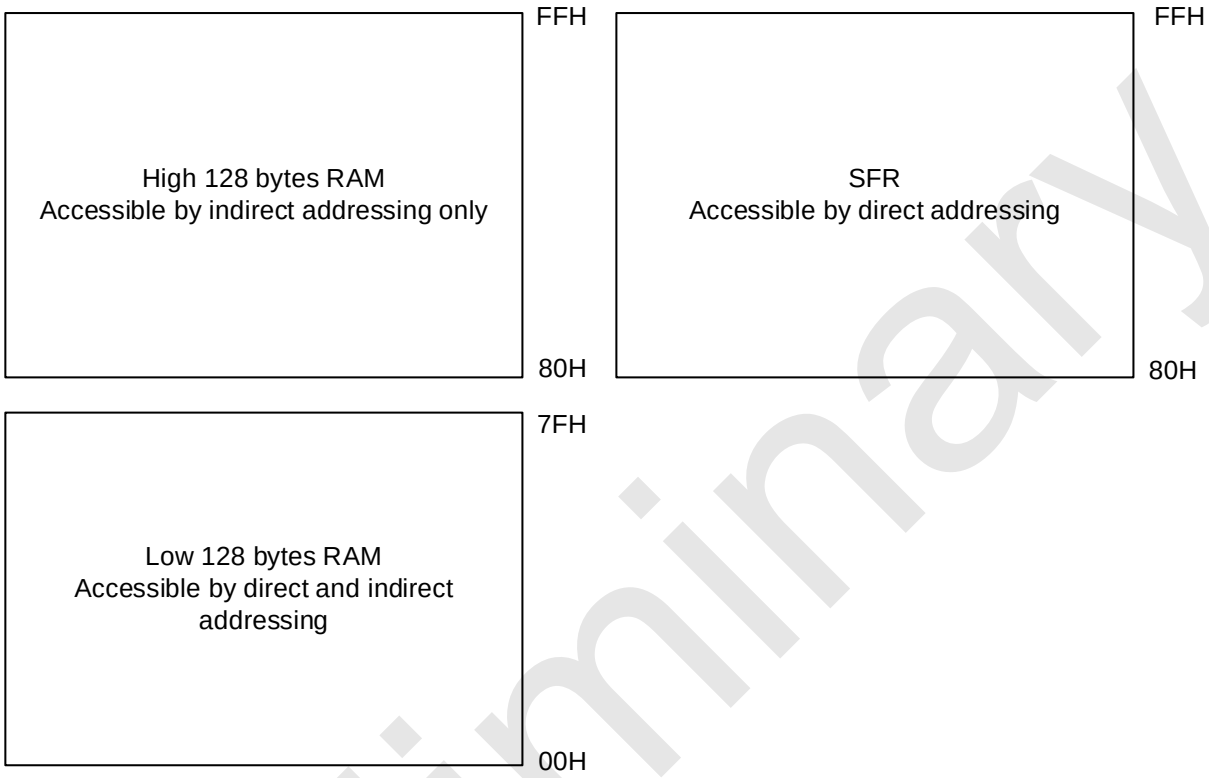
Low 128 bytes SRAM area is divided into three parts:

① Register bank 0 ~ 3, address from 00H to 1FH. The active bank is selected by bits RS1 and RS0 of PSW register. 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

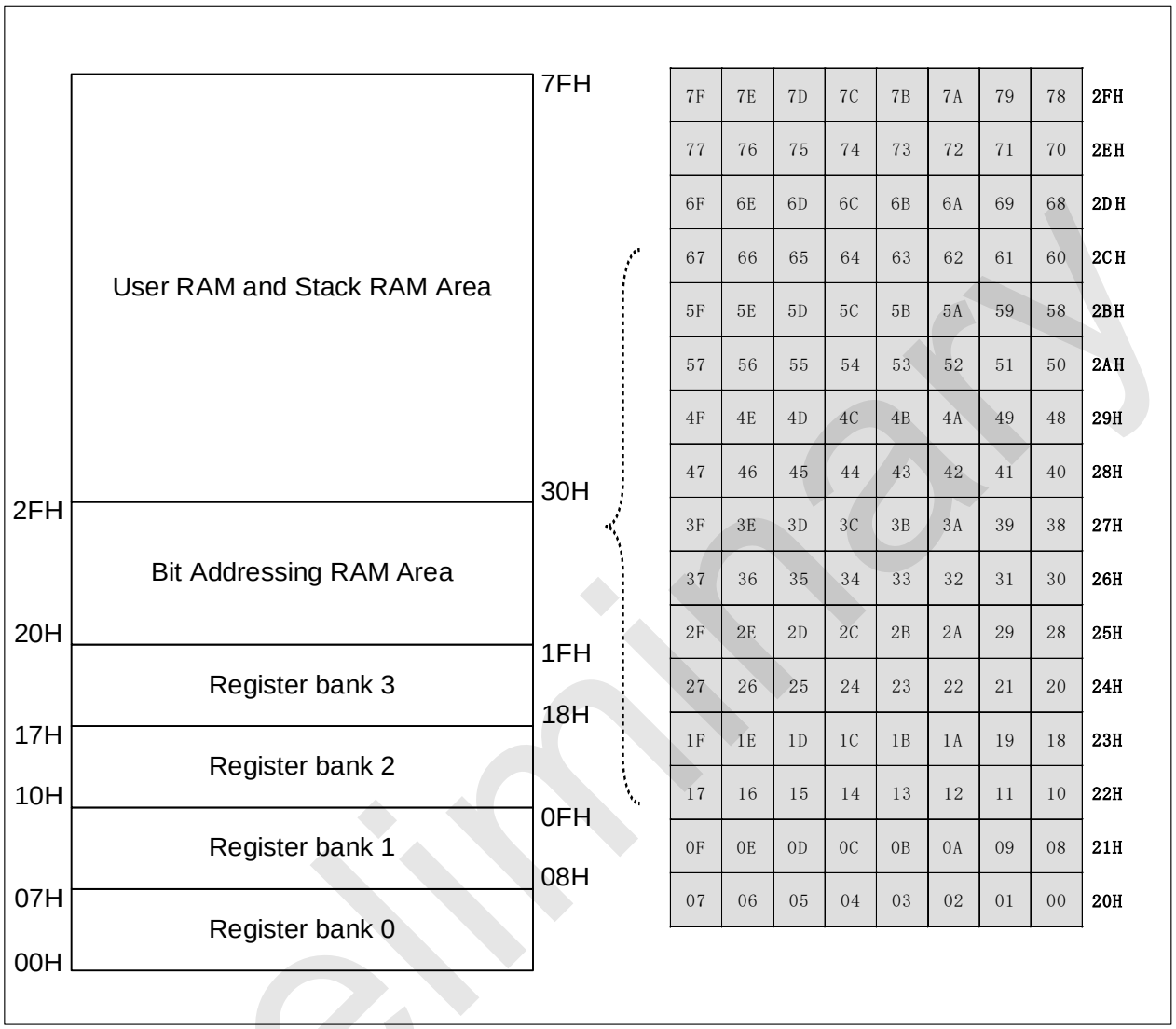
bit address is from 00H to 7FH (bitwise addressing is different from normal SRAM byte-oriented addressing), which can be distinguished by instructions in program;

③ User RAM and stack area, the 8-bit stack pointer will point to stack area after the SC92F730X reset; in general, users can set initial value in initializer, which is recommended to configure in the unit interval from E0H to FFH.



256 bytes RAM Structure Diagram

Low 128 bytes RAM structure is shown below:



SRAM Structure Diagram

5.3.2 External 256 bytes SRAM

The external 256 bytes RAM (SRAM) can be accessed by instruction ‘’MOVX @DPTR’’ or instruction MOVX A ,@Ri or MOVX @Ri,A.

6 Special Function Register (SFR)

6.1 SFR Mapping

The SC92F730X 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 SC92F730X 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	-	-	-	-	-	-	OPINX	OPREG
F0h	B	IAPKEY	IAPADL	IAPADH	IAPADE	IAPDAT	IAPCTL	-
E8h	-	-	-	-	-	-	-	-
E0h	ACC	-	-	-	-	-	-	-
D8h	-	-	-	-	-	PWMDTY3	PWMDTY4	PWMDTY5
D0h	PSW	-	PWMCON	PWMPRD	-	PWMDTY0	PWMDTY1	PWMDTY2
C8h	T2CON	-	RCAP2L	RCAP2H	TL2	TH2	BTMCON	WDTCON
C0h	-	-	-	-	-	-	INT2F	INT2R
B8h	IP	IP1	INT0F	INT0R	-	-	-	-
B0h	-	-	-	-	-	-	-	-
A8h	IE	IE1	-	ADCCFG0	ADCCFG1	ADCCON	ADCVL	ADCVH
A0h	P2	P2CON	P2PH	-	-	-	-	-
98h	SCON	SBUF	P0CON	P0PH	P0VO	-	-	-
90h	P1	P1CON	P1PH	-	-	-	-	IOHCON

88h	TCON	TMOD	TL0	TL1	TH0	TH1	TMCON	OTCON
80h	P0	SP	DPL	DPH	-	-	-	PCON
	Bit Addressable	Not Bit Addressable						

Note：Hollow space of SFR refers to the fact that there is no such register RAM, it is not recommended for user to use.

6.2 SFR Instructions

For a description of each SFR, see the following table:

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
P0	80H	P0 Data Register	-	-	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	-	-	-	-	0000xxxxb
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	-	-	-	-	VOIRS[1: 0]		-	-	xxxx00xxb
P1	90H	P1 Data Register	P17	P16	-	-	P13	P12	P11	P10	00xx0000b
P1CON	91H	P1 I/O Control Register	P1C7	P1C6	-	-	P1C3	P1C2	P1C1	P1C0	00xx0000b
P1PH	92H	P1 Pull-up Resistor Control Register	P1H7	P1H6	-	-	P1H3	P1H2	P1H1	P1H0	00xx0000b
IOHCON	97H	IOH Setup Register	P2H[1: 0]		P2L[1: 0]		P0H[1: 0]		P0L[1: 0]		00000000b
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	-	-	P0C5	P0C4	P0C3	P0C2	P0C1	P0C0	xx000000b
P0PH	9BH	P0 Pull-up Resistor Control Register	-	-	P0H5	P0H4	P0H3	P0H2	P0H1	P0H0	xx000000b

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
P0VO	9CH	P0 Port LCD Voltage Output Register	-	-	-	P04VO	P03VO	P02VO	P01VO	P00VO	xxx00000b
P2	A0H	P2 Data Register	P27	P26	P25	P24	-	-	P21	P20	0000xx00b
P2CON	A1H	P2 I/O Control Register	P2C7	P2C6	P2C5	P2C4	-	-	P2C1	P2C0	0000xx00b
P2PH	A2H	P2 Pull-up Resistor Control Register	P2H7	P2H6	P2H5	P2H4	-	-	P2H1	P2H0	0000xx00b
IE	A8H	Interrupt Enable Register	EA	EADC	ET2	EUART	ET1	-	ET0	EINT0	00000x00b
IE1	A9H	Interrupt Enable Register 1	-	-	-	-	EINT2	EBTM	EPWM	-	xxxx000xb
ADCCFG0	ABH	ADC Configuration Register 0	EAIN7	EAIN6	EAIN5	EAIN4	-	-	EAIN1	EAIN0	0000xx00b
ADCCFG1	ACH	ADC Configuration Register 1	-	-	-	-	-	-	EAIN9	EAIN8	xxxxxx00b
ADCCON	ADH	ADC Control Register	ADCEN	ADCS	LOWSP	EOC/ ADCIF	ADCIS[3:0]				00000000b
ADCVL	AEH	ADC Result Register	ADCV[3: 0]				-	-	-	-	0000xxxxb
ADCVH	AFH	ADC Result Register	ADCV[11: 4]								00000000b
IP	B8H	Interrupt Priority Control Register	-	IPADC	IPT2	IPUART	IPT1	-	IPT0	IPINT0	x0000x00b
IP1	B9H	Interrupt Priority Control Register 1	-	-	-	-	IPINT2	IPBTM	IPPWM	-	xxxx000xb
INT0F	BAH	INT0 Falling Edge Interrupt Control Register	-	-	-	-	INT0F3	INT0F2	INT0F1	INT0F0	xxxx0000b
INT0R	BBH	INT0 Rising Edge Interrupt Control Register	-	-	-	-	INT0R3	INT0R2	INT0R1	INT0R0	xxxx0000b
INT2F	C6H	INT2 Falling Edge Interrupt Control Register	-	-	INT2F5	INT2F4	-	-	INT2F1	INT2F0	xx00xx00b
INT2R	C7H	INT2 Rising Edge Interrupt Control Register	-	-	INT2R5	INT2R4	-	-	INT2R1	INT2R0	xx00xx00b
T2CON	C8H	Timer2 Control Register	TF2	-	RCLK	TCLK	-	TR2	-	-	0x00x0xxb
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
PWMCFG0	D1H	PWM Setup Register 0	-	-	INV2	INV1	INV0	ENPWM5	ENPWM4	ENPWM3	xx000000b
PWMCON	D2H	PWM Control Register	ENPWM	PWMIF	ENPWM2	ENPWM1	ENPWM0	PWMCKS[2:0]			00000000b
PWMPRD	D3H	PWM Period Setting Register	PWMPRD[7:0]								00000000b

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
PWMCFG1	D4H	PWM Setup Register 1	-	-	INV5	INV4	INV3	-	-	-	xx000xxxb
PWMDTY0	D5H	PWM0 duty cycle setting register	PDT0[7:0]								00000000b
PWMDTY1	D6H	PWM1 duty cycle setting register	PDT1[7:0]								00000000b
PWMDTY2	D7H	PWM2 duty cycle setting register	PDT2[7:0]								00000000b
PWMDTY3	DDH	PWM3 duty cycle setting register	PDT3[7:0]								00000000b
PWMDTY4	DEH	PWM4 duty cycle setting register	PDT4[7:0]								00000000b
PWMDTY5	DFH	PWM5 duty cycle setting register	PDT5[7:0]								00000000b
ACC	E0H	Accumulator	ACC[7: 0]								00000000b
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[12: 8]					xxx00000b
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
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

Bit Number	Bit Mnemonic	Description
7	**CY**	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
6	**AC**	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
5	**F0**	User flag bit
4 ~ 3	**RS1、RS0**	Register banks selection bits
2	**OV**	Overflow flag bit
1	**F1**	F1 flag bit User customized flag
0	**P**	Parity flag bit. This flag bit is the parity value of the number of 1 in accumulator ACC.

RS1	RS0	Current Selected Register banks 0 ~ 3
0	0	Group 0 (00H ~ 07H)
0	1	Group 1 (08H ~ 0FH)
1	0	Group 2 (10H ~ 17H)
1	1	Group 3 (18H ~ 1FH)

		1: Odd number of number of 1 in ACC 0: Even number of number of 1 in ACC (including 0)

Data Pointer DPTR (82H, 83H)

The Data pointer DPTR is a 16-bit dedicated register, which is composed of Low byte DPL (82H) and High byte DPH (83H). DPTR is the only register in the traditional 8051-based MCU 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 SC92F730X 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 voltage is built in the IC, which is used as ADC internal reference voltage. The user can search for specific configuration contents in 16 Analog-to-digital converter (ADC).

7.2 Power-on Reset

After the SC92F730X 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 SC92F730X will always be in reset mode. There will not be a valid clock until the voltage supplied to the SC92F730X 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 SC92F730X. 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 SC92F730X 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 SC92F730X has 4 kinds of reset modes:

① External RST reset

② Low-voltage reset (LVR)

③ Power-on reset (POR)

④ Watchdog (WDT) reset.

7.3.1 External Reset

External reset is to supply a certain width reset pulse signal to the SC92F730X from the RST pin to realize the SC92F730X reset.

RST/INT01/P1.1 have reset function, user can configure P1.1 pin as RST (reset pin) or not in Customer Option via PC program software before programming.

7.3.2 Low-voltage Reset (LVR)

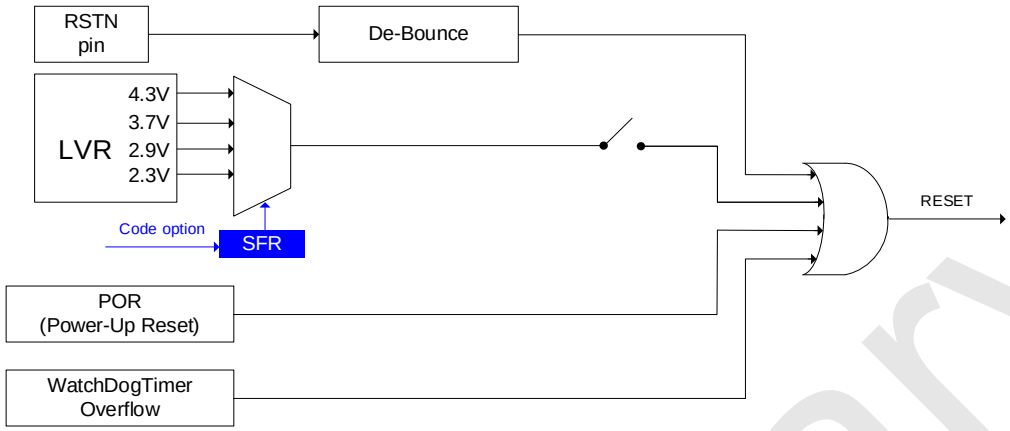
The SC92F730X provides a low-voltage reset circuit. There are 4-level LVR voltage options: 4.3V, 3.7V, 2.9V and 2.3V. The default is the Option value written by user. When the VDD voltage is lower than the threshold voltage of the low voltage reset and the duration is longer than TLVR(TLVR is the shaking time of LVR, about 30μs), a reset is generated.

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

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

Bit Number	Bit Mnemonic	Description
2	**DISLVR**	LVR enable bit 0: LVR enable 1: LVR disable
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 SC92F730X Resetting Part is shown below:



The SC92F730X Reset Diagram

7.3.3 Power-on Reset (POR)

The SC92F730X provides a power-on reset circuit. When power voltage VDD is up to POR reset voltage, the system will be reset automatically.

7.3.4 Watchdog Reset (WDT)

The SC92F730X 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	-	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
R/W	R/W	-	R/W		R/W	R/W	R/W	
POR	n	x	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 register of PORT is FFh. 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	P0PH	xx000000b
B	00000000b	P0VO	xxx00000b
PSW	00000000b	P1	00xx0000b

Mnemonic	Reset value	Mnemonic	Reset value
SP	00000111b	P1CON	00xx0000b
DPL	00000000b	P1PH	00xx0000b
DPH	00000000b	P2	0000xx00b
PCON	0xxxxx00b	P2CON	0000xx00b
ADCCFG0	0000xx00b	P2PH	0000xx00b
ADCCFG1	xxxxxx00b	PWMCFG0	xx000000b
ADCCON	00000000b	PWMCFG1	xx000xxxb
ADCVH	00000000b	PWMCON	00000000b
ADCVL	0000xxxxb	PWMDTY0	00000000b
BTMCON	00xx0000b	PWMDTY1	00000000b
IAPADE	00000000b	PWMDTY2	00000000b
IAPADH	xxx00000b	PWMDTY3	00000000b
IAPADL	00000000b	PWMDTY4	00000000b
IAPCTL	xxxx0000b	PWMDTY5	00000000b
IAPDAT	00000000b	PWMPRD	00000000b
IAPKEY	00000000b	RCAP2H	00000000b
IE	00000x00b	RCAP2L	00000000b
IE1	xxxx000xb	SBUF	00000000b
INT0R	xxxx0000b	SCON	00000000b
INT2R	xx00xx00b	TCON	0000xxxxb
INT0F	xxxx0000b	TMCON	xxxxx000b
INT2F	xx00xx00b	TMOD	x000x000b
IP	x0000x00b	TH0	00000000b
IP1	xxxx000xb	TL0	00000000b
OTCON	xxxx00xxb	TH1	00000000b
OPINX	00000000b	TL1	00000000b
OPREG	nnnnnnnnb	T2CON	0x00x0xxb
IOHCON	00000000b	TH2	00000000b

Mnemonic	Reset value	Mnemonic	Reset value
P0	xx000000b	TL2	00000000b
P0CON	xx000000b	WDTCON	xxx0x000b

7.4 High-speed RC Oscillator

The SC92F730X has a built-in adjustable high-precision HRC. HRC is precisely calibrated to 24 MHz@5V/25℃ when delivery. The user can set system clock as 24/12/6/2MHz 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 (2.4V ~ 5.5V) and temperature drifting (-40℃ ~ 85℃), the deviation is within ±1%.

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

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	ENWDT	-	SCLKS[1: 0]		DISRST	DISLVR	LVRS[1: 0]	
R/W	R/W	-	R/W		R/W	R/W	R/W	
POR	n	x	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 SC92F730X has a special function: the user can modify SFR value to adjust frequency of HRC within certain scope.

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
The value of OP_HRCR[7:0] when power-on can make sure HRC precisely works at 24/12/6/2MHz (according to Code Option), there may be difference in OP_HRCR[s] of each IC .User can change HRC by modifying the value of this register.
When initial value is OP_HRCR[s], IC system clock works at 24/12/6/2MHz. For each change of 1 for OP_HRCR [7: 0], the change of HRC is about 0.23%@12MHz.
The relationship between OP_HRCR [7: 0] and output frequency HRC is shown as follows:
<table>
<tr><td>OP_HRCR [7:0]</td><td>HRC actual output frequency
(taking 12M as an example)</td></tr>
<tr><td>OP_HRCR [s]-n</td><td>12000* (1-0.23%*n)kHz</td></tr>
<tr><td>...</td><td>....</td></tr>
<tr><td>OP_HRCR [s]-2</td><td>12000* (1-0.23%*2) = 11944.8kHz</td></tr>
<tr><td>OP_HRCR [s]-1</td><td>12000* (1-0.23%*1) = 11972.4kHz</td></tr>
<tr><td>OP_HRCR [s]</td><td>12000kHz</td></tr>
<tr><td>OP_HRCR [s]+1</td><td>12000* (1+0.23%*1) = 12027.6kHz</td></tr>
<tr><td>OP_HRCR [s]+2</td><td>12000* (1+0.23%*2) = 12055.2kHz</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>OP_HRCR [s]+n</td><td>12000* (1+0.23%*n)kHz</td></tr>
</table>
Note:
1. The value of OP_HRCR[7:0] after each power-on of the IC is the value of HRC frequency closest to 24MHz; the user can modify the value of HRC after each power-on by means of EEPROM to make HRC work at the frequency the user needs;
2. To guarantee IC operating reliably, the operating frequency of IC shall not exceed 24MHz;
3. The user shall confirm the change of HRC frequency will not influence other functions.
</td></tr>
</table>

7.5 Low-speed RC Oscillator

The SC92F730X is equipped with a built-in 32kHz RC oscillation circuit, which can be set as clock source of low-frequency clock timer Base Timer and WDT. This 32kHz low-frequency oscillator can be activated by enable Base Timer or WDT.

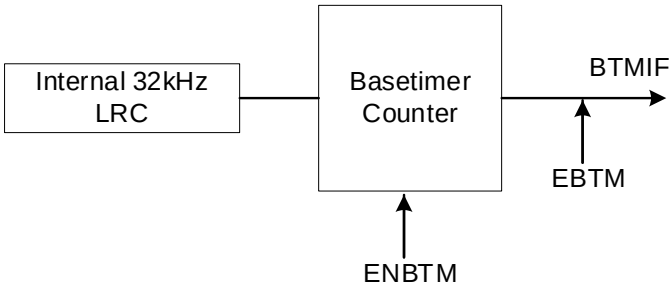
Base Timer, a low-frequency clock timer which can wake up CPU from STOP mode and generate interrupt.

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

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	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 and cleared manually by software.
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 Others: reserved
5 ~ 4	-	Reserved

Base timer structure is shown as follows:



Base timer structure

7.6 Power Saving Modes

The SC92F730X 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. In STOP mod , the user can wake up SC92F730X by external interrupt INT0, INT2, and low frequency clock interrupt, user also can wake up STOP by external reset.

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 Mnemonic	SMOD	-	-	-	-	-	STOP	IDL
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

Bit Number	Bit Mnemonic	Description
		and clock continue to operate and all CPU states are saved before entering IDLE mode

Note：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 PCON bit1 STOP bit to 1, 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 PCON bits1 STOP bit to 1, 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 SC92F730X 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 SC92F730X 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

Relative addressing is to add current value in program counter (PC) and 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.

9 Interrupt

The SC92F730X provides 9 interrupt sources: Timer0, Timer1, Timer2, INT0, INT2, ADC, PWM, UART, and Base Timer. These 9 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 two 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 SC92F730X 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	-	EINT0	IPINT0	0003H	1 (high)	0	-	Yes
Timer0	Timer0 overflow	TF0	ET0	IPT0	000BH	2	1	H/W Auto	No
Timer1	Timer1 overflow	TF1	ET1	IPT1	001BH	3	3	H/W Auto	No
UART	Receiving or transmitting completed	RI/TI	EUART	IPUART	0023H	4	4	Must be cleared by user	No
Timer2	Timer2 overflow	TF2	ET2	IPT2	002BH	5	5	Must be cleared by user	No
ADC	ADC conversion completed	ADCIF	EADC	IPADC	0033H	6	6	Must be cleared by user	No
PWM	PWM overflow	PWMIF	EPWM	IPPWM	0043H	7	8	Must be cleared by user	No
BTM	Base timer overflow	BTMIF	EBTM	IPBTM	004BH	8	9	H/W Auto	Yes
INT2	External interrupt 2 conditions compliant	-	EINT2	IPINT2	0053H	9	10	-	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 a frame of data is completed in UART reception or transmission, the RI or TI bit will be automatically set to “1” by hardware, and a UART interrupt will be generated. After a UART interrupt occurs, the hardware will not automatically clear the RI/TI bit, and this bit must be cleared by software

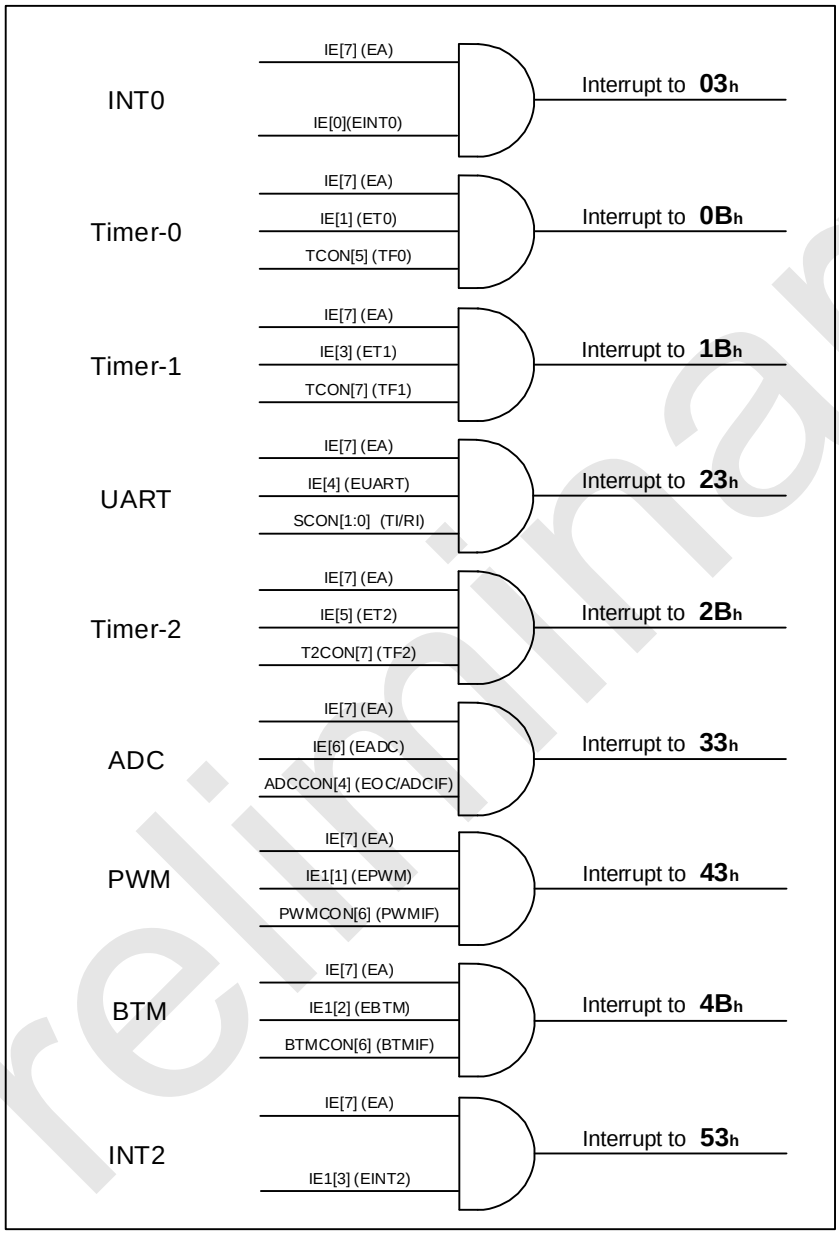
ADC Interrupt: After ADC conversion is completed, ADC interrupt generates, whose interrupt flag is the ADC conversion completion flag EOC/ADCIF (ADCCON.4). When user starts ADCS conversion, EOC will be reset automatically by hardware. Once conversion completes, EOC would be set to “1” automatically by hardware. User should clear the ADC interrupt flag by software when the interrupt service routine is executed after ADC interrupt generates.

PWM Interrupt: When PWM counter overflows (beyond PWMPD), The PWMIF bit will be set to 1 automatically by hardware, PWM interrupt generates. After a PWM interrupt occurs, the hardware will not automatically clear the PWMIF bit, and this bit must be cleared by software

External Interrupt INT0 and INT2: When any external interrupt pin meets the interrupt conditions, external interrupt generates. There are 4 external interrupt sources for INT0 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 and INT2 can also wake up STOP mode of microcontroller unit.

9.2 Interrupt Structure Diagram

The SC92F730X interrupt structure is shown in the figure below:



The SC92F730X Interrupt Structure and Vector

9.3 Interrupt Priority

The SC92F730X 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 SC92F730X 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	-	ET0	EINT0
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	R/W
POR	0	0	0	0	0	x	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
4	**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
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
2	**-**	Reserved

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

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

Bit Number	Bit Mnemonic	Description
6	**IPADC**	ADC interrupt priority selection bit 0: ADC interrupt priority is low

Bit Number	Bit Mnemonic	Description
		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
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
7,2	**-**	Reserved

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

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

Bit Number	Bit Mnemonic	Description
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

Bit Number	Bit Mnemonic	Description
1	**EPWM**	PWM interrupt enabling control bit 0: Disable PWM interrupt 1: Enable interrupt upon PWM counting overflows (counting to PWMPRD)
7 ~ 4,0	**-**	Reserved

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

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

Bit Number	Bit Mnemonic	Description
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
7 ~ 4,0	**-**	Reserved

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

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	-	-	INT0F3	INT0F2	INT0F1	INT0F0
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	**INT0Fn** **(n=0 ~ 3)**	INT0 falling edge interrupt control bit 0: INT0n falling edge interrupt disable 1: INT0n falling edge interrupt enable
7 ~ 4	**-**	Reserved

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

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	-	-	INT0R3	INT0R2	INT0R1	INT0R0
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	**INT0Rn** **(n=0 ~ 3)**	INT0 rising edge interrupt control bit 0: INT0n rising edge interrupt disable 1: INT0n rising edge interrupt enable
7 ~ 4	**-**	Reserved

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

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	-	-	INT2F5	INT2F4	-	-	INT2F1	INT2F0
R/W	-	-	R/W	R/W	-	-	R/W	R/W
POR	x	x	0	0	x	x	0	0

Bit Number	Bit Mnemonic	Description
5 ~ 4,2 ~ 1	**INT2Fn** **(n=1 ~ 2,4 ~ 5)**	INT2 falling edge interrupt control bit 0: INT2n falling edge interrupt disable 1: INT2n falling edge interrupt enable
7 ~ 6,3 ~ 2	**-**	Reserved

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

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	INT2R5	INT2R4	-	-	INT2R1	INT2R0
R/W	-	-	R/W	R/W	-	-	R/W	R/W
POR	x	x	0	0	x	x	0	0

Bit Number	Bit Mnemonic	Description
5 ~ 4,1 ~ 0	**INT2Rn** **(n=0 ~ 1,4 ~ 5)**	INT2 rising edge interrupt control bit 0: INT2n rising edge interrupt disable 1: INT2n rising edge interrupt enable
7 ~ 6,3 ~ 2	**-**	Reserved

10 Timer/Counter T0 and T1

The SC92F730X has two 16-bit Timer/Counters, Timer0 (T0) and Time1 (T1), with two operating modes: counter mode and timer mode. There is a control bit C/Tx which can select T0 and T1 as a timer or a counter. They are both essentially an addition counter, just the source of the count is different. 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 P1.2/T0 and P1.3/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) 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	-	-	-	-	0000xxxxb
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	-	-	-	-
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	**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.
3 ~ 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/P1.3).

<table>
<tr><th>Bit Number</th><th>Bit Mnemonic</th><th>Description</th></tr>
<tr><td>5 ~ 4</td><td>M11, M01</td><td>Timer1 operating mode
<table>
<tr><th>Mode</th><th>M11</th><th>M01</th><th>Operation</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>13-bit TIMER/Counter, TL1 high 3 bits invalid</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>16-bit Timer/Counter,both TL1 and TH1 are valid.</td></tr>
<tr><td>2</td><td>1</td><td>0</td><td>8-bit Auto-Reload Mode.
TH1 holds a value which is reloaded into 8-bit Timer/CounterTL1 each time it overflows.</td></tr>
<tr><td>3</td><td>1</td><td>1</td><td>Timer/Counter 1 stop counting</td></tr>
</table></td></tr>
<tr><td>2</td><td>C/T0</td><td>Timer or Counter selector 0
0: Cleared for Timer operation (input from internal system clock fsys).
1: Set for Counter operation (input from external pin T0/P1.2).</td></tr>
<tr><td>1 ~ 0</td><td>M10, M00</td><td>Timer0 operating mode
<table>
<tr><th>Mode</th><th>M10</th><th>M00</th><th>Operation</th></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,both TL0 and TH0 are valid.</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.
Timer 0 now acts as a dual 8-bit timer/counter
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	EUART	ET1	-	ET0	EINT0
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	R/W
POR	0	0	0	0	0	x	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	-	IPT0	IPINT0
R/W	-	R/W	R/W	R/W	R/W	-	R/W	R/W
POR	x	0	0	0	0	x	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.

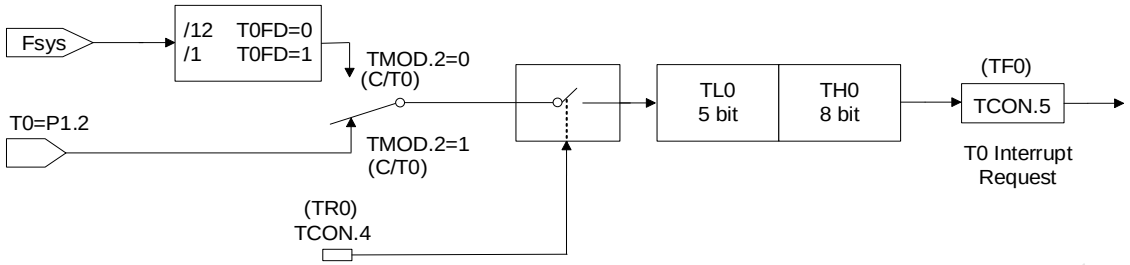
Operating mode 0: 13-bit Timer/Counter

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 (P1.2) 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.

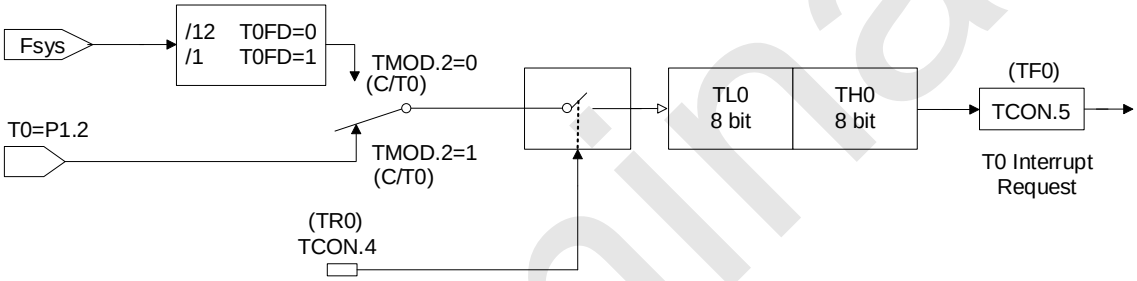
When configured as a timer, the SFR T0FD is used to select fractional frequency ratio of clock source.



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.



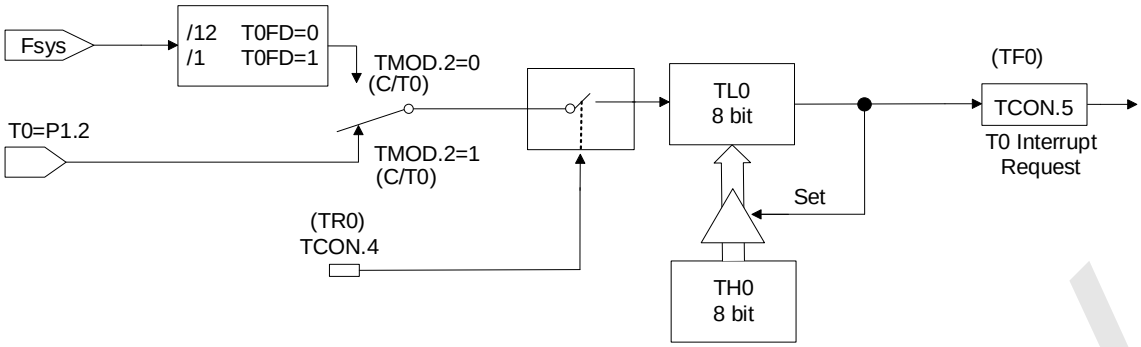
Operating mode 1: 16-bit Timer/Counter

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.

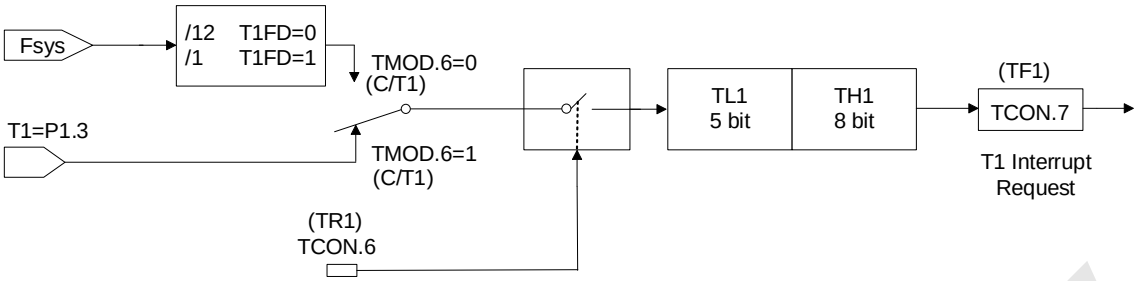
Operating mode 0: 13-bit Timer/Counter

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 to 1. 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 (P1.3) 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.

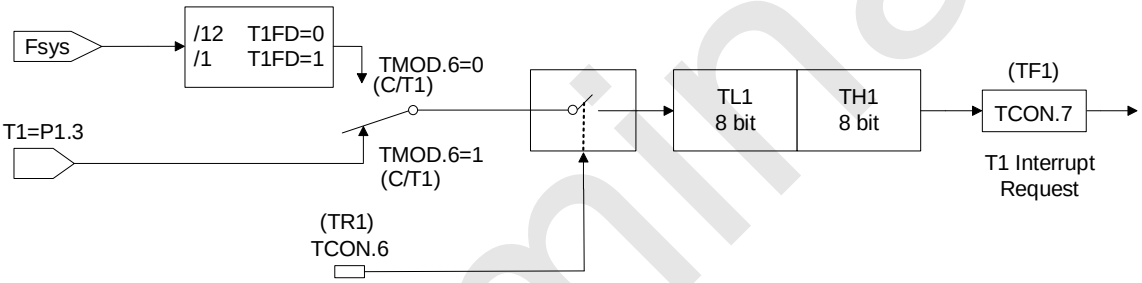
When configured as timer, the SFR T1FD is used to select fractional frequency ratio of clock source.



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.



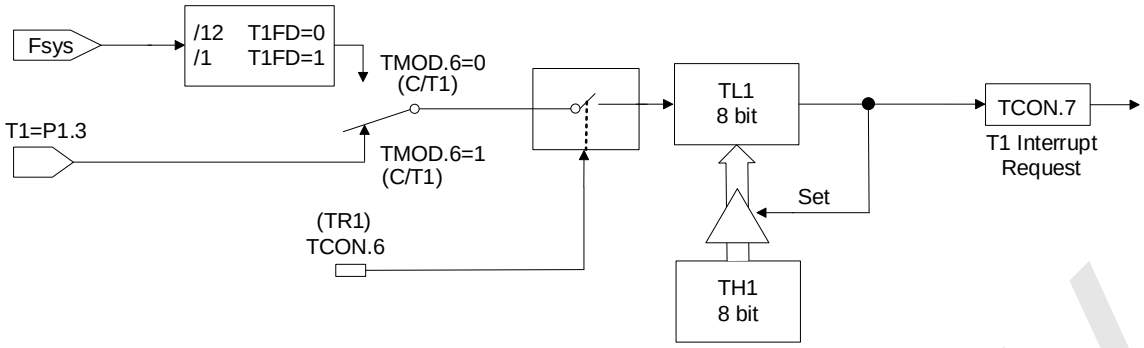
Operating mode 0: 16-bit Timer/Counter

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 SC92F730X microcontroller unit works as Timer and it is an adding counter in nature. The clock source of T2 comes from system clock or frequency division clock. TR2 is the counting switch of Timer T2. Only when TR2 = 1, will T2 be enabled for counting.

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 2 operating modes:

① Mode 1: 16-bit automatic reload timer mode

② Mode 2: Baud rate generator mode

11.1 T2-related Registers

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
T2CON	C8H	Timer2 Control Register	TF2	-	RCLK	TCLK	-	TR2	-	-	0x00x0xxb
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	-	RCLK	TCLK	-	TR2	-	-
R/W	R/W	-	R/W	R/W	-	R/W	-	-
POR	0	x	0	0	x	0	x	x

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)

Bit Number	Bit Mnemonic	Description
5	**RCLK**	UART receiving clock control bit 0: Timer1 generates receiving baud rate 1: Timer2 generates receiving baud rate
4	**TCLK**	UART transmitting clock control bit 0: Timer1 generates transmitting baud rate 1: Timer2 generates transmitting baud rate
2	**TR2**	Timer2 start/stop control bit 0: Stop Timer2 1: Start Timer2
6,3,1 ~ 0	**-**	Fix at 0

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	-	ET0	EINT0
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	R/W
POR	0	0	0	0	0	x	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	-	IPT0	IPINT0
R/W	-	R/W	R/W	R/W	R/W	-	R/W	R/W
POR	x	0	0	0	0	x	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”

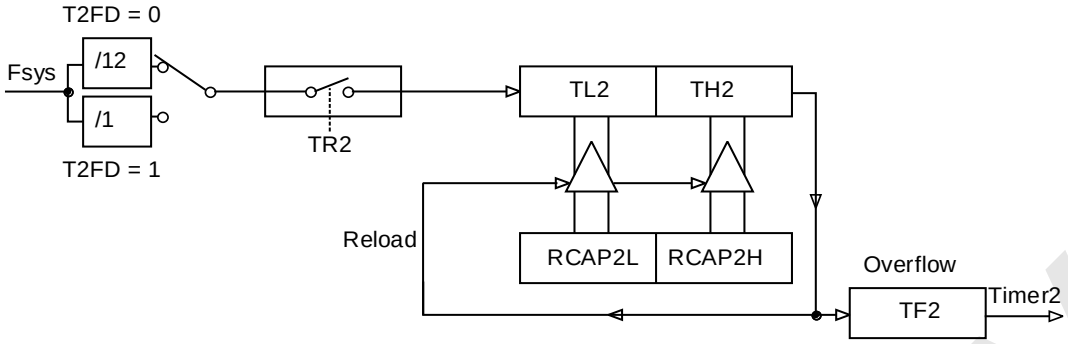
11.2 T2 Operating Modes

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

TR2	**RCLK**	**TCLK**	**Mode**	
1	0	0	1	16-bit automatic reload timer
1	1	X	2	Baud Rate Generator
	X	1		
0	X	X	X	Timer2 stops

Operating Mode 1: 16-bit Automatic Reload Timer

In the 16-bit automatic reload mode, 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.



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

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.

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.

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

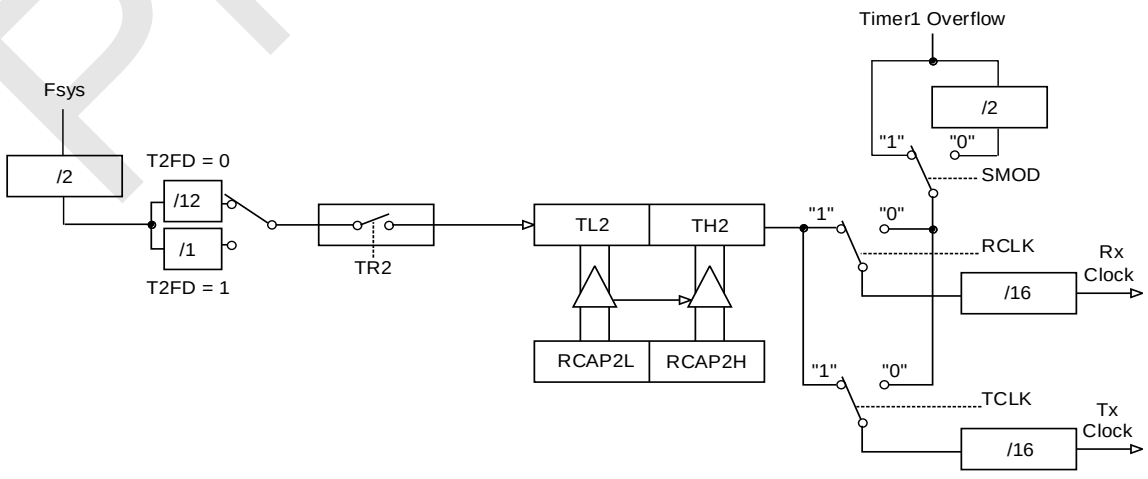
$$\text{BaudRate} = \frac{1}{16} \times \frac{\text{fn2}}{(65536 - [\text{RCAP2H}, \text{RCAP2L}]) \times 2}$$

Including, fn2 is the clock frequency of Timer2

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

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

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



Mode 2: Baud Rate Generator

Note:

1. **TF2 can be set by software, and only software and hardware reset can clear IT**
2. **When EA = 1 and ET2 = 1, setting up TF2 to 1 can arouse interrupt of Timer2;**
3. **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 PWM

The SC92F730X provides an independent counter, which is able to support 6-channel PWM output: PWM0 ~ 5.

The SC92F730X PWM has the following functions:

① 8-bit precision;

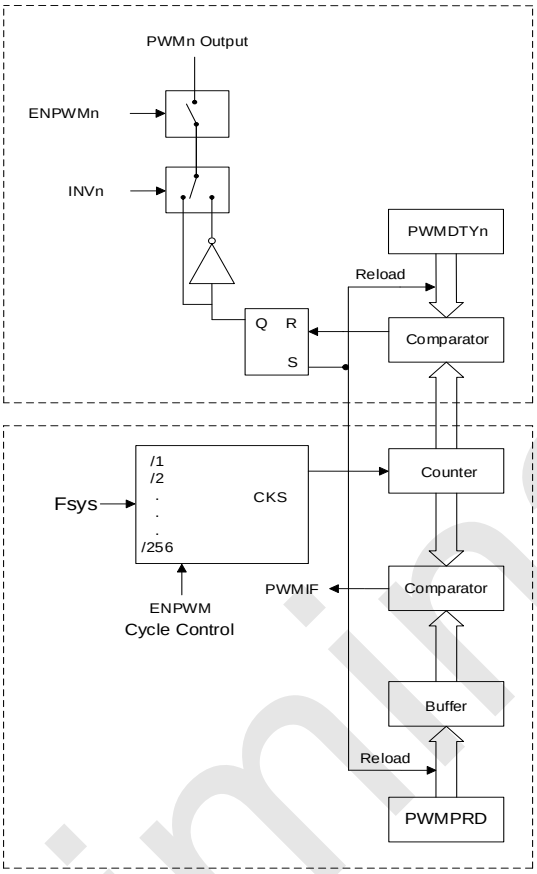
② PWM0 ~ 5 shared the same clock cycle, but the duty cycle of each PWM channel can be configured separately;

③ Output can be configured in forward or reverse direction;

④ Provide one PWM overflow interrupt.

The PWM of SC92F730X supports adjustable cycle and duty cycle. The PWM-related settings for PWM0~5 are controlled by register PWMCON. PWMCFG0 and PWMCFG1 determine the polarity of PWM output waveforms and select the output IO. PWMPRD sets the common cycle for PWM, while PWMDTY05 respectively control the duty cycle for PWM0~5

12.1 PWM block Diagram



The SC92F730X PWM block Diagram

12.2 PWM-related Registers

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
PWMCFG0	D1H	PWM Configuration Register 0	-	-	INV2	INV1	INV0	ENPWM5	ENPWM4	ENPWM3	xx000000b
PWMCON	D2H	PWM Control Register	ENPWM	PWMIF	ENPWM2	ENPWM1	ENPWM0	PWMCKS[2:0]			00000000b
PWMPRD	D3H	PWM Period Setting Register	PWMPRD[7 : 0]								00000000b
PWMCFG1	D4H	PWM Configuration Register 1	-	-	INV5	INV4	INV3	-	-	-	xx000xxxb
PWMDTY0	D5H	PWM0 Duty Cycle Configuration Register	PDT0[7 : 0]								00000000b
PWMDTY1	D6H	PWM1 Duty Cycle Configuration Register	PDT1[7 : 0]								00000000b
PWMDTY2	D7H	PWM2 Duty Cycle Configuration Register	PDT2[7 : 0]								00000000b

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
PWMDTY3	DDH	PWM3 Duty Cycle Configuration Register	PDT3[7 : 0]								00000000b
PWMDTY4	DEH	PWM4 Duty Cycle Configuration Register	PDT4[7 : 0]								00000000b
PWMDTY5	DFH	PWM5 Duty Cycle Configuration Register	PDT5[7 : 0]								00000000b
IE1	A9H	Interrupt Enable register 1	-	-	-	-	EINT2	EBTM	EPWM	-	xxxx000xb
IP1	B9H	Interrupt Priority Control Register 1	-	-	-	-	IPINT2	IPBTM	IPPWM	-	xxxx000xb

12.3 PWM General Configuration Registers

PWMCON (D2H) PWM Control Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	ENPWM	PWMIF	ENPWM2	ENPWM1	ENPWM0	PWMCKS[2:0]		
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: Enable Clock to enter PWM unit and PWM starts to work 0: PWM unit stops operating and PWM counter resets to zero. PWMn still connects to output pin. If using other functions multiplexed with PWMn output pin, set ENPWMn to 0
6	**PWMIF**	PWM interrupt flag When PWM counter overflows (that is to say, the figure exceeds PWMPRD), this bit will be automatically set to 1 by hardware. If at this time IE1[1] (EPWM) is set to 1 as well, PWM interrupt generates. **Note：Six PWMs share the same period and the same PWM interrupt vector.**
5	**ENPWM2**	PWM2 functional switch control bit 0: PWM2 do not output to IO 1: PWM2 output to IO
4	**ENPWM1**	PWM1 functional switch control bit

Bit Number	Bit Mnemonic	Description
		0: PWM1 do not output to IO 1: PWM1 output to IO
3	**ENPWM0**	PWM0 functional switch control bit 0: PWM0 do not output to IO 1: PWM0 output to IO
2 ~ 0	**PWMCKS[2:0]**	PWM (PWM clock source selector) 000: f_{SYS} 001: $f_{SYS}/2$ 010: $f_{SYS}/4$ 011: $f_{SYS}/8$ 100: $f_{SYS}/32$ 101: $f_{SYS}/64$ 110: $f_{SYS}/128$ 111: $f_{SYS}/256$

PWMPRD[7:0] serves as the shared cycle setting controller for the 6 PWM channels. Whenever the PWM counter reaches the pre-set value of PWMPRD[7:0], the next PWM CLK arrival causes the counter to reset to 00h. In other words, the cycle for PWM0~5 is calculated as (PWMPRD[7:0] + 1) multiplied by the PWM clock.

The counting time of the PWM counter is controlled by PWMCKS[2:0], which allows the selection of different numbers of system clocks to count one unit (pre-scaler selector). That means choosing the divisor for the PWM counter clock source, where the source is the system clock fSYS divided by the specified divisor. Additionally, PWM05 can be configured for inverted PWM output using INV05 in PWMCFG0 and PWMCFG1

PWMPRD (D3H) PWM Period Configuration Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	PWMPRD[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	**PWMPRD[7 : 0]**	The shared period setting for the 6 PWM channels; These value represents the (period - 1) of the PWM waveforms for PWM0~5. In other words, the actual period of the PWM output is calculated as :(PWMPRD[7:0] + 1)*PWM clock

PWMCFG0 (D1H) PWM Configuration Register 0 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	INV2	INV1	INV0	ENPWM5	ENPWM4	ENPWM3
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	**INV2**	PWM2 output reverse control bit 0: PWM2 output not invert 1: PWM2 output reverse
4	**INV1**	PWM1 output reverse control bit 0: PWM1 output not invert 1: PWM1 output reverse
3	**INV0**	PWM0 output reverse control bit 0: PWM0 output not invert 1: PWM0 output reverse
2	**ENPWM5**	PWM5 functional switch control bit 0: PWM5 do not output to IO 1: PWM5 output to IO
1	**ENPWM4**	PWM4 functional switch control bit 0: PWM4 do not output to IO 1: PWM4 output to IO
0	**ENPWM3**	PWM3 functional switch control bit 0: PWM3 do not output to IO 1: PWM3 output to IO
7 ~ 6	**-**	Reserved

PWMCFG1 (D4H) PWM Configuration Register 1 (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	INV5	INV4	INV3	-	-	-
R/W	-	-	R/W	R/W	R/W	-	-	-

Bit Number	7	6	5	4	3	2	1	0
POR	x	x	0	0	0	x	x	x

Bit Number	Bit Mnemonic	Description
5	**INV5**	PWM5 output reverse control bit 0: PWM5 output not invert 1: PWM5 output reverse
4	**INV4**	PWM4 output reverse control bit 0: PWM4 output not invert 1: PWM4 output reverse
3	**INV3**	PWM3 output reverse control bit 0: PWM3 output not invert 1: PWM3 output reverse
7 ~ 6,2 ~ 0	**-**	Reserved

PWMDTY0 (D5H) PWM0 Duty Cycle Configuration Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	PDT0[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	**PDT0[7 : 0]**	PWM0 duty cycle length configuration of high 8 bits: High level width of PWM0 is (PDT0[7: 0]) PWM clocks.

PWMDTY1 (D6H) PWM1 Duty Cycle Configuration Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	PDT1[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	**PDT1[7 : 0]**	PWM1 duty cycle length configuration of high 8 bits: High level width of PWM1 is (PDT1[7: 0]) PWM clocks.

PWMDTY2 (D7H) PWM2 Duty Cycle Configuration Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	PDT2[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	**PDT2[7 : 0]**	PWM2 duty cycle length configuration of high 8 bits: High level width of PWM2 is (PDT2[7: 0]) PWM clocks.

PWMDTY3 (DDH) PWM3 Duty Cycle Configuration Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	PDT3[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	**PDT3[7 : 0]**	PWM3 duty cycle length configuration of high 8 bits: High level width of PWM3 is (PDT3[7: 0]) PWM clocks.

PWMDTY4 (DEH) PWM4 Duty Cycle Configuration Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	PDT4[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	**PDT4[7 : 0]**	PWM4 duty cycle length configuration of high 8 bits: High level width of PWM4 is (PDT4[7: 0]) PWM clocks.

PWMDTY5 (DFH) PWM5 Duty Cycle Configuration Register (Read/Write)

Bit Number	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit Mnemonic	PDT5[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	**PDT5[7 : 0]**	Independent Mode: PWM5 duty cycle length configuration of high 8 bits; High level width of PWM5 is (PDT5[7: 0]) PWM clocks.

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

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

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	-	-	-	-	IPINT2	IPBTM	IPPWM	-
R/W	-	-	-	-	R/W	R/W	R/W	-
POR	x	x	x	x	0	0	0	x

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”

Note:

4. **The ENPWM bit controls whether the PWM module is operational**
5. **The ENPWMn bit selects whether PWMn is used as GPIO or as PWMn output**
6. **The EPWM bit controls whether PWM is allowed to generate interrupts**
7. **If ENPWM is set to 1, the PWM module is activated, but if ENPWMn is set to 0, PWM output is turned off and used as a GPIO. In this case, the PWM module can be used as an 8-bit timer, and if EPWM is set to 1, PWM interrupts will still occur**
8. **All 6 PWM channels share the same cycle, and an overflow generates a PWM interrupt with the same interrupt vector**

12.4 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 (PDTn). But note that changing the value of PDTn will change the duty cycle immediately.

② Period Change features

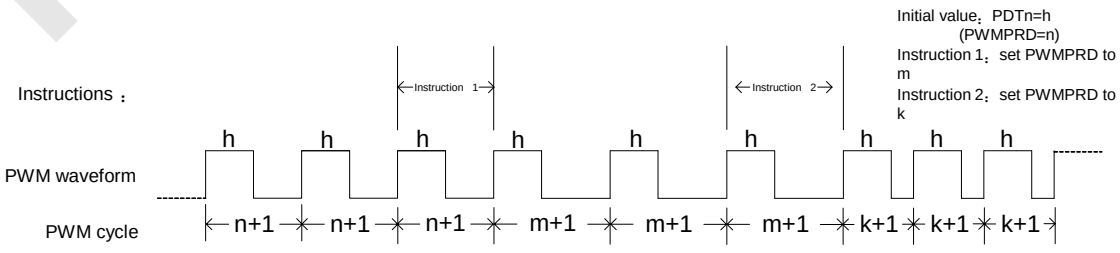


Diagram for Period Change Features

When PWMn outputs waveform, if it is required to change the period, the user can change the value of period configuration registers PWMPRD. Unlike changing the duty cycle, change the value of PWMPRD will not change the period immediately. It is required to wait until the end of this period and change in the next period. Refer to the figure above.

③ Relationship between Period and Duty cycle

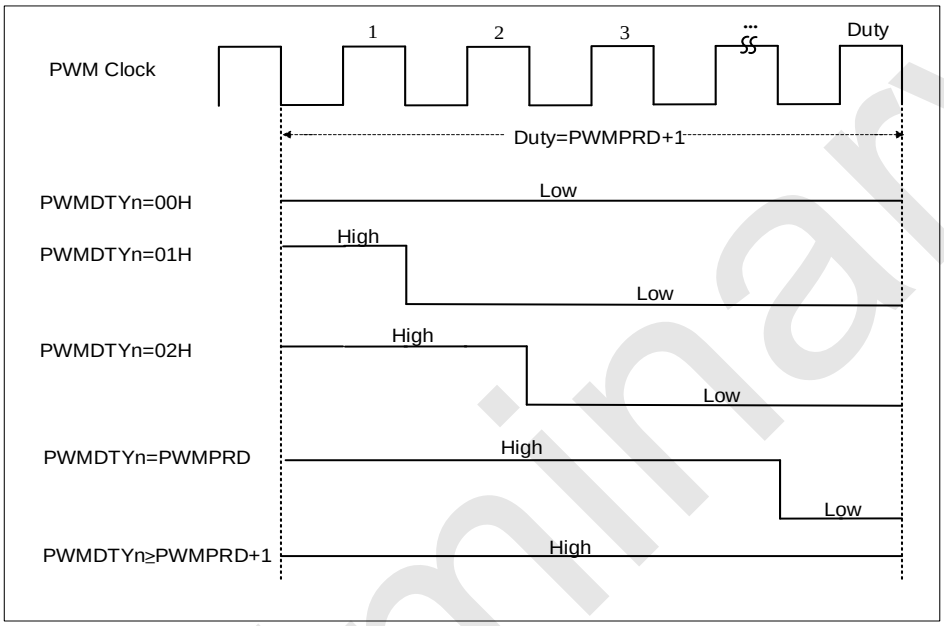


Diagram for Relationship between Period and Duty cycle

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

13 General-purpose I/O (GPIO)

The SC92F730X offers up to 18 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 18 IOs are shared with other functions, including P0.0 ~ P0.4 can be used as LCD COM driver by configuring output voltage as 1/2 V_{DD}. Under output state, what I/O port read from the value of port data register is the actual state value of the port.

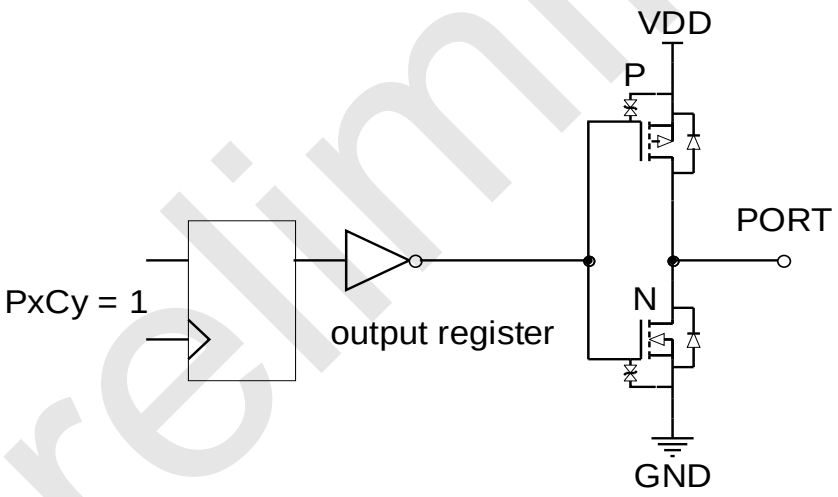
Note：Unused IO port or IO port with no package pin shall be configured as strong push-pull output mode.

13.1 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 22mA and low output for the current larger than 70mA

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

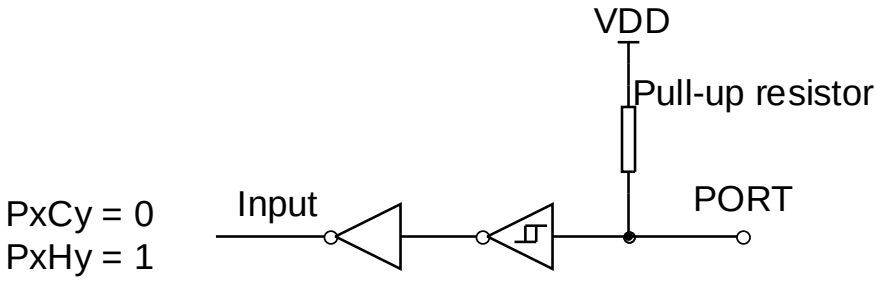


Strong Push-pull Output Mode

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.

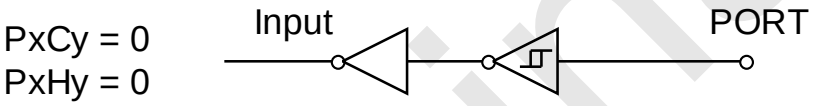
The port structure diagram for pull-up input mode is shown below:



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

13.2 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	-	-	P0C5	P0C4	P0C3	P0C2	P0C1	P0C0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	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	-	-	P0H5	P0H4	P0H3	P0H2	P0H1	P0H0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	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	-	-	P1C3	P1C2	P1C1	P1C0
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
POR	0	0	x	x	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	-	-	P1H3	P1H2	P1H1	P1H0
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
POR	0	0	x	x	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	-	-	P2C1	P2C0
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
POR	0	0	0	0	x	x	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	-	-	P2H1	P2H0
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
POR	0	0	0	0	x	x	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 0	**PxCy** **(x=0 ~ 2, y=0 ~ 7)**	Px port input/output control bits 0: Pxy in input mode(default at power-up) 1: Pxy in strong push-pull mode

Bit Number	Bit Mnemonic	Description
7 ~ 0	**PxHy** **(x=0 ~ 2, y=0 ~ 7)**	Px port pull-up resister enable bit 0: Pxy in high-impedance input mode(default at power-up), pull-up resister disable 1: Pxy pull-up resister enable

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

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	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
POR	x	x	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	-	-	P1.3	P1.2	P1.1	P1.0
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
POR	0	0	x	x	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	-	-	P2.1	P2.0
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
POR	0	0	0	0	x	x	0	0

IOHCON (97H) IOH Configuration Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	P2H[1: 0]		P2L[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

Bit Number	7	6	5	4	3	2	1	0
POR	0	0	0	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
7 ~ 6	**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);
5 ~ 4	**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);
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);

14 Software LCD Driver

The P0.0 ~ P0.4 of the SC92F730X can be used as the COM port of the software LCD. In addition to the normal IO functions, these IOs can also output 1/2VDD. The user can select the corresponding IO as the LCD driver COM according to the usage.

14.1 Software LCD Drives-related Registers

LCD Driver Related SFR Register Description:

P0VO (9CH) P0 port LCD voltage output register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	P04VO	P03VO	P02VO	P01VO	P00VO
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	0	0	0	0	0

P0yVO (y=0 ~ 4)	P0y	P0y selection output port
0	X	Ordinary IO port
1	1	The output voltage of P0y port is 1/2V_{DD}.

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

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

Bit Number	Bit Mnemonic	Description
3 ~ 2	**VOIRS[1: 0]**	Selection bits of voltage dividing resistance of LCD voltage output port (suitable driving according to LCD screen size) 00: Disable internal voltage divider resistor. (Energy saving) 01: Set the internal partial resistance to 12.5K 10: Set the internal partial resistance to 37.5K 11: Set the internal partial resistance to 87.5K

15 UART

15.1 UART-related Registers

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: Mode 2, 11-bit full-duplex asynchronous communication composing of 1 starting bit, 8 data bits 1 programmable 9th bit and 1 stopping bit; 11: 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 2 and 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 2 and 3, 9th bit of receiving data
2	**RB8**	Only valid for mode 2 and 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 0: When SM0~1 = 00, serial port operates under clock of 1/12 system clock; when SM0~1 = 10, serial port operates under clock of 1/64 system clock 1: When SM0~1 = 00, serial port operates under clock of 1/4 system clock; when SM0~1 = 10, serial port operates under clock of 1/32 system clock

15.2 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 is the 8-bit auto-reload registers of Timer1, SMOD is the baud rate multiplier of UART, and [RCAP2H、RCAP2L] are the 16-bit reload registers of Timer2.

1. When Timer1 is used as baud rate generator, it works in Mode 2:

$$\text{BaudRate} = \frac{2^{\text{SMOD}}}{[\text{TH1}, \text{TL1}]} \times \frac{\text{fn1}}{(256 - \text{TH1}) \times 2}$$

fn1 is the clock frequency of Timer1:

$$\text{fn1} = \frac{\text{fsys}}{12}; \quad \text{T1FD = 0}$$

$$\text{fn1} = \text{fsys}; \quad \text{T1FD = 1}$$

2. When Timer2 is used as baud rate generator:

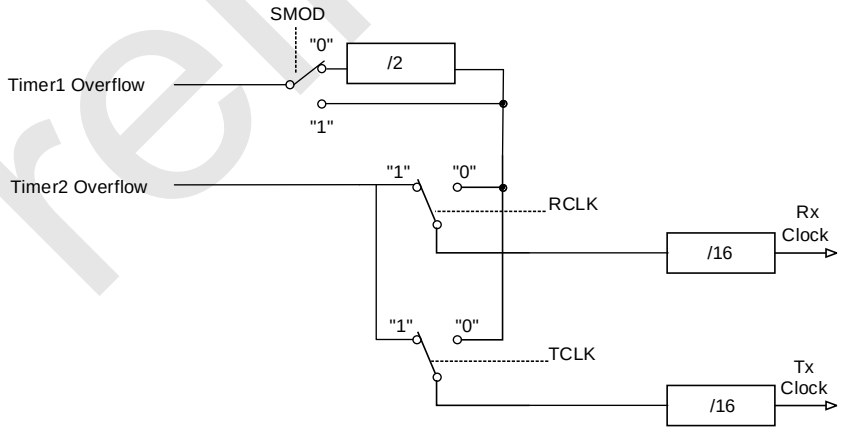
$$\text{BaudRate} = \frac{1}{16} \times \frac{\text{fn2}}{(65536 - [\text{RCAP2H}, \text{RCAP2L}] \times 2)}$$

fn2 is the clock frequency of Timer2:

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

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

The schematic diagrams for baud rate generators in mode 1 and mode 3 are as follows:



In Mode 2, the baud rate is fixed at either 1/32 or 1/64 of the system clock, determined by the state of the SMOD bit (PCON.7). When the SMOD bit is 0, the baud rate is set to 1/64 of the system clock. When the SMOD bit is 1, the baud rate is set to 1/32 of the system clock

16 Analog-to-Digital Converter (ADC)

The SC92F730X has a 12-bit high-precision successive approximation ADC with 9-channel, the external 8 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 VDD);

② Precise 2.4V reference output from internal Regulator

Note：Users are advised to use the interrupt method when converting ADC, check the DEMO for details.

16.1 ADC-related Registers

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

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	ADCEN	ADCS	LOWSP	EOC/ADCIF	ADCIS[3: 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”. **Note：After writing “1” to ADCS, do not write to the ADCCON register until the interrupt flag EOC/ADCIF is set.**
5	**LOWSP**	ADC Sampling Clocks Selector 0: Select clock frequency of ADC as 2MHz 1: Select clock frequency of ADC as 333kHz Explanations: The total time conversion time of ADC is T_{ADC}, consists of the sampling

		time and the conversion time. The conversion time is fixed at 950ns. The total time for the ADC from sampling to completing the conversion is calculated as follows: ● When LOWSP is 0 (LOWSP=0): T_{ADC1} = 6 * (1/2MHz) + 950ns ≈ 4us ● When LOWSP is 1 (LOWSP=1): T_{ADC2} = 6 * (1/333kHz) + 950ns ≈ 19us
4	**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.
3 ~ 0	**ADCIS[3: 0]**	ADC Input Selection Bits 0000: Select AIN0 as ADC input 0001: Select AIN1 as ADC input 0100: Select AIN4 as ADC input 0101: Select AIN5 as ADC input 0110: Select AIN6 as ADC input 0111: Select AIN7 as ADC input 1000: Select AIN8 as ADC input 1001: Select AIN9 as ADC input 1111: ADC input is 1/4 V_{DD}, used for measuring power voltage Others: 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	-	-	EAIN1	EAIN0
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
POR	0	0	0	0	x	x	0	0

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

Bit Number	7	6	5	4	3	2	1	0

Bit Mnemonic	-	-	-	-	-	-	EAIN9	EAIN8
R/W	-	-	-	-	-	-	R/W	R/W
POR	x	x	x	x	x	x	0	0

Bit Number	Bit Mnemonic	Description
0	**EAINx** **(x=0 ~ 1, 4 ~ 9)**	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	-	-	-	IAPS[1: 0]		-	-
R/W	R/W	-	-	-	R/W	R/W	-	-
POR	n	x	x	x	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	-	ET0	EINT0
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	R/W
POR	0	0	0	0	0	x	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	-	IPT0	IPINT0
R/W	-	R/W	R/W	-	R/W	-	R/W	R/W
POR	x	0	0	x	0	x	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"

16.2 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.

17 EEPROM and IAP Operations

There are two options for the SC92F730X IAP operating scope:

EEPROM and IAP operating modes are shown below:

1. 128 bytes EEPROM can be used as data storage;
2. The Code area of IC ROM and 128 bytes of EEPROM can be used for IAP operations, which is mainly used for remote program updating.

Note：The number of erasure of EEPROM is 100,000 times. The user should not exceed the rated burn number of EEPROM, otherwise there will be an exception!

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	-	-	-	IAPS[1: 0]		-	-
R/W	R/W	-	-	-	R/W	R/W	-	-
POR	n	x	x	x	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 (1E00H ~ 1FFFH) 10: Last 1k code memory allows IAP operation (1C00H ~ 1FFFH) 11: All code memory allows IAP operation (0000H ~ 1FFFH)

17.1 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

Mnemonic	Add	Description	7	6	5	4	3	2	1	0	POR
IAPADH	F3H	IAP Write Address High Register	-		-		-		IAPADR[12: 8]		xxx00000b
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 EEPROM / IAP function; ② If no writing command is received after n system clocks, EEPROM / 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) IAP Write Address High Register (Read/Write)

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	-	-	-	IAPADR[12: 8]				
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	0	0	0	0	0

Bit Number	Bit Mnemonic	Description
4 ~ 0	**IAPADR[12: 8]**	EEPROM/IAP writing address high 5-bit
7 ~ 5	-	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 0x01: Perform reading operation on user's ID area, but can't perform writing operation 0x02: MOVC for EEPROM **Note：The number of erasure of EEPROM is 100,000 times. The user should not exceed the rated burn number of EEPROM, otherwise there will be an exception!** Other: Reserved

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

Bit Number	7	6	5	4	3	2	1	0
Bit Mnemonic	IAPDAT[7: 0]							

Bit Number	7	6	5	4	3	2	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 ~ 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 4mS@24/12/6/2 MHz 01: Configure CPU HOLD TIME 2mS@24/12/6/2 MHz 10: Configure CPU HOLD TIME 1mS@24/12/6/2 MHz 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 only the last interrupt can be saved among several interrupts.** 1. Select 10 for the Flash ROM IAP operation. **Note: Users should set the LVR to 3.7V or higher and ensure that the VDD voltage range is between 3.7V and 5.5V when performing IAP operations on Flash ROM** 2. Select 00 or 01 for the EEPROM IAP operation. **Note: Users should ensure that the VDD voltage range is between 2.4V and 5.5V when performing IAP operations on EEPROM applications**
1 ~ 0	**CMD[1: 0]**	EEPROM / IAP writing operating command 10: Write Other: 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 EEPROM/IAP operation!**

17.2 EEPROM / IAP Operating Procedures:

Writing procedure of the SC92F730X 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[12: 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.

Note：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.

17.2.1 128 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, 1ms@24/12/6/2 MHz;
_nop_ ();                   //Wait (at least 8 _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
```

17.2.2 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, 1ms@24/12/6/2MHz;

_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 8K 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.

18 Electrical Characteristics

18.1 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	℃
I_{VDD}	Current value flowing through VDD	-	200	mA
I_{VSS}	Current value flowing through VSS	-	200	mA

18.2 Recommended Operating Conditions

Symbol	Parameter	Min Value	Max Value	Unit	System Clock Frequency
V_{DD}	Operating voltage 1	3.7	5.5	V	24MHz
	Operating voltage 2	2.4	5.5	V	12/6/2MHz
$V_{DD(Flash)}$	Flash ROM operating voltage during IAP operation	3.7	5.5	V	-
$V_{DD(EEPROM)}$	EEPROM operating voltage during IAP operation	2.4	5.5	V	-
T_A	Ambient temperature	-40	85	℃	-

18.3 DC Characteristics

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

Symbol	Parameter	Min Value	Typical Value	Max Value	Unit	Testing Conditions
Current						
I_{op1}	Operating current	-	4	-	mA	f_{SYS} =24MHz

Symbol	Parameter	Min Value	Typical Value	Max Value	Unit	Testing Conditions
I_{op2}	Operating current	-	3.2	-	mA	f_{SYS} =12MHz
I_{op3}	Operating current	-	2.7	-	mA	f_{SYS} =6MHz
I_{op4}	Operating current	-	2.4	-	mA	f_{SYS} =2MHz
I_{pd1}	Standby Current (Power Down Mode)	-	3	6	μA	
I_{IDL1}	Standby Current (IDLE Mode)	-	2.2	-	mA	
I_{BTM}	Base Timer Operating Current	-	1	2	μA	BTMFS[3: 0]= 1000 One interrupt occurs for every 4.0 seconds
I_{WDT}	WDT Current	-	1	2	μA	WDTCKS[2: 0]= 000 WDT overflows every 500ms
I/O 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/tDIO UART Enter RX INT0, INT2 Timer clock input
V_{IL2}	Input low voltage	-0.2	-	$0.2V_{DD}$	V	
I_{OL1}	Output low current	-	37	-	mA	V_{Pin}=0.4V
I_{OL2}	Output low current	-	70	-	mA	V_{Pin}=0.8V
I_{OH1}	Output high current P1	-	22	-	mA	V_{Pin}=4.3V
I_{OH2}	Output high current P1	-	11	-	mA	V_{Pin}=4.7V
I_{OH3}	Output high current P0/P2	-	22	-	mA	V_{Pin}=4.3V Pxyz = 0, I_{OH} level 0
	Output high current P0/P2	-	11	-	mA	V_{Pin}=4.3V

Symbol	Parameter	Min Value	Typical Value	Max Value	Unit	Testing Conditions
						Pxyz = 1, I_{OH} level 1
	Output high current P0/P2	-	5	-	mA	V_{Pin}=4.3V Pxyz = 2, I_{OH} level 2
	Output high current P0/P2	-	2.2	-	mA	V_{Pin}=4.3V Pxyz = 3, I_{OH} level 3
I_{OH4}	Output high current P0/P2	-	11	-	mA	V_{Pin}=4.7V Pxyz = 0, I_{OH} level 0
	Output high current P0/P2	-	5.5	-	mA	V_{Pin}=4.7V Pxyz = 1, I_{OH} level 1
	Output high current P0/P2	-	2.2	-	mA	V_{Pin}=4.7V Pxyz = 2, I_{OH} level 2
	Output high current P0/P2	-	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	2.38	2.40	2.42	V	T_A=-40 ~ 85℃

(V_{DD} = 3.3V, T_A = +25℃, unless otherwise specified)

Symbol	Parameter	Min Value	Typical Value	Max Value	Unit	Conditions
Current						
I_{op5}	Operating current	-	3.2	-	mA	f_{SYS} =24MHz
I_{op6}	Operating current	-	2.6	-	mA	f_{SYS} =12MHz
I_{op7}	Operating current	-	2.3	-	mA	f_{SYS} =6MHz
I_{op8}	Operating current	-	2	-	mA	f_{SYS} =2MHz
I_{pd2}	Standby Current (Power Down Mode)	-	3	6	uA	
I_{IDL2}	Standby Current (IDLE Mode)	-	1.9	-	mA	
I/O Port Features						

Symbol	Parameter	Min Value	Typical Value	Max Value	Unit	Conditions
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/tDIO UART Enter RX INT0, INT2 Timer clock input
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	-	50	-	mA	V_{Pin}=0.8V
I_{OH5}	Output high current P1	-	7	-	mA	V_{Pin}=3.0V
I_{OH6}	Output high current P0/P2	-	7	-	mA	V_{Pin}=3.0V Pxyz = 0, I_{OH} level 0
	Output high current P0/P2	-	3.5	-	mA	V_{Pin}=3.0V Pxyz = 1, I_{OH} level 1
	Output high current P0/P2	-	1.4	-	mA	V_{Pin}=3.0V Pxyz = 2, I_{OH} level 2
	Output high current P0/P2	-	0.7	-	mA	V_{Pin}=3.0V Pxyz = 3, I_{OH} level 3
R_{PH2}	Pull-up resistance	-	52	-	kΩ	
Internal calibrated 2.4V as ADC reference voltage						
V_{DD24}	Internal reference 2.4V voltage output	2.38	2.40	2.42	V	TA=-40 ~ 85℃

18.4 AC Characteristics

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

Symbol	Parameter	Min	Typical	Max	Unit	Conditions
T_{POR}	Power On Reset time	-	15	-	ms	
T_{PDW}	Power Down Mode waking-up time	-	65	130	μs	
T_{Reset}	Reset Pulse Width	18	-	-	μs	Valid for Low level

Symbol	Parameter	Min	Typical	Max	Unit	Conditions
T_{LVR}	LVR anti-shake time	-	30	-	μs	
f_{HRC}	RC oscillation stability	23.76	24	24.24	MHz	V_{DD}=2.4 ~ 5.5V T_A=-40 ~ 85 ℃

18.5 ADC Characteristics

(T_A= 25℃, unless otherwise specified)

Symbol	Parameter	Min	Typical	Max	Unit	Conditions
V_{AD1}	Supply Voltage 1	2.7	5.0	5.5	V	V_{ref}=2.4V
V_{AD2}	Supply Voltage 2	2.4	5.0	5.5	V	V_{ref}=V_{DD}
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	-	±4	-	LSB	V_{DD}=5V V_{REF}=5V
INL	Integral nonlinear error	-	±4	-	LSB	
E_Z	Offset error	-	±1	-	LSB	
E_F	Full scale error	-	4	-	LSB	
E_{AD}	Total absolute error	-	±4	-	LSB	
T_{ADC1}	ADC conversion time 1	-	4	-	μs	ADC Clock = 2MHz
T_{ADC2}	ADC conversion time 2	-	19	-	μs	ADC Clock = 333kHz

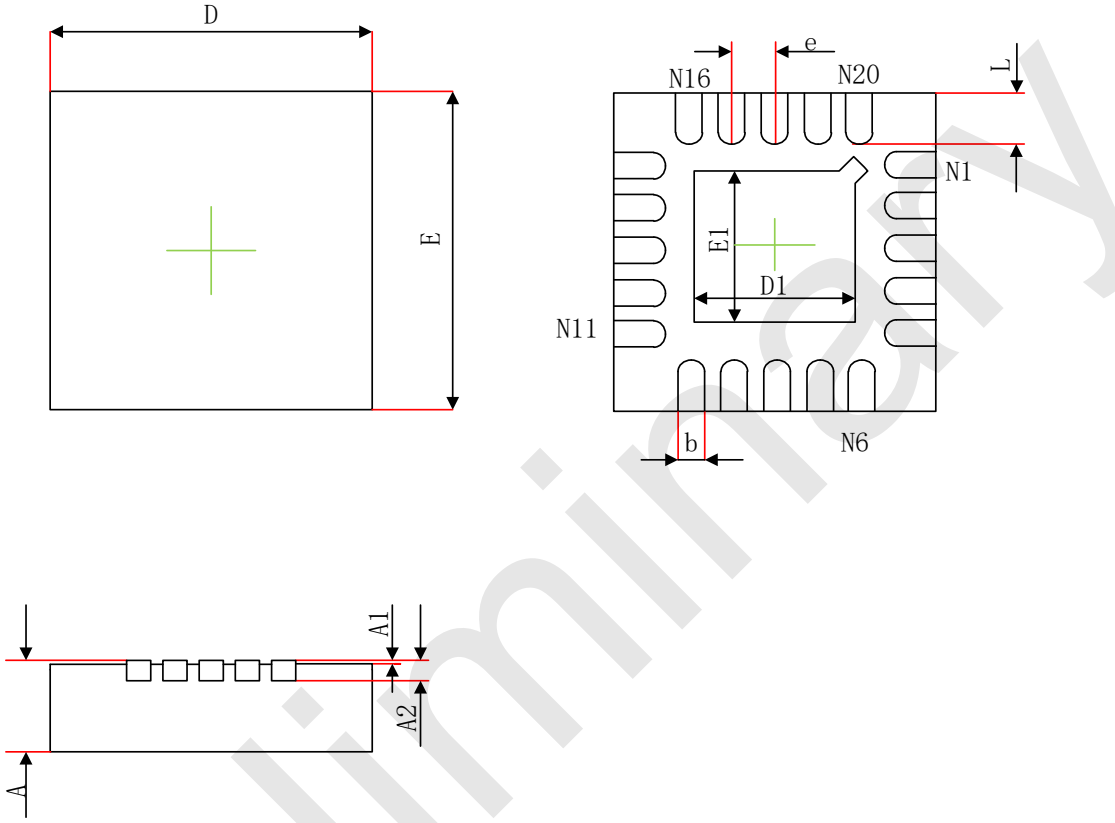
19 Ordering Information

PRODUCT NO	PKG	PACKING
SC92F7300M08U	SOP8	TUBE
SC92F7301M16U	SOP16	TUBE
SC92F7302M20U	SOP20	TUBE
SC92F7302X20U	TSSOP20	TUBE
SC92F7302Q20R	QFN20	TRAY

20 Packaging Information

SC92F7302Q20R

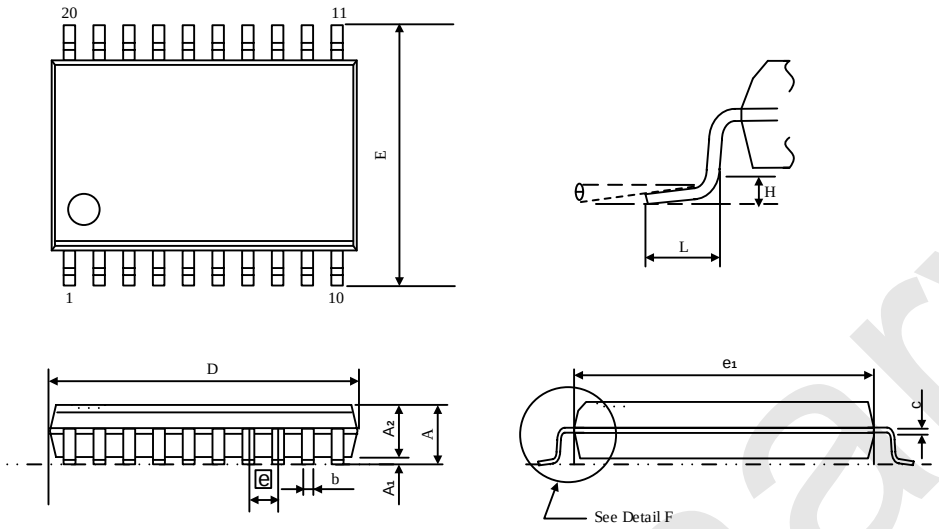
QFN20(4X4) Overall Dimension Unit:mm



Symbol	mm		
	Min	Normal	Max
A	0.700	0.750	0.800
A1	0	-	0.050
A2	0.153	0.203	0.253
b	0.180	0.250	0.300
D	3.900	4.000	4.100
D1	1.900	2.000	2.100
E	3.900	4.000	4.100
E1	1.900	2.000	2.100
e	0.450	0.500	0.550
L	0.390	0.400	0.410

SC92F7302X20U

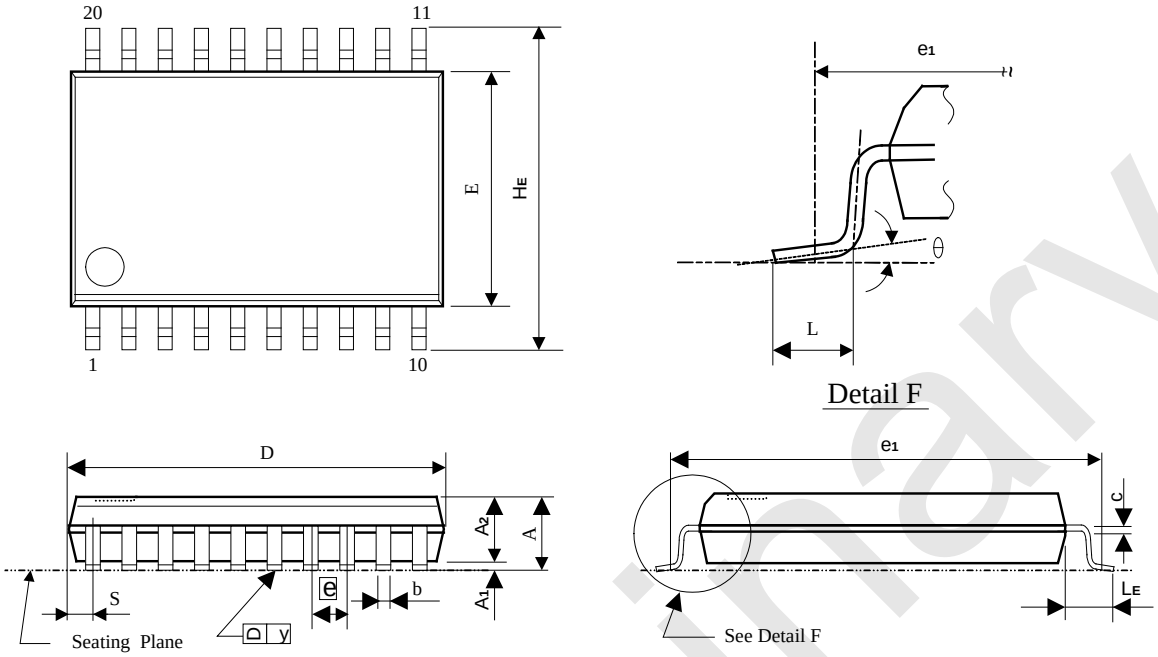
TSSOP20L Overall Dimension **Unit:mm**



Symbol	mm		
	Min	Normal	Max
A	-	-	1.200
A1	0.050	-	0.150
A2	0.800	-	1.050
b	0.190	-	0.300
c	0.090	-	0.200
D	6.400	-	6.600
E	6.20	-	6.60
e1	4.300	-	4.500
e	0.65(BSC)		
L	-	-	1.00
θ	0°	-	8°
H	0.05		0.15

SC92F7302M20U

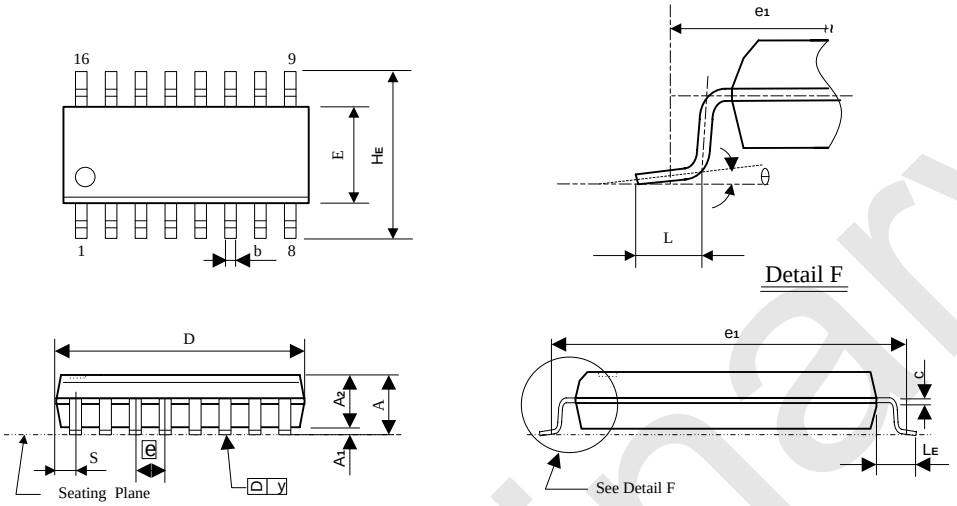
SOP20L(300mil) Overall Dimension **Unit:mm**



Symbol	mm		
	Min	Normal	Max
A	2.40	2.56	2.65
A1	0.100	0.200	0.300
A2	2.240	2.340	2.440
b	0.35	--	0.47
c	0.25	--	0.31
D	12.60	12.80	13.00
E	7.30	7.50	7.70
HE	10.100	10.300	10.500
e	1.27(BSC)		
L	0.700	0.850	1.000
LE	1.30	1.40	1.50
θ	0°	-	8°

SC92F7301M16U

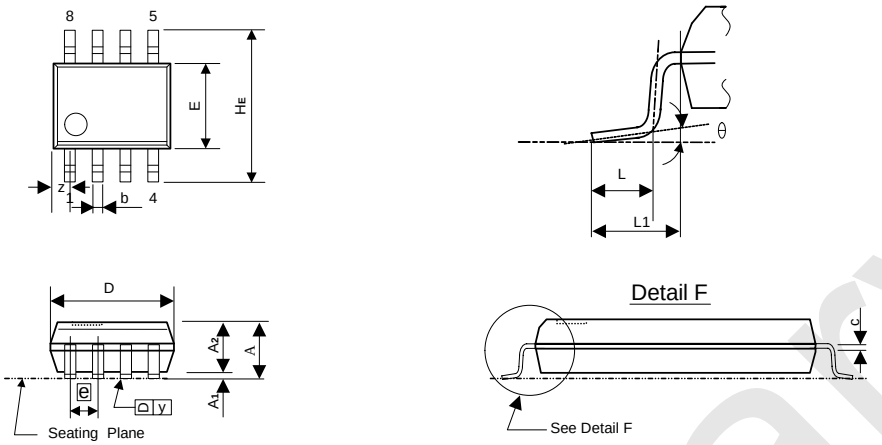
SOP16L(150mil) Overall Dimension **Unit:mm**



Symbol	mm		
	Min	Normal	Max
A	1.500	1.625	1.750
A1	0.050	0.1375	0.225
A2	1.30	1.45	1.55
b	0.38	0.43	0.48
c	0.20	0.23	0.26
D	9.70	9.90	10.10
E	3.70	3.90	4.10
HE	5.80	6.00	6.20
e	1.27(BSC)		
L	0.50	0.65	0.80
LE	0.95	1.05	1.15
θ	0°	-	8°

SC92F7300M08U

SOP8L(150mil) Overall Dimension Unit:mm



Symbol	mm		
	Min	Normal	Max
A	1.500	1.625	1.750
A1	0.100	0.1625	0.225
A2	1.30	1.425	1.55
b	0.39	0.435	0.48
c	0.20	0.23	0.26
D	4.70	4.90	5.10
E	3.70	3.90	4.10
HE	5.80	6.00	6.20
e	1.270(BSC)		
L	0.50	0.65	0.80
L1	0.95	1.05	1.15
θ	0°	-	8°

21 Revision History

Version	Notes	Date
V0.1	Initial Release.	2024.02.01

Important Notice

Shenzhen SinOne Microelectronics Co., Ltd. (hereinafter referred to as SinOne) reserves the right to change, correct, enhance, modify and improve SinOne products, documents or services at any time without prior notice. SinOne considers the information provided to be accurate and reliable. The information in this document will be used in February 2024. In the actual production design, please refer to the latest data manual of each product and other relevant materials.