高速 1T 8051 内核 Flash MCU，256 bytes SRAM，8 Kbytes Flash，128 bytes 独立 EEPROM，12 位 ADC，5 路 8 位 PWM，3 个定时器，UART

1 总体描述

SC92F730X 系列是一颗增强型的 1T 8051 内核工业级 Flash 微控制器，指令系统完全兼容传统 8051 产品系列。

SC92F730X 集成有 8 Kbytes Flash ROM、256 bytes SRAM、128 bytes EEPROM、最多 12 个 GP I/O、6 个 IO 可外部中断、3 个 16 位定时器、5 路 12 位高精度 ADC、5 路独立 8 位 PWM、IO 驱动分级控制（P0 和 P2 口）、内部 1%高精度高频 12/6/2MHz 振荡器和±4%精度低频 128kHz 振荡器、UART 等通讯接口等资源。为提高可靠性及简化客户电路，SC92F730X 内部也集成有 4 级可选电压 LVR、2.4V 基准 ADC 参考电压、WDT 等高可靠电路。SC92F730X 具有非常优异的抗干扰性能，非常适合应用于各种物联网控制、大小智能家电和智能家居、充电器、电源、航模、对讲机、无线通讯、游戏机等工业控制和消费应用领域。

2 主要功能

工作电压： 2.4V~5.5V

工作温度： -40 ~ 85℃

封装：

SC92F7309X14U（TSSOP14）

SC92F7308BS10U（MSOP10）

内核： 高速 1T 8051

Flash ROM： 8 Kbytes Flash ROM（MOVC 禁止寻址 0000H~00FFH）可重复写入 1 万次

IAP：可 code option 成 0K、0.5K、1K 或 8K

EEPROM：独立的 128 bytes，可重复写入 10 万次，10 年以上保存寿命

SRAM：内部 256 bytes

系统时钟（f_{SYS}）：

- 内建高频 24MHz 振荡器（f_{HRC}）
- IC 工作的系统时钟可通过编程器选择设定为 12/6/2MHz (2.4-5.5V)
- 频率误差： 跨越 (4.0V~5.5V) 及 (-20 ~ 85℃) 应用环境, 不超过 ±1%。

内建低频 **128kHz LRC** 振荡器：

- 可作为 BaseTimer 的时钟源，并唤醒 STOP
- 可作为 WDT 的时钟源
- 频率误差： 跨越 (4.0V ~ 5.5V) 及 (-20 ~ 85℃) 应用环境，频率误差不超过 ±4%

低电压复位（**LVR**）：

- 复位电压有 4 级可选：分别是：4.3V、3.7V、2.9V、2.3V
- 缺省值为用户烧写 Code Option 所选值

Flash 烧写接口：

- 2 线烧写接口

中断（**INT**）：

- Timer0， Timer1， Timer2， INT0， INT2， ADC，PWM，UART，Base Timer 共 9 个中断源
- 外部中断有 2 个中断向量，共 6 个中断口，全部可设上升沿、下降沿、双沿中断
- 两级中断优先级可设

数字外围：

- 最大 12 个双向可独立控制的 I/O 口，可独立设定上拉电阻
- P0、P2 口源驱动能力分四级控制
- 全部 IO 具有大灌电流驱动能力（47mA）
- 11 位 WDT，可选时钟分频比
- 3 个标准 80C51 定时器 Timer0、Timer1 和 Timer2
- 5 路共用周期、单独可调占空比的 8 位 PWM
- 4 个 IO 可作为 1/2 BIAS 的 LCD COM 输出
- 1 个独立 UART 通信口

模拟外围：

- 5 路 12 位±2LSB ADC
  - 内建基准的 2.4V 参考电压
  - ADC 的参考电压有 2 种选择，分别是 V_{DD} 以及内部 2.4V
  - 内部一路 ADC 可直接测量 V_{DD} 电压
  - 可设 ADC 转换完成中断

省电模式：

- IDLE Mode，可由任何中断唤醒
- STOP Mode，可由 INT0、2 和 BaseTimer 唤醒

目录

1 总体描述 1

2 主要功能 1

目录 2

3 管脚定义 5

3.1 管脚配置 5

3.2 管脚定义 5

4 内部框图 7

5 FLASH ROM 和 SRAM 结构 8

5.1 flash rom 8

5.2 Customer Option 区域(用户烧写设置) 9
5.2.1 Option 相关 SFR 操作说明 10

5.3 sram 11
5.3.1 256 bytes SRAM 11

6 特殊功能寄存器(SFR) 13

6.1 SFR 映像 13

6.2 SFR 说明 14
6.2.1 8051 CPU 内核常用特殊功能寄存器介绍 15

7 电源、复位和时钟 16

7.1 电源电路 16

7.2 上电复位过程 16
7.2.1 复位阶段 16
7.2.2 调入信息阶段 16
7.2.3 正常操作阶段 16

7.3 复位方式 16
7.3.1 外部 RST 复位 16
7.3.2 低电压复位 LVR 16
7.3.3 上电复位 POR 17
7.3.4 看门狗复位 WDT 17
7.3.5 复位初始状态 18

7.4 高频系统时钟电路 19

7.5 低频振荡器及低频时钟定时器 21

7.6 STOP 模式和 IDLE 模式 22

8 中央处理单元 CPU 及指令系统 23

8.1 CPU 23

8.2 寻址方式 23
8.2.1 立即寻址 23
8.2.2 直接寻址 23
8.2.3 间接寻址 23
8.2.4 寄存器寻址 23
8.2.5 相对寻址 23
8.2.6 变址寻址 23
8.2.7 位寻址 23

9 INTERRUPT 中断 24

9.1 中断源、向量 24

9.2 中断结构图 25

9.3 中断优先级 26

9.4 中断处理流程 26

9.5 中断相关 SFR 寄存器 26

10 定时器 TIMER0 、TIMER1 29

10.1 T0 和 T1 相关特殊功能寄存器 29

10.2 T0 工作模式 31

10.3 T1 工作模式 33

11 定时器 TIMER2 34

11.1 T2 相关特殊功能寄存器 34

11.2 T2 工作模式 35

12 PWM 38

12.1 PWM 结构框图 38

12.2 PWM 相关 SFR 寄存器 39

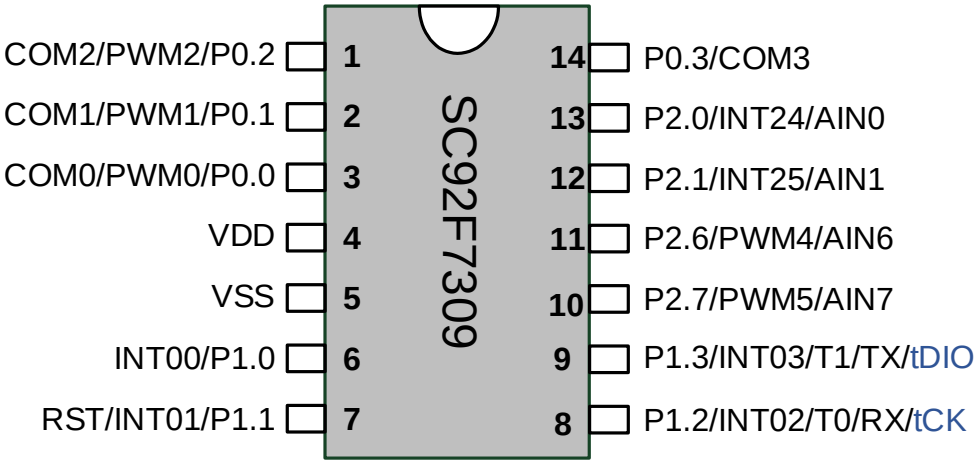
12.3 PWM 波形及用法 42

13 GP I/O 44

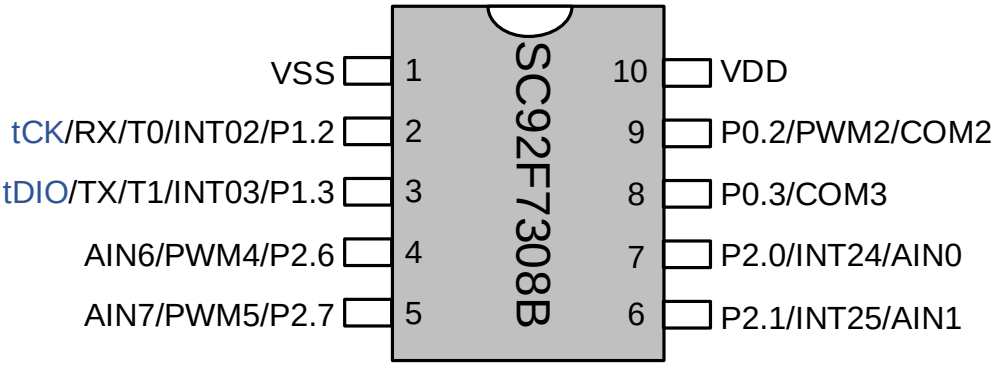
13.1 GPIO 结构图 44
13.2 I/O 端口相关寄存器 45
14 软件 LCD 驱动 47
14.1 软件 LCD 驱动相关寄存器 47
15 UART 48
15.1 UART 相关寄存器 48
15.2 串口通信的波特率 49
16 模数转换 ADC 50
16.1 ADC 相关寄存器 50
16.2 ADC 转换步骤 52
17 EEPROM 及 IAP 操作 53
17.1 EEPROM / IAP 操作相关寄存器 53
17.2 EEPROM / IAP 操作流程 55
17.2.1 128 bytes 独立 EEPROM 操作例程 55
17.2.2 8 Kbytes CODE 区域 IAP 操作例程 56
18 电气特性 57
18.1 极限参数 57
18.2 推荐工作条件 57
18.3 直流电气特性 57
18.4 交流电气特性 58
18.5 ADC 电气特性 59
19 订购信息 60
20 封装信息 61
21 规格更改记录 63

3 管脚定义

3.1 管脚配置



SC92F7309 管脚配置图（TSSOP14）



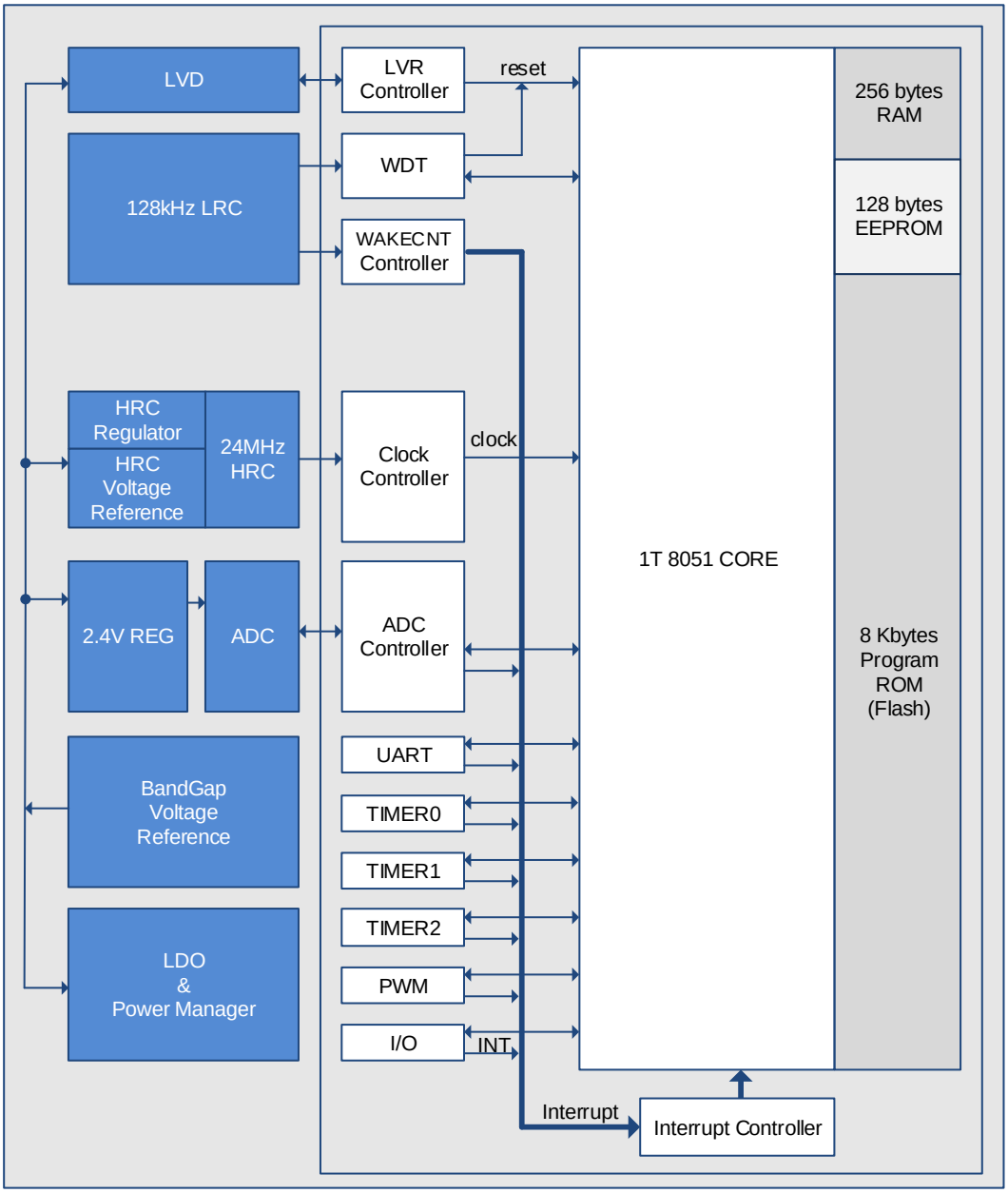
SC92F7308B 管脚配置图（MSOP10）

3.2 管脚定义

脚位数		管脚名称	类型	功能说明
14PIN	10PIN			
1	9	**P0.2/PWM2/COM2**	I/O	P0.2: GPIO P0.2 PWM2: PWM2 输出口 COM2: LCD 驱动公共端 COM2
2	-	**P0.1/PWM1/COM1**	I/O	P0.1: GPIO P0.1 PWM1: PWM1 输出口 COM1: LCD 驱动公共端 COM1
3	-	**P0.0/PWM0/COM0**	I/O	P0.0: GPIO P0.0 PWM0: PWM0 输出口 COM0: LCD 驱动公共端 COM0
4	10	**VDD**	Power	电源

5	**1**	**VSS**	Power	接地
6	**-**	**P1.0/INT00**	I/O	P1.0: GPIO P1.0 INT00：外部中断 0 的输入 0
7	**-**	**P1.1/INT01/RST**	I/O	P1.1: GPIO P1.1 INT01: 外部中断 0 的输入 1 RST: 复位管脚
8	**2**	**P1.2/INT02/T0/RX/tCK**	I/O	P1.2: GPIO P1.2 INT02: 外部中断 0 的输入 2 T0: 计数器 0 外部输入 RX: UART 接收 tCK: 烧录和仿真口时钟线
9	**3**	**P1.3/INT03/T1/TX/tDIO**	I/O	P1.3: GPIO P1.3 INT03: 外部中断 0 的输入 3 T1: 计数器 1 外部输入 TX: UART 发送 tDIO: 烧录和仿真口数据线
10	**5**	**P2.7/PWM5/AIN7**	I/O	P2.7: GPIO P2.7 PWM5: PWM5 输出口 AIN7: ADC 输入通道 7
11	**4**	**P2.6/PWM4/AIN6**	I/O	P2.6: GPIO P2.6 PWM4: PWM4 输出口 AIN6: ADC 输入通道 6
12	**6**	**P2.1/INT25/AIN1**	I/O	P2.1: GPIO P2.1 INT25: 外部中断 2 的输入 5 AIN1: ADC 输入通道 1
13	**7**	**P2.0/INT24/AIN0**	I/O	P2.0: GPIO P2.0 INT24: 外部中断 2 的输入 4 AIN0: ADC 输入通道 0
14	**8**	**P0.3/COM3**	I/O	P0.3: GPIO P0.3 COM3: LCD 驱动公共端 COM3

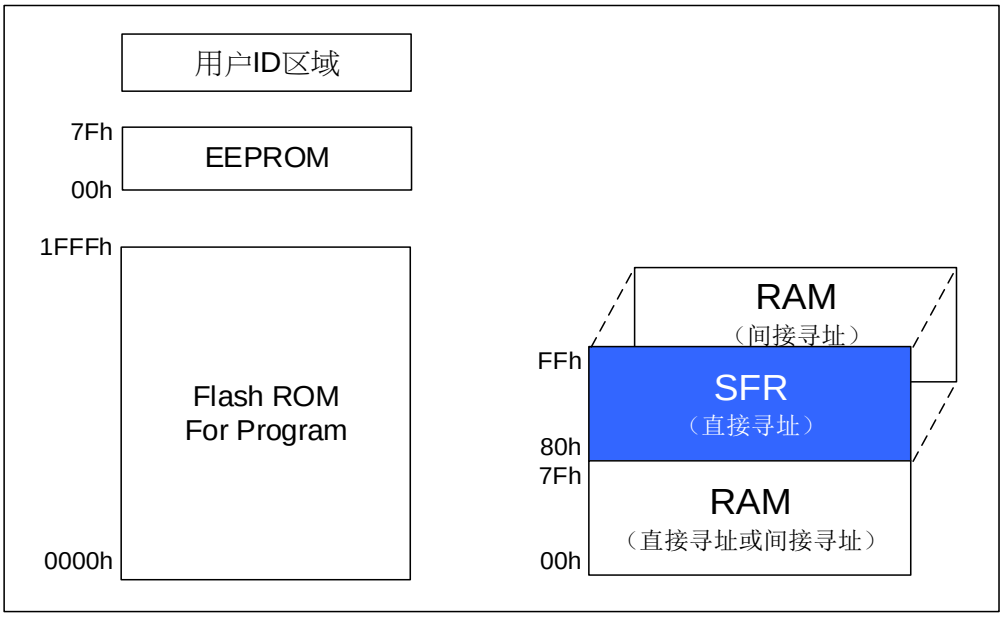
4 内部框图



SC92F730X BLOCK DIAGRAM

5 FLASH ROM 和 SRAM 结构

SC92F730X 的 Flash ROM 和 SRAM 结构如下：



Flash ROM 和 SRAM 结构图

5.1 FLASH ROM

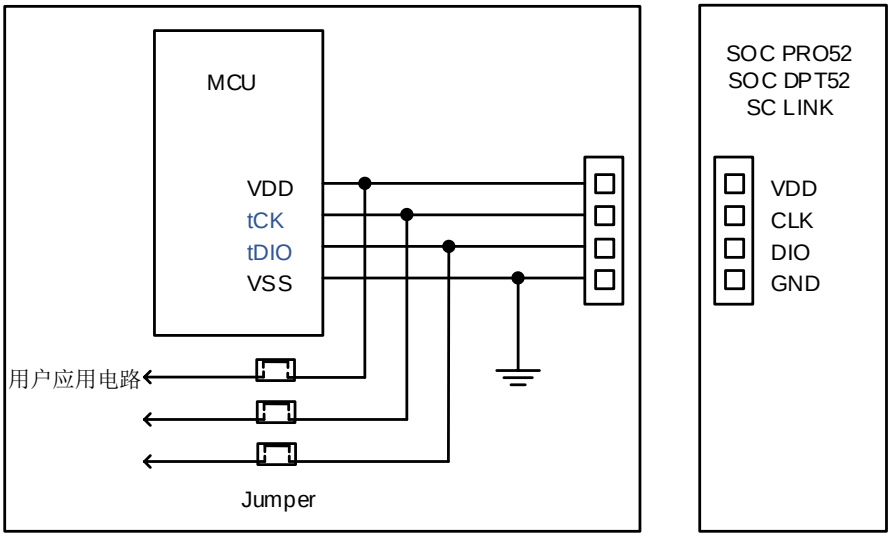
SC92F730X 有 8 Kbytes 的 Flash ROM，ROM 地址为 0000H~1FFFH。此 8 Kbytes Flash ROM 可反复擦写 1 万次，可通过 SinOne 提供的专用 ICP 烧写器(SOC Pro52/DPT52/SC LINK)来进行编程及擦除。地址为 0000H~00FFH 地址的 256 bytes 区间 MOVC 指令不可寻址。

EEPROM 为独立于 8 Kbytes ROM 之外的一块区间，其地址为 00H~7FH，可在程序中对其进行单 byte 读写操作，具体操作方法参考 17 EEPROM 及 IAP 操作。

用户 ID 区域：出厂时写入用户 ID，用户只可对其进行读操作，具体操作方式参考 17 EEPROM 及 IAP 操作。

SC92F730X 的 8 Kbytes Flash ROM 能提供查空 BLANK、编程 PROGRAM、校验 VERIFY 和擦除 ERASE 功能，但不提供读取 READ 的功能。此 Flash ROM 和 EEPROM 通常写入前无需进行擦除操作，直接写入数据即可实现新数据的覆盖。

SC92F730X 的 Flash ROM 通过 tDIO、tCK、V_{DD}、VSS 来进行编程，具体连接关系如下：



ICP 模式 Flash Writer 编程连接示意图

5.2 CUSTOMER OPTION 区域(用户烧写设置)

SC92F730X 内部有单独的一块 Flash 区域用于保存客户的上电初始值设置，此区域称为 Customer Option 区域。用户在烧写 IC 时将此部分代码写入 IC 内部，IC 在复位初始化时，就会将此设置调入 SFR 作为初始设置。

OPINX 值	OPREG										上电初始值
	符号	说明	7	6	5	4	3	2	1	0	
83H@FFH	OP_HRCR	高频 RC 震荡频率调节(细调)	OP_HRCR[7:0]								nnnnnnnnb
C1H@FFH	OP_CTM0	Customer Option 寄存器 0	ENWDT	-	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]		nxnnnnnnb
C2H@FFH	OP_CTM1	Customer Option 寄存器 1	VREFS	-	-	-	IAPS[1:0]		-	-	nxxxnnxxb

OP_HRCR (83H@FFH) 系统时钟改变寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	OP_HRCR[7:0]							
读/写	读/写							
上电初始值	n	n	n	n	n	n	n	n

位编号	位符号	说明
7~0	**OP_HRCR[7:0]**	内部高频 RC 频率调校 中心值 10000000b 对应 HRC 中心频率，数值变大频率加快，数值变小频率变慢。

OP_CTM0 (C1H@FFH) Customer Option 寄存器 0(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ENWDT	-	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
读/写	读/写	-	读/写		读/写	读/写	读/写	
上电初始值	n	x	n		n	n	n	

位编号	位符号	说明
7	**ENWDT**	WDT 开关

		0：WDT 无效 1：WDT 有效（但 IC 在执行 IAP 过程中 WDT 停止计数）
5~4	**SCLKS[1:0]**	系统时钟频率选择： 00：保留； 01：系统时钟频率为高频振荡器频率除以 2； 10：系统时钟频率为高频振荡器频率除以 4； 11：系统时钟频率为高频振荡器频率除以 12。
3	**DISRST**	IO/RST 复位切换控制 0 ：P1.1 当复位脚使用 1 ：P1.1 当正常的 I/O 管脚使用
2	**DISLVR**	LVR 使能设置 0：LVR 正常使用 1：LVR 无效
1~0	**LVRS [1:0]**	LVR 电压选择控制 11： 4.3V 复位 10： 3.7V 复位 01： 2.9V 复位 00： 2.3V 复位

OP_CTM1 (C2H@FFH) Customer Option 寄存器 1(读/写)

位编号	7	6	5	4	3	2	1	0
符号	VREFS	-	-	-	IAPS[1:0]		-	-
读/写	读/写	-	-	-	读/写	读/写	-	-
上电初始值	n	x	x	x	n	n	x	x

位编号	位符号	说明
7	**VREFS**	**参考电压选择(初始值从 Code Option 调入，用户可修改设置)** 0：设定 ADC 的 VREF 为 V_{DD} 1：设定 ADC 的 VREF 为 内部准确的 2.4V
3~2	**IAPS[1:0]**	IAP 空间范围选择 00: Code 区域禁止 IAP 操作，仅 EEPROM 区域可作为数据存储使用 01: 最后 0.5K Code 区域允许 IAP 操作(1E00H ~1FFFH) 10: 最后 1K Code 区域允许 IAP 操作(1C00H~1FFFH) 11: 全部 Code 区域允许 IAP 操作(0000H~1FFFH)

5.2.1 OPTION 相关 SFR 操作说明

Option 相关 SFR 的读写操作由 OPINX 和 OPREG 两个寄存器进行控制，各 Option SFR 的具体位置由 OPINX 确定，各 Option SFR 的写入值由 OPREG 确定：

符号	地址	说明		上电初始值
OPINX	FEH	Option 指针	OPINX[7:0]	00000000b
OPREG	FFH	Option 寄存器	OPREG[7:0]	nnnnnnnnb

操作 Option 相关 SFR 时 OPINX 寄存器存放相关 OPTION 寄存器的地址，OPREG 寄存器存放对应的值。例如：要将 OP_HRCR 配置为 0x01，具体操作方法如下：

C 语言例程：

```
OPINX = 0x83;        //将 OP_HRCR 的地址写入 OPINX 寄存器
OPREG = 0x01;        //对 OPREG 寄存器写入 0x01（待写入 OP_HRCR 寄存器的值）
```

汇编例程：

```
MOV OPINX, #83H      ; 将 OP_HRCR 的地址写入 OPINX 寄存器
MOV OPREG, #01H      ; 对 OPREG 寄存器写入 0x01（待写入 OP_HRCR 寄存器的值）
```

注意：禁止向 OPINX 寄存器写入 Customer Option 区域 SFR 地址之外的数值！否则会造成系统运行异常！

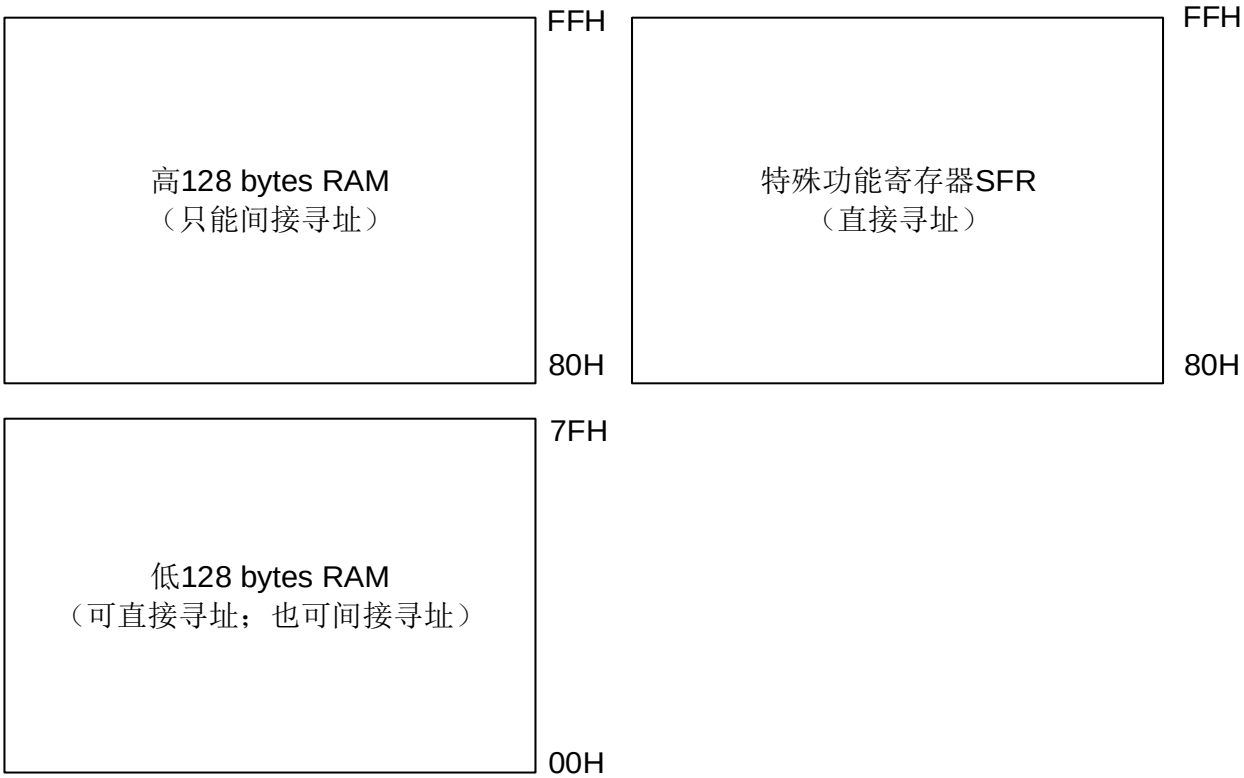
5.3 SRAM

SC92F730X 单片机的 SRAM 为内部的 256 bytes RAM。地址范围为 00H~FFH，其中高 128 bytes(地址 80H~FFH)只能间接寻址，低 128 bytes(地址 00H~7FH)可直接寻址也可间接寻址。

特殊功能寄存器 SFR 的地址也是 80H~FFH。但 SFR 同内部高 128 bytes SRAM 的区别是：SFR 寄存器是直接寻址，而内部高 128 bytes SRAM 只能是间接寻址。

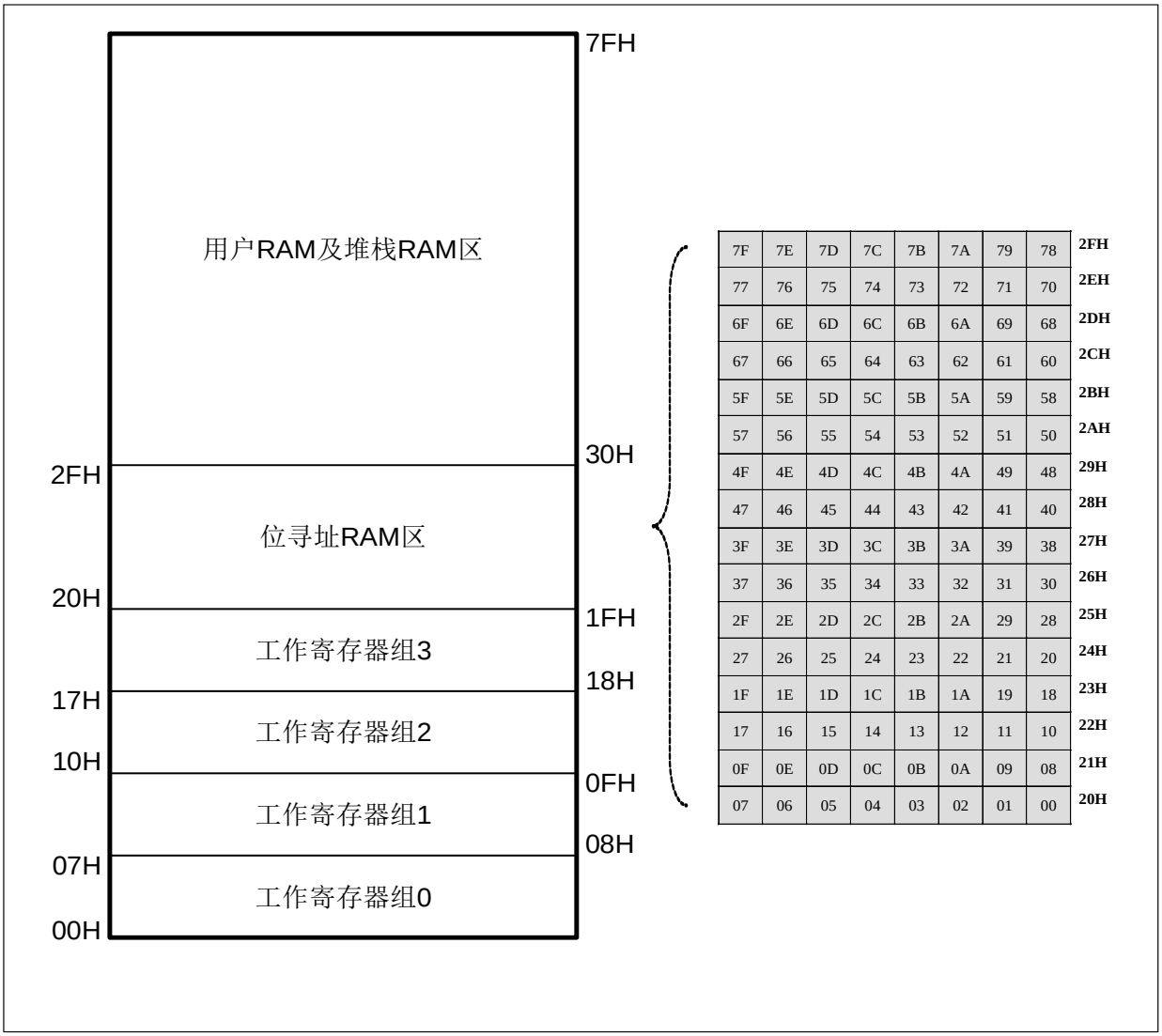
5.3.1 256 BYTES SRAM

低 128 bytes SRAM 区可分为三部分：①工作寄存器组 0~3，地址 00H~1FH,程序状态字寄存器 PSW 中的 RS0、RS1 组合决定了当前使用的工作寄存器，使用工作寄存器组 0~3 可加快运算的速度；②位寻址区 20H~2FH，此区域用户可以用作普通 RAM 也可用作按位寻址 RAM；按位寻址时，位的地址为 00H~7FH，(此地址按位编地址，不同于通用 SRAM 按字节编地址)，程序中可由指令区分；③用户 RAM 和堆栈区，SC92F730X 复位过后，8 位的堆栈指针指向堆栈区，用户一般会在初始化程序时设置初值，建议设置在 E0H~FFH 的单元区间。



256 bytes RAM 结构图

低 128 bytes RAM 结构如下：



SRAM 结构图

6 特殊功能寄存器(SFR)

6.1 SFR 映像

SC92F730X 系列有一些特殊功能寄存器，我们称为 SFR。这些 SFR 寄存器的地址位于 80H~FFH，有些可以位寻址，有些不能位寻址。能够进行位寻址操作的寄存器的地址末位数都是“0”或“8”，这些寄存器在需要改变单个位的数值时非常方便。所有的 SFR 特殊功能寄存器都必须使用直接寻址方式寻址。

SC92F730X 的特殊功能寄存器名称及地址如下表：

	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	-	-	-	-	-	-	PWMDTY4	PWMDTY5
D0h	PSW	PWMCFG0	PWMCON	PWMPRD	PWMCFG1	PWMDTY0	PWMDTY1	PWMDTY2
C8h	T2CON	-	RCAP2L	RCAP2H	TL2	TH2	BTMCON	WDTCON
C0h	-	-	-	-	-	-	INT2F	INT2R
B8h	IP	IP1	INT0F	INT0R	-	-	-	-
B0h	-	-	-	-	-	-	-	-
A8h	IE	IE1	-	ADCCFG0	-	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
	可位寻址	不可位寻址						

说明：

1. SFR 寄存器中空的部分代表没有此寄存器 RAM，不建议用户使用。

6.2 SFR 说明

特殊功能寄存器 SFR 的具体解释说明如下：

符号	地址	说明	7	6	5	4	3	2	1	0	上电初始值
P0	80H	P0 口数据寄存器	-	-	-	-	P03	P02	P01	P00	xxxx0000b
SP	81H	堆栈指针	SP[7:0]								00000111b
DPL	82H	DPTR 数据指针低位	DPL[7:0]								00000000b
DPH	83H	DPTR 数据指针高位	DPH[7:0]								00000000b
PCON	87H	电源管理控制寄存器	SMOD	-	-	-	-	-	STOP	IDL	0xxxxx00b
TCON	88H	定时器控制寄存器	TF1	TR1	TF0	TR0	-	-	-	-	0000xxxxb
TMOD	89H	定时器工作模式寄存器	-	C/T1	M11	M01	-	C/T0	M10	M00	x000x000b
TL0	8AH	定时器 0 低 8 位	TL0[7:0]								00000000b
TL1	8BH	定时器 1 低 8 位	TL1[7:0]								00000000b
TH0	8CH	定时器 0 高 8 位	TH0[7:0]								00000000b
TH1	8DH	定时器 1 高 8 位	TH1[7:0]								00000000b
TMCON	8EH	定时器频率控制寄存器	-	-	-	-	-	T2FD	T1FD	T0FD	xxxxx000b
OTCON	8FH	输出控制寄存器	-	-	-	-	VOIRS[1:0]		-	-	xxxx00xxb
P1	90H	P1 口数据寄存器	-	-	-	-	P13	P12	P11	P10	xxxx0000b
P1CON	91H	P1 口输入/输出控制寄存器	-	-	-	-	P1C3	P1C2	P1C1	P1C0	xxxx0000b
P1PH	92H	P1 口上拉电阻控制寄存器	-	-	-	-	P1H3	P1H2	P1H1	P1H0	xxxx0000b
IOHCON	97H	输出电流设置寄存器	P2H[1:0]		P2L[1:0]		P0H[1:0]		P0L[1:0]		00000000b
SCON	98H	串口控制寄存器	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00000000b
SBUF	99H	串口数据缓存寄存器	SBUF[7:0]								00000000b
P0CON	9AH	P0 口输入/输出控制寄存器	-	-	-	-	P0C3	P0C2	P0C1	P0C0	xxxx0000b
P0PH	9BH	P0 口上拉电阻控制寄存器	-	-	-	-	P0H3	P0H2	P0H1	P0H0	xxxx0000b
P0VO	9CH	P0 口 LCD 电压输出寄存器	-	-	-	-	P03VO	P02VO	P01VO	P00VO	xxxx0000b
P2	A0H	P2 口数据寄存器	P27	P26	-	-	-	-	P21	P20	00xxxx00b
P2CON	A1H	P2 口输入/输出控制寄存器	P2C7	P2C6	-	-	-	-	P2C1	P2C0	00xxxx00b
P2PH	A2H	P2 口上拉电阻控制寄存器	P2H7	P2H6	-	-	-	-	P2H1	P2H0	00xxxx00b
IE	A8H	中断使能寄存器	EA	EADC	ET2	EUART	ET1	-	ET0	EINT0	00000x00b
IE1	A9H	中断使能寄存器 1	-	-	-	-	EINT2	EBTM	EPWM	-	xxxx000xb
ADCCFG0	ABH	ADC 设置寄存器 0	EAIN7	EAIN6	-	-	-	-	EAIN1	EAIN0	00xxxx00b
ADCCON	ADH	ADC 控制寄存器	ADCEN	ADCS	LOWSP	EOC/ ADCIF	ADCIS[3:0]				00000000b
ADCVL	AEH	ADC 结果寄存器	ADCV[3:0]				-	-	-	-	0000xxxxb
ADCVH	AFH	ADC 结果寄存器	ADCV[11:4]								00000000b
IP	B8H	中断优先级控制寄存器	-	IPADC	IPT2	IPUART	IPT1	-	IPT0	IPINT0	x0000x00b
IP1	B9H	中断优先级控制寄存器 1	-	-	-	-	IPINT2	IPBTM	IPPWM	-	xxxx000xb
INT0F	BAH	INT0 下降沿中断控制寄存器	-	-	-	-	INT0F3	INT0F2	INT0F1	INT0F0	xxxx0000b
INT0R	BBH	INT0 上升沿中断控制寄存器	-	-	-	-	INT0R3	INT0R2	INT0R1	INT0R0	xxxx0000b
INT2F	C6H	INT2 下降沿中断控制寄存器	-	-	INT2F5	INT2F4	-	-	-	-	xx00xxxxb
INT2R	C7H	INT2 上升沿中断控制寄存器	-	-	INT2R5	INT2R4	-	-	-	-	xx00xxxxb
T2CON	C8H	定时器 2 控制寄存器	TF2	-	RCLK	TCLK	-	TR2	-	-	0x00x0xxb
RCAP2L	CAH	定时器 2 重载低 8 位	RCAP2L[7:0]								00000000b
RCAP2H	CBH	定时器 2 重载高 8 位	RCAP2H[7:0]								00000000b
TL2	CCH	定时器 2 低 8 位	TL2[7:0]								00000000b
TH2	CDH	定时器 2 高 8 位	TH2[7:0]								00000000b
BTMCON	CEH	低频定时器控制寄存器	ENBTM	BTMIF	-	-	BTMFS[3:0]				00xx0000b
WDTCON	CFH	WDT 控制寄存器	-	-	-	CLRWDT	-	WDTCKS[2:0]			xxx0x000b
PSW	D0H	程序状态字寄存器	CY	AC	F0	RS1	RS0	OV	F1	P	00000000b
PWMCFG0	D1H	PWM 设置寄存器 0	-	-	INV2	INV1	INV0	ENPWM5	ENPWM4	-	xx00000xb
PWMCON	D2H	PWM 控制寄存器	ENPWM	PWMIF	ENPWM2	ENPWM1	ENPWM0	PWMCKS[2:0]			00000000b
PWMPRD	D3H	PWM 周期设置寄存器	PWMPRD[7:0]								00000000b
PWMCFG1	D4H	PWM 设置寄存器 1	-	-	INV5	INV4	-	-	-	-	xx00xxxxb
PWMDTY0	D5H	PWM0 占空比设置寄存器	PDT0[7:0]								00000000b
PWMDTY1	D6H	PWM1 占空比设置寄存器	PDT1[7:0]								00000000b
PWMDTY2	D7H	PWM2 占空比设置寄存器	PDT2[7:0]								00000000b
PWMDTY4	DEH	PWM4 占空比设置寄存器	PDT4[7:0]								00000000b
PWMDTY5	DFH	PWM5 占空比设置寄存器	PDT5[7:0]								00000000b

ACC	E0H	累加器	ACC[7:0]						00000000b
B	F0H	B 寄存器	B[7:0]						00000000b
IAPKEY	F1H	IAP 保护寄存器	IAPKEY[7:0]						00000000b
IAPADL	F2H	IAP 写入地址低位寄存器	IAPADR[7:0]						00000000b
IAPADH	F3H	IAP 写入地址高位寄存器	-	-	-	IAPADR[12:8]			xxx00000b
IAPADE	F4H	IAP 写入扩展地址寄存器	IAPADER[7:0]						00000000b
IAPDAT	F5H	IAP 数据寄存器	IAPDAT[7:0]						00000000b
IAPCTL	F6H	IAP 控制寄存器	-	-	-	-	PAYTIMES[1:0]	CMD[1:0]	xxxx0000b
OPINX	FEH	Option 指针	OPINX[7:0]						00000000b
OPREG	FFH	Option 寄存器	OPREG[7:0]						nnnnnnnnb

6.2.1 8051 CPU 内核常用特殊功能寄存器介绍

程序计数器 PC

程序计数器 PC 不属于 SFR 寄存器。PC 有 16 位，是用来控制指令执行顺序的寄存器。单片机上电或者复位后，PC 值为 0000H，也即是说单片机程序从 0000H 地址开始执行程序。

累加器 ACC (E0H)

累加器 ACC 是 8051 内核单片机的最常用的寄存器之一，指令系统中采用 A 作为助记符。常用来存放参加计算或者逻辑运算的操作数及结果。

B 寄存器 (F0H)

B 寄存器在乘除法运算中必须与累加器 A 配合使用。乘法指令 MUL A，B 把累加器 A 和寄存器 B 中的 8 位无符号数相乘，所得的 16 位乘积的低位字节放在 A 中，高位字节放在 B 中。除法指令 DIV A，B 是用 A 除以 B，整数商放在 A 中，余数放在 B 中。寄存器 B 还可以作为通用的暂存寄存器使用。

堆栈指针 SP (81H)

堆栈指针是一个 8 位的专用寄存器，它指示出堆栈顶部在通用 RAM 中的位置。单片机复位后，SP 初始值为 07H，即堆栈会从 08H 开始向上增加。08H~1FH 为工作寄存器组 1~3。

PSW (D0H) 程序状态字寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	CY	AC	F0	RS1	RS0	OV	F1	P
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

<table>
<tr><th>位编号</th><th>位符号</th><th>说明</th></tr>
<tr><td>7</td><td>CY</td><td>标志位
1：加法运算最高位有进位，或者减法运算最高位有借位时
0：加法运算最高位无进位，或者减法运算最高位无借位时</td></tr>
<tr><td>6</td><td>AC</td><td>进位辅助标志位(可在 BCD 码加减法运算时方便调整)
1：加法运算时在 bit3 位有进位，或减法运算在 bit3 位有借位时
0：无借位、进位</td></tr>
<tr><td>5</td><td>F0</td><td>用户标志位</td></tr>
<tr><td>4~3</td><td>RS1、RS0</td><td>工作寄存器组选择位:
<table>
<tr><td>RS1</td><td>RS0</td><td>当前使用的工作寄存器组 0~3</td></tr>
<tr><td>0</td><td>0</td><td>组 0 (00H~07H)</td></tr>
<tr><td>0</td><td>1</td><td>组 1 (08H~0FH)</td></tr>
<tr><td>1</td><td>0</td><td>组 2 (10H~17H)</td></tr>
<tr><td>1</td><td>1</td><td>组 3 (18H~1FH)</td></tr>
</table></td></tr>
<tr><td>2</td><td>OV</td><td>溢出标志位</td></tr>
<tr><td>1</td><td>F1</td><td>F1 标志
用户自定义标志</td></tr>
</table>

0	**P**	奇偶标志位。此标志位为累加器 ACC 中 1 的个数的奇偶值。 1：ACC 中 1 的个数为奇数 0：ACC 中 1 的个数为偶数(包括 0 个)

数据指针 DPTR (82H、83H)

数据指针 DPTR 是一个 16 位的专用寄存器，由低 8 位 DPL（82H）和高 8 位 DPH（83H）组成。DPTR 是以传统 8051 内核单片机中唯一可以直接进行 16 位操作的寄存器，也可以分别对 DPL 和 DPH 按字节进行操作。

7 电源、复位和时钟

7.1 电源电路

SC92F730X 电源核心包括了 BG、LDO、POR、LVR 等电路，可实现在 2.4~5.5V 范围内可靠工作。此外，IC 内建了一个经调校过的精准 2.4V 电压，可用作 ADC 内部参考电压。用户可在 16 模数转换 ADC 查找具体设置内容。

7.2 上电复位过程

SC92F730X 上电后，在客户端软件执行前，会经过以下的过程：

- 复位阶段
- 调入信息阶段
- 正常操作阶段

7.2.1 复位阶段

是指 SC92F730X 会一直处于复位的情况，直到供应给 SC92F730X 的电压高过某一电压，内部才开始有效的 Clock。复位阶段的时间长短和外部电源的上升速度有关，外部电源达到内建 POR 电压后，复位阶段才会完成。

7.2.2 调入信息阶段

在 SC92F730X 内部有一个预热计数器。在复位阶段期间，此预热计数器一直被清为 0，直到电压过了 POR 电压后，内部 RC 振荡器开始起振，该预热计数器开始计数。当内部的预热计数器计数到一定数目后，每隔一定数量个 HRC clock 就会从 Flash ROM 中的 IFB（包含 Code Option）读出一个 byte 数据存放到内部系统寄存器中。直到预热完成后，该复位信号才会结束。

7.2.3 正常操作阶段

结束调入信息阶段后，SC92F730X 开始从 Flash 中读取指令代码即进入正常操作阶段。这时的 LVR 电压值是用户写入 Code Option 的设置值。

7.3 复位方式

SC92F730X 有 4 种复位方式：①外部 RST 复位②低电压复位 LVR③上电复位 POR④看门狗 WDT 复位。

7.3.1 外部 RST 复位

外部 RST 复位就是从外部 RST 给 SC92F730X 一定宽度的复位脉冲信号，来实现 SC92F730X 的复位。

RST/INT01/P1.1 具有复位功能，用户在烧录程序前可以通过烧录上位机软件配置 Customer Option 项来选择将其修改为非复位脚。

7.3.2 低电压复位 LVR

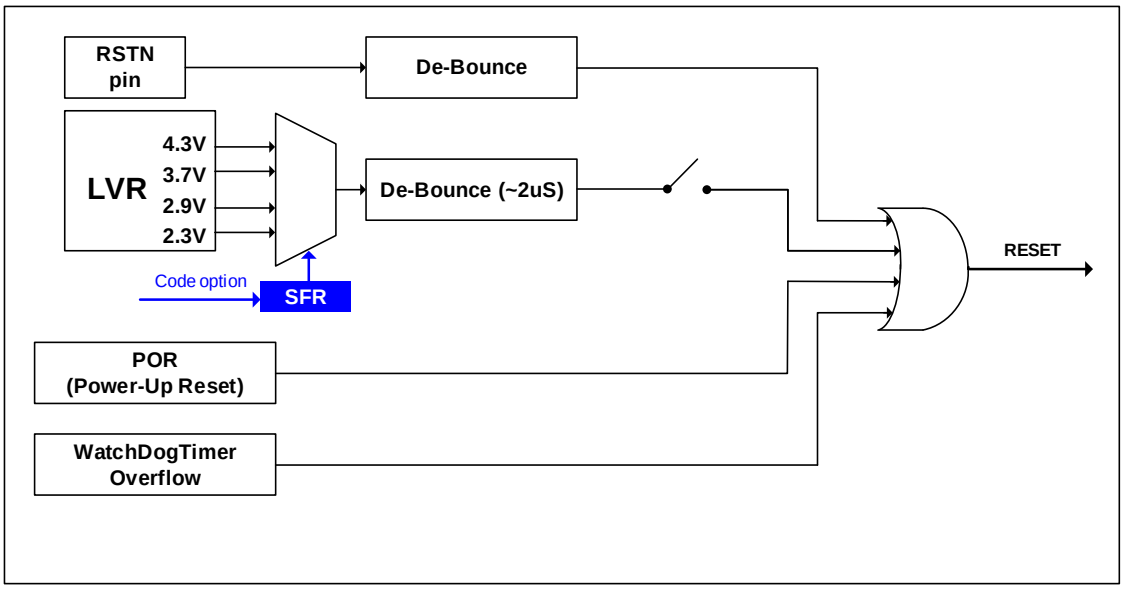
SC92F730X 内建了一个低电压复位电路。而复位的门限电压有 4 种选择：4.3V、3.7V、2.9V、2.3V。缺省值 Default 是用户写入的 Option 值。

OP_CTM0 (C1H@FFH) Customer Option 寄存器 0(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ENWDT	-	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
读/写	读/写	-	读/写		读/写	读/写	读/写	
上电初始值	n	x	n		n	n	n	

位编号	位符号	说明
2	**DISLVR**	LVR 使能设置 0：LVR 正常使用 1：LVR 无效
1~0	**LVRS [1:0]**	LVR 电压选择控制 11： 4.3V 复位 10： 3.7V 复位 01： 2.9V 复位 00： 2.3V 复位

SC92F730X 的复位部分电路结构图如下：



SC92F730X 复位电路图

7.3.3 上电复位 POR

SC92F730X 内部有上电复位电路，当电源电压 V_{DD} 达到 POR 复位电压时，系统自动复位。

7.3.4 看门狗复位 WDT

SC92F730X 有一个 WDT，其时钟源为内部的 128kHz 振荡器。用户可以通过编程器的 Code Option 选择是否开启看门狗复位功能。

OP_CTM0 (C1H@FFH) Customer Option 寄存器 0(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ENWDT	-	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
读/写	读/写	-	读/写		读/写	读/写	读/写	
上电初始值	n	x	n		n	n	n	

位编号	位符号	说明
7	**ENWDT**	WDT 开关(此位由系统将用户 Code Option 所设的值调入) 1: WDT 开始工作 0: WDT 关闭

WDTCON (CFH) WDT 控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	CLRWDT	-	WDTCKS[2:0]		
读/写	-	-	-	读/写	-	读/写		
上电初始值	x	x	x	0	x	0	0	0

位编号	位符号	说明
4	**CLRWDT**	WDT 清“0”位(写 1 有效) 1 ： WDT 计数器从 0 开始计数 此位由系统硬件自动置 0
2~0	**WDTCKS [2:0]**	看门狗时钟选择

WDTCKS[2:0]	WDT 溢出时间
000	500ms
001	250ms
010	125ms
011	62.5ms
100	31.5ms
101	15.75ms
110	7.88ms
111	3.94ms

7~5,3	**-**	保留

7.3.5 复位初始状态

当 SC92F730X 处于复位状态时，多数寄存器会回到其初始状态。看门狗 WDT 处于关闭的状态。程序计数器 PC 初始值为 0000h，堆栈指针 SP 初始值为 07h。“热启动”的 Reset（如 WDT、LVR 等）不会影响到 SRAM，SRAM 值始终是复位前的值。SRAM 内容的丢失会发生在电源电压低到 RAM 无法保存为止。

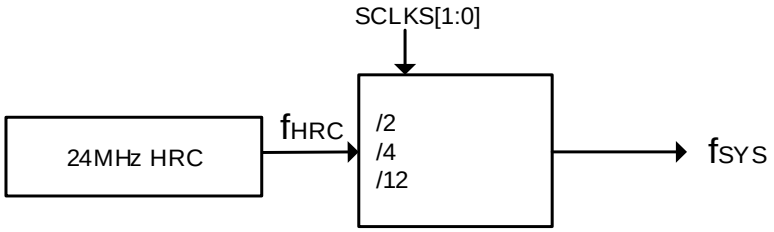
SFR 寄存器的上电复位初始值如下表：

SFR 名称	初始值	SFR 名称	初始值
ACC	00000000b	P0PH	xxxx0000b
B	00000000b	P0VO	xxxx0000b
PSW	00000000b	P1	xxxx0000b
SP	00000111b	P1CON	xxxx0000b
DPL	00000000b	P1PH	xxxx0000b
DPH	00000000b	P2	00xxxx00b
PCON	0xxxxx00b	P2CON	00xxxx00b
ADCCFG0	00xxxx00b	P2PH	00xxxx00b
ADCCON	00000000b	PWMCFG0	xx00000xb
ADCVH	00000000b	PWMCFG1	xx00xxxxb
ADCVL	0000xxxxb	PWMCON	00000000b
BTMCON	00xx0000b	PWMDTY0	00000000b
IAPADE	00000000b	PWMDTY1	00000000b

IAPADH	xxx00000b	PWMDTY2	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	xx00xxxxb	TCON	0000xxxxb
INT0F	xxxx0000b	TMCON	xxxxx000b
INT2F	xx00xxxxb	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
P0	xxxx0000b	TL2	00000000b
P0CON	xxxx0000b	WDTCON	xxx0x000b

7.4 高频系统时钟电路

SC92F730X 内建了一个振荡频率可调的高精度 HRC，HRC 出厂时被精确地调校至 24MHz@5V/25℃，用户可以通过编程器的 Code Option 将系统时钟设置为 12/6/2MHz 使用。调校过程是过滤掉制程上的偏差对精度所造成的影响。此 HRC 受工作的环境温度和工作电压影响会有一定的漂移，对于压漂（4.0V~5.5V）以及(-20℃~85℃)的温漂会在 ±1% 以内。



SC92F730X 内部时钟关系

OP_CTM0 (C1H@FFH) Customer Option 寄存器 0(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ENWDT	-	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
读/写	读/写	-	读/写		读/写	读/写	读/写	
上电初始值	n	x	n		n	n	n	

位编号	位符号	说明
5~4	**SCLKS[1:0]**	系统时钟频率选择： 00：保留； 01：系统时钟频率为高频振荡器频率除以 2；

		10：系统时钟频率为高频振荡器频率除以 4； 11：系统时钟频率为高频振荡器频率除以 12。

SC92F730X 有一个特殊的功能：用户可修改 SFR 的值实现 HRC 频率在一定范围的调整。用户可以通过配置 OP_HRCR 寄存器实现，该寄存器的配置方法可参考章节：5.2.1 Option 相关 SFR 操作说明。

OP_HRCR (83H@FFH) 系统时钟改变寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	OP_HRCR[7:0]							
读/写	读/写							
上电初始值	n	n	n	n	n	n	n	n

位编号	位符号	说明
7~0	**OP_HRCR[7:0]**	**HRC 频率改变寄存器** 用户可通过修改此寄存器的值实现高频振荡器频率 f_{HRC} 的改变，进而改变 IC 的系统时钟频率 f_{SYS}： 1. OP_HRCR[7:0]上电后的初始值 OP_HRCR[s]是一个固定值，以确保 f_{HRC} 为 24MHz，每颗 IC 的 OP_HRCR[s]都可能会有差异 2. 初始值为 OP_HRCR [s] 时 IC 的系统时钟频率 f_{SYS} 可通过 Option 项设置为准确的 12/6/2MHz，OP_HRCR [7:0]每改变 1 则 f_{SYS} 频率改变约 0.23% OP_HRCR [7:0]和 f_{SYS} 输出频率的关系如下：

OP_HRCR [7:0]值	f_{SYS} 实际输出频率(12M 为例)
OP_HRCR [s]-n	12000*(1-0.23%*n)kHz
...	....
OP_HRCR [s]-2	12000*(1-0.23%*2) = 11944.8kHz
OP_HRCR [s]-1	12000*(1-0.23%*1) = 11972.4kHz
OP_HRCR [s]	12000kHz
OP_HRCR [s]+1	12000*(1+0.23%*1) = 12027.6kHz
OP_HRCR [s]+2	12000*(1+0.23%*2) = 12055.2kHz
...	...
OP_HRCR [s]+n	12000*(1+0.23%*n)kHz

说明：
1. IC 每次上电后 OP_HRCR[7:0]的值都是高频振荡器频率 f_{HRC} 最接近 24MHz 的值；用户可借助 EEPROM 在每次上电后修正 HRC 的值以让 IC 的系统时钟频率 f_{SYS} 工作在用户需要的频率；
2. 为保证 IC 工作可靠，IC 最高工作频率请勿超过 12MHz 的 10%，即 13.2MHz；
3. 请用户确认 HRC 频率的改变不会影响其它功能。

7.5 低频振荡器及低频时钟定时器

SC92F730X 内建一个频率为 128kHz 的 RC 振荡电路，作为低频时钟定时器 Base Timer 和 WDT 的时钟源。开启 Base Timer 或使能 WDT 均可启动 128kHz 低频振荡器。

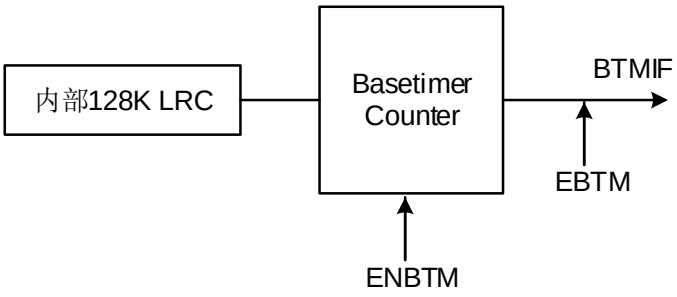
低频时钟定时器 Base Timer 可以把 CPU 从 STOP mode 唤醒，并且产生中断。

BTMCON (CEH) 低频定时器控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ENBTM	BTMIF	-	-	BTMFS[3:0]			
读/写	读/写	读/写	-	-	读/写			
上电初始值	0	0	x	x	0	0	0	0

位编号	位符号	说明
7	**ENBTM**	低频 Base Timer 启动控制 0： Base Timer 不启动 1： Base Timer 启动
6	**BTMIF**	Base Timer 中断申请标志 当 CPU 接受 Base Timer 的中断后, 此标志位会被硬件自动清除。
3~0	**BTMFS [3:0]**	低频时钟中断频率选择 0000：每 15.625ms 产生一个中断 0001：每 31.25ms 产生一个中断 0010：每 62.5ms 产生一个中断 0011：每 125ms 产生一个中断 0100：每 0.25 秒产生一个 中断 0101：每 0.5 秒产生一个 中断 0110：每 1.0 秒产生一个 中断 0111：每 2.0 秒产生一个 中断 1000：每 4.0 秒产生一个 中断 其它：保留
5~4	**-**	保留

Base Timer 结构图如下：



Base Timer 结构图

7.6 STOP 模式和 IDLE 模式

SC92F730X 提供了一个特殊功能寄存器 PCON。配置该寄存器的 bit0 和 bit1 可控制 MCU 进入不同的工作模式。

对 PCON.1 写入 1，内部的高频系统时钟就会停止，进到 STOP 模式，达到省电功能。在 STOP 模式下，用户可以通过外部中断 INT0、INT2 和低频时钟中断把 SC92F730X 唤醒，也可以通过外部复位将 STOP 唤醒。

对 PCON.0 写入 1，程序停止运行，进入 IDLE 模式，但外部设备及时钟继续运行，进入 IDLE 模式前所有 CPU 状态都被保存。IDLE 模式可由任何中断唤醒。

PCON (87H) 电源管理控制寄存器(只写、*不可读 *)

位编号	7	6	5	4	3	2	1	0
符号	SMOD	-	-	-	-	-	STOP	IDL
读/写	只写	-	-	-	-	-	只写	只写
上电初始值	0	x	x	x	x	x	0	0

位编号	位符号	说明
1	**STOP**	STOP 模式控制 0： 正常操作模式 1： 节能模式， 高频振荡器停止工作，低频振荡器及 WDT 可根据设定选择工作与否。
0	**IDL**	IDLE 模式控制 0： 正常操作模式 1： 节能模式， 程序停止运行，但外部设备及时钟继续运行，进入 IDLE 模式前所有 CPU 状态都被保存。

注意：

配置 MCU 进入 STOP 或 IDLE 模式时，对 PCON 寄存器进行配置操作的语句后面要加上 8 个 NOP 指令，不能直接跟其它指令，否则在唤醒后无法正常执行后续的指令！

例如：设置 MCU 进入 STOP 模式：

C 语言例程：

```
#include”intrins.h”

PCON |= 0x02;      //PCON 的 bit1 STOP 位写 1，配置 MCU 进入 STOP 模式
_nop_();           //至少需要 8 个_nop_()
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
……
```

汇编例程：

```
ORL PCON,#02H      ; PCON 的 bit1 STOP 位写 1，配置 MCU 进入 STOP 模式
NOP                ; 至少需要 8 个 NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
……
```

8 中央处理单元 CPU 及指令系统

8.1 CPU

SC92F730X 所用的 CPU 是一个 高速的 1T 标准 8051 内核，其指令完全兼容传统 8051 内核单片机。

8.2 寻址方式

SC92F730X 的 1T 8051 CPU 指令的寻址方式有：①立即寻址②直接寻址③间接寻址④寄存器寻址⑤相对寻址⑥变址寻址⑦位寻址

8.2.1 立即寻址

立即寻址也称为立即数，它是在指令操作数中直接给出参加运算的操作数，指令举例如下：

MOV A, #50H （这条指令是将立即数 50H 送到累加器 A 中）

8.2.2 直接寻址

在直接寻址方式中，指令操作数域给出的是参加运算操作数的地址。直接寻址方式只能用来表示特殊功能寄存器、内部数据寄存器和位地址空间。其中特殊功能寄存器和位地址空间只能用直接寻址方式访问。举例如下：

ANL 50H, #91H （表示 50H 单元中的数与立即数 91H 相“与”，结果存放在 50H 单元中。其中 50H 为直接地址，表示内部数据寄存器 RAM 中的一个单元。）

8.2.3 间接寻址

间接寻址采用 R0 或 R1 前添加“@”符号来表示。假设 R1 中的数据是 40H，内部数据存储器 40H 单元的数据为 55H,则指令为

MOV A, @R1 （ 把数据 55H 传送至累加器 A）。

8.2.4 寄存器寻址

寄存器寻址时对选定的工作寄存器 R7~R0、累加器 A、通用寄存器 B、地址寄存器和进位 C 中的数进行操作。其中寄存器 R7~R0 由指令码的低 3 位表示，ACC、B、DPTR 及进位位 C 隐含在指令码中。因此，寄存器寻址也包含一种隐含寻址方式。寄存器工作区的选择由程序状态字寄存器 PSW 中的 RS1、RS0 来决定。指令操作数指定的寄存器均指当前工作区的寄存器。

INC R0 是指（R0）+1→R0

8.2.5 相对寻址

相对寻址是将程序计数器 PC 中的当前值与指令第二字节给出的数相加，其结果作为转移指令的转移地址。转移地址也成为转移目的地址，PC 中的当前值成为基地址，指令第二字节给出的数成为偏移量。由于目的地址是相对于 PC 中的基地址而言，所以这种寻址方式成为相对寻址。偏移量为带符号的数，所能表示的范围为+127~-128.这种寻址方式主要用于转移指令。

JC ＄+50H

表示若进位位 C 为 0，则程序计数器 PC 中的内容不改变，即不转移。若进位位 C 为 1，则以 PC 中的当前值及基地址，加上偏移量 50H 后所得到的结果作为该转移指令的目的地址。

8.2.6 变址寻址

在变址寻址方式中，指令操作数制定一个存放变址基址的变址寄存器。变址寻址时，偏移量与变址基值相加，其结果作为操作数的地址。变址寄存器有程序计数器 PC 和地址寄存器 DPTR。

MOVC A, @A+DPTR

表示累加器 A 为偏移量寄存器，其内容与地址寄存器 DPTR 中的内容相加，其结果作为操作数的地址，取出该单元中的数送入累加器 A 中。

8.2.7 位寻址

位寻址是指对一些可进行位操作的内部数据存储器 RAM 和特殊功能寄存器进行位操作时的寻址方式。在进行位操作时，借助于进位位 C 作为位操作累加器，指令操作数直接给出该位的地址，然后根据操作码的性质对该位进行位操作。位地址与字节直接寻址中的字节地址编码方式完全一样，主要由操作指令的性质加以区分，使用时

应特别注意。

MOV C, 20H (将地址为 20H 的位操作寄存器值送入进位位 C 中。)

9 INTERRUPT 中断

SC92F730X 单片机提供 9 个中断源：Timer0，Timer1，Timer2，INT0，INT2，ADC，PWM，UART，Base Timer。这 9 个中断源分为 2 个中断优先级，并可以单独分别设置为高优先级或者低优先级。两个外部中断可以分别设定其中每个中断源的触发条件为上升、下降或上下沿，每个中断分别有独立的优先级设置位、中断标志、中断向量和使能位，总的使能位 EA 可以实现所有中断的打开或者关闭。

9.1 中断源、向量

SC92F730X 的中断源、中断向量、及相关控制位列表如下：

中断源	中断发生时间	中断标志	中断使能控制	中断优先权控制	中断向量	查询优先级	中断号（C51）	标志清除方式	能否唤醒 STOP
INT0	外部中断 0 条件符合	-	EINT0	IPINT0	0003H	1 (高)	0	-	能
Timer0	Timer0 溢出	TF0	ET0	IPT0	000BH	2	1	H/W Auto	不能
Timer1	Timer1 溢出	TF1	ET1	IPT1	001BH	3	3	H/W Auto	不能
UART	接收或发送完成	RI/TI	EUART	IPUART	0023H	4	4	必须用户清除	不能
Timer2	Timer2 溢出	TF2	ET2	IPT2	002BH	5	5	必须用户清除	不能
ADC	ADC 转换完成	ADCIF	EADC	IPADC	0033H	6	6	必须用户清除	不能
PWM	PWM 溢出	PWMIF	EPWM	IPPWM	0043H	7	8	H/W Auto	不能
BTM	Base timer 溢出	BTMIF	EBTM	IPBTM	004BH	8	9	H/W Auto	能
INT2	外部中断 2 条件符合	-	EINT2	IPINT2	0053H	9	10	-	能

在 EA=1 及各中断使能控制为 1 的情况下，各中断发生情况如下：

定时器中断：Timer0 和 Timer1 溢出时会产生中断并将中断标志 TF0 和 TF1 置为“1”，当单片机执行该定时器中断时，中断标志 TF0 和 TF1 会被硬件自动清“0”。Timer2 溢出时会产生中断并将中断标志 TF2 置为“1”，在 Timer2 中断发生后，硬件并不会自动清除 TF2 位，此 bit 必须由使用者的软件负责清除。

UART 中断：当 UART 接收或发送一帧数据完成时 RI 或 TI 位会被硬件自动置“1”，UART 中断产生。在 UART 中断发生后，硬件并不会自动清除 RI/TI 位，此 bit 必须由使用者的软件负责清除。

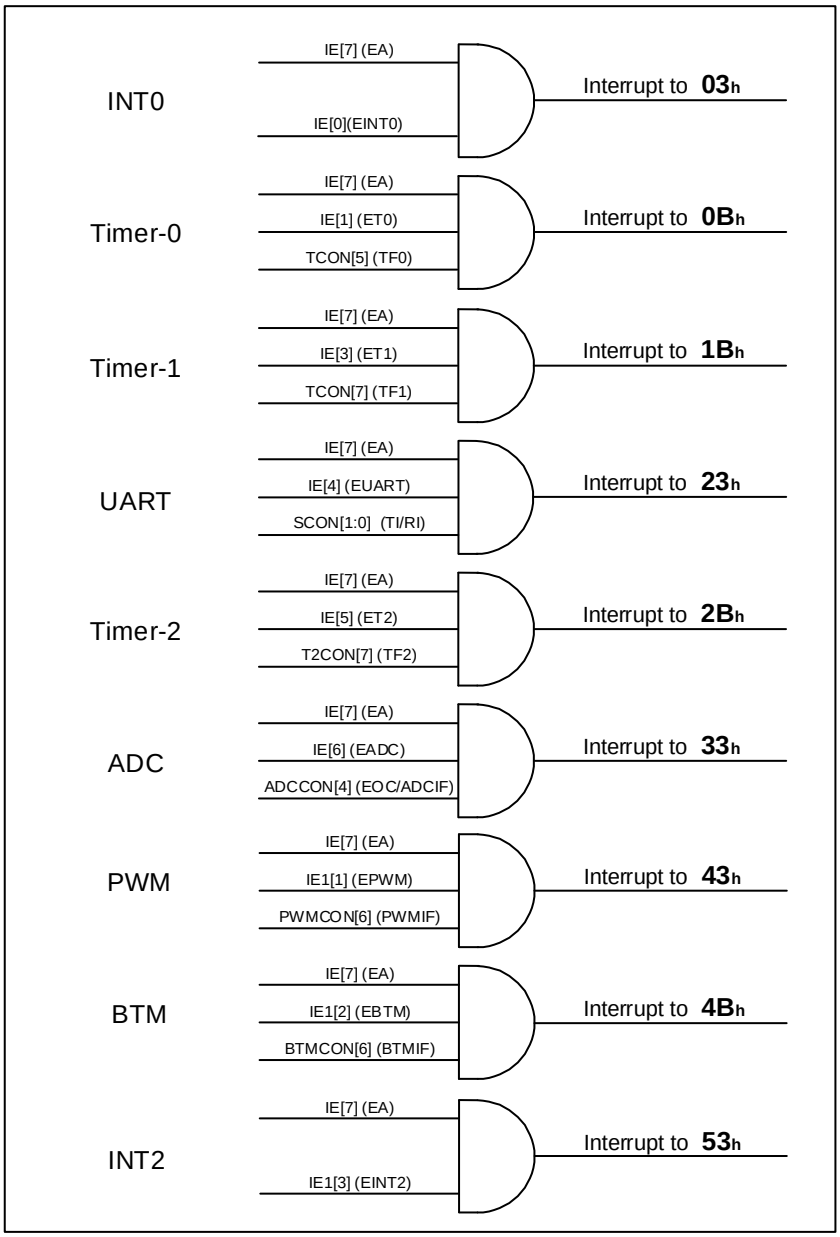
ADC 中断：ADC 中断的发生时间为 ADC 转换完成时，其中断标志就是 ADC 转换结束标志 EOC/ADCIF（ADCCON.4）。当使用者设定 ADCS 开始转换后，EOC 会被硬件自动清除为 “0” ；当转换完成后，EOC 会被硬件自动置为 “1” 。使用者在 ADC 中断发生之后，进入中断服务程序时，必须用软件去清除它。

PWM 中断：当 PWM 计数器溢出时(也就是说: 计数器数到超过 PWMPRD 时)，PWMIF 位(PWM Interrupt Flag)会被硬件自动置“1”，PWM 中断产生。当单片机执行该 PWM 中断时，中断标志 PWMIF 会被硬件自动清“0”。

外部中断 INT0、INT2：当外部中断口有中断条件发生时，外部中断就发生了。INT0 有四个外部中断源，INT2 有两个外部中断源，用户可以根据需要设成上沿、下沿或者双沿中断，可通过设置 SFR （INTxF 和 INTxR）来实现。用户可通过 IP 寄存器来设置每个中断的优先级级别。外部中断 INT0、INT2 还可以唤醒单片机的 STOP。

9.2 中断结构图

SC92F730X 的中断结构如下图所示：



SC92F730X 中断结构和向量

9.3 中断优先级

SC92F730X 单片机的中断具有两个中断优先级，这些中断源的请求可编程为高优先级中断或者低优先级中断，即可实现两级中断服务程序的嵌套。一个正在执行的低优先级中断能被高优先级中断请求所中断，但不能被另一个同一优先级的中断请求所中断，一直执行到结束，遇到返回指令 RETI，返回主程序后再执行一条指令才能响应新的中断请求。

也就是说：

① 低优先级中断可被高优先级中断请求所中断，反之不能；

② 任何一种中断，在响应过程中，不能被同一优先级的中断请求所中断。

中断查询顺序：SC92F730X 单片机的同一优先级中断，如果同时来几个中断，则中断响应的优先顺序同 C51 中的中断查询号相同，即查询号小的会优先响应，查询号大的会慢响应。

9.4 中断处理流程

当一个中断产生并且被 CPU 响应，则主程序运行被中断，将执行下述操作

① 当前正在执行的指令执行完；

② PC 值被压入堆栈，保护现场；

③ 中断向量地址载入程序计数器 PC；

④ 执行相应的中断服务程序；

⑤ 中断服务程序结束并 RETI；

⑥ 将 PC 值退栈，并返回执行中断前的程序。

在此过程中，系统不会立即执行其它同一优先级的中断，但会保留所发生的中断请求，在当前中断处理结束后，转去执行新的中断请求。

9.5 中断相关 SFR 寄存器

IE (A8H) 中断使能寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	EA	EADC	ET2	EUART	ET1	-	ET0	EINT0
读/写	读/写	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	0	0	0	0	0	x	0	0

位编号	位符号	说明
7	**EA**	中断使能的总控制 0: 关闭所有的中断 1: 打开所有的中断
6	**EADC**	ADC 中断使能控制 0：关闭 ADC 中断 1：允许 ADC 转换完成时产生中断
5	**ET2**	Timer2 中断使能控制 0：关闭 TIMER2 中断 1：允许 TIMER2 中断
4	**EUART**	UART 中断使能控制 0：关闭 UART 中断 1：允许 UART 中断
3	**ET1**	Timer1 中断使能控制 0：关闭 TIMER1 中断 1：允许 TIMER1 中断
1	**ET0**	Timer0 中断使能控制 0：关闭 TIMER0 中断 1：允许 TIMER0 中断

0	**EINT0**	外部中断 0 使能控制 0：关闭 INT0 中断 1：打开 INT0 中断
2	**-**	保留

IP (B8H) 中断优先级控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	IPADC	IPT2	IPUART	IPT1	-	IPT0	IPINT0
读/写	-	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	x	0	0	0	0	x	0	0

位编号	位符号	说明
6	**IPADC**	ADC 中断优先权选择 0：ADC 中断优先权为低 1：ADC 中断优先权为高
5	**IPT2**	Timer2 中断优先权选择 0：Timer2 中断优先权为低 1：Timer2 中断优先权为高
4	**IPUART**	UART 中断优先权选择 0：UART 中断优先权为低 1：UART 中断优先权为高
3	**IPT1**	Timer1 中断优先权选择 0：Timer1 中断优先权为低 1：Timer1 中断优先权为高
1	**IPT0**	Timer0 中断优先权选择 0：Timer0 中断优先权为低 1：Timer0 中断优先权为高
0	**IPINT0**	INT0 计数器中断优先权选择 0：INT0 中断优先权为低 1：INT0 中断优先权为高
7,2	**-**	保留

IE1 (A9H) 中断使能寄存器 1(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	EINT2	EBTM	EPWM	-
读/写	-	-	-	-	读/写	读/写	读/写	-
上电初始值	x	x	x	x	0	0	0	x

位编号	位符号	说明
3	**EINT2**	外部中断 2 使能控制 0：关闭 INT2 中断 1：打开 INT2 中断
2	**EBTM**	Base Timer 中断使能控制 0： 关闭 Base Timer 中断 1： 允许 Base Timer 中断
1	**EPWM**	PWM 中断使能控制 0：关闭 PWM 中断 1：允许 PWM 计数溢出（数到 PWMPRD）时产生中断
7~4,0	**-**	保留

IP1 (B9H) 中断优先级控制寄存器 1(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	IPINT2	IPBTM	IPPWM	-
读/写	-	-	-	-	读/写	读/写	读/写	-
上电初始值	x	x	x	x	0	0	0	x

位编号	位符号	说明
3	**IPINT2**	INT2 计数器中断优先权选择 0：INT2 中断优先权为低 1：INT2 中断优先权为高
2	**IPBTM**	Base Timer 中断优先权选择 0：Base Timer 中断优先权为低 1：Base Timer 中断优先权为高
1	**IPPWM**	PWM 中断使能选择 0： PWM 中断优先权为低 1： PWM 中断优先权为高
7~4,0	**-**	保留

INT0F (BAH) INT0 下降沿中断控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	INT0F3	INT0F2	INT0F1	INT0F0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

位编号	位符号	说明
3~1	**INT0Fn (n=0~3)**	INT0 下降沿中断控制 0 : INT0n 下降沿中断关闭 1: INT0n 下降沿中断使能
7~4	**-**	保留

INT0R (BBH) INT0 上升沿中断控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	INT0R3	INT0R2	INT0R1	INT0R0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

位编号	位符号	说明
3~1	**INT0Rn (n=0~3)**	INT0 上升沿中断控制 0 : INT0n 上升沿中断关闭 1: INT0n 上升沿中断使能
7~4	**-**	保留

INT2F (C6H) INT2 下降沿中断控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	INT2F5	INT2F4	-	-	-	-
读/写	-	-	读/写	读/写	-	-	-	-
上电初始值	x	x	0	0	x	x	x	x

位编号	位符号	说明
5~4	**INT2Fn (n=4~5)**	INT2 下降沿中断控制 0 : INT2n 下降沿中断关闭

		1: INT2n 下降沿中断使能
7~6,3~0	-	保留

INT2R (C7H) INT2 上升沿中断控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	INT2R5	INT2R4	-	-	-	-
读/写	-	-	读/写	读/写	-	-	-	-
上电初始值	x	x	0	0	x	x	x	x

位编号	位符号	说明
5~4	**INT2Rn (n=4~5)**	INT2 上升沿中断控制 0 : INT2n 上升沿中断关闭 1: INT2n 上升沿中断使能
7~6,3~0	-	保留

10 定时器 TIMER0 、TIMER1

T0 和 T1 是 SC92F730X 单片机内部的两个 16 位定时器/计数器，它们具有计数方式和定时方式两种工作模式。特殊功能寄存器 TMOD 中有一个控制位 C/Tx 来选择 T0 和 T1 是定时器还是计数器。它们本质上都是一个加法计数器，只是计数的来源不同。定时器的来源为系统时钟或者其分频时钟，但计数器的来源为外部管脚的输入脉冲。只有在 TRx=1 的时候，T0 和 T1 才会被打开计数。

计数器模式下，P1.2/T0 和 P1.3/T1 管脚上的每一个脉冲，T0 和 T1 的计数值分别增加 1。

定时器模式下，可通过特殊功能寄存器 TMCON 来选择 T0 和 T1 的计数来源是 $f_{SYS}/12$ 或 f_{SYS} (f_{SYS} 为系统时钟)。

定时器/计数器 T0 有 4 种工作模式，定时器/计数器 T1 有 3 种工作模式（模式三不存在）：

① 模式 0：13 位定时器/计数器模式
② 模式 1：16 位定时器/计数器模式
③ 模式 2： 8 位自动重载模式
④ 模式 3：两个 8 位定时器/计数器模式

在上述模式中，T0 和 T1 的模式 0、1、2 都相同，模式 3 不同。

10.1 T0 和 T1 相关特殊功能寄存器

符号	地址	说明	7	6	5	4	3	2	1	0	Reset 值
TCON	88H	定时器控制寄存器	TF1	TR1	TF0	TR0	-	-	-	-	0000xxxxb
TMOD	89H	定时器工作模式寄存器	-	C/T1	M11	M01	-	C/T0	M10	M00	x000x000b
TL0	8AH	定时器 0 低 8 位	TL0[7:0]								00000000b
TL1	8BH	定时器 1 低 8 位	TL1[7:0]								00000000b
TH0	8CH	定时器 0 高 8 位	TH0[7:0]								00000000b
TH1	8DH	定时器 1 高 8 位	TH1[7:0]								00000000b
TMCON	8EH	定时器频率控制寄存器	-	-	-	-	-	T2FD	T1FD	T0FD	xxxxx000b

各寄存器的解释说明如下：

TCON (88H) 定时器控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	TF1	TR1	TF0	TR0	-	-	-	-
读/写	读/写	读/写	读/写	读/写	-	-	-	-
上电初始值	0	0	0	0	x	x	x	x

位编号	位符号	说明
7	**TF1**	T1 溢出中断请求标志。T1 产生溢出，发生中断时，硬件将 TF1 置为“1”，申请中断，CPU 响应时，硬件清“0”。
6	**TR1**	定时器 T1 的运行控制位。此位由软件置 1 和清 0。当 TR1=1 时，允许 T1 开始计数。TR1=0 时禁止 T1 计数。
5	**TF0**	T0 溢出中断请求标志。T0 产生溢出，发生中断时，硬件将 TF0 置为“1”，申请中断，CPU 响应时，硬件清“0”。
4	**TR0**	定时器 T0 的运行控制位。此位由软件置位和清 0。当 TR0=1 时，允许 T0 开始计数。TR0=0 时禁止 T0 计数。
3~0	**-**	保留

TMOD (89H) 定时器工作模式寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	C/T1	M11	M01	-	C/T0	M10	M00
读/写	-	读/写	读/写	读/写	-	读/写	读/写	读/写
上电初始值	x	0	0	0	x	0	0	0
	T1				T0			

位编号	位符号	说明
6	**C/T1**	TMOD[6]控制定时器 1 0：定时器，T1 计数来源于 f_{SYS} 分频 1：计数器，T1 计数来源于外部管脚 T1/P1.3
5~4	**M11,M01**	定时器/计数器 1 模式选择 00： 13 位定时器/计数器，TL1 高 3 位无效 01： 16 位定时器/计数器，TL1 和 TH1 全 10： 8 位自动重载定时器，溢出时将 TH1 存放的值自动重装入 TL1 11： 定时器/计数器 1 无效(停止计数)
2	**C/T0**	TMOD[2]控制定时器 0 0：定时器，T0 计数来源于 f_{SYS} 分频 1：计数器，T0 计数来源于外部管脚 T0/P1.2
1~0	**M10,M00**	定时器/计数器 0 模式选择 00： 13 位定时器/计数器，TL0 高 3 位无效 01： 16 位定时器/计数器，TL0 和 TH0 全 10： 8 位自动重载定时器，溢出时将 TH0 存放的值自动重装入 TL0 11 ：定时器 0 此时作为双 8 位定时器/计数器。TL0 作为一个 8 位定时器/计数器，通过标准定时器 0 的控制位控制；TH0 仅作为一个 8 位定时器，由定时器 1 的控制位控制。
7,3	**-**	保留

TMOD 寄存器中 TMOD[0]~TMOD[2]是设置 T0 的工作模式；TMOD[4]~TMOD[6]是设置 T1 的工作模式。

定时器和计数器 Tx 功能由特殊功能寄存器 TMOD 的控制位 C/Tx 来选择，M0x 和 M1x 都是用来选择 Tx 的工作模式。TRx 作为 T0 和 T1 的开关控制，只有 TRx=1 时 T0 和 T1 才打开。

TMCON (8EH) 定时器频率控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	-	T2FD	T1FD	T0FD
读/写	-	-	-	-	-	读/写	读/写	读/写
上电初始值	x	x	x	x	x	0	0	0

位编号	位符号	说明
1	**T1FD**	T1 输入频率选择控制 0：T1 频率源自于 $f_{SYS}/12$ 1：T1 频率源自于 f_{SYS}
0	**T0FD**	T0 输入频率选择控制 0：T0 频率源自于 $f_{SYS}/12$ 1：T0 频率源自于 f_{SYS}

IE (A8H) 中断使能寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	EA	EADC	ET2	EUART	ET1	-	ET0	EINT0
读/写	读/写	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	0	0	0	0	0	x	0	0

位编号	位符号	说明
3	**ET1**	Timer1 中断使能控制 0： 关闭 TIMER1 中断 1： 允许 TIMER1 中断
1	**ET0**	Timer0 中断使能控制 0： 关闭 TIMER0 中断 1： 允许 TIMER0 中断

IP (B8H) 中断优先级控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	IPADC	IPT2	IPUART	IPT1	-	IPT0	IPINT0
读/写	-	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	x	0	0	0	0	x	0	0

位编号	位符号	说明
3	**IPT1**	Timer1 中断优先权 0：设定 Timer 1 的中断优先权是 “低” 1：设定 Timer 1 的中断优先权是 “高”
1	**IPT0**	Timer0 中断优先权 0：设定 Timer 0 的中断优先权是 “低” 1：设定 Timer 0 的中断优先权是 “高”

10.2 T0 工作模式

通过对寄存器 TMOD 中的 M10、M00(TMOD[1]、TMOD[0])的设置，定时器/计数器 0 可实现 4 种不同的工作模式 。

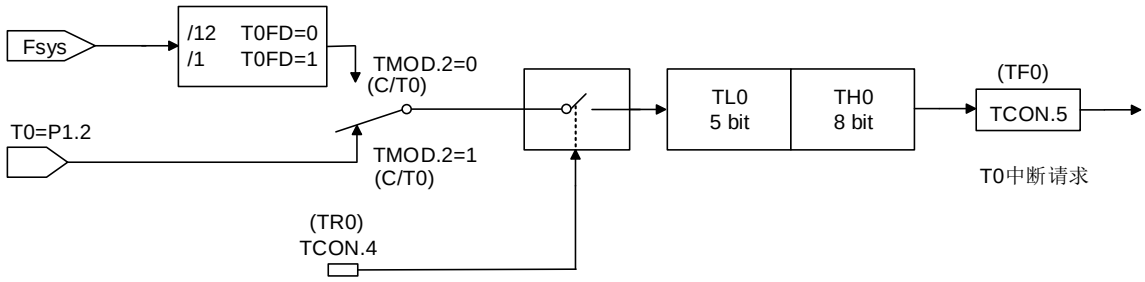
工作模式 0: 13 位计数器/定时器。

TH0 寄存器存放 13 位计数器/定时器的高 8 位(TH0.7~TH0.0), TL0 存放低 5 位(TL0.4~TL0.0)。TL0 的高三位(TL0.7~TL0.5)是不确定值，读取时应被忽略掉。当 13 位定时器/计数器递增溢出时，系统会将定时器溢出标志 TF0 置 1。如果定时器 0 中断被允许，将会产生一个中断。

C/T0 位选择计数器/定时器的时钟输入源。如果 C/T0=1，定时器 0 输入脚 T0(P1.2)的电平从高到低的变化，会使定时器 0 数据寄存器加 1。如果 C/T0=0，选择系统时钟的分频为定时器 0 的时钟源。

当 TR0 置 1 打开定时器 T0。TR0 置 1 并不强行复位定时器，意味着如果 TR0 置 1，定时器寄存器将从上次 TR0 清 0 时的值开始计数。所以，在允许定时器之前，应该设定定时器寄存器的初始值。

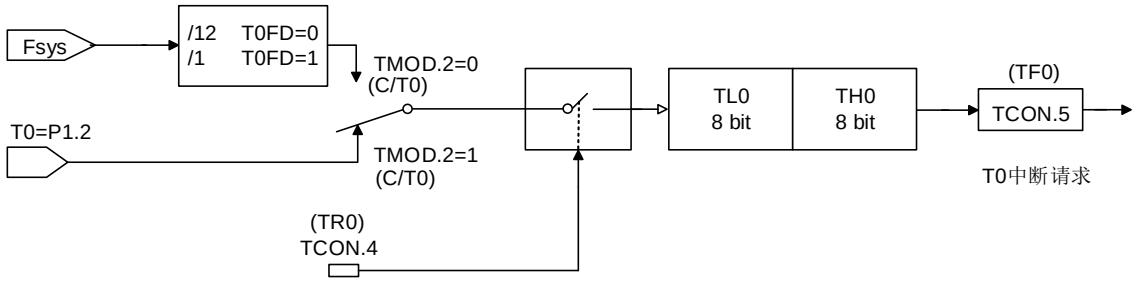
当作为定时器应用时，可配置 T0FD 来选择时钟源的分频比例。



定时器/计数器工作模式 0: 13 位定时器/计数器

工作模式 1: 16 位计数器/定时器

除了使用 16 位(TL0 的 8 位数据全部有效)计数器/定时器之外，模式 1 和模式 0 的运行方式相同。打开和配置计数器/定时器方式也相同。



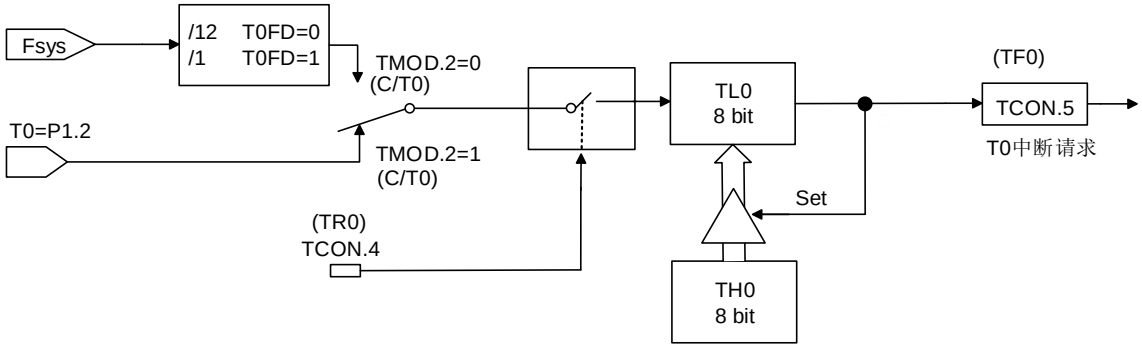
定时器/计数器工作模式 0: 16 位定时器/计数器

工作模式 2: 8 位自动重载计数器/定时器

在工作模式 2 中, 定时器 0 是 8 位自动重载计数器/定时器。TL0 存放计数值，TH0 存放重载值。当在 TL0 中的计数器溢出至 0x00 时，定时器溢出标志 TF0 被置 1，寄存器 TH0 的值被重载入寄存器 TL0 中。如果定时器中断使能，当 TF0 置 1 时将产生一个中断，但在 TH0 中的重载值不会改变。在允许定时器正确计数开始之前，TL0 必须初始化为所需要的值。

除了自动重载功能外，工作模式 2 中的计数器/定时器的使能和配置方式同模式 0 和 1 是相同的。

当作为定时器应用时，可配置寄存器 TMCON.0(T0FD)来选择定时器时钟源被系统时钟 f_{SYS} 分频的比例。



定时器/计数器工作模式 2: 自动重载的 8 位定时器/计数器

工作模式 3: 两个 8 位计数器/定时器(仅限于定时器 0)

在工作模式 3 中，定时器 0 用作两个独立的 8 位计数器/定时器，分别由 TL0 和 TH0 控制。TL0 通过定时器 0 的控制位(在 TCON 中)和状态位(在 TMOD 中)：TR0、C/T0、TF0 控制。定时器 0 可通过 T0 的 TMOD.2(C/T0)来选择是定时器模式还是计数器模式。

TH0 通过定时器 1 的控制 TCON 来设置相关的控制，但 TH0 仅被限定为定时器模式，无法通过 TMOD.2(C/T0)来设定为计数器模式。TH0 由定时器控制位 TR1 的控制使能，需设定 TR1=1。当发生溢出及产生中断时，TF1 会置 1，并按 T1 发生中断来进行相应的处理。

在 T0 被设为工作模式 3 时，TH0 定时器占用了 T1 的中断资源及 TCON 中寄存器，T1 的 16 位计数器会停止计数，相当于“TR1=0”。当采用 TH0 定时器工作时，需设置 TR1=1。

10.3 T1 工作模式

通过对寄存器 TMOD 中的 M11、M01(TMOD[5]、TMOD[4])的设置，定时器/计数器 1 可实现 3 种不同的工作模式 。

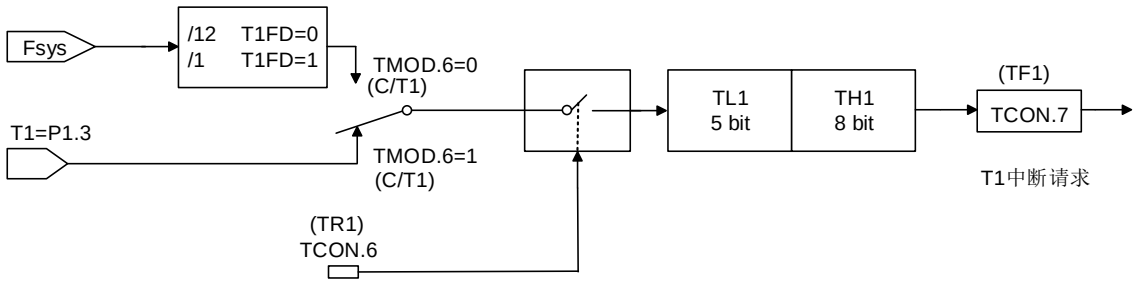
工作模式 0: 13 位计数器/定时器。

TH1 寄存器存放 13 位计数器/定时器的高 8 位(TH1.7~TH1.0)；TL1 存放低 5 位(TL1.4~TL1.0)。TL1 的高三位(TL1.7~TL1.5)是不确定值，读取时应被忽略掉。当 13 位定时器计数器递增溢出时，系统会将定时器溢出标志 TF1 置 1。如果定时器 1 中断被允许，将会产生一个中断。C/T1 位选择计数器/定时器的时钟源。

如果 C/T1=1，定时器 1 输入脚 T1(P1.3)的电平从高到低的变化，会使定时器 1 数据寄存器加 1。如果 C/T1=0，选择系统时钟的分频为定时器 1 的时钟源。

TR1 置 1 打开定时器。TR1 置 1 并不强行复位定时器，意味着如果 TR1 置 1，定时器寄存器将从上次 TR1 清 0 时的值开始计数。所以，在允许定时器之前，应该设定定时器寄存器的初始值。

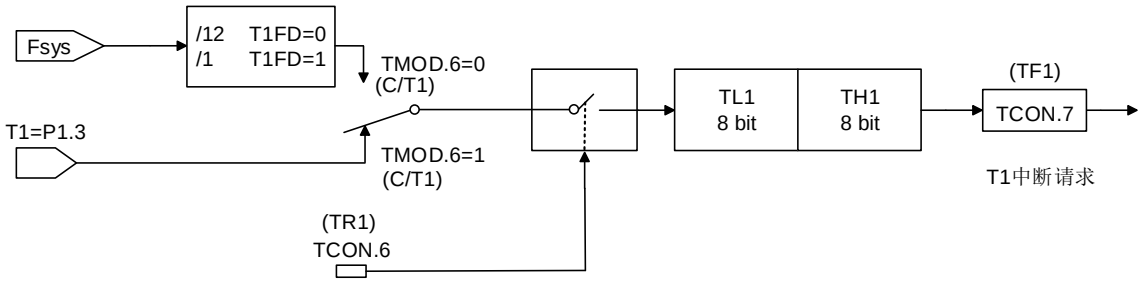
当作为定时器应用时，可配置 T1FD 来选择时钟源的分频比例。



定时器/计数器工作模式 0: 13 位定时器/计数器

工作模式 1:16 位计数器/定时器

除了使用 16 位(TL1 的 8 位数据全部有效)计数器/定时器之外，模式 1 和模式 0 的运行方式相同。打开和配置计数器/定时器方式也相同。



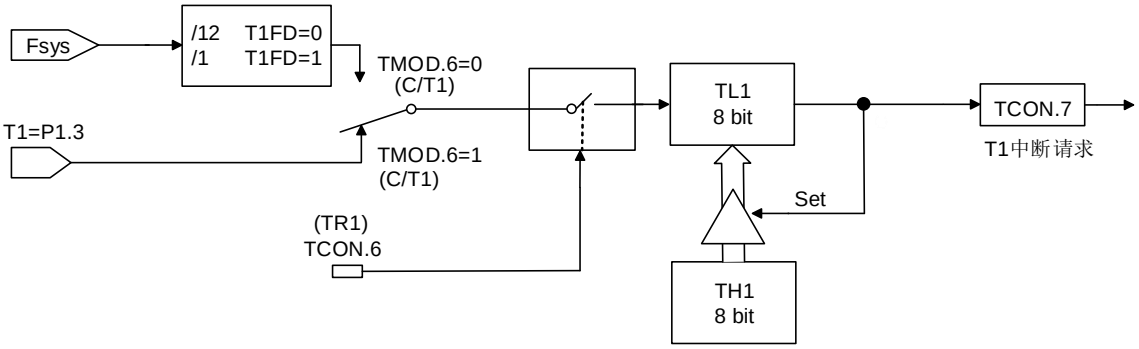
定时器/计数器工作模式 0: 16 位定时器/计数器

工作模式 2: 8 位自动重载计数器/计数器

在工作模式 2 中，定时器 1 是 8 位自动重载计数器/定时器。TL1 存放计数值，TH1 存放重载值。当在 TL1 中的计数器溢出至 0x00 时，定时器溢出标志 TF1 被置 1，寄存器 TH1 的值被重载入寄存器 TL1 中。如果定时器中断使能，当 TF1 置 1 时将产生一个中断，但在 TH1 中的重载值不会改变。在允许定时器正确计数开始之前，TL1 必须初始化为所需要的值。

除了自动重载功能外，工作模式 2 中的计数器/定时器的使能和配置方式同方式 0 和 1 是相同的。

当作为定时器应用时，可配置寄存器 TMCON.4(T1FD)来选择定时器时钟源被系统时钟 f_{SYS} 分频的比例。



定时器/计数器工作模式 2: 自动重载的 8 位定时器/计数器

11 定时器 TIMER2

SC92F730X 单片机内部的 Timer2 作为定时器本质上都是一个加法计数器，定时器的时钟来源为系统时钟或者其分频时钟。TR2 是 T2 计数的开关控制，只有在 TR2=1 的时候，T2 才会被打开计数。

定时器模式下，可通过特殊功能寄存器 TMCON 来选择 T2 的计数来源是 $f_{SYS}/12$ 或 f_{SYS}。

定时器/计数器 T2 有 2 种工作模式：

① 模式 1：16 位自动重载定时器模式

② 模式 2： 波特率发生器模式

11.1 T2 相关特殊功能寄存器

符号	地址	说明	7	6	5	4	3	2	1	0	Reset 值
T2CON	C8H	定时器 2 控制寄存器	TF2	-	RCLK	TCLK	-	TR2	-	-	0x00x0xxb
RCAP2L	CAH	定时器 2 重载低 8 位	RCAP2L[7:0]								00000000b
RCAP2H	CBH	定时器 2 重载高 8 位	RCAP2H[7:0]								00000000b
TL2	CCH	定时器 2 低 8 位	TL2[7:0]								00000000b
TH2	CDH	定时器 2 高 8 位	TH2[7:0]								00000000b
TMCON	8EH	定时器频率控制寄存器	-	-	-	-	-	T2FD	T1FD	T0FD	xxxxx000b

各寄存器的解释说明如下：

T2CON (C8H) 定时器 2 控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	TF2	-	RCLK	TCLK	-	TR2	-	-
读/写	读/写	-	读/写	读/写	-	读/写	-	-
上电初始值	0	x	0	0	x	0	x	x

位编号	位符号	说明
7	**TF2**	定时器 2 溢出标志位 0：无溢出(必须由软件清 0) 1：溢出(如果 RCLK = 0 和 TCLK = 0，由硬件设 1)
5	**RCLK**	UART 接收时钟控制位 0：定时器 1 产生接收波特率 1：定时器 2 产生接收波特率
4	**TCLK**	UART 发送时钟控制位 0：定时器 1 产生发送波特率 1：定时器 2 产生发送波特率

2	**TR2**	定时器 2 开始/停止控制位 0：停止定时器 2 1：开始定时器 2
6,3,1~0	**-**	**固定写 0**

TMCON (8EH) 定时器频率控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	-	T2FD	T1FD	T0FD
读/写	-	-	-	-	-	读/写	读/写	读/写
上电初始值	x	x	x	x	x	0	0	0

位编号	位符号	说明
2	**T2FD**	T2 输入频率选择控制 0：T2 频率源自于 $f_{SYS}/12$ 1：T2 频率源自于 f_{SYS}

IE (A8H) 中断使能寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	EA	EADC	ET2	EUART	ET1	-	ET0	EINT0
读/写	读/写	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	0	0	0	0	0	x	0	0

位编号	位符号	说明
5	**ET2**	Timer2 中断使能控制 0： 关闭 TIMER2 中断 1： 允许 TIMER2 中断

IP (B8H) 中断优先级控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	IPADC	IPT2	IPUART	IPT1	-	IPT0	IPINT0
读/写	-	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	x	0	0	0	0	x	0	0

位编号	位符号	说明
5	**IPT2**	Timer2 中断优先权 0：设定 Timer 2 的中断优先权是 “低” 1：设定 Timer 2 的中断优先权是 “高”

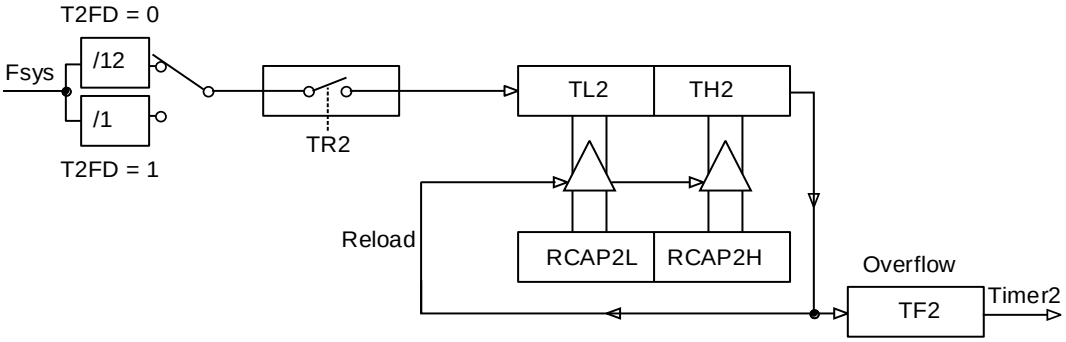
11.2 T2 工作模式

定时器 2 工作模式与配置方式如下表：

TR2	RCLK	TCLK	方式	
1	0	0	1	16 位自动重载定时器
1	1	X	2	波特率发生器
	X	1		
0	X	X	X	定时器 2 停止

工作模式 1: 16 位自动重载定时器

在 16 位自动重载方式下，定时器 2 递增到 0xFFFFH，在溢出后置起 TF2 位，同时定时器自动将用户软件写好的寄存器 RCAP2H 和 RCAP2L 的 16 位值装入 TH2 和 TL2 寄存器。



模式 1: 16 位自动重载 DCEN = 0

工作模式 2: 波特率发生器

通设置 T2CON 寄存器中的 TCLK 和/或 RCLK 选择定时器 2 作为波特率发生器。接收器和发送器的波特率可以不同。如果定时器 2 作为接收器或发送器，则定时器 1 相应的作为另一种的波特率发生器

设置 T2CON 寄存器中的 TCLK 和/或 RCLK 使定时器 2 进入波特率发生器方式，该方式与自动重载方式相似

定时器 2 的溢出会使 RCAP2H 和 RCAP2L 寄存器中的值重载入定时器 2 计数，但不会产生中断

在 UART 方式 1 和 3 中的波特率由定时器 2 的溢出率根据下列方程式决定：

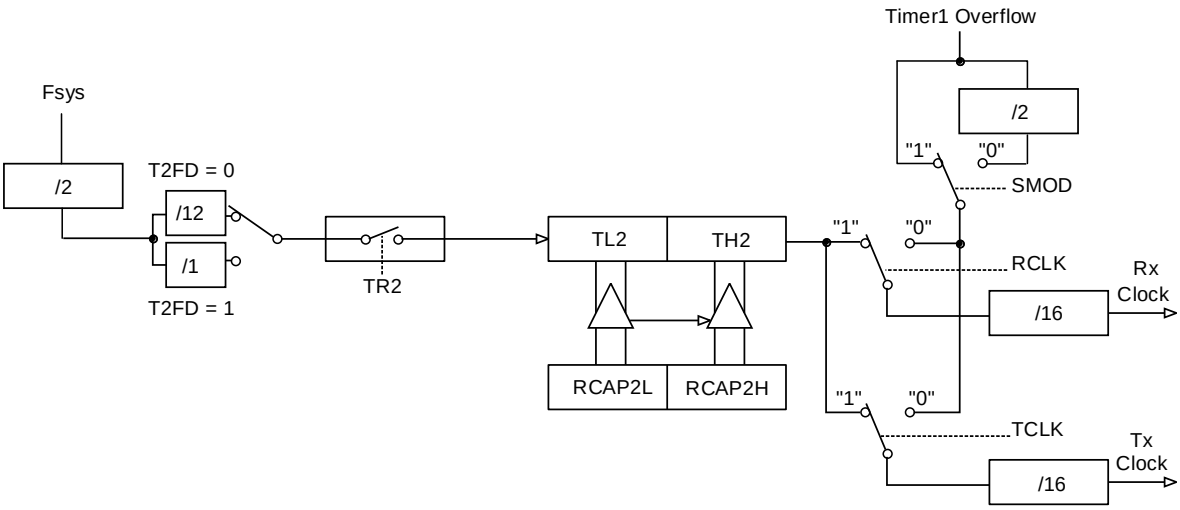
$$\text{BaudRate} = \frac{1}{16} \times \frac{\text{fn2}}{(65536 - [\text{RCAP2H}, \text{RCAP2L}]) \times 2}$$

其中， fn2 为定时器 2 时钟频率：

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

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

定时器 2 作为波特率发生器的原理图如下：



模式 2：波特率发生器

注意：

1. 当事件发生时或其它任何时间都能由软件设置 TF2 为 1，只有软件以及硬件复位才能使之清 0；
2. 当 EA = 1 且 ET2 = 1 时，设置 TF2 为 1 能引起定时器 2 中断；
3. 当定时器 2 作为波特率发生器时，写入 TH2/TL2 或 RCAP2H/RCAP2L 会影响波特率的准确性，引起通信出错。

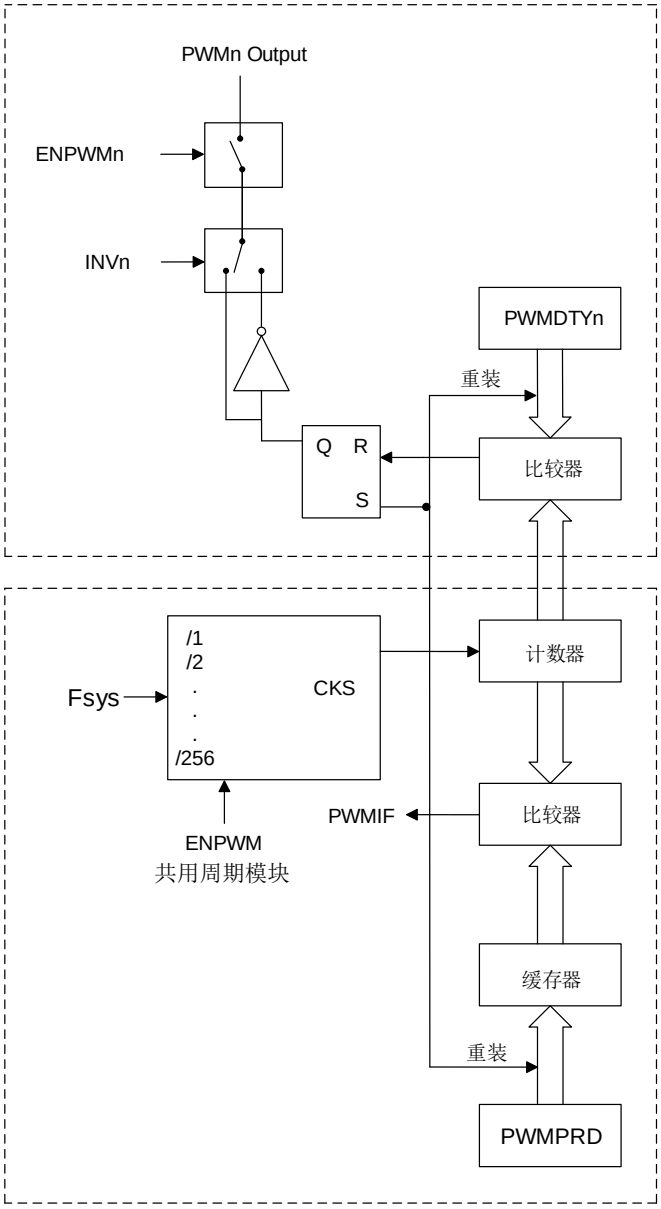
12 PWM

SC92F730X 提供了一个独立的计数器, 它可以支持 5 路的 PWM 输出：PWM0~2，PWM4~5。
SC92F730X 的 PWM 具有的功能为：

① 8 位 PWM 精度；
② PWM0~2，PWM4~5 周期相同，但占空比可单独设置；
③ 输出可设置正反向；
④ 提供 1 个 PWM 溢出的中断。

SC92F730X 的 PWM 可支持周期及占空比的调整，寄存器 PWMCON 控制 PWM0~2，PWM4~5 相关设置，PWMCFG0、PWMCFG1 设置 PWM 输出波形的极性及输出 IO 的选择，PWMPRD 设置 PWM 共同的周期，PWMDTY0~2，PWMDTY4~5 分别控制 PWM0~2，PWM4~5 的占空比。

12.1 PWM 结构框图



SC92F730X PWM 结构框图

12.2 PWM 相关 SFR 寄存器

符号	地址	说明	7	6	5	4	3	2	1	0	Reset 值
PWMCFG0	D1H	PWM 设置寄存器 0	-	-	INV2	INV1	INV0	ENPWM5	ENPWM4	-	xx00000xb
PWMCON	D2H	PWM 控制寄存器	ENPWM	PWMIF	ENPWM2	ENPWM1	ENPWM0	PWMCKS[2:0]			00000000b
PWMPRD	D3H	PWM 周期设置寄存器	PWMPRD[7:0]								00000000b
PWMCFG1	D4H	PWM 设置寄存器 1	-	-	INV5	INV4	-	-	-	-	xx00xxxxb
PWMDTY0	D5H	PWM0 占空比设置寄存器	PDT0[7:0]								00000000b
PWMDTY1	D6H	PWM1 占空比设置寄存器	PDT1[7:0]								00000000b
PWMDTY2	D7H	PWM2 占空比设置寄存器	PDT2[7:0]								00000000b
PWMDTY4	DEH	PWM4 占空比设置寄存器	PDT4[7:0]								00000000b
PWMDTY5	DFH	PWM5 占空比设置寄存器	PDT5[7:0]								00000000b
IE1	A9H	中断使能寄存器	-	-	-	-	EINT2	EBTM	EPWM	-	xxxx000xb
IP1	B9H	中断优先级控制寄存器 1	-	-	-	-	IPINT2	IPBTM	IPPWM	-	xxxx000xb

PWMCON (D2H)PWM 控制寄存器 (读/写)

位编号	7	6	5	4	3	2	1	0
符号	ENPWM	PWMIF	ENPWM2	ENPWM1	ENPWM0	PWMCKS[2:0]		
读/写	读/写	读/写	读/写	读/写	读/写	读/写		
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	**ENPWM**	PWM 模块开关控制(Enable PWM) 1：允许 Clock 进到 PWM 单元, 开始 PWM 的工作 0：PWM 单元停止工作，PWM 计数器清零。**PWMn 仍接到输出口，若要使用与 PWMn 输出口复用的其它功能，应将 ENPWMn 置 0**
6	**PWMIF**	PWM 中断请求标志位(PWM Interrupt Flag) 当 PWM 计数器溢出时(也就是说: 数到超过 PWMPRD 时)， 此位会被硬件自动设定成 1。如果此时 IE1[1] (EPWM) 也是被设定成 1，PWM 的中断产生。
5	**ENPWM2**	PWM2 功能开关 1：PWM2 输出到 IO 0：PWM2 不输出到 IO
4	**ENPWM1**	PWM1 功能开关 1：PWM1 输出到 IO 0：PWM1 不输出到 IO
3	**ENPWM0**	PWM0 功能开关 1：PWM0 输出到 IO 0：PWM0 不输出到 IO
2~0	**PWMCKS[2:0]**	PWM 时钟源选择(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] 是五路 PWM 共享的周期设置控制器。 每当 PWM 计数器数到 PWMPRD[7:0]预先设置的值时，下一个 PWM CLK 到来时该计数器会跳数到 00_h，也就是说 PWM0~2，PWM4~5 的周期都是 (PWMPRD[7:0] +

1)*PWM 时钟。

PWM 计数器的计数时间可由 PWMCKS[2:0]所控制，分别可以选择不同个数的系统时钟去计数一个单位(pre-scalar selector)，即选择 PWM 计数器时钟源被系统时钟 f_{SYS} 分频的分频比。PWM0~2，PWM4~5 还可以被 PWMCFG0、PWMCFG1 中的 INV0~2，INV4~5 来选择 PWM 输出是否反向。

PWMPRD (D3H) PWM 周期设置寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	**PWMPRD[7:0]**							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**PWMPRD[7:0]**	五路 PWM 共用的周期设置； 此数值代表 PWM0~2，PWM4~5 输出波形的 (**周期 – 1**); 也就是说 PWM 输出的周期值为(PWMPRD[7:0] + 1)* PWM 时钟；

PWMCFG0 (D1H) PWM 设置寄存器 0(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	INV2	INV1	INV0	ENPWM5	ENPWM4	-
读/写	-	-	读/写	读/写	读/写	读/写	读/写	-
上电初始值	x	x	0	0	0	0	0	x

位编号	位符号	说明
5	**INV2**	PWM2 输出反向控制 1 ： 把 PWM2 的输出反向 0 ： PWM2 的输出不反向
4	**INV1**	PWM1 输出反向控制 1 ： 把 PWM1 的输出反向 0 ： PWM1 的输出不反向
3	**INV0**	PWM0 输出反向控制 1 ： 把 PWM0 的输出反向 0 ： PWM0 的输出不反向
2	**ENPWM5**	PWM5 功能开关 1：PWM5 输出到 IO 0：PWM5 不输出到 IO
1	**ENPWM4**	PWM4 功能开关 1：PWM4 输出到 IO 0：PWM4 不输出到 IO
7~6,0	**-**	保留

PWMCFG1 (D4H) PWM 设置寄存器 1(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	INV5	INV4	-	-	-	-
读/写	-	-	读/写	读/写	-	-	-	-
上电初始值	x	x	0	0	x	x	x	x

位编号	位符号	说明
5	**INV5**	PWM5 输出反向控制 1 ： 把 PWM5 的输出反向 0 ： PWM5 的输出不反向

4	**INV4**	PWM4 输出反向控制 1：把 PWM4 的输出反向 0：PWM4 的输出不反向
7~6, 3~0	**-**	保留

PWMDTY0 (D5H) PWM0 占空比设置寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	**PDT0[7:0]**							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**PDT0 [7:0]**	PWM0 占空比长度设置； PWM0 的高电平宽度 是 (PDT0[7:0])个 PWM 时钟

PWMDTY1 (D6H) PWM1 占空比设置寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	**PDT1[7:0]**							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**PDT1[7:0]**	PWM1 占空比长度设置； PWM1 的高电平宽度 是 (PDT1[7:0])个 PWM 时钟

PWMDTY2 (D7H) PWM2 占空比设置寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	**PDT2[7:0]**							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**PDT2[7:0]**	PWM2 占空比长度设置； PWM2 的高电平宽度 是 (PDT2[7:0])个 PWM 时钟

PWMDTY4 (DEH) PWM4 占空比设置寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	**PDT4[7:0]**							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**PDT4[7:0]**	PWM4 占空比长度设置； PWM4 的高电平宽度 是 (PDT4[7:0])个 PWM 时钟

PWMDTY5 (D7H) PWM5 占空比设置寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	**PDT5[7:0]**							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**PDT5[7:0]**	PWM5 占空比长度设置； PWM5 的高电平宽度 是 (PDT5[7:0])个 PWM 时钟

IE1 (A9H) 中断使能寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	EINT2	EBTM	EPWM	-
读/写	-	-	-	-	读/写	读/写	读/写	-
上电初始值	x	x	x	x	0	0	0	x

位编号	位符号	说明
1	**EPWM**	PWM 中断使能控制 0：关闭 PWM 中断 1：允许 PWM 计数器溢出时产生中断

IP1 (B9H) 中断优先级控制寄存器 1(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	IPINT2	IPBTM	IPPWM	-
读/写	-	-	-	-	读/写	读/写	读/写	-
上电初始值	x	x	x	x	0	0	0	x

位编号	位符号	说明
1	**IPPWM**	PWM 中断优先权选择 0：设定 PWM 的中断优先级是 “低” 1：设定 PWM 的中断优先级是 “高”

注意事项：

1. ENPWM 位能控制 PWM 模块是否工作。
2. ENPWMn 位能选择 PWMn 口作为 GPIO 还是作为 PWMn 输出。
3. EPWM(IE1.1)位能控制 PWM 是否被允许产生中断。
4. 如果 ENPWM 置 1，PWM 模块被打开，但 ENPWMn=0，PWM 输出被关闭并作为 GPIO 口。此时 PWM 模块可以作为一个 8 位 Timer 使用，此时 EPWM(IE1.1)被置 1，PWM 仍然会产生中断。
5. 五个 PWM 共用周期，溢出时产生 PWM 中断是同一中断向量。

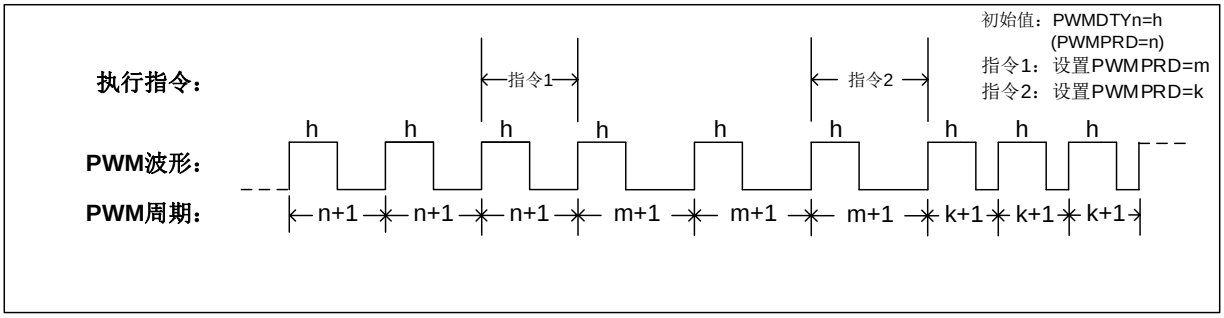
12.3 PWM 波形及用法

各 SFR 参数改变对 PWM 波形影响如下所述：

① 占空比变化特性

当 PWMn 输出波形时，若需改变占空比，可通过改变高电平设置寄存器(PWMDTYn)的值实现。但需要注意，更改 PWMDTYn 的值，占空比会即时生效。

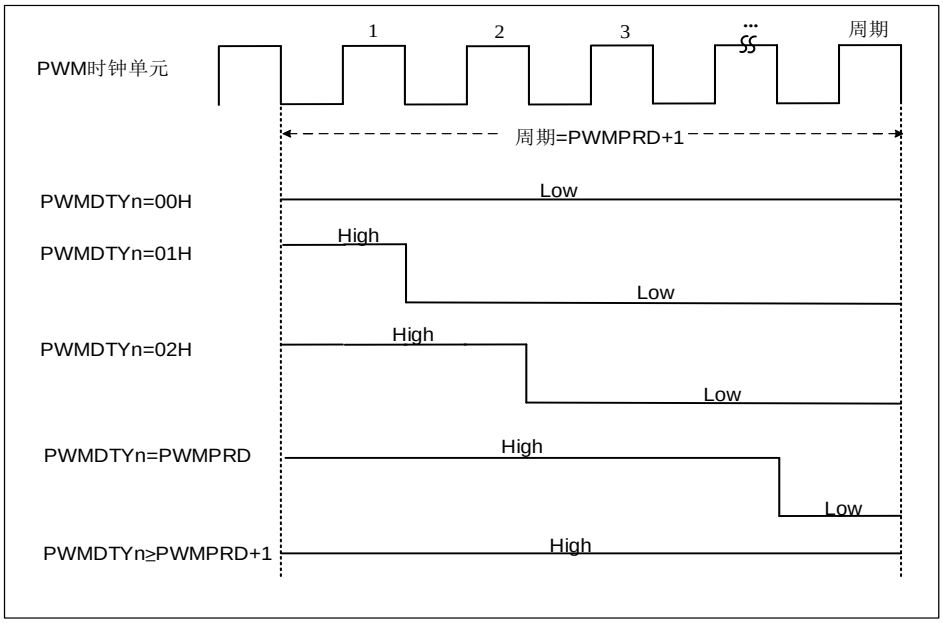
② 周期变化特性



周期变化特性图

当 PWMn 输出波形时，若需改变周期，可通过改变周期设置寄存器 PWMPRD 的值实现。同改变占空比一样，更改 PWMPRD 的值，周期不会立即改变，而是等待本周期结束，在下个周期改变，参考上图所示。

③ 周期和占空比的关系



周期与占空比关系图

周期和占空比的关系如上图所示。该结果的前提是 PWMn(n=0~2,4~5) 输出反向控制(INVn)初始为 0，若需得到相反结果，可置 INVn 为 1。

13 GP I/O

SC92F730X 提供了最多 12 个可控制的双向 GPIO 端口，输入输出控制寄存器用来控制各端口的输入输出状态，当端口作为输入时，每个 I/O 端口带有由 PxPHy 控制的内部上拉电阻。此 12 个 IO 同其他功能复用，其中 P0.0~P0.3 可以通过设置输出二分之一 V_{DD} 的电压，可用来作为 LCD 显示的 COM 驱动。I/O 端口在输出状态下，读到的是端口数据寄存器里的值。

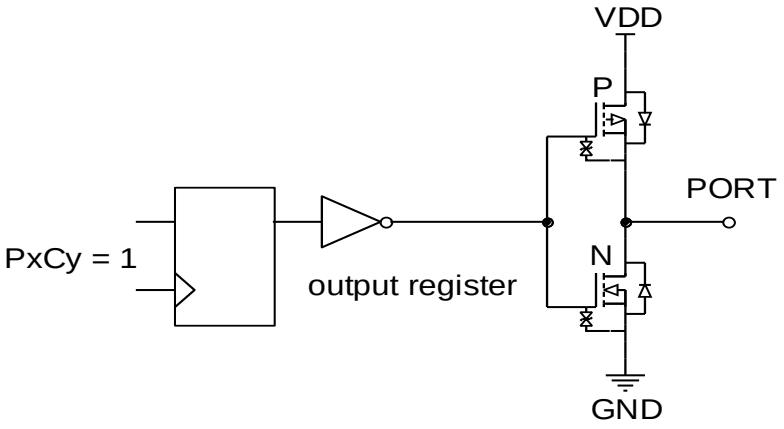
注意：未使用及封装未引出的 IO 口均要设置为强推挽输出模式。

13.1 GPIO 结构图

强推挽输出模式

强推挽输出模式下，能够提供持续的大电流驱动：大于 16mA 的输出高，大于 47mA 的输出低。

强推挽输出模式的端口结构示意图如下：

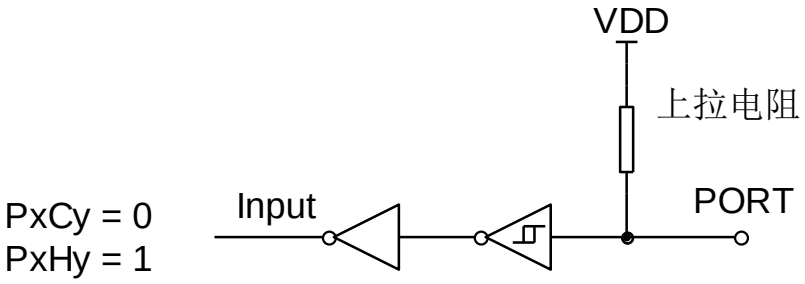


强推挽输出模式

带上拉的输入模式

带上拉的输入模式下，输入口上恒定接一个上拉电阻，仅当输入口上电平被拉低时，才会检测到低电平信号。

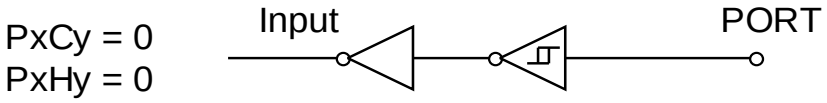
带上拉的输入模式的端口结构示意图如下：



带上拉的输入模式

高阻输入模式(Input only)

高阻输入模式的端口结构示意图如下所示：



高阻输入模式

13.2 I/O 端口相关寄存器

P0CON (9AH) P0 口输入/输出控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	P0C3	P0C2	P0C1	P0C0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

P0PH (9BH) P0 口上拉电阻控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	P0H3	P0H2	P0H1	P0H0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

P1CON (91H) P1 口输入/输出控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	P1C3	P1C2	P1C1	P1C0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

P1PH (92H) P1 口上拉电阻控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	P1H3	P1H2	P1H1	P1H0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

P2CON (A1H) P2 口输入/输出控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	P2C7	P2C6	-	-	-	-	P2C1	P2C0
读/写	读/写	读/写	-	-	-	-	读/写	读/写
上电初始值	0	0	x	x	x	x	0	0

P2PH (A2H) P2 口上拉电阻控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	P2H7	P2H6	-	-	-	-	P2H1	P2H0
读/写	读/写	读/写	-	-	-	-	读/写	读/写
上电初始值	0	0	x	x	x	x	0	0

位编号	位符号	说明
7~0	**PxCy** **（x=0~2, y=0~7）**	Px 口输入输出控制： 0：Pxy 为输入模式（上电初始值） 1：Pxy 为强推挽输出模式
7~0	**PxHy**	Px 口上拉电阻设置，仅在 PxCy=0 时有效：

	（x=0~2, y=0~7）	0：Pxy 为高阻输入模式（上电初始值），上拉电阻关闭； 1：Pxy 上拉电阻打开

P0 (80H) P0 口数据寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	P0.3	P0.2	P0.1	P0.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

P1 (90H) P1 口数据寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	P1.3	P1.2	P1.1	P1.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

P2 (A0H) P2 口数据寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	P2.7	P2.6	-	-	-	-	P2.1	P2.0
读/写	读/写	读/写	-	-	-	-	读/写	读/写
上电初始值	0	0	x	x	x	x	0	0

IOHCON(97H) 输出电流设置寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	P2H[1:0]		P2L[1:0]		P0H[1:0]		P0L[1:0]	
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~6	**P2H[1:0]**	P2 高四位 IOH 设置 00：设置 P2 高四位 I_{OH} 等级 0（最大）； 01：设置 P2 高四位 I_{OH} 等级 1； 10：设置 P2 高四位 I_{OH} 等级 2； 11：设置 P2 高四位 I_{OH} 等级 3（最小）；
5~4	**P2L[1:0]**	P2 低四位 IOH 设置 00：设置 P2 低四位 I_{OH} 等级 0（最大）； 01：设置 P2 低四位 I_{OH} 等级 1； 10：设置 P2 低四位 I_{OH} 等级 2； 11：设置 P2 低四位 I_{OH} 等级 3（最小）；
3~2	**P0H[1:0]**	P0 高四位 IOH 设置 00：设置 P0 高四位 I_{OH} 等级 0（最大）； 01：设置 P0 高四位 I_{OH} 等级 1； 10：设置 P0 高四位 I_{OH} 等级 2； 11：设置 P0 高四位 I_{OH} 等级 3（最小）；
1~0	**P0L[1:0]**	P0 低四位 IOH 设置 00：设置 P0 低四位 I_{OH} 等级 0（最大）； 01：设置 P0 低四位 I_{OH} 等级 1； 10：设置 P0 低四位 I_{OH} 等级 2； 11：设置 P0 低四位 I_{OH} 等级 3（最小）；

14 软件 LCD 驱动

SC92F730X 的 P0.0~P0.3 可作为软件 LCD 的 COM 口，这些 IO 除了正常 IO 功能外，还可输出 1/2V_{DD} 电压。用户可根据使用情况，选择相应 IO 作为 LCD 驱动的 COM。

14.1 软件 LCD 驱动相关寄存器

LCD 驱动相关 SFR 寄存器说明如下：

P0VO (9CH) P0 口 LCD 电压输出寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	P03VO	P02VO	P01VO	P00VO
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

P0yVO (y=0~3)	P0y	P0y 口输出选择
0	x	普通 IO 口
1	1	打开 Pxy 口的 LCD 电压输出功能, Pxy 输出电压为 1/2V_{DD}

OTCON (8FH) 输出控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	VOIRS[1:0]		-	-
读/写	-	-	-	-	读/写	读/写	-	-
上电初始值	x	x	x	x	0	0	x	x

位编号	位符号	说明
3~2	**VOIRS[1:0]**	**LCD 电压输出口分压电阻选择（根据 LCD 屏大小选择适合的驱动）** 00：关闭内部分压电阻（省电） 01：设定内部分压电阻为 12.5K 10：设定内部分压电阻为 37.5K 11：设定内部分压电阻为 87.5K

15 UART

15.1 UART 相关寄存器

SCON (98H) 串口控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~6	**SM0~1**	串行通信模式控制位 00：模式 0，8 位半双工同步通信模式，在 RX 引脚上收发串行数据。TX 引脚用作发送移位时钟。每帧收发 8 位，低位先接收或发送； 01：模式 1，10 位全双工异步通信，由 1 个起始位，8 个数据位和 1 个停止位组成，通信波特率可变； 10：模式 2，11 位全双工异步通信，由 1 个起始位，8 个数据位，一个可编程的第 9 位和 1 个停止位组成； 11：模式 3，11 位全双工异步通信，由 1 个起始位，8 个数据位，一个可编程的第 9 位和 1 个停止位组成，通信波特率可变。
5	**SM2**	串行通信模式控制位 2，此控制位只对模式 2，3 有效 0：每收到一个完整的数据帧就置位 RI 产生中断请求； 1：收到一个完整的数据帧时，只有当 RB8=1 时才会置位 RI 产生中断请求。
4	**REN**	接收允许控制位 0：不允许接收数据； 1：允许接收数据。
3	**TB8**	只对模式 2、3 有效，为发送数据的第 9 位
2	**RB8**	只对模式 2、3 有效，为接收数据的第 9 位
1	**TI**	发送中断标志位
0	**RI**	接收中断标志位

SBUF (99H) 串口数据缓存寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	SBUF[7:0]							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**SBUF[7:0]**	**串口数据缓存寄存器** SBUF 包含两个寄存器：一个发送移位寄存器和一个接收锁存器，写入 SBUF 的数据将送至发送移位寄存器，并启动发送流程，读 SBUF 将返回接收锁存器中的内容。

PCON (87H) 电源管理控制寄存器(只写、*不可读 *)

位编号	7	6	5	4	3	2	1	0
符号	SMOD	-	-	-	-	-	STOP	IDL
读/写	-	-	-	-	-	-	只写	只写
上电初始值	x	x	x	x	x	x	0	0

位编号	位符号	说明
7	**SMOD**	**波特率倍率设置位** 0：SM0~1 = 00 时，串行端口在系统时钟的 1/12 下运行、SM0~1 = 10 时，串行端口在系统时钟的 1/64 下运行； 1：SM0~1 = 00 时，串行端口在系统时钟的 1/4 下运行、SM0~1 = 10 时，串行端口在系统时钟的 1/32 下运行。

15.2 串口通信的波特率

方式 0 中，波特率可编程为系统时钟的 1/12 或 1/4，由 SMOD(PCON.7)位决定。当 SMOD 为 0 时，串行端口在系统时钟的 1/12 下运行。当 SMOD 为 1 时，串行端口在系统时钟的 1/4 下运行。

在方式 1 和方式 3 中，波特率可选择来至定时器 1 或定时器 2 的溢出率。

分别置 TCLK(T2CON.4)和 RCLK(T2CON.5)位为 1 来选择定时器 2 作为 TX 和 RX 的波特时钟源(详见定时器章节)。无论 TCLK 还是 RCLK 为逻辑 1，定时器 2 都为波特率发生器方式。如果 TCLK 和 RCLK 为逻辑 0，定时器 1 作为 Tx 和 Rx 的波特时钟源。

方式 1 和方式 3 波特率公式如下所示，其中 TH1 是定时器 1 的 8 位自动重载寄存器，SMOD 为 UART 的波特率二倍频器，[RCAP2H、RCAP2L]是定时器 2 的 16 位重载入寄存器。

1. 用定时器 1 作为波特率发生器，定时器 1 工作在模式 2：

$$\text{BaudRate} = \frac{2^{SMOD}}{16} \times \frac{\text{fn1}}{(256 - \text{TH1}) \times 2}$$

其中， fn1 为定时器 1 时钟频率：

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

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

2. 用定时器 2 作为波特率发生器：

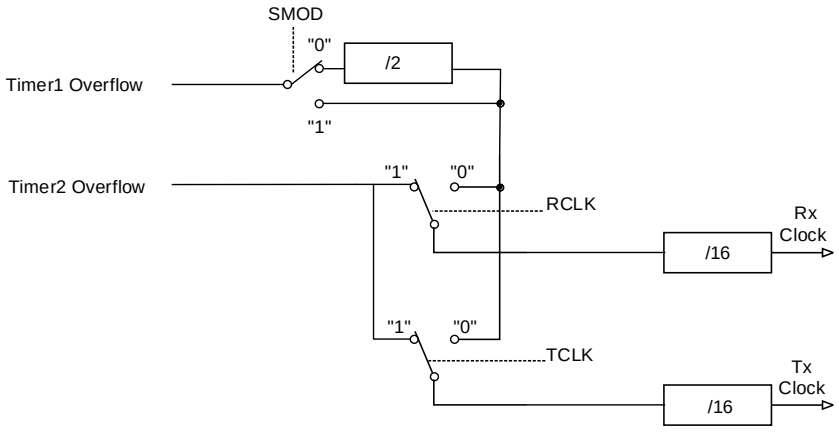
$$\text{BaudRate} = \frac{1}{16} \times \frac{\text{fn2}}{(65536 - [\text{RCAP2H}, \text{RCAP2L}]) \times 2}$$

其中， fn2 为定时器 2 时钟频率：

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

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

方式 1 和方式 3 波特率发生器的原理图如下：



方式 1 和方式 3 波特率发生器原理

在方式 2 中，波特率固定为系统时钟的 1/32 或 1/64，由 SMOD 位(PCON.7)决定。当 SMOD 位为 0 时，波特率为系统时钟的 1/64。当 SMOD 位为 1 时，波特率为系统时钟的 1/32。

16 模数转换 ADC

SC92F730X 内建一个 12-bit 5 通道的高精度逐次逼近型 ADC ，外部的 4 路 ADC 和 IO 口的其它功能复用。内部还有一个通道可选择到 1/4 V_{DD}，配合内部 2.4V 参考电压用于测量 V_{DD} 电压。

ADC 的参考电压可以有 2 种选择：

① 是 V_{DD} 管脚（即直接是内部的 V_{DD}）；

② 是内部 Regulator 输出的参考电压精准的 2.4V。

16.1 ADC 相关寄存器

ADCCON (ADH) ADC 控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ADCEN	ADCS	LOWSP	EOC/ADCIF	ADCIS[3:0]			
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	n

位编号	位符号	说明
7	**ADCEN**	启动 ADC 的电源 0：关闭 ADC 模块电源 1：开启 ADC 模块电源
6	**ADCS**	ADC 开始触发控制（ADC Start） 对此 bit 写 “1”，开始做一次 ADC 的转换，即该位只是 ADC 转换的触发信号。此位只可写入 1 有效。 **注意：对 ADCS 写“1”后，到中断标志 EOC/ADCIF 置起前不要对 ADCCON 寄存器进行写操作**
5	**LOWSP**	ADC 采样时钟频率选择(ADC Sampling Clocks Selector) 0： 设定 ADC 所使用的 clock 频率为 2MHz 1： 设定 ADC 所使用的 clock 频率为 333kHz LOWSP 控制的是 ADC 的采样时钟频率，ADC 的转换时钟频率固定为 2MHz，且不受 LOWSP 位的影响 ADC 需经历 6 个 ADC 采样时钟加上 14 个 ADC 转换时钟的时间才能完成从采样到转换的整个过程，因此在实际使用中，ADC 从采样到完成转换的总时间计算如下： LOWSP=0：T_{ADC1}=6*(1/2MHz)+14*(1/2 MHz)=10us； LOWSP=1：T_{ADC2}=6*(1/333kHz)+14*(1/2 MHz)=25us。
4	**EOC /ADCIF**	转换完成/ADC 中断请求标志(End Of Conversion / ADC Interrupt Flag) 0：转换尚未完成 1：ADC 转换完成。需用户软件清除 ADC 转换完成标志 EOC：当使用者设定 ADCS 开始转换后，此位会被硬件自动清除为 0； 当转换完成后, 此位会被硬件自动置为 1； ADC 中断请求标志 ADCIF：此位同时也当作是 ADC 中断的中断请求标志，如果用户使能 ADC 中断， 那么在 ADC 的 中断 发生后, 用户必须用软件清除此位。
3~0	**ADCIS[3:0]**	ADC 输入通道选择(ADC Input Selector) 0000：选用 AIN0 为 ADC 的输入 0001：选用 AIN1 为 ADC 的输入 0110：选用 AIN6 为 ADC 的输入 0111：选用 AIN7 为 ADC 的输入 1111： ADC 输入为 1/4 V_{DD}，可用于测量电源电压

		其它：保留

ADCCFG0 (ABH) ADC 设置寄存器 0(读/写)

位编号	7	6	5	4	3	2	1	0
符号	EAIN7	EAIN6	-	-	-	-	EAIN1	EAIN0
读/写	读/写	读/写	-	-	-	-	读/写	读/写
上电初始值	0	0	x	x	x	x	0	0

位编号	位符号	说明
0	**EAINx (x=0~1, 6~7)**	**ADC 端口设置寄存器** 0：设定 AINx 为 IO 口 1：设定 AINx 为 ADC 输入，并自动将上拉电阻移除。

OP_CTM1 (C2H@FFH) Customer Option 寄存器 1(读/写)

位编号	7	6	5	4	3	2	1	0
符号	VREFS	-	-	-	IAPS[1:0]		-	-
读/写	读/写	-	-	-	读/写	读/写	-	-
上电初始值	n	x	x	x	n	n	x	x

位编号	位符号	说明
7	**VREFS**	**参考电压选择(初始值从 Code Option 调入，用户可修改设置)** 0：设定 ADC 的 VREF 为 V_{DD} 1：设定 ADC 的 VREF 为 内部准确的 2.4V

ADCVL (AEH) ADC 转换数值寄存器(低位)(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ADCV[3:0]				-	-	-	-
读/写	读/写	读/写	读/写	读/写	-	-	-	-
上电初始值	0	0	0	0	x	x	x	x

ADCVH (AFH) ADC 转换数值寄存器(高位)(读/写)

位编号	7	6	5	4	3	2	1	0
符号	ADCV[11:4]							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
11~4	**ADCV[11:4]**	ADC 转换值的高 8 位数值
3~0	**ADCV[3:0]**	ADC 转换值的低 4 位数值

IE (A8H) 中断使能寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	EA	EADC	ET2	EUART	ET1	-	ET0	EINT0
读/写	读/写	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	0	0	0	0	0	x	0	0

位编号	位符号	说明
6	**EADC**	ADC 中断使能控制 0： 不允许 EOC/ADCIF 产生中断 1： 允许 EOC/ADCIF 产生 中断

IP (B8H) 中断优先级控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	IPADC	IPT2	IPUART	IPT1	-	IPT0	IPINT0
读/写	-	读/写	读/写	读/写	读/写	-	读/写	读/写
上电初始值	x	0	0	0	0	x	0	0

位编号	位符号	说明
6	**IPADC**	ADC 中断优先权选择 0：设定 ADC 的中断优先级是 “低” 1：设定 ADC 的中断优先级是 “高”

16.2 ADC 转换步骤

用户实际进行 ADC 转换所需要的操作步骤如下：

① 设定 ADC 输入管脚；（设定 AINx 对应的位为 ADC 输入，通常 ADC 管脚会预先固定）
② 设定 ADC 参考电压 Vref, 设定 ADC 转换所用的频率；
③ 开启 ADC 模块电源；
④ 选择 ADC 输入通道；(设置 ADCIS 位，选择 ADC 输入通道)；
⑤ 启动 ADCS，转换开始；
⑥ 等待 EOC/ADCIF=1，如果 ADC 中断使能，则 ADC 中断会产生，用户需要软件清 0 EOC/ADCIF 标志；
⑦ 从 ADCVH、ADCVL 获得 12 位数据，先高位后低位，一次转换完成；
⑧ 如不换输入通道，则重复 5~7 的步骤，进行下一次转换。

注意事项：在设定 IE[6](EADC)前，使用者最好用软件先清除 EOC/ADCIF，并且在 ADC 中断服务程序执行完时，也清除该 EOC/ADCIF，以避免不断的产生 ADC 中断。

17 EEPROM 及 IAP 操作

SC92F730X 的 IAP 操作空间范围有两种模式可选：

EEPROM 及 IAP 的操作模式如下：

1. 128 bytes EEPROM 可以作为数据存储使用；
2. IC 的 Code 区域（范围可选）及 128 bytes EEPROM 内都可进行 In Application Programming(IAP)操作，主要用作远程程序更新使用。

EEPROM 及 IAP 操作模式选择作为 Code Option 在编程器写入 IC 时选择：

OP_CTM1 (C2H@FFH) Customer Option 寄存器 1(读/写)

位编号	7	6	5	4	3	2	1	0
符号	VREFS	-	-	-	IAPS[1:0]		-	-
读/写	读/写	-	-	-	读/写	读/写	-	-
上电初始值	n	x	x	x	n	n	x	x

位编号	位符号	说明
3~2	**IAPS[1:0]**	IAP 空间范围选择 00: Code 区域禁止 IAP 操作，仅 EEPROM 区域可作为数据存储使用 01: 最后 0.5K Code 区域允许 IAP 操作(1E00H ~1FFFH) 10: 最后 1K Code 区域允许 IAP 操作(1C00H~1FFFH) 11: 全部 Code 区域允许 IAP 操作(0000H~1FFFH)

17.1 EEPROM / IAP 操作相关寄存器

EEPROM / IAP 操作相关寄存器说明：

符号	地址	说明	7	6	5	4	3	2	1	0	Reset 值
IAPKEY	F1H	IAP 保护寄存器	IAPKEY[7:0]								00000000b
IAPADL	F2H	IAP 写入地址低位寄存器	IAPADR[7:0]								00000000b
IAPADH	F3H	IAP 写入地址高位寄存器	-	-	-	IAPADR[12:8]					xxx00000b
IAPADE	F4H	IAP 写入扩展地址寄存器	IAPADER[7:0]								00000000b
IAPDAT	F5H	IAP 数据寄存器	IAPDAT[7:0]								00000000b
IAPCTL	F6H	IAP 控制寄存器	-	-	-	-	PAYTIMES [1:0]		CMD[1:0]		xxxx0000b

IAPKEY (F1H) IAP 保护寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	IAPKEY[7:0]							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**IAPKEY[7:0]**	打开 EEPROM / IAP 功能及操作时限设置 写入一个非零值 n，代表： ① 打开 EEPROM / IAP 功能； ② n 个系统时钟后如果接收不到写入命令，则 EEPROM / IAP 功能被重新关闭。

IAPADL (F2H) IAP 写入地址低位寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	IAPADR[7:0]							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**IAPADR[7:0]**	EEPROM / IAP 写入地址的低 8 位

IAPADH (F3H) IAP 写入地址高位寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	IAPADR[12:8]				
读/写	-	-	-	读/写	读/写	读/写	读/写	读/写
上电初始值	x	x	x	0	0	0	0	0

位编号	位符号	说明
4~0	**IAPADR[12:8]**	EEPROM / IAP 写入地址的高 5 位
7~6	**-**	保留

IAPADE (F4H) IAP 写入扩展地址寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	IAPADER[7:0]							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**IAPADER[7:0]**	IAP 扩展地址： 0x00：MOVC 和 IAP 烧写都针对 Code 进行 0x01：针对用户 ID 区域进行读操作，不可进行写操作 0x02：MOVC 和写入都针对 EEPROM 进行 其它：保留

IAPDAT (F5H) IAP 数据寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	IAPDAT[7:0]							
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	**IAPDAT**	EEPROM / IAP 写入的数据

IAPCTL(F6H)IAP 控制寄存器

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	PAYTIMES[1:0]		CMD[1:0]	
读/写	-	-	-	-	读/写	读/写	读/写	读/写
上电初始值	x	x	x	x	0	0	0	0

位编号	位符号	说明
3~2	**PAYTIMES[1:0]**	EEPROM / IAP 写入操作时，CPU Hold Time 时间长度设定 00：设定 CPU HOLD TIME 4mS@12/6/2MHz 01：设定 CPU HOLD TIME 2mS@12/6/2MHz 10：设定 CPU HOLD TIME 1mS@12/6/2MHz 11：保留 说明：CPU Hold 的是 PC 指针，其他功能模块继续工作；中断标志会被保存，并在 Hold 结束后进入中断，但多次的中断只能保留最后一次。

		选择建议：V_{DD} 在 2.7V~5.5V，可选择 10 V_{DD} 在 2.4V~5.5V，可选择 01 或者 00
1~0	**CMD[1:0]**	EEPROM / IAP 写入操作命令 10 ：写入 其它：保留 **注意：EEPROM / IAP 写操作的语句后面务必要加上至少 8 个 NOP 指令，以保证 EEPROM / IAP 操作完成后可正常执行后续的指令！**

17.2 EEPROM / IAP 操作流程

SC92F730X 的 EEPROM / IAP 的 写入流程如下:

① 写入 IAPADE[7:0] ，0x00：选择 Code 区，进行 IAP 操作；0x02：选择 EEPROM 区，进行 EEPROM 读写操作；
② 写入 IAPDAT[7:0]（准备好 EEPROM / IAP 写入的数据）；
③ 写入 {IAPADR[12:8] ，IAPADR[7:0]}（准备好 EEPROM / IAP 操作的目标地址）；
④ 写入 IAPKEY[7:0] 写入一个非 0 的值 n（打开 EEPROM / IAP 保护，且在 n 个系统时钟内没收到写入命令 EEPROM / IAP 会被关闭）；
⑤ 写入 IAPCTL[3:0]（设定 CPU Hold 时间，写入 CMD[1:0]为 1、0，CPU Hold 并启动 EEPROM / IAP 写入）；
⑥ EEPROM / IAP 写入结束，CPU 继续后续操作。

注意：编程 IC 时，若通过 Code Option 选择了“Code 区域禁止 IAP 操作”，则 IAPADE[7:0]=0x00 时（选择 Code 区），IAP 不可操作，即数据无法写入，仅可通过 MOVC 指令读取数据。

17.2.1 128 BYTES 独立 EEPROM 操作例程

```
#include ”intrins.h”
unsigned char EE_Add;
unsigned char EE_Data;
unsigned char code * POINT =0x0000;
```

EEPROM 写操作 C 的 Demo 程序：

```
EA = 0;                 //关总中断
IAPADE = 0x02;          //选择 EEPROM 区域
IAPDAT = EE_Data;       //送数据到 EEPROM 数据寄存器
IAPADH = 0x00;          //写入 EEPROM 目标地址高位值
IAPADL = EE_Add;        //写入 EEPROM 目标地址低位值
IAPKEY = 0xF0;          //此值可根据实际调整；需保证本条指令执行后到对 IAPCTL 赋值前，
                        //时间间隔需小于 240（0xf0）个系统时钟，否则 IAP 功能关闭;
                        // 开启中断时要特别注意
IAPCTL = 0x0A;          //执行 EEPROM 写入操作，1ms@12M/6M/2M;
_nop_();                //等待(至少需要 8 个_nop_())
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();

IAPADE = 0x00;          //返回 ROM 区域
EA = 1;                 //开总中断
```

EEPROM 读操作 C 的 Demo 程序：

```
EA = 0;                          //关总中断
IAPADE =  0x02;                  //选择 EEPROM 区域
EE_Data = *( POINT +EE_Add);          //读取 IAP_Add 的值到 IAP_Data
IAPADE =  0x00;                  //返回 ROM 区域, 防止 MOVC 操作到 EEPROM
EA = 1;                          //开总中断
```

17.2.2 8 KBYTES CODE 区域 IAP 操作例程

```
#include ”intrins.h”
unsigned int IAP_Add;
unsigned char IAP_Data;
unsigned char code * POINT =0x0000;
```

IAP 写操作 C 的 Demo 程序：

```
IAPADE = 0x00;                   //选择 Code 区域
IAPDAT = IAP_Data;               //送数据到 IAP 数据寄存器
IAPADH = (unsigned char)((IAP_Add >> 8));      //写入 IAP 目标地址高位值
IAPADL = (unsigned char)IAP_Add;               //写入 IAP 目标地址低位值
IAPKEY = 0xF0;                   //此值可根据实际调整；需保证本条指令执行后到对 IAPCTL 赋值前，
                                 //时间间隔需小于 240（0xf0）个系统时钟，否则 IAP 功能关闭;
                                 // 开启中断时要特别注意
IAPCTL = 0x0A;                   //执行 IAP 写入操作，1ms@12M/6M/2M;
_nop_();                         //等待(至少需要 8 个_nop_())
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
_nop_();
```

IAP 读操作 C 的 Demo 程序：

```
IAPADE = 0x00;                   //选择 Code 区域
IAP_Data = *( POINT+IAP_Add);         //读取 IAP_Add 的值到 IAP_Data
```

注意：8 Kbytes Code 区域内的 IAP 操作有一定的风险，需要用户在软件中做相应的安全处理措施，**如果操作不当可能会造成用户程序被改写！**除非用户必需此功能(比如用于远程程序更新等)，不建议用户使用。

18 电气特性

18.1 极限参数

符号	参数	最小值	最大值	UNIT
VDD/VSS	直流供电电压	-0.3	5.5	V
Voltage ON any Pin	任一管脚输入/输出电压	-0.3	V_{DD}+0.3	V
T_A	工作环境温度	-40	85	℃
T_{STG}	储存温度	-55	125	℃

18.2 推荐工作条件

符号	参数	最小值	最大值	UNIT
V_{DD}	工作电压	2.4	5.5	V
T_A	工作环境温度	-40	85	℃

18.3 直流电气特性

(V_{DD} = 5V，T_A = +25℃，除非另有说明)

符号	参数	最小值	典型值	最大值	单位	测试条件
电流						
I_{op1}	工作电流	-	7	-	mA	f_{SYS}=12MHz
I_{op2}	工作电流	-	6	-	mA	f_{SYS}=6MHz
I_{op3}	工作电流	-	5	-	mA	f_{SYS}=2MHz
I_{pd1}	待机电流（Power Down 模式）	-	0.7	1.0	μA	
I_{IDL}	待机电流（IDLE 模式）	-	6.7	-	mA	
I_{BTM}	Base Timer 工作电流	-	6	8	μA	BTMFS[3:0]= 1000 每 4.0 秒产生一个中断
I_{WDT}	WDT 电流	-	4	6	μA	WDTCKS[2:0]= 000 WDT 溢出时间 500ms
IO 口特性						
V_{IH1}	输入高电压	0.7V_{DD}	-	V_{DD}+0.3	V	
V_{IL1}	输入低电压	-0.3	-	0.3V_{DD}	V	
V_{IH2}	输入高电压	0.8V_{DD}	-	V_{DD}	V	施密特触发输入: RST/tCK/SCK
V_{IL2}	输入低电压	-0.2	-	0.2V_{DD}	V	
I_{OL1}	输出低电流	-	28	-	mA	V_{Pin}=0.4V
I_{OL2}	输出低电流	-	47	-	mA	V_{Pin}=0.8V
I_{OH1}	输出高电流 Pxyz=0	-	16	-	mA	V_{Pin}=4.3V P0/P2 I_{OH} 等级 0 及 P1
	输出高电流 Pxyz=1	-	13	-	mA	V_{Pin}=4.3V P0/P2 I_{OH} 等级 1
	输出高电流 Pxyz=2	-	9	-	mA	V_{Pin}=4.3V P0/P2 I_{OH} 等级 2

	输出高电流 Pxyz=3	-	5	-	mA	V_{Pin}=4.3V P0/P2 I_{OH} 等级 3
I_{OH2}	输出高电流 Pxyz=0	-	7	-	mA	V_{Pin}=4.7V P0/P2 I_{OH} 等级 0 及 P1
	输出高电流 Pxyz=1	-	6	-	mA	V_{Pin}=4.7V P0/P2 I_{OH} 等级 1
	输出高电流 Pxyz=2	-	4	-	mA	V_{Pin}=4.7V P0/P2 I_{OH} 等级 2
	输出高电流 Pxyz=3	-	2	-	mA	V_{Pin}=4.7V P0/P2 I_{OH} 等级 3
R_{PH1}	上拉电阻	-	30	-	kΩ	
做为 ADC 参考电压的内部基准 2.4V						
V_{DD24}	内部基准 2.4V 电压输出	2.37	2.40	2.45	V	T_A=-40~85℃

(V_{DD} = 3.3V，T_A = +25℃，除非另有说明)

符号	参数	最小值	典型值	最大值	单位	测试条件
电流						
I_{op4}	工作电流	-	5.0	-	mA	f_{SYS}=12MHz
I_{op5}	工作电流	-	4.2	-	mA	f_{SYS}=6MHz
I_{op6}	工作电流	-	3.6	-	mA	f_{SYS}=2MHz
I_{pd2}	待机电流 (Power Down 模式)	-	0.7	1	uA	
I_{IDL2}	待机电流 (IDLE 模式)	-	4.7	-	mA	
IO 口特性						
V_{IH3}	输入高电压	0.7_{DD}	-	V_{DD}+0.3	V	
V_{IL3}	输入低电压	-0.3	-	$0.3V_{DD}$	V	
V_{IH4}	输入高电压	$0.8V_{DD}$	-	V_{DD}	V	施密特触发输入: RST/tCK/SCK
V_{IL4}	输入低电压	-0.2	-	$0.2V_{DD}$	V	
I_{OL3}	输出低电流	-	20	-	mA	V_{Pin} =0.4V
I_{OL4}	输出低电流	-	38	-	mA	V_{Pin} =0.8V
I_{OH3}	输出高电流	-	7	-	mA	V_{Pin} =3.0V
R_{PH2}	上拉电阻	-	52	-	kΩ	
做为 ADC 参考电压的内部基准 2.4V						
V_{DD24}	内部基准 2.4V 电压输出	2.37	2.40	2.45	V	T_A =-40~85℃

18.4 交流电气特性

(V_{DD} = 2.4V ~ 5.5V，T_A = 25℃，除非另有说明)

符号	参数	最小值	典型值	最大值	单位	测试条件
T_{POR}	Power On Reset 时间	-	5	10	ms	
T_{PDW}	Power Down 模式唤醒时间	-	1	1.5	ms	
T_{Reset}	复位脉冲宽度	18	-	-	us	低电平有效
f_{HRC}	RC 振荡稳定性	23.76	24	24.24	MHz	V_{DD}=4.0~5.5V T_A=-20~85 ℃

18.5 ADC 电气特性

(T_A = 25℃，除非另有说明)

符号	参数	最小值	典型值	最大值	单位	测试条件
V_{AD}	供电电压	3.0	5.0	5.5	V	
NR	精度	-	12	-	bit	GND≤V_{AIN}≤V_{DD}
V_{AIN}	ADC 输入电压	GND	-	V_{DD}	V	
R_{AIN}	ADC 输入电阻	1	-	-	MΩ	V_{IN}=5V
I_{ADC1}	ADC 转换电流 1	-	-	2	mA	ADC 模块打开 V_{DD}=5V
I_{ADC2}	ADC 转换电流 2	-	-	1.8	mA	ADC 模块打开 V_{DD}=3.3V
DNL	微分非线性误差	-	±1	-	LSB	V_{DD}=5V V_{REF}=5V
INL	积分非线性误差	-	±1	-	LSB	
E_Z	偏移量误差	-	±10	-	LSB	
E_F	满刻度误差	-	0	-	LSB	
E_{AD}	总绝对误差	-	±10	-	LSB	
T_{ADC1}	ADC 转换时间	-	10	-	us	ADC Clock = 2MHz
T_{ADC2}	ADC 转换时间	-	25	-	us	ADC Clock = 333kHz

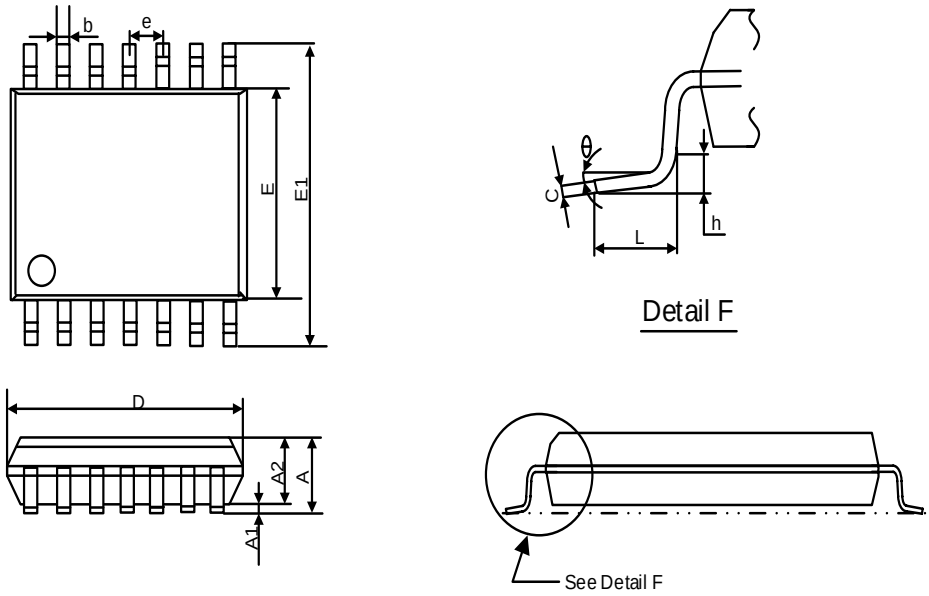
19 订购信息

产品编号	封装	包装
SC92F7309X14U	TSSOP14	管装
SC92F7308BS10U	MSOP10	管装

20 封装信息

SC92F7309X14U

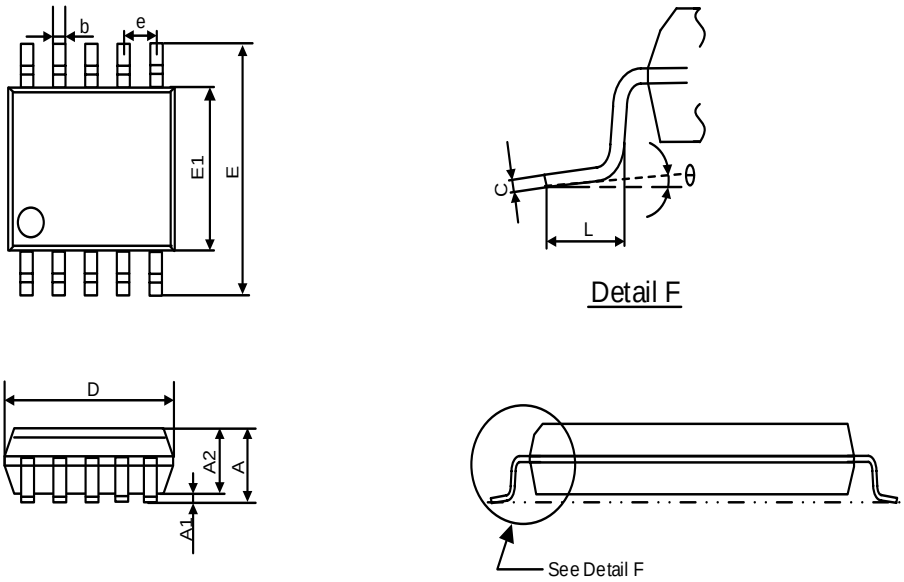
TSSOP14 外形尺寸　　单位：毫米



符号	mm(毫米)	
	最小	最大
D	4.900	5.100
E	4.300	4.500
b	0.190	0.300
c	0.090	0.200
E1	6.250	6.550
A	-	1.200
A2	0.800	1.000
A1	0.050	0.150
e	0.65（BSC）	
L	0.500	0.700
h	0.25（TYP）	
θ	1°	7°

SC92F7308BS10U

MSOP10 外形尺寸　单位：毫米



符号	mm(毫米)	
	最小	最大
A	-	1.100
A1	0.020	0.150
A2	0.750	0.950
b	0.180	0.330
c	0.090	0.230
D	2.900	3.100
e	0.500（BSC）	
E	4.750	5.050
E1	2.900	3.100
L	0.400	0.800
θ	0°	6°

21 规格更改记录

版本	记录	日期
V1.0	初版	2020 年 10 月