

高速 1T 8051 内核 Flash MCU，0.5 Kbytes SRAM，8 Kbytes Flash，256 bytes 独立 EEPROM，17 通道可低功耗高灵敏度触控电路，4 个定时器，三选一通口 SSI，Checksum

## 1 总体描述

SC92F6372/6371/6378/6370（以下简称 SC92F637X）是一颗增强型的高速 1T 8051 内核工业级集成触控按键功能的 Flash 微控制器，指令系统完全兼容传统 8051 产品系列。

SC92F637X 内建一个 17 通道的可低功耗高灵敏度电容触控电路，触控电路可选择在 STOP Mode 下运行。SC92F637X 还集成有 8 Kbytes Flash ROM、0.5 Kbytes SRAM、256 bytes EEPROM、最多 18 个 GP I/O、7 个 IO 可外部中断、Timer0~3 共 4 个 16 位定时器、4 路常规 PWM(Timer2/3 各两路输出)、内部±1%高精度高频 32 MHz 振荡器和低频 32 kHz 振荡器、一个 UART/SPI/TWI 三选一通口 SSI。为提高可靠性及简化客户电路，SC92F637X 内部也集成有 4 级可选电压 LVR。SC92F637X 具有非常优异的抗干扰性能和性能极好的触控按键性能，非常适合应用于各种使用场合的触控按键和主控控制，如大小智能家电和智能家居、物联网、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 ~ 105°C

### EMS

- ESD
  - HBM: JS-001-2023 Class 3A
  - CDM: ANSI/ESDA/JEDEC JS-002-2022 Class C3
- EFT
  - EN6100-4-4 Level 4

### 封装：

- SOP20/TSSOP20/QFN20(3X3)
- SOP16
- MSOP10
- SOP8

内核：1T 8051

### Flash ROM：

- 8 Kbytes Flash ROM
- 可重复烧写 1000 次，10 年以上保存寿命

### EEPROM：

- 256 bytes，无需擦除，10,000 次写入，10 年以上保存寿命
- 用户在对 EEPROM 进行操作时，需保证 VDD 电压范围在 2.5V~5.5V 之间

SRAM：内部 256 bytes+外部 256 bytes

### 系统时钟 (fsys)：

- 内建高频 32 MHz 振荡器 (fHRC)
  - 作为系统时钟源时，fsys 可通过编程器选择设定

为 1/2/4/12 这四种分频中的一种

- 频率误差：跨越 (2.4V~5.5V) 及 (-40 ~ 85°C) 应用环境，不超过 ±1%
- 频率误差：跨越 (2.4V~5.5V) 及 (-40 ~ 105°C) 应用环境，不超过 ±1.5%

### 内建低频 32kHz LRC 振荡器：

- 可作为 Base Timer 的时钟源，可唤醒 STOP
- 可作为 WDT 的时钟源

### 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
- 缺省值为为用户烧写 Code Option 所选值

### Flash 烧写和仿真：

- 2 线 JTAG 烧写和仿真接口

### 中断 (INT)：

- Timer0, Timer1, Timer2, Timer3, INT0, INT1, SSI, Base Timer, TK 共 9 个中断源
- 外部中断有 2 个中断向量，共 7 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 18 个双向可独立控制的 I/O 口，可独立设定上拉电阻
- IO 口灌电流：
  - P20 / P21 / P24 / P25 / P26 / P05 六个端口具有大灌流驱动能力 100mA@0.8V
  - 除 P20 / P21 / P24 / P25 / P26 / P05 六个端口之外的所有端口灌电流驱动能力 80mA@0.8V
- 11 位 WDT，可选时钟分频比
- 4 个标准 80C51 定时器 Timer0、Timer1、Timer2 和 Timer3
- Timer2/3 可实现 Capture 功能
- 1 个三合一 SSI
  - UART 通信口、SPI 通信口和 TWI 通信口
  - 两组信号口：TX/SDA 和 RX/SCL 可支持全管脚映射

### 模拟外围：

- 17 通道可低功耗高灵敏度触控电路：
  - 高灵敏度模式可适应隔空按键触控、接近感应可适应隔空按键触控、接近感应等对灵敏度要求较高的触控应用
  - CMOD 管脚需对地接入 103 电容
  - 具有很强的抗干扰性，可通过 10V 动态 CS 测试
  - 支持低功耗模式
  - 全套开发支持：高灵活触控软件库，智能化调试软件

### 省电模式：

- IDLE Mode，可由任何中断唤醒
- STOP Mode，由 INT0、INT1、BaseTimer 和 TK 唤醒

## 92 系列产品命名规则

名称	SC	92	F	6	3	7	2	X	M	20	U
序号	①	②	③	④	⑤	⑥	⑦	⑧	⑨	⑩	⑪

序号	含义
①	Sinone Chip 缩写
②	产品系列名称
③	产品类型 (F: Flash MCU)
④	系列号: 5: GP 系列, 6: TK 系列
⑤	ROM Size: 1 为 2K, 2 为 4K, 3 为 8K, 4 为 16K, 5 为 32K...
⑥	子系列编号: 0~9, A~Z
⑦	引脚数: 0: 8pin, 1: 16pin, 2: 20pin, 3: 28pin, 5: 32pin, 6: 44pin, 7: 48pin, 8: 64pin/10pin, 9: 100pin/24pin,
⑧	版本号: (缺省、B、C、D)
⑨	封装形式: (D: DIP; M: SOP; X: TSSOP; F: QFP; P: LQFP; Q: QFN; K: SKDIP; S: MSOP/SSOP);
⑩	引脚数
⑪	包装方式: (U: 管装; R: 盘装; T: 卷带)

## 目录

1	总体描述 .....	1
2	主要功能 .....	1
92	系列产品命名规则 .....	2
	目录.....	3
3	管脚定义 .....	7
3.1	管脚配置.....	7
3.2	管脚定义.....	9
4	内部框图 .....	11
5	FLASH ROM 和 SRAM 结构.....	12
5.1	FLASH ROM.....	12
5.2	256 bytes EEPROM.....	12
5.3	用户 ID 区域.....	12
5.4	烧写和仿真 .....	13
5.4.1	JTAG 专用模式.....	13
5.4.2	常规模式 ( JTAG 专用口无效 ) .....	13
5.5	EEPROM 区域及用户 ID 区域读写操作.....	14
5.5.1	EEPROM 读写操作相关寄存器 .....	14
5.5.2	EEPROM 操作流程 .....	15
5.6	Customer Option 区域 (用户烧写设置) .....	16
5.6.1	Option 相关 SFR 操作说明 .....	17
5.7	SRAM .....	18
5.7.1	内部 256 bytes SRAM.....	18
5.7.2	外部 256 bytes SRAM.....	19
6	特殊功能寄存器(SFR) .....	20
6.1	SFR 映像.....	20
6.2	SFR 说明.....	21
6.2.1	11 bytes 寄存器扩展 RAM .....	22
6.3	8051 CPU 内核常用特殊功能寄存器介绍 .....	23

6.3.1	程序计数器 PC .....	23
6.3.2	累加器 ACC (E0H) .....	23
6.3.3	B 寄存器 (F0H) .....	23
6.3.4	堆栈指针 SP (81H) .....	23
6.3.5	PSW (D0H) 程序状态字寄存器(读/写) .....	23
6.3.6	数据指针 DPTR (82H、83H) .....	24
<b>7</b>	<b>电源、复位和时钟 .....</b>	<b>25</b>
7.1	电源电路 .....	25
7.2	上电复位过程 .....	25
7.2.1	复位阶段 .....	25
7.2.2	调入信息阶段 .....	25
7.2.3	正常操作阶段 .....	25
7.3	复位方式 .....	25
7.3.1	外部 RST 复位 .....	25
7.3.2	低电压复位 LVR .....	26
7.3.3	上电复位 POR .....	26
7.3.4	看门狗复位 WDT .....	26
7.3.5	软件复位 .....	27
7.3.6	复位初始状态 .....	27
7.4	高频系统时钟电路 .....	28
7.5	低频振荡器及低频时钟定时器 .....	29
7.6	STOP 模式和 IDLE 模式 .....	30
<b>8</b>	<b>中央处理单元 CPU 及指令系统 .....</b>	<b>32</b>
8.1	CPU .....	32
8.2	寻址方式 .....	32
8.2.1	立即寻址 .....	32
8.2.2	直接寻址 .....	32
8.2.3	间接寻址 .....	32
8.2.4	寄存器寻址 .....	32
8.2.5	相对寻址 .....	32
8.2.6	变址寻址 .....	32
8.2.7	位寻址 .....	32
<b>9</b>	<b>INTERRUPT 中断 .....</b>	<b>34</b>
9.1	中断源、向量 .....	34
9.2	中断结构图 .....	35
9.3	中断优先级 .....	35
9.4	中断处理流程 .....	36
9.5	中断相关 SFR 寄存器 .....	36

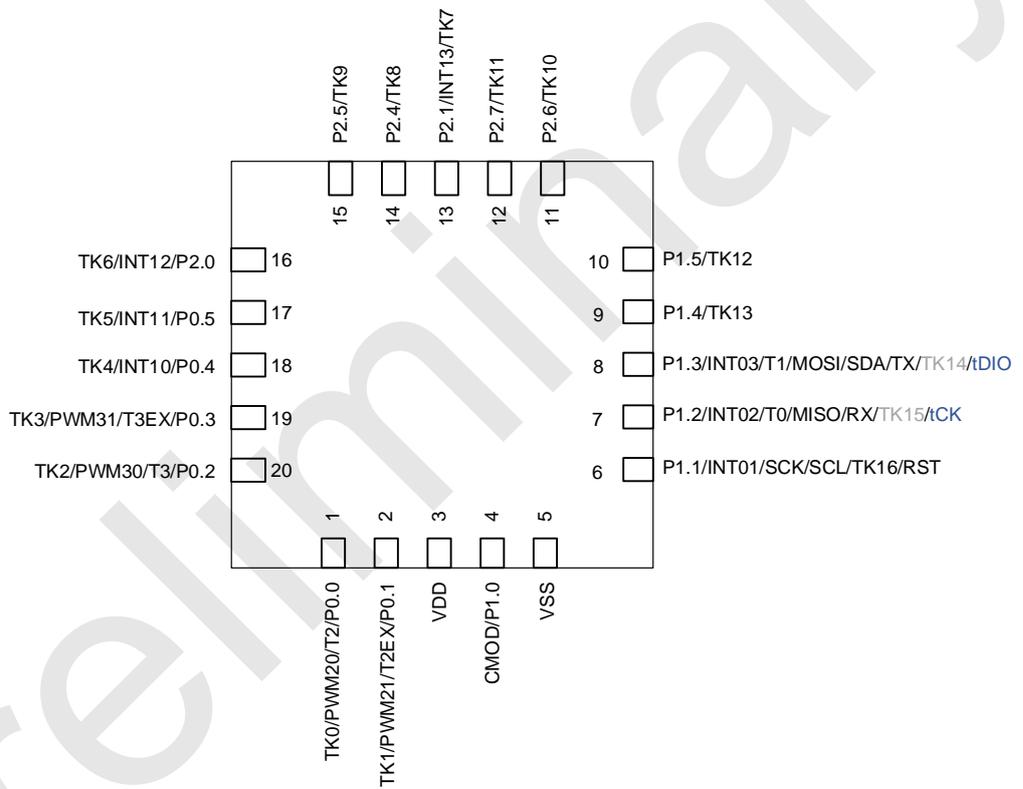
<b>10</b>	<b>定时器 TIMER0/1 .....</b>	<b>40</b>
10.1	T0 和 T1 相关特殊功能寄存器 .....	40
10.2	T0 工作模式 .....	42
10.2.1	工作模式 0: 13 位计数器/定时器 .....	42
10.2.2	工作模式 1: 16 位计数器/定时器 .....	42
10.2.3	工作模式 2: 8 位自动重载计数器/定时器 .....	43
10.2.4	工作模式 3: 两个 8 位计数器/定时器(仅限于定时器 0) .....	43
10.3	T1 工作模式 .....	43
10.3.1	工作模式 0: 13 位计数器/定时器 .....	43
10.3.2	工作模式 1: 16 位计数器/定时器 .....	44
10.3.3	工作模式 2: 8 位自动重载计数器/计数器 .....	44
<b>11</b>	<b>定时器 TIMER2/3 .....</b>	<b>46</b>
11.1	T2/3 相关特殊功能寄存器 .....	46
11.2	定时器 Timer2 .....	46
11.3	定时器 Timer3 .....	48
11.4	T2/3 工作模式 .....	49
11.5	Timer2/3 工作模式说明 .....	50
11.5.1	工作模式 0: 16 位捕获 .....	50
11.5.2	工作模式 1: 16 位自动重载定时器 .....	50
11.5.3	工作模式 3: 可编程时钟输出 .....	51
<b>12</b>	<b>常规脉冲宽度调制计数器 PWM2/3 .....</b>	<b>53</b>
12.1	PWM2/3 相关寄存器 .....	53
12.2	PWM2/3 占空比变化特性 .....	55
12.3	PWM2/3 周期变化特性 .....	55
<b>13</b>	<b>GP I/O .....</b>	<b>56</b>
13.1	GPIO 结构图 .....	56
13.1.1	强推挽输出模式 .....	56
13.1.2	带上拉的输入模式 .....	56
13.1.3	高阻输入模式(Input only) .....	56
13.2	I/O 端口相关寄存器 .....	57
<b>14</b>	<b>SPI/TWI/UART 三选一串行接口 SSI .....</b>	<b>59</b>
14.1	SPI .....	59
14.1.1	SPI 操作相关寄存器 .....	59
14.1.2	信号描述 .....	60
14.1.3	工作模式 .....	61

14.1.4	传送形式 .....	62
14.1.5	出错检测 .....	63
<b>14.2</b>	<b>TWI.....</b>	<b>63</b>
14.2.1	信号描述 .....	64
14.2.2	工作模式 .....	64
14.2.3	操作步骤 .....	66
<b>14.3</b>	<b>UART.....</b>	<b>66</b>
<b>15</b>	<b>高灵敏度触控电路 .....</b>	<b>68</b>
15.1	触控电路的耗电模式 .....	68
<b>16</b>	<b>CHECK SUM 模块 .....</b>	<b>69</b>
16.1	Check Sum 校验操作相关寄存器.....	69
<b>17</b>	<b>电气特性 .....</b>	<b>70</b>
17.1	极限参数 .....	70
17.2	推荐工作条件 .....	70
17.3	FLASH ROM 参数 .....	70
17.4	EEPROM 参数.....	70
17.5	直流电气特性 .....	71
17.6	IO 电气特性.....	71
17.7	交流电气特性 .....	72
17.7.1	32 MHZ HRC 频率误差曲线图.....	73
<b>18</b>	<b>应用电路 .....</b>	<b>74</b>
<b>19</b>	<b>订购信息 .....</b>	<b>75</b>
<b>20</b>	<b>封装信息 .....</b>	<b>76</b>
<b>21</b>	<b>规格更改记录 .....</b>	<b>82</b>
<b>22</b>	<b>声明.....</b>	<b>83</b>

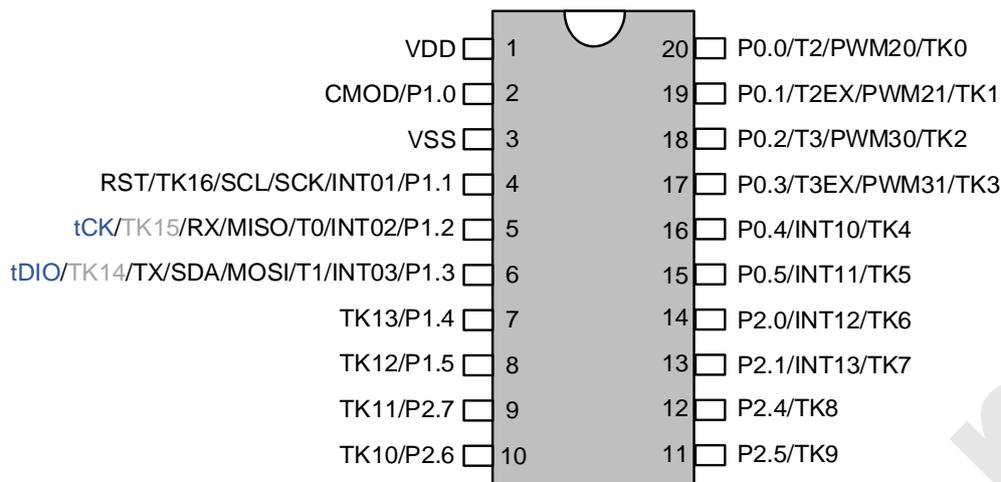
### 3 管脚定义

#### 3.1 管脚配置

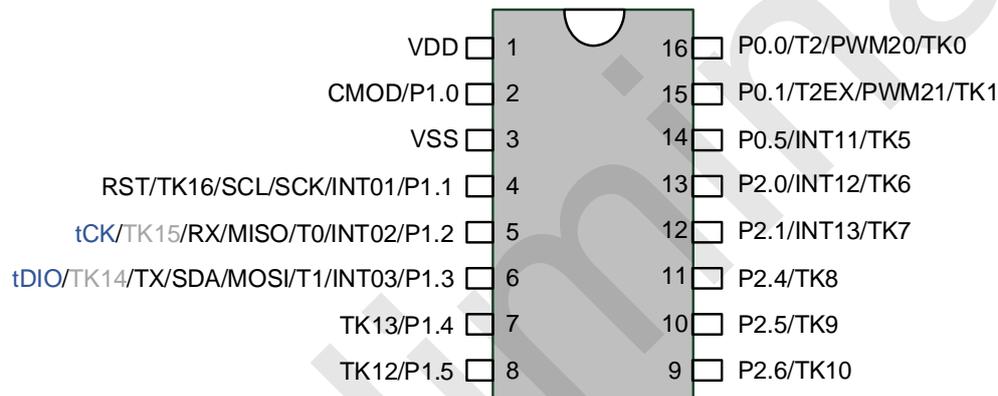
特别说明：SC92F637X 的 TK14/TK15 与 TK 调试通信口复用，若需使用 TK 调试功能，请尽量避免使用 TK14/TK15!



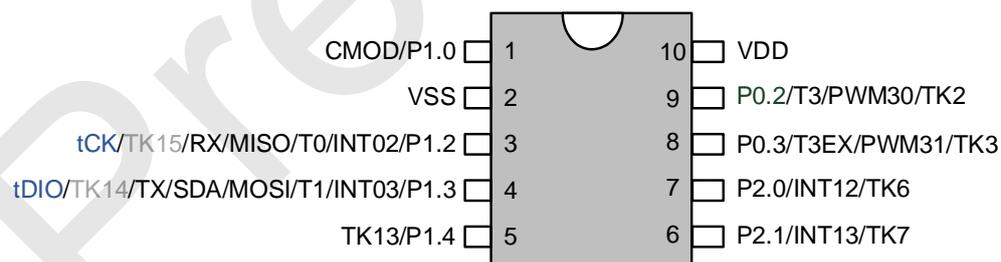
QFN20 管脚配置图



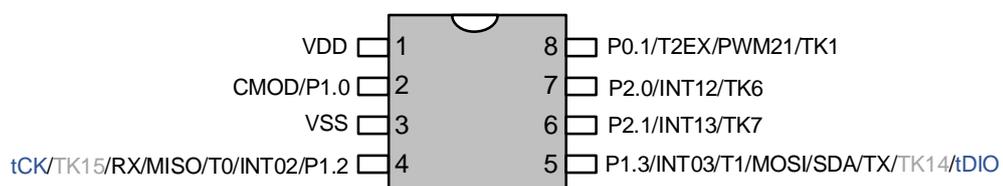
SOP20/TSSOP20 管脚配置图



SOP16 管脚配置图



MSOP10 管脚配置图



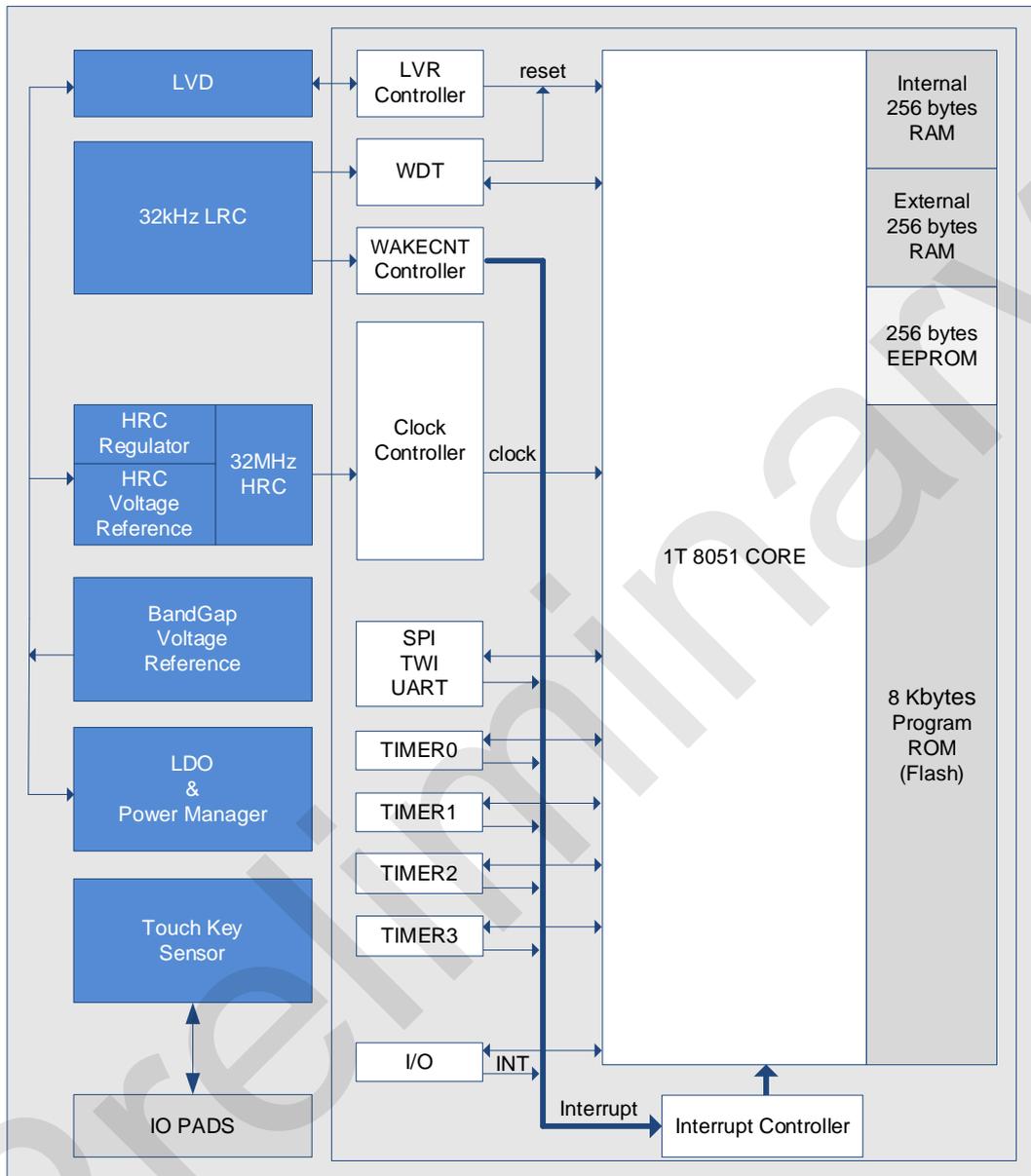
SOP8 管脚配置图

### 3.2 管脚定义

管脚编号					管脚名称	管脚类型	功能说明
QFN20	SOP20/ TSSOP20	SOP16	MSOP10	SOP20			
3	1	1	10	1	VDD	Power	电源
4	2	2	1	2	P1.0/CMOD	I/O	P1.0: GPIO P1.0 CMOD: Touch Key 触控外接电容
5	3	3	2	3	VSS	Power	接地
6	4	4	-	-	P1.1/INT01/SCK/SCL/TK16/RST	I/O	P1.1: GPIO P1.1 INT01: 外部中断 0 的输入 1 SCK: SPI 的 SCK SCL: TWI 的 SCL TK16: TK 的通道 16 RST: 外部复位引脚
7	5	5	3	4	P1.2/INT02/T0/MISO/RX/TK15/iCK	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 MISO: SPI 主输入从输出 RX: UART 接收 TK15: TK 的通道 15, 若需使用 TK 调试功能, 请尽量避免使用此 TK 通道! iCK: 烧录和仿真口时钟线
8	6	6	4	5	P1.3/INT03/T1/MOSI/TX/SDA/TK14/iDIO	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 MOSI: SPI 主输出从输入 TX: UART 发送 SDA: TWI 的 SDA TK14: TK 的通道 14, 若需使用 TK 调试功能, 请尽量避免使用此 TK 通道! iDIO: 烧录和仿真口数据线
9	7	7	5	-	P1.4/TK13	I/O	P1.4: GPIO P1.4 TK13: TK 的通道 13
10	8	8	-	-	P1.5/TK12	I/O	P1.5: GPIO P1.5 TK12: TK 的通道 12
12	9	-	-	-	P2.7/TK11	I/O	P2.7: GPIO P2.7 TK11: TK 的通道 11
11	10	9	-	-	P2.6/TK10	I/O	P2.6: GPIO P2.6 TK10: TK 的通道 10
15	11	10	-	-	P2.5/TK9	I/O	P2.5: GPIO P2.5 TK9: TK 的通道 9
14	12	11	-	-	P2.4/TK8	I/O	P2.4: GPIO P2.4 TK8: TK 的通道 8
13	13	12	6	6	P2.1/INT13/TK7	I/O	P2.1: GPIO P2.1 INT13: 外部中断 1 的输入

							3 TK7: TK 的通道 7
16	14	13	7	7	P2.0/INT12/TK6	I/O	P2.0: GPIO P2.0 INT12: 外部中断 1 的输入 2 TK6: TK 的通道 6
17	15	14	-	-	P0.5/INT11/TK5	I/O	P0.5: GPIO P0.5 INT11: 外部中断 1 的输入 1 TK5: TK 的通道 5
18	16	-	-	-	P0.4/INT10/TK4	I/O	P0.4: GPIO P0.4 INT10: 外部中断 1 的输入 0 TK4: TK 的通道 4
19	17	-	8	-	P0.3/T3EX/PWM31/TK3	I/O	P0.3: GPIO P0.3 T3EX: 定时器 3 外部捕获信号输入 PWM31: PWM31 输出口 TK3: TK 的通道 3
20	18	-	9	-	P0.2/T3/PWM30/TK2	I/O	P0.2: GPIO P0.4 T3: 计数器 3 外部输入 PWM30: PWM30 输出口 TK2: TK 的通道 2
2	19	15	-	8	P0.1/T2EX/PWM21/TK1	I/O	P0.1: GPIO P0.1 T2EX: 定时器 2 外部捕获信号输入 PWM21: PWM21 输出口 TK1: TK 的通道 1
1	20	16	-	-	P0.0/T2/PWM20/TK0	I/O	P0.0: GPIO P0.0 T2: 计数器 2 外部输入 PWM20: PWM20 输出口 TK0: TK 的通道 0

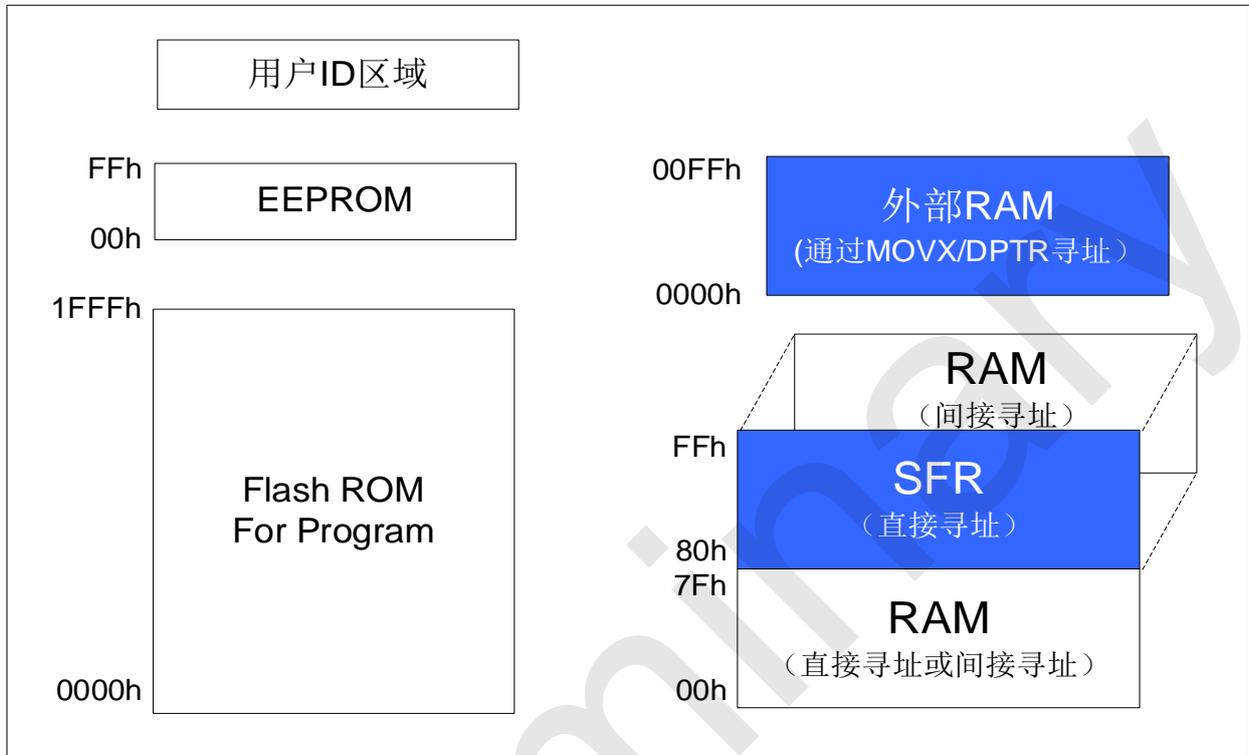
## 4 内部框图



SC92F637X BLOCK DIAGRAM

## 5 Flash ROM 和 SRAM 结构

SC92F637X 的 Flash ROM 和 SRAM 结构如下：



Flash ROM 和 SRAM 结构图

### 5.1 FLASH ROM

SC92F637X 有 8 Kbytes 的 Flash ROM，ROM 地址为 0000H~1FFFFH。可通过赛元提供的专用 ICP 烧写器（SC LINK PRO）进行编程、擦除和仿真。此 8 Kbytes Flash ROM 特性如下：

- 可通过赛元提供的专用 ICP 烧写器进行：查空（BLANK）、编程（PROGRAM）、校验（VERIFY）和擦除（ERASE）功能；
- 可重复烧写 1000 次，10 年以上保存寿命；

### 5.2 256 bytes EEPROM

SC92F637X 的 EEPROM 为独立于 8 Kbytes Flash ROM 之外的一块区间，其地址为 00H~FFH，可通过赛元提供的专用 ICP 烧写器（SC LINK PRO）进行编程及擦除，也可在程序中对其进行单 byte 读写操作，具体操作方法参考 [EEPROM 区域及用户 ID 区域读写操作](#)。此 256 bytes EEPROM 特性如下：

- EEPROM 擦写次数为 1 万次，用户擦写不要超过 EEPROM 的额定烧写次数，否则会出现异常！
- 105°C 环境下数据可保存 10 年以上；
- 用户在对 EEPROM 进行写操作的时，需保证 V<sub>DD</sub> 电压范围在 2.5V~5.5V 之间。

### 5.3 用户 ID 区域

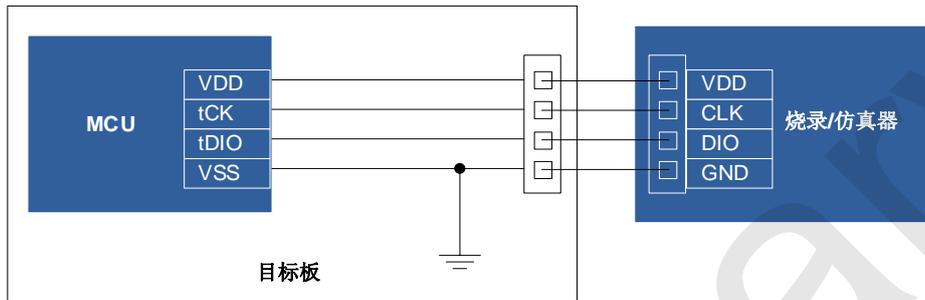
用户 ID 区域，出厂时写入用户定制 ID，用户可对其进行读操作，具体操作方法参考 [EEPROM 区域及用户 ID 区域读写操作](#)，但禁止对用户 ID 区域进行写操作。

## 5.4 烧写和仿真

SC92F637X 的烧写仿真接口 **tDIO**、**tCK** 为 JTAG 烧写仿真信号，在 SC92F637X 上的管脚定义如下：

- P1.3/**tDIO**、P1.2/**tCK**

上电初始，用户可通过 **tDIO**、**tCK**、VDD 和 VSS 四线对 SC92F637X 的 Flash ROM 区域或 EEPROM 区域进行烧写，也可对 SC92F637X 的 Flash ROM 区域进行仿真，JTAG 烧写仿真连接示意图如下：



JTAG 模式烧写仿真连接示意图

用户可通过 Customer Option 中的 DISJTG 项，选择 **tDIO**、**tCK** 所在的端口为 JTAG 专用模式或常规模式，相关 Customer Option 寄存器如下：

### 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	-	-	-	DISJTG	-	-	-	-
读/写	-	-	-	读/写	-	-	-	-
上电初始值	x	x	x	n	x	x	x	x

位编号	位符号	说明
4	DISJTG	IO/JTAG 口切换控制 0：JTAG 模式使能，P1.2、P1.3 只能作为 <b>tCK</b> / <b>tDIO</b> 使用 1：常规模式（Normal），JTAG 功能无效

### 5.4.1 JTAG 专用模式

DISJTG=0，JTAG 功能开启，**tDIO**、**tCK** 所在的端口与之复用的其它功能不可用，此模式一般用于在线调试阶段，方便用户仿真调试，JTAG 专用模式生效后，芯片无需重新上下电即可直接进入烧录或仿真模式。

### 5.4.2 常规模式 ( JTAG 专用口无效 )

DISJTG=1，**tDIO**、**tCK** 所在的端口为常规模式，JTAG 功能不可用，与之复用的其它功能可正常使用。此模式可防止烧录口占用 MCU 管脚，方便用户最大化利用 MCU 资源。

**注意：**当 JTAG 专用口无效的配置设定成功后，芯片必须彻底下电再重新上电后才能进入烧录或仿真模式，这样就会影响到带电模式下的烧录和仿真。赛元建议用户在量产烧录时选择 JTAG 专用口无效的配置，在研发调试阶段选择 JTAG 模式。

## 5.5 EEPROM 区域及用户 ID 区域读写操作

### 5.5.1 EEPROM 读写操作相关寄存器

EEPROM 区域及用户 ID 区域相关寄存器说明:

符号	地址	说明	7	6	5	4	3	2	1	0	Reset 值
IAPKEY	F1H	EEPROM 保护寄存器	IAPKEY[7:0]								0000000b
IAPAD	F2H	EEPROM 写入地址寄存器	IAPADR[7:0]								0000000b
IAPADE	F4H	EEPROM 写入扩展地址寄存器	IAPADER[7:0]								0000000b
IAPDAT	F5H	EEPROM 数据寄存器	IAPDAT[7:0]								0000000b
IAPCTL	F6H	EEPROM 控制寄存器	-	-	-	-	PAYTIMES [1:0]	CMD	EBUSY		xxx0000b

#### IAPKEY (F1H) EEPROM 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	<b>IAPKEY[7:0]</b>	打开 EEPROM 功能及操作时限设置 写入一个非零值 n, 代表: ① 打开 EEPROM 功能; ② n 个系统时钟后如果接收不到写入命令, 则 EEPROM 功能被重新关闭。

#### IAPAD (F2H) EEPROM 写入地址寄存器(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	<b>IAPADR[7:0]</b>	EEPROM 写入地址

#### IAPADE (F4H) EEPROM 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	<b>IAPADER[7:0]</b>	IAP 扩展地址: 0x01: 针对用户 ID 区域进行读操作, 不可进行写操作 0x02: 针对 EEPROM 区域进行读写操作。注意: EEPROM 擦写次数为 1 万次, 用户擦写不要超过 EEPROM 的额定烧写次数, 否则会出现异常! 其它: 保留

#### IAPDAT (F5H) EEPROM 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	<b>IAPDAT</b>	EEPROM 写入的数据

**IAPCTL (F6H) EEPROM 控制寄存器(读/写)**

位编号	7	6	5	4	3	2	1	0
符号	-	-	-	-	PAYTIMES[1:0]		CMD	EBUSY
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	<b>PAYTIMES[1:0]</b>	EEPROM 区域 写入操作时，写入时间长度设定 00: 设定写入时间为 4ms@V <sub>DD</sub> =5V 01: 设定写入时间为 2ms@V <sub>DD</sub> =5V 10: 设定写入时间为 1ms@V <sub>DD</sub> =5V 11: 保留  说明：在进行 EEPROM 写入操作时，CPU 可正常运行 注意：用户在对 EEPROM 进行操作时，需保证 V <sub>DD</sub> 电压范围在 2.5V~5.5V 之间
1	<b>CMD</b>	EEPROM 命令使能控制位，该位只可写，不可读 1: 执行写入或擦除操作命令 0: 无操作
0	<b>EBUSY</b>	EEPROM 读写等待状态位，该位只可读 0: EEPROM 操作空闲/已完成 1: EEPROM 操作进行中

### 5.5.2 EEPROM 操作流程

SC92F637X 的 EEPROM 操作流程请参照《SC92F637X\_IAP 操作库资料包》。

#### 5.5.2.1 256 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 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 的值得到 IAP_Data
IAPADE = 0x00; //返回 ROM 区域, 防止 MOV C 操作到 EEPROM
EA = 1; //开总中断
```

##### EEPROM 写操作 C 的 Demo 程序:

用户如需进行 EEPROM 写操作，请参照《SC92F637X\_IAP 操作库资料包》

## 5.6 Customer Option 区域（用户烧写设置）

SC92F637X 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 Code 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。

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，如下表所示：

符号	地址	说明	7	6	5	4	3	2	1	0
OP_HRCR	83H@FFH	系统时钟改变寄存器	-	-	OP_HRCR[5:0]					
OP_CTM0	C1H@FFH	Customer Option 寄存器 0	ENWDT	DRV_E H	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
OP_CTM1	C2H@FFH	Customer Option 寄存器 1	-	-	-	DISJTG	-	-	-	-

### 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[5:0]					
读/写	-	-	读/写					
上电初始值	x	x	n	n	n	n	n	n

位编号	位符号	说明																				
5~0	<b>OP_HRCR[5:0]</b>	<p><b>HRC 频率改变寄存器</b></p> <p>用户可通过修改此寄存器的值实现高频振荡器频率 <math>f_{HRC}</math> 的改变，进而改变 IC 的系统时钟频率 <math>f_{SYS}</math>：</p> <ol style="list-style-type: none"> <li>OP_HRCR[5:0] 上电后的初始值 OP_HRCR[s] 是一个固定值，以确保 <math>f_{HRC}</math> 为 32 MHz，每颗 IC 的 OP_HRCR[s] 都可能会有差异</li> <li>初始值为 OP_HRCR [s] 时 IC 的系统时钟频率 <math>f_{SYS}</math> 可通过 Option 项设置为准确的 32 / 16 / 8 / 2.66 MHz，OP_HRCR[5:0] 的设置值每改变 1 则 <math>f_{SYS}</math> 频率改变约 0.50%</li> </ol> <p>OP_HRCR [5:0] 和 <math>f_{SYS}</math> 输出频率的关系如下：</p> <table border="1"> <thead> <tr> <th>OP_HRCR [5:0] 值</th> <th><math>f_{SYS}</math> 实际输出频率(32 MHz 为例)</th> </tr> </thead> <tbody> <tr> <td>OP_HRCR [s]-n</td> <td><math>320000 * (1 - 0.50% * n)</math> kHz</td> </tr> <tr> <td>...</td> <td>....</td> </tr> <tr> <td>OP_HRCR [s]-2</td> <td><math>320000 * (1 - 0.50% * 2) = 31680</math> kHz</td> </tr> <tr> <td>OP_HRCR [s]-1</td> <td><math>320000 * (1 - 0.50% * 1) = 31840</math> kHz</td> </tr> <tr> <td>OP_HRCR [s]</td> <td>320000 kHz</td> </tr> <tr> <td>OP_HRCR [s]+1</td> <td><math>320000 * (1 + 0.50% * 1) = 32160</math> kHz</td> </tr> <tr> <td>OP_HRCR [s]+2</td> <td><math>320000 * (1 + 0.50% * 2) = 32320</math> kHz</td> </tr> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>OP_HRCR [s]+n</td> <td><math>320000 * (1 + 0.50% * n)</math> kHz</td> </tr> </tbody> </table> <p>注意：</p> <ol style="list-style-type: none"> <li>IC 每次上电后 OP_HRCR[5:0] 的值都是高频振荡器频率 <math>f_{HRC}</math> 最接近 32 MHz 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 <math>f_{SYS}</math> 工作在用户需要的频率；</li> <li>为保证 IC 工作可靠，IC 最高工作频率请勿超过 32 MHz 的 10%，即 35.2 MHz；</li> <li>请用户确认 HRC 频率的改变不会影响其它功能。</li> </ol>	OP_HRCR [5:0] 值	$f_{SYS}$ 实际输出频率(32 MHz 为例)	OP_HRCR [s]-n	$320000 * (1 - 0.50% * n)$ kHz	...	....	OP_HRCR [s]-2	$320000 * (1 - 0.50% * 2) = 31680$ kHz	OP_HRCR [s]-1	$320000 * (1 - 0.50% * 1) = 31840$ kHz	OP_HRCR [s]	320000 kHz	OP_HRCR [s]+1	$320000 * (1 + 0.50% * 1) = 32160$ kHz	OP_HRCR [s]+2	$320000 * (1 + 0.50% * 2) = 32320$ kHz	...	...	OP_HRCR [s]+n	$320000 * (1 + 0.50% * n)$ kHz
OP_HRCR [5:0] 值	$f_{SYS}$ 实际输出频率(32 MHz 为例)																					
OP_HRCR [s]-n	$320000 * (1 - 0.50% * n)$ kHz																					
...	....																					
OP_HRCR [s]-2	$320000 * (1 - 0.50% * 2) = 31680$ kHz																					
OP_HRCR [s]-1	$320000 * (1 - 0.50% * 1) = 31840$ kHz																					
OP_HRCR [s]	320000 kHz																					
OP_HRCR [s]+1	$320000 * (1 + 0.50% * 1) = 32160$ kHz																					
OP_HRCR [s]+2	$320000 * (1 + 0.50% * 2) = 32320$ kHz																					
...	...																					
OP_HRCR [s]+n	$320000 * (1 + 0.50% * n)$ kHz																					
7~6	-	保留																				

### 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	DRV_EH	SCLKS[1:0]		DISRST	DISLVR	LVRs[1:0]	
读/写	读/写	读/写	读/写		读/写	读/写	读/写	
上电初始值	n	n	n		n	n	n	

位编号	位符号	说明
7	<b>ENWDT</b>	WDT 开关 1: WDT 开始工作 0: WDT 关闭
6	<b>DRV_EH</b>	P22~P26 所在端口灌电流 I <sub>OL</sub> 驱动能力选择: 0: P20~P21, P24~P26, P05 对应的端口, 其灌电流 I <sub>OL</sub> 始终为常规驱动能力。 1: P20~P21, P24~P26, P05 对应的端口, 其灌电流 I <sub>OL</sub> 始终为强驱动能力。
5~4	<b>SCLKS[1:0]</b>	系统时钟频率选择: 00: 系统时钟频率为高频振荡器频率除以 1; 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	<b>DISRST</b>	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	<b>DISLVR</b>	LVR 使能设置 0: LVR 正常使用 1: LVR 无效
1~0	<b>LVRs[1:0]</b>	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	-	-	-	DISJTG	-	-	-	-
读/写	-	-	-	读/写	-	-	-	-
上电初始值	x	x	x	n	x	x	x	x

位编号	位符号	说明
4	<b>DISJTG</b>	IO/JTAG 口切换控制 0: JTAG 模式使能, P1.2、P1.3 只能作为 tCK/tDIO 使用 1: 常规模式 (Normal), JTAG 功能无效

**5.6.1 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]	0000000b
OPREG	FFH	Option 寄存器	OPREG[7:0]	nnnnnnnb

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 = 0x04; //对 OPREG 寄存器写入 0x04 (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,#04H ;对 OPREG 寄存器写入 0x04 (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！**

## 5.7 SRAM

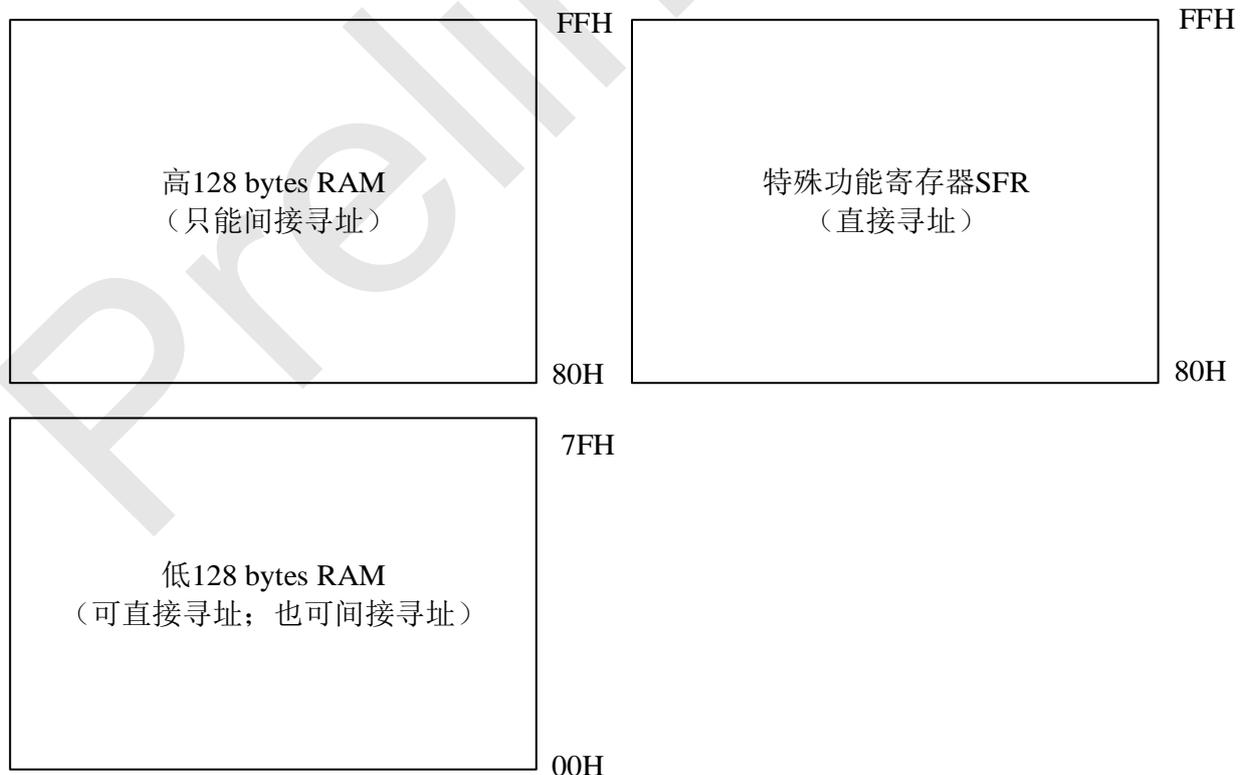
SC92F637X 单片机内部集成了 512 bytes 的 SRAM，分为内部 256 bytes RAM 和外部 256 bytes RAM。内部 RAM 的地址范围为 00H~FFH，其中高 128 bytes（地址 80H~FFH）只能间接寻址，低 128 bytes（地址 00H~7FH）可直接寻址也可间接寻址。

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。

外部 RAM 的地址为 00H~FFH，但需通过 MOVX 指令来寻址。

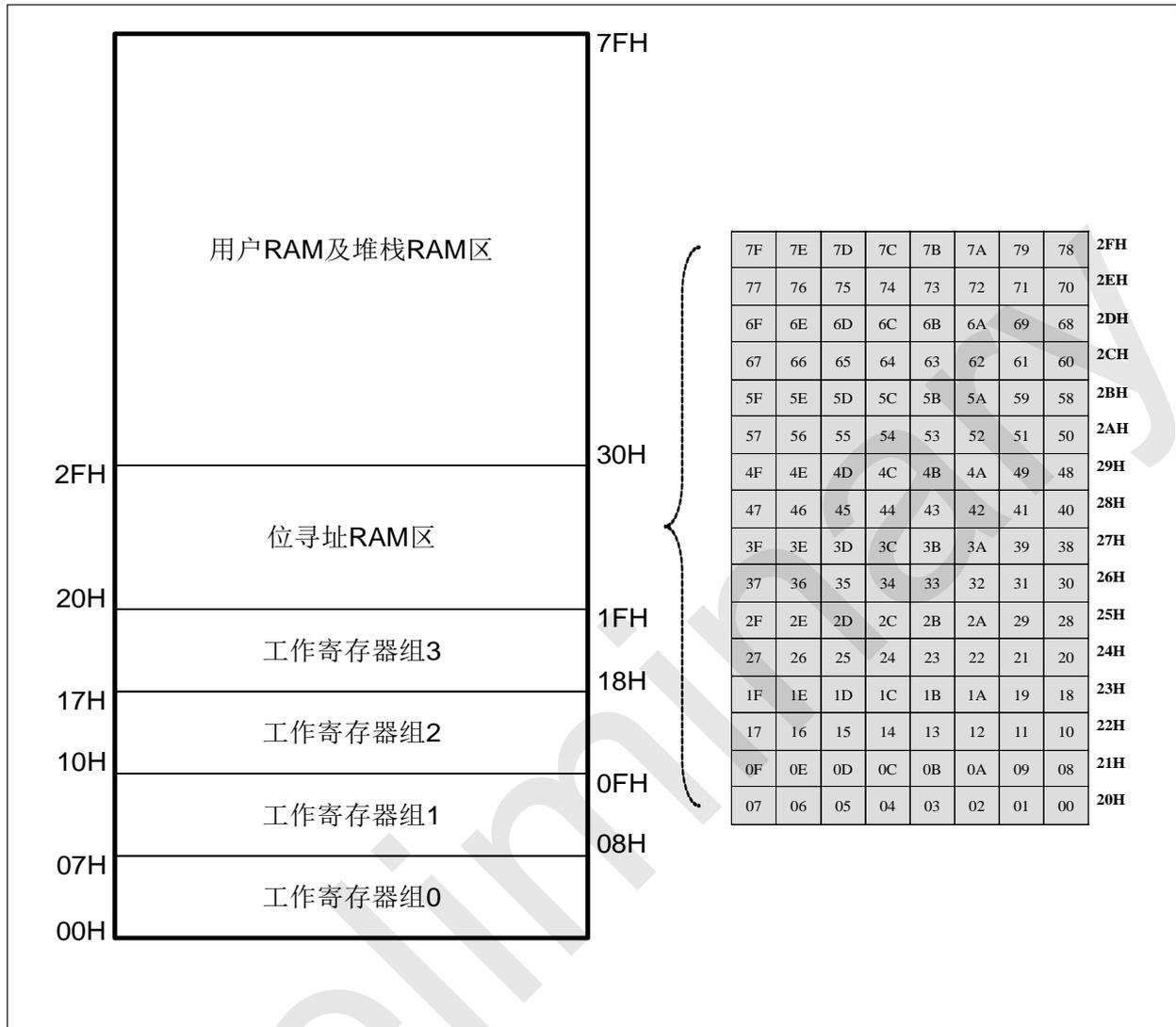
### 5.7.1 内部 256 BYTES SRAM

内部低 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，SC92F637X 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。



内部 256 bytes RAM 结构图

内部低 128 bytes RAM 结构如下:



SRAM 结构图

### 5.7.2 外部 256 BYTES SRAM

可通过 MOVX @DPTR, A 来访问外部 256 字节 RAM; 也可以使用 MOVX A, @Ri 或 MOVX @Ri, A 来访问外部 256 字节 RAM, Ri 寄存器存放外部 SRAM 的 8 位地址。

## 6 特殊功能寄存器(SFR)

### 6.1 SFR 映像

SC92F637X 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。

SC92F637X 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	-	-	-	BTMCON	CHKSUML	CHKSUMH	OPINX	OPREG
F0h	B	IAPKEY	IAPAD	-	IAPADE	IAPDAT	IAPCTL	-
E8h	-	-	-	-	-	-	-	OPERCON
E0h	ACC	-	-	-	-	-	-	-
D8h	-	-	-	-	-	-	-	-
D0h	PSW	-	-	-	-	-	-	-
C8h	TXCON	TXMOD	RCAPXL	RCAPXH	TLX	THX	TXINX	WDTCON
C0h	-	-	-	-	-	-	-	-
B8h	IP	IP1	INT0F	INT0R	INT1F	INT1R	-	-
B0h	-	-	-	-	-	-	-	-
A8h	IE	IE1	-	-	-	-	-	-
A0h	P2	P2CON	P2PH	-	-	-	-	-
98h	-	-	P0CON	P0PH	-	SSCON0	SSCON1	SSDAT
90h	P1	P1CON	P1PH	-	-	SSCON2	-	-
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。
2. SFR 中的 F1H~FFH 为系统配置使用的特殊功能寄存器，用户使用可能会导致系统异常，用户在初始化系统时，不能对这些寄存器进行清零或其它操作。

## 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	-	-	P05	P04	P03	P02	P01	P00	xx00000b
SP	81H	堆栈指针	SP[7:0]								0000111b
DPL	82H	DPTR 数据指针低位	DPL[7:0]								0000000b
DPH	83H	DPTR 数据指针高位	DPH[7:0]								0000000b
PCON	87H	电源管理控制寄存器	-	-	-	-	RST	-	STOP	IDL	xxxx0x00b
TCON	88H	定时器控制寄存器	TF1	TR1	TF0	TR0	IE1	-	IE0	-	0000x0xb
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]								0000000b
TL1	8BH	定时器 1 低 8 位	TL1[7:0]								0000000b
TH0	8CH	定时器 0 高 8 位	TH0[7:0]								0000000b
TH1	8DH	定时器 1 高 8 位	TH1[7:0]								0000000b
TMCON	8EH	定时器频率控制寄存器	-	-	-	-	-	-	T1FD	T0FD	xxxxxx00b
OTCON	8FH	输出控制寄存器	SSMOD[1:0]		-	-	-	-	-	-	00xxxxxb
P1	90H	P1 口数据寄存器	-	-	P15	P14	P13	P12	P11	P10	xx00000b
P1CON	91H	P1 口输入/输出控制寄存器	-	-	P1C5	P1C4	P1C3	P1C2	P1C1	P1C0	xx00000b
P1PH	92H	P1 口上拉电阻控制寄存器	-	-	P1H5	P1H4	P1H3	P1H2	P1H1	P1H0	xx00000b
SSCON2	95H	SSI 控制寄存器 2	SSCON2[7:0]								0000000b
P0CON	9AH	P0 口输入/输出控制寄存器	-	-	P0C5	P0C4	P0C3	P0C2	P0C1	P0C0	xx00000b
P0PH	9BH	P0 口上拉电阻控制寄存器	-	-	P0H5	P0H4	P0H3	P0H2	P0H1	P0H0	xx00000b
SSCON0	9DH	SSI 控制寄存器 0	SSCON0[7:0]								0000000b
SSCON1	9EH	SSI 控制寄存器 1	SSCON1[7:0]								0000000b
SSDAT	9FH	SSI 数据寄存器	SSD[7:0]								0000000b
P2	A0H	P2 口数据寄存器	P27	P26	P25	P24	-	-	P21	P20	0000xx00b
P2CON	A1H	P2 口输入/输出控制寄存器	P2C7	P2C6	P2C5	P2C4	-	-	P2C1	P2C0	0000xx00b
P2PH	A2H	P2 口上拉电阻控制寄存器	P2H7	P2H6	P2H5	P2H4	-	-	P2H1	P2H0	0000xx00b
IE	A8H	中断使能寄存器	EA	-	ET2	-	ET1	EINT1	ET0	EINT0	0x0x0000b
IE1	A9H	中断使能寄存器 1	-	ET3	-	ETK	-	EBTM	-	ESSI	x0x000x0b
IP	B8H	中断优先级控制寄存器	-	-	IPT2	-	IPT1	IPINT1	IPT0	IPINT0	xx0x0000b
IP1	B9H	中断优先级控制寄存器 1	-	IPT3	-	IPTK	-	IPBTM	-	IPSSI	x0x000x0b
INT0F	BAH	INT0 下降沿中断控制寄存器	-	-	-	-	INT0F3	INT0F2	INT0F1	-	xxxx000xb
INT0R	BBH	INT0 上升沿中断控制寄存器	-	-	-	-	INT0R3	INT0R2	INT0R1	-	xxxx000xb
INT1F	BCH	INT1 下降沿中断控制寄存器	-	-	-	-	INT1F3	INT1F2	INT1F1	INT1F0	xxxx0000b
INT1R	BDH	INT1 上升沿中断控制寄存器	-	-	-	-	INT1R3	INT1R2	INT1R1	INT1R0	xxxx0000b
TXCON	C8H	定时器 2/3 控制寄存器	TFX	EXFX	-	-	EXENX	TRX	C/TX	CP/RLX	00xx0000b
TXMOD	C9H	定时器 2/3 工作模式寄存器	TXFD	-	EPWMN1	EPWMN0	INVN1	INVN0	TXOE	DCXEN	0x000000b
RCAPXL	CAH	定时器 2/3 重载低 8 位	RCAPXL[7:0]								0000000b
RCAPXH	CBH	定时器 2/3 重载高 8 位	RCAPXH[7:0]								0000000b
TLX	CCH	定时器 2/3 低 8 位	TLX[7:0]								0000000b
THX	CDH	定时器 2/3 高 8 位	THX[7:0]								0000000b
TXINX	CEH	定时器 2/3 控制寄存器指针	-	-	-	-	-	TXINX[2:0]		-	xxxxx010b
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
ACC	E0H	累加器	ACC[7:0]								00000000b
OPERCON	EFH	运算控制寄存器	-	-	-	-	-	-	-	CHKSUMS	xxxxxxx0b
B	F0H	B 寄存器	B[7:0]								00000000b
IAPKEY	F1H	EEPROM 保护寄存器	IAPKEY[7:0]								00000000b
IAPAD	F2H	EEPROM 写入地址寄存器	IAPADR[7:0]								00000000b
IAPADE	F4H	EEPROM 写入扩展地址寄存器	IAPADER[7:0]								00000000b
IAPDAT	F5H	EEPROM 数据寄存器	IAPDAT[7:0]								00000000b
IAPCTL	F6H	EEPROM 控制寄存器	-	-	-	-	PAYTIMES[1:0]		CMD	EBUSY	xxxx0000b
BTMCON	FBH	低频定时器控制寄存器	ENBTM	BTMIF	-	-	BTMFS[3:0]			-	00xx0000b
CHKSUML	FCH	Check Sum 结果寄存器低位	CHKSUML[7:0]								00000000b
CHKSUMH	FDH	Check Sum 结果寄存器高位	CHKSUMH[7:0]								00000000b
OPINX	FEH	Option 指针	OPINX[7:0]								00000000b

OPREG	FFH	Option 寄存器	OPREG[7:0]	nnnnnnnb
-------	-----	------------	------------	----------

### 6.2.1 11 BYTES 寄存器扩展 RAM

RAM 分区	地址	寄存器名	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
PWM20~PWM31 Duty 8 bytes	100H	PWM2 DUTY	PDT20[15:8]							
	101H		PDT20[7:0]							
	102H		PDT21[15:8]							
	103H		PDT21[7:0]							
	104H	PWM3 DUTY	PDT30[15:8]							
	105H		PDT30[7:0]							
	106H		PDT31[15:8]							
	107H		PDT31[7:0]							
端口映射寄存器 3 bytes	108H	SPOS_TX	-	-	-	TXSEL[4:0]				
	109H	SPOS_RX	-	-	-	RXSEL[4:0]				
	10AH	SPOS_SCK				SCKSEL[4:0]				

#### 6.2.1.1 PWM2/3 占空比调节寄存器（读/写）

地址	7	6	5	4	3	2	1	0	上电初始值
100H	PDT20[15:8]								0000000b
101H	PDT20[7:0]								0000000b
102H	PDT21[15:8]								0000000b
103H	PDT21[7:0]								0000000b
104H	PDT30[15:8]								0000000b
105H	PDT30[7:0]								0000000b
106H	PDT31[15:8]								0000000b
107H	PDT31[7:0]								0000000b

#### 6.2.1.2 端口映射寄存器（读/写）

SC92F637X 的 SSICK、SSITX 和 SSIRX 端口支持全管脚映射，用户可通过以下寄存器设置映射端口位置。

注意：以下三个地址实际有效位仅 5bits，为 bit4~0:

符号	地址	功能	7	6	5	4	3	2	1	0	初始值
SPOS_SSITX	108H	TX/SDA/MOSI 映射寄存器	-	-	-	TXSEL[4:0]				xxx01110b	
SPOS_SSIRX	109H	RX/MISO 映射寄存器	-	-	-	RXSEL[4:0]				xxx01111b	
SPOS_SISCK	10AH	SCK/SCL 映射寄存器				SCKSEL[4:0]				xxx1000b	

#### 6.2.1.3 信号映射设置

信号映射端口 SSICK、SSITX 和 SSIRX 在三选一串口模块选择 SPI/TWI/UART 时对应的信号口功能如下：

- 选择 SPI 功能：SSICK 对应 SCK，SSITX 对应 MOSI，SSIRX 对应 MISO
- 选择 TWI 功能：SSICK 对应 SCL，SSITX 对应 SDA
- 选择 UART 功能：SSITX 对应 TX，SSIRX 对应 RX

若用户需要设置映射管脚，可通过对相应端口的映射寄存器写入值实现。如若要将 SCL 端口映射到管脚 P0.1，需要先对应 RAM 中 SCK/SCL 映射寄存器的地址定义一个变量，如 SPOS\_SISCK，之后对变量 SPOS\_SISCK 写入值 0x01 即可实现映射。具体映射管脚及其对应设定值如下表所示，其中包含各端口默认映射管脚信息：

序号	设定值	管脚映射	默认端口-SSI
0	00000	P0.0	-
1	00001	P0.1	-
2	00010	P0.2	-
3	00011	P0.3	-
4	00100	P0.4	-
5	00101	P0.5	-
6	00110	P2.0	-
7	00111	P2.1	-
8	01000	P2.4	-
9	01001	P2.5	-
10	01010	P2.6	-
11	01011	P2.7	-
12	01100	P1.5	-
13	01101	P1.4	-

14	01110	P1.3	TX/SDA/MOSI
15	01111	P1.2	RX/MISO
16	10000	P1.1	SCK/SCL
17	10001	P1.0	-
X	其它	保留	-

## 6.3 8051 CPU 内核常用特殊功能寄存器介绍

### 6.3.1 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。

### 6.3.2 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。

### 6.3.3 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。

### 6.3.4 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。

### 6.3.5 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

位编号	位符号	说明															
7	CY	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															
6	AC	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															
5	F0	用户标志位															
4~3	RS1、RS0	工作寄存器组选择位: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>RS1</th> <th>RS0</th> <th>当前使用的工作寄存器组 0~3</th> </tr> </thead> <tbody> <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> </tbody> </table>	RS1	RS0	当前使用的工作寄存器组 0~3	0	0	组 0 (00H~07H)	0	1	组 1 (08H~0FH)	1	0	组 2 (10H~17H)	1	1	组 3 (18H~1FH)
RS1	RS0	当前使用的工作寄存器组 0~3															
0	0	组 0 (00H~07H)															
0	1	组 1 (08H~0FH)															
1	0	组 2 (10H~17H)															
1	1	组 3 (18H~1FH)															
2	OV	溢出标志位															
1	F1	F1 标志 用户自定义标志															
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)
--	--	---

### 6.3.6 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

SC92F637X 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。

### 7.2 上电复位过程

SC92F637X 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 SC92F637X 会一直处于复位的情况，直到供应给 SC92F637X 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 SC92F637X 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

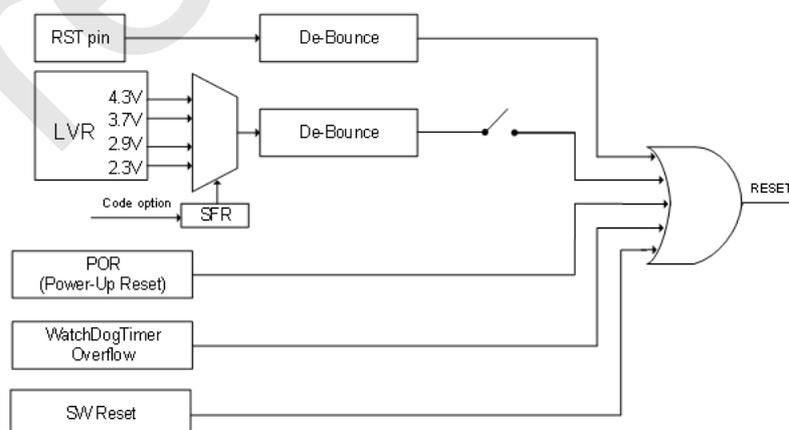
结束调入信息阶段后，SC92F637X 开始从 Flash 中读取指令代码即进入正常操作阶段。这时的 LVR 电压值是用户写入 Code Option 的设置值。

### 7.3 复位方式

SC92F637X 有 5 种复位方式，前四种为硬件复位：

1. 外部 RST 复位
2. 低电压复位 LVR
3. 上电复位 POR
4. 看门狗 WDT 复位
5. 软件复位

SC92F637X 的复位部分电路结构图如下：



SC92F637X 复位电路图

#### 7.3.1 外部 RST 复位

外部 RST 复位就是从外部 RST 给 SC92F637X 一定宽度的复位脉冲信号，来实现 SC92F637X 的复位。

用户在烧录程序前可通过烧录上位机软件配置 Customer Option 项将 P1.1 管脚配置为 RST（复位脚）使用。

### 7.3.2 低电压复位 LVR

SC92F637X 内建了一个低电压复位电路。而复位的门限电压内部有 4 种选择：4.3V、3.7V、2.9V、2.3V，缺省值 Default 是用户写入的 Option 值。当 VDD 电压小于低电压复位的门限电压，且持续时间大于  $T_{LVR}$  时，会产生复位。其中， $T_{LVR}$  是 LVR 的去抖时间，约 30 $\mu$ s。

#### 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	DRV_EH	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
读/写	读/写	读/写	读/写		读/写	读/写	读/写	
上电初始值	n	n	n		n	n	n	

位编号	位符号	说明
2	<b>DISLVR</b>	LVR 使能设置 0: LVR 正常使用 1: LVR 无效
1~0	<b>LVRS[1:0]</b>	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

### 7.3.3 上电复位 POR

SC92F637X 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

SC92F637X 有一个 WDT，其时钟源为内部的 32kHz 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	DRV_EH	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
读/写	读/写	读/写	读/写		读/写	读/写	读/写	
上电初始值	n	n	n		n	n	n	

位编号	位符号	说明
7	<b>ENWDT</b>	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) 看门狗控制寄存器(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	<b>CLRWDT</b>	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 软件复位

**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	-	-	-	-	RST	-	STOP	IDL
读/写	-	-	-	-	只写	-	只写	只写
上电初始值	x	x	x	x	n	x	0	0

位编号	位符号	说明
3	RST	软件 reset 控制位： 写状态： 0: 程序正常运行； 1: 此位被写“1”后 CPU 立刻 reset

### 7.3.6 复位初始状态

当 SC92F637X 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、软件复位等）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。

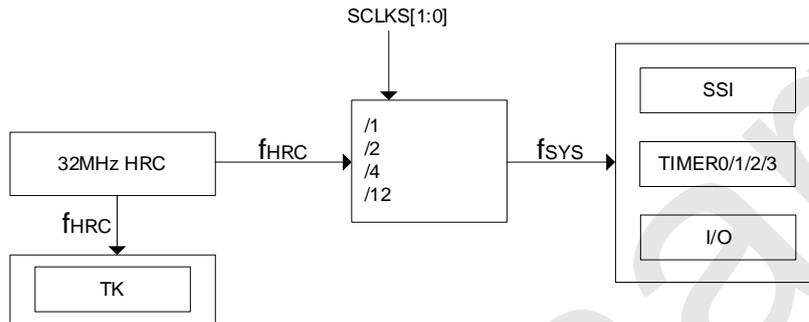
SFR 寄存器的上电复位初始值，详情见：[6.2 SFR 说明](#)。

## 7.4 高频系统时钟电路

SC92F637X 内建了一个振荡频率可调的高精度 HRC 作为系统时钟。HRC 出厂时被精确地调校至 32MHz@5V/25°C，用户可以通过编程器的 Code Option 将系统时钟设置为 32 / 16 / 8 / 2.66 MHz 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些的漂移：

- 对于压漂（2.4V~5.5V）以及(-40°C~85°C)的温漂一般状况会在 ±1% 以内。
- 对于压漂（2.4V~5.5V）以及（-40°C~105°C）的温漂一般状况会在 ±1.5% 以内



SC92F637X 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	DRV_EH	SCLKS[1:0]		DISRST	DISLVR	LVRS[1:0]	
读/写	读/写	读/写	读/写		读/写	读/写	读/写	
上电初始值	n	n	n		n	n	n	

位编号	位符号	说明
5~4	<b>SCLKS[1:0]</b>	系统时钟频率选择： 00：系统时钟频率为高频振荡器频率除以 1； 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。

SC92F637X 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[5:0]					
读/写	-	-	读/写					
上电初始值	x	x	n	n	n	n	n	n

位编号	位符号	说明
5~0	<b>OP_HRCR[5:0]</b>	<b>HRC 频率改变寄存器</b> 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 $f_{SYS}$ ： 3. OP_HRCR[5:0] 上电后的初始值 OP_HRCR[s] 是一个固定值，以确保 $f_{HRC}$ 为 32 MHz，每颗 IC 的 OP_HRCR[s] 都可能会有差异 4. 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 32 / 16 / 8 / 2.66 MHz，OP_HRCR[5:0] 的设置值

		<p>每改变 1 则 <math>f_{sys}</math> 频率改变约 0.50%</p> <p>OP_HRCR [5:0]和 <math>f_{sys}</math> 输出频率的关系如下:</p> <table border="1"> <tr> <td>OP_HRCR [5:0]值</td> <td><math>f_{sys}</math> 实际输出频率(32 MHz 为例)</td> </tr> <tr> <td>OP_HRCR [s]-n</td> <td><math>320000*(1-0.50\%*n)</math>kHz</td> </tr> <tr> <td>...</td> <td>....</td> </tr> <tr> <td>OP_HRCR [s]-2</td> <td><math>320000*(1-0.50\%*2) = 31680</math>kHz</td> </tr> <tr> <td>OP_HRCR [s]-1</td> <td><math>320000*(1-0.50\%*1) = 31840</math>kHz</td> </tr> <tr> <td>OP_HRCR [s]</td> <td>320000kHz</td> </tr> <tr> <td>OP_HRCR [s]+1</td> <td><math>320000*(1+0.50\%*1) = 32160</math>kHz</td> </tr> <tr> <td>OP_HRCR [s]+2</td> <td><math>320000*(1+0.50\%*2) = 32320</math>kHz</td> </tr> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>OP_HRCR [s]+n</td> <td><math>320000*(1+0.50\%*n)</math>kHz</td> </tr> </table> <p>注意:</p> <ol style="list-style-type: none"> <li>1. IC 每次上电后 OP_HRCR[5:0]的值都是高频振荡器频率 <math>f_{HRC}</math> 最接近 32 MHz 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 <math>f_{sys}</math> 工作在用户需要的频率;</li> <li>2. 为保证 IC 工作可靠, IC 最高工作频率请勿超过 32 MHz 的 10%, 即 35.2 MHz;</li> <li>3. 请用用户确认 HRC 频率的改变不会影响其它功能。</li> </ol>	OP_HRCR [5:0]值	$f_{sys}$ 实际输出频率(32 MHz 为例)	OP_HRCR [s]-n	$320000*(1-0.50\%*n)$ kHz	...	....	OP_HRCR [s]-2	$320000*(1-0.50\%*2) = 31680$ kHz	OP_HRCR [s]-1	$320000*(1-0.50\%*1) = 31840$ kHz	OP_HRCR [s]	320000kHz	OP_HRCR [s]+1	$320000*(1+0.50\%*1) = 32160$ kHz	OP_HRCR [s]+2	$320000*(1+0.50\%*2) = 32320$ kHz	...	...	OP_HRCR [s]+n	$320000*(1+0.50\%*n)$ kHz
OP_HRCR [5:0]值	$f_{sys}$ 实际输出频率(32 MHz 为例)																					
OP_HRCR [s]-n	$320000*(1-0.50\%*n)$ kHz																					
...	....																					
OP_HRCR [s]-2	$320000*(1-0.50\%*2) = 31680$ kHz																					
OP_HRCR [s]-1	$320000*(1-0.50\%*1) = 31840$ kHz																					
OP_HRCR [s]	320000kHz																					
OP_HRCR [s]+1	$320000*(1+0.50\%*1) = 32160$ kHz																					
OP_HRCR [s]+2	$320000*(1+0.50\%*2) = 32320$ kHz																					
...	...																					
OP_HRCR [s]+n	$320000*(1+0.50\%*n)$ kHz																					
7~6	-	保留																				

## 7.5 低频振荡器及低频时钟定时器

SC92F637X 内建一个频率为 32kHz 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 32kHz 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 (FBH) 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	<b>ENBTM</b>	低频 Base Timer 启动控制 0: Base Timer 不启动 1: Base Timer 启动
6	<b>BTMIF</b>	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	<b>BTMFS [3:0]</b>	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 1001: 每 8.0 秒产生一个中断

		1010: 每 16.0 秒产生一个中断 1011: 每 32.0 秒产生一个中断 1100~1111: 保留
5~4	-	保留

## 7.6 STOP 模式和 IDLE 模式

SC92F637X 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、INT1、低频时钟中断和 TK 中断把 SC92F637X 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	-	-	-	-	RST	-	STOP	IDL
读/写	-	-	-	-	只写	-	只写	只写
上电初始值	x	x	x	x	0	x	0	0

位编号	位符号	说明
1	<b>STOP</b>	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	<b>IDL</b>	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_();
_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  
.....

Preliminary

## 8 中央处理单元 CPU 及指令系统

### 8.1 CPU

SC92F637X 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

SC92F637X 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寻址，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

寄存器寻址是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，指令操作数指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。)

Preliminary

## 9 INTERRUPT 中断

SC92F637X 单片机提供 9 个中断源：Timer0, Timer1, Timer2, Timer3, INT0, INT1, SSI, Base Timer, TK。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。两组外部中断 INT0 和 INT1 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，总的中断使能位 EA 可以实现所有中断的打开或者关闭。

### 9.1 中断源、向量

SC92F637X 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	IE0	EINT0	IPINT0	0003H	1 (高)	0	H/W Auto	能
Timer0	Timer0 溢出	TF0	ET0	IPT0	000BH	2	1	H/W Auto	不能
INT1	外部中断 1 条件符合	IE1	EINT1	IPINT1	0013H	3	2	H/W Auto	能
Timer1	Timer1 溢出	TF1	ET1	IPT1	001BH	4	3	H/W Auto	不能
Timer2	Timer2 溢出	TFX	ET2	IPT2	002BH	5	5	必须用户清除	不能
SSI	接收或发送完成	SPIF/TWIF	ESSI	IPSSI	003BH	6	7	必须用户清除	不能
BTM	Base timer 溢出	BTMIF	EBTM	IPBTM	004BH	7	9	H/W Auto	能
TK	Touch Key 计数器溢出	TKIF	ETK	IPTK	005BH	8	11	H/W Auto	能
Timer3	Timer3 溢出	TFX	ET3	IPT3	006BH	9	13	必须用户清除	不能

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：

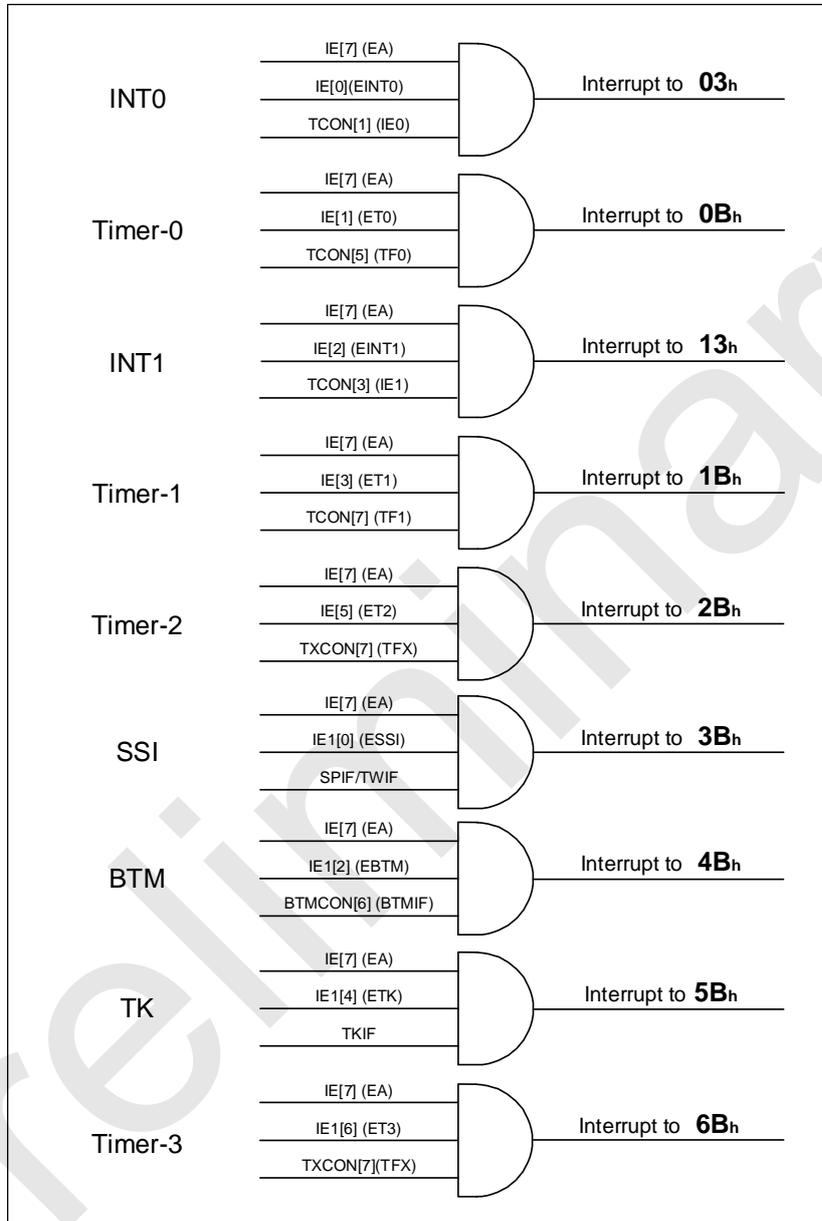
定时器中断：Timer0 和 Timer1 溢出时会产生中断并将中断标志 TF0 和 TF1 置为“1”，当单片机执行该定时器中断时，中断标志 TF0 和 TF1 会被硬件自动清“0”。Timer2/3 溢出时会产生中断并将中断标志 TF2/TF3 置为“1”，在 Timer2/3 中断发生后，硬件并不会自动清除 TF2/TF3 位，此 bit 必须由使用者的软件负责清除。

SSI 中断：当 SSI 接收或发送一帧数据完成时 SPIF/TWIF 位会被硬件自动置“1”，SSI 中断产生。当单片机执行该 SSI 中断时，中断标志 SPIF/TWIF 必须由使用者的软件负责清除。

外部中断 INT0 和 INT1：当外部中断口有中断条件发生时，外部中断就发生了。INT0 有三个外部中断源，INT1 有四个外部中断源，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 和 INT1 还可以唤醒单片机的 STOP。

## 9.2 中断结构图

SC92F637X 的中断结构如下图所示：



SC92F637X 中断结构和向量

## 9.3 中断优先级

SC92F637X 单片机的中断具有两个中断优先级，这些中断源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：SC92F637X 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	-	ET2	-	ET1	EINT1	ET0	EINT0
读/写	读/写	-	读/写	-	读/写	读/写	读/写	读/写
上电初始值	0	x	0	x	0	0	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
5	ET2	Timer2 中断使能控制 0: 关闭 TIMER2 中断 1: 允许 TIMER2 中断
3	ET1	Timer1 中断使能控制 0: 关闭 TIMER1 中断 1: 允许 TIMER1 中断
2	EINT1	外部中断 1 使能控制 0: 关闭 INT1 中断 1: 打开 INT1 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 INTO 中断 1: 打开 INTO 中断
6,4	-	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	-	-	IPT2	-	IPT1	IPINT1	IPT0	IPINT0
读/写	-	-	读/写	-	读/写	读/写	读/写	读/写
上电初始值	x	x	0	x	0	0	0	0

位编号	位符号	说明
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

3	<b>IPT1</b>	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
2	<b>IPINT1</b>	INT1 计数器中断优先权选择 0: INT1 中断优先权为低 1: INT1 中断优先权为高
1	<b>IPT0</b>	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	<b>IPINT0</b>	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
7~6,4	-	保留

**IE1 (A9H) 中断控制寄存器 1(读/写)**

位编号	7	6	5	4	3	2	1	0
符号	-	ET3	-	ETK	-	EBTM	-	ESSI
读/写	-	读/写	-	读写	-	读/写	-	读/写
上电初始值	x	0	x	0	x	0	x	0

位编号	位符号	说明
6	<b>ET3</b>	Timer3 中断使能控制 0: 关闭 Timer3 中断 1: 允许 Timer3 中断
4	<b>ETK</b>	Touch Key 中断使能控制 0: 关闭 Touch Key 中断 1: 打开 Touch Key 中断
2	<b>EBTM</b>	Base Timer 中断使能控制 0: 关闭 Base Timer 中断 1: 允许 Base Timer 中断
0	<b>ESSI</b>	三合一串口中断使能控制 0: 关闭串口中断 1: 允许串口中断
7,5,3,1	-	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	-	IPT3	-	IPTK	-	IPBTM	-	IPSSI
读/写	-	读/写	-	读/写	-	读/写	-	读/写
上电初始值	x	0	x	0	x	0	x	0

位编号	位符号	说明
6	<b>IPT3</b>	Timer3 中断优先权选择 0: Timer3 中断优先权为低 1: Timer3 中断优先权为高
4	<b>IPTK</b>	Touch Key 中断优先权选择 0: Touch Key 中断优先权为低 1: Touch Key 中断优先权为高
2	<b>IPBTM</b>	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

0	<b>IPSSI</b>	三合一串口中断优先权选择 0: SSI 中断优先权为低 1: SSI 中断优先权为高
7,5,3,1	-	保留

**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	IE1	-	IE0	-
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	x

位编号	位符号	说明
3	<b>IE1</b>	INT1 溢出中断请求标志。INT1 产生溢出，发生中断时，硬件将 IE1 置为“1”，申请中断，CPU 响应时，硬件清“0”。
1	<b>IE0</b>	INT0 溢出中断请求标志。INT0 产生溢出，发生中断时，硬件将 IE0 置为“1”，申请中断，CPU 响应时，硬件清“0”。
2,0	-	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	-
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
3~1	<b>INT0Fn</b> (n=1~3)	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
7~4,0	-	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	-
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
3~1	<b>INT0Rn</b> (n=1~3)	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
7~4,0	-	保留

**INT1F (BCH) INT1 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	-	-	-	-	INT1F3	INT1F2	INT1F1	INT1F0
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
3~0	<b>INT1Fn</b> (n=0~3)	INT1 下降沿中断控制 0: INT1n 下降沿中断关闭

		1: INT1n 下降沿中断使能
7~4	-	保留

**INT1R (BDH) INT1 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	-	-	-	-	INT1R3	INT1R2	INT1R1	INT1R0
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
3~0	<b>INT1Rn (n=0~3)</b>	INT1 上升沿中断控制 0: INT1n 上升沿中断关闭 1: INT1n 上升沿中断使能
7~4	-	保留

## 10 定时器 Timer0/1

SC92F637X 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 TCON 来选择 T0 和 T1 的计数来源是 f<sub>sys</sub>/12 或 f<sub>sys</sub> (f<sub>sys</sub> 为分频后的系统时钟)。

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	IE1	-	IE0	-	00000x0xb
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	-	-	-	-	-	-	T1FD	T0FD	xxxxxx00b

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	IE1	-	IE0	-
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	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。
2,0	-	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	<b>C/T1</b>	TMOD[6]控制定时器 1 0: 定时器, T1 计数来源于 f <sub>sys</sub> 分频 1: 计数器, T1 计数来源于外部管脚 T1/P1.3	
5~4	<b>M11,M01</b>	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 全有效 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	<b>C/T0</b>	TMOD[2]控制定时器 0 0: 定时器, T0 计数来源于 f <sub>sys</sub> 分频 1: 计数器, T0 计数来源于外部管脚 T0/P1.2	
1~0	<b>M10,M00</b>	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 全有效 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	-	-	-	-	-	-	T1FD	T0FD
读/写	-	-	-	-	-	-	读/写	读/写
上电初始值	x	x	x	x	x	x	0	0

位编号	位符号	说明
1	<b>T1FD</b>	T1 输入频率选择控制 0: T1 频率源自于 f <sub>sys</sub> /12 1: T1 频率源自于 f <sub>sys</sub>
0	<b>T0FD</b>	T0 输入频率选择控制 0: T0 频率源自于 f <sub>sys</sub> /12 1: T0 频率源自于 f <sub>sys</sub>

#### 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	-	ET2	-	ET1	EINT1	ET0	EINT0
读/写	读/写	-	读/写	-	读/写	读/写	读/写	读/写
上电初始值	0	x	0	x	0	0	0	0

位编号	位符号	说明
3	<b>ET1</b>	Timer1 中断使能控制 0: 关闭 TIMER1 中断 1: 允许 TIMER1 中断

1	<b>ET0</b>	Timer0 中断使能控制 0: 关闭 TIMERO 中断 1: 允许 TIMERO 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	-	-	IPT2	-	IPT1	IPINT1	IPT0	IPINT0
读/写	-	-	读/写	-	读/写	读/写	读/写	读/写
上电初始值	x	x	0	x	0	0	0	0

位编号	位符号	说明
3	<b>IPT1</b>	Timer1 中断优先级 0: 设定 Timer 1 的中断优先级是“低” 1: 设定 Timer 1 的中断优先级是“高”
1	<b>IPT0</b>	Timer0 中断优先级 0: 设定 Timer 0 的中断优先级是“低” 1: 设定 Timer 0 的中断优先级是“高”

## 10.2 T0 工作模式

通过对寄存器 TMOD 中的 M10、M00(TM0D[1]、TM0D[0])的设置，定时器/计数器 0 可实现 4 种不同的工作模式。

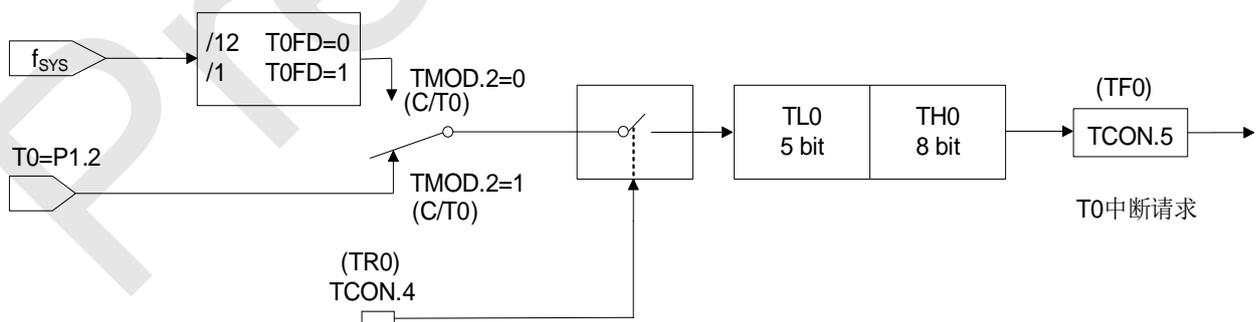
### 10.2.1 工作模式 0: 13 位计数器/定时器。

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。

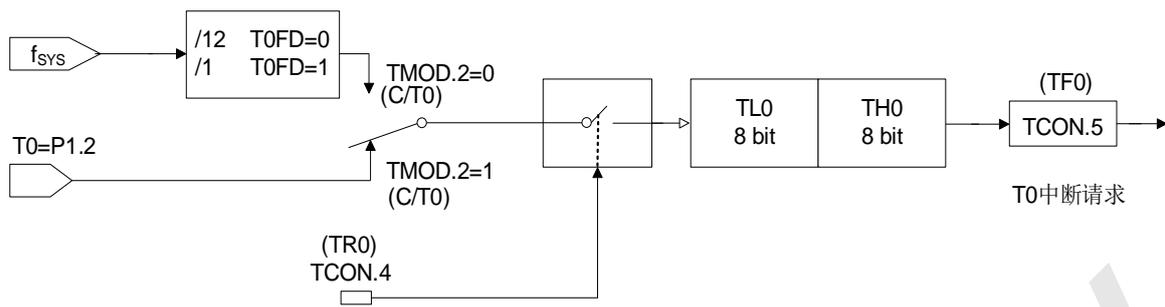
当作为定时器应用时，可配置 T0FD 来选择时钟源的分频比例。



定时器/计数器工作模式 0: 13 位定时器/计数器

### 10.2.2 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。



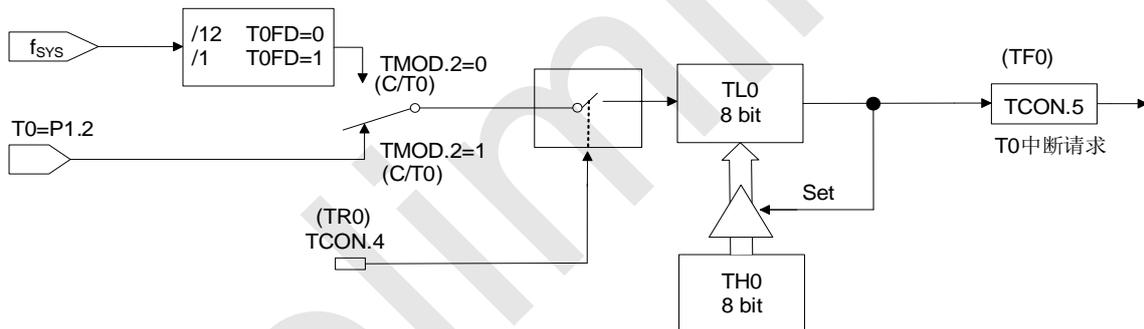
定时器/计数器工作模式 0: 16 位定时器/计数器

### 10.2.3 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

### 10.2.4 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(TM0D[5]、TM0D[4])的设置, 定时器/计数器 1 可实现 3 种不同的工作模式。

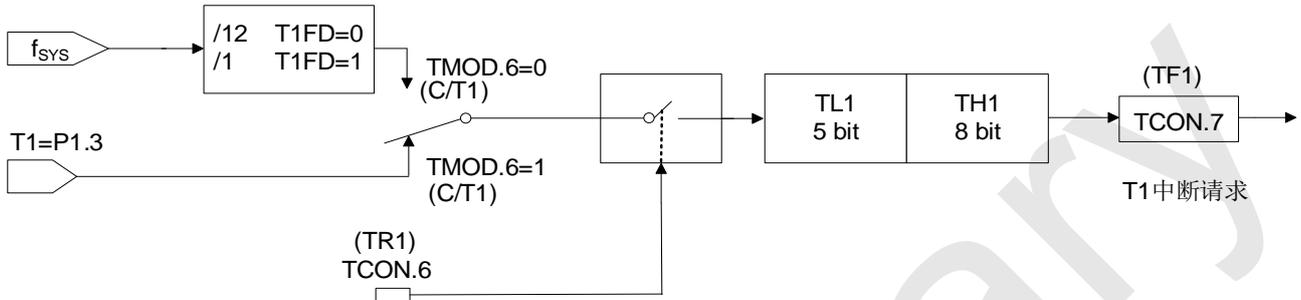
### 10.3.1 工作模式 0: 13 位计数器/定时器。

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。

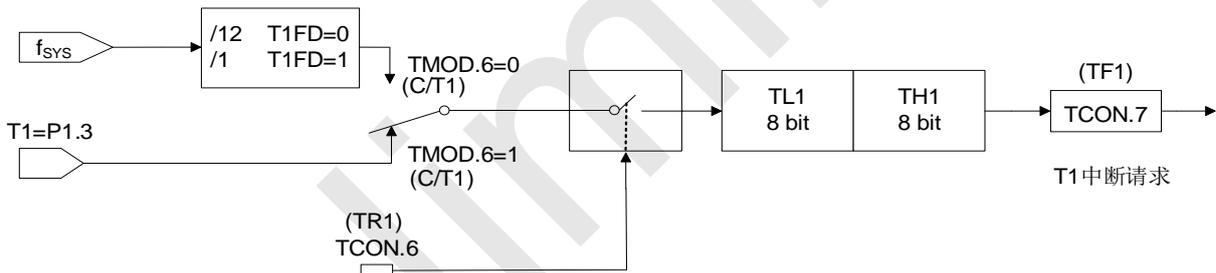
当作为定时器应用时, 可配置 T1FD 来选择时钟源的分频比例。



定时器/计数器工作模式 0: 13 位定时器/计数器

### 10.3.2 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。



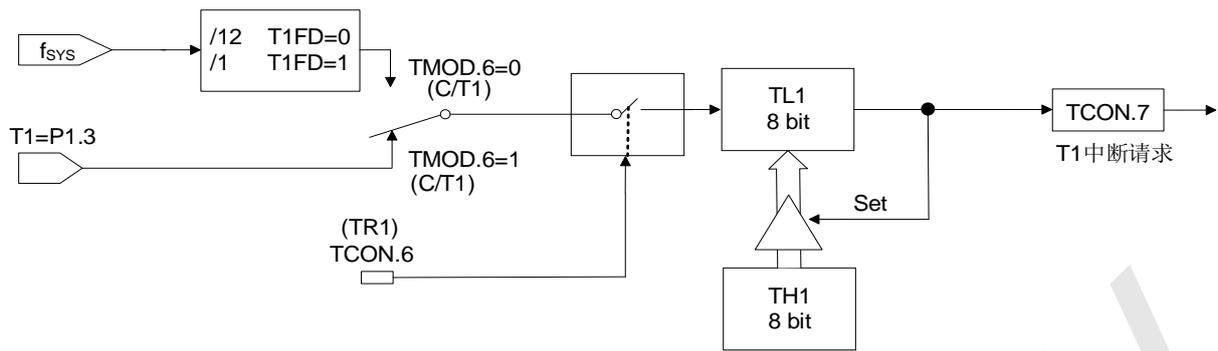
定时器/计数器工作模式 0: 16 位定时器/计数器

### 10.3.3 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 T1FD 来选择定时器时钟源被系统时钟 fsys 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 Timer2/3

SC92F637X 单片机内部的 Timer2/3 是两个独立的 Timer，其中 Timer2 有 4 种工作模式，Timer3 有 3 种工作模式。

Timer2/3 的控制寄存器共用同一组地址（C8H-CDH），用户可通过 TXINX[2:0]将 TimerX 寄存器组(TXCON / TXMOD / RCAPXL / RCAPXH / TLX / THX)指向 Timer2/3，从而实现一组寄存器配置两个独立 Timer 的功能。

**注意：**只有在 TXINX[2:0]配置成功后 TimerX 寄存器组才会指向用户指定的 Timer2/3，此时操作 TimeX 寄存器组才是对相应 Timer 的有效操作。

### 11.1 T2/3 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	
TXINX	CEH	定时器 2/3 控制寄存器指针	-	-	-	-	-	TXINX[2:0]			xxxx010b	
TXCON	C8H	定时器 2/3 控制寄存器	TFX	EXFX	-	-	EXENX	TRX	C/TX	CP/RLX	00xx0000b	
TXMOD	C9H	定时器 2/3 工作模式寄存器	TXFD	-	EPWM N1	EPWM N0	INVN1	INVN0	TXOE	DCXEN	0xxxx00b	
RCAPXL	CAH	定时器 2/3 重载低 8 位	RCAPXL[7:0]									00000000b
RCAPXH	CBH	定时器 2/3 重载高 8 位	RCAPXH[7:0]									00000000b
TLX	CCH	定时器 2/3 低 8 位	TLX[7:0]									00000000b
THX	CDH	定时器 2/3 高 8 位	THX[7:0]									00000000b
TMCON	8EH	定时器频率控制寄存器	USMDX[1:0]	-	-	-	-	-	T1FD	T0FD	00xxx00b	

#### TXINX (CEH) 定时器 2/3 控制寄存器指针(读/写)

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

位编号	位符号	说明
2~0	<b>TXINX[2:0]</b>	定时器 2/3 控制寄存器指针 010: TimerX 寄存器组: TXCON / TXMOD / RCAPXL / RCAPXH / TLX / THX 指向 T2 011: TimerX 寄存器组指向 T3 其他: 保留
7~3	-	保留

### 11.2 定时器 Timer2

SC92F637X 单片机内部的 Timer2 具有计数方式和定时方式两种工作模式。特殊功能寄存器 TXCON 中有一个控制位 C/TX 来选择 T2 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。TRX 是 T2 在定时器/计数器模式计数的开关控制，只有在 TR2=1 的时候，T2 才会被打开计数。

计数器模式下，T2 管脚上的每一个脉冲，T2 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 TXMOD.7(TXFD)来选择 T2 的计数来源是 fsys/12 或 fsys。

定时器/计数器 T2 有 3 种工作模式：

- ① 模式 0: 16 位捕获模式
- ② 模式 1: 16 位自动重载定时器模式
- ③ 模式 3: 可编程时钟输出模式

TXINX[2:0] = 010，TimerX 寄存器组指向 Timer2，各寄存器的解释说明如下：

#### TXCON (C8H) 定时器 2 控制寄存器(读/写) (TXINX[2:0] = 010)

位编号	7	6	5	4	3	2	1	0
符号	TFX	EXFX	-	-	EXENX	TRX	C/TX	CP/RLX
读/写	读/写	读/写	-	-	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	<b>TFX</b>	定时器 2 溢出标志位 0: 无溢出(必须由软件清 0) 1: 溢出
6	<b>EXFX</b>	T2EX 引脚外部事件输入(下降沿)被检测到的标志位 0: 无外部事件输入(必须由软件清 0) 1: 检测到外部输入(如果 EXENX = 1, 由硬件设 1)
3	<b>EXENX</b>	T2EX 引脚上的外部事件输入(下降沿)用作重载/捕获触发器允许/禁止控制: 0: 忽略 T2EX 引脚上的事件 1: 检测到 T2EX 引脚上一个下降沿, 产生一个捕获或重载
2	<b>TRX</b>	定时器 2 开始/停止控制位 0: 停止定时器 2/停止 PWM2 计数器 1: 开始定时器 2/开启 PWM2 计数器
1	<b>C/TX</b>	定时器 2 定时器/计数器方式选定位 2 0: 定时器方式, T2 引脚用作 I/O 端口 1: 计数器方式
0	<b>CP/RLX</b>	捕获/重载方式选定位 0: 16 位带重载功能的定时器/计数器 1: 16 位带捕获功能的定时器/计数器, T2EX 为定时器 2 外部捕获信号输入口

**TXMOD (C9H) 定时器 2 工作模式寄存器(读/写) (TXINX[2:0] = 010)**

位编号	7	6	5	4	3	2	1	0
符号	TXFD	-	EPWM21	EPWM20	INV21	INV20	TXOE	DCEN
读/写	读/写	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	x	0	0	0	0	0	0

位编号	位符号	说明
7	<b>TXFD</b>	T2 输入频率选择控制 0: T2 频率源自于 f <sub>sys</sub> /12 1: T2 频率源自于 f <sub>sys</sub>
1	<b>TXOE</b>	定时器 2 输出允许位 0: 设置 T2 作为时钟输入或 I/O 端口 1: 设置 T2 作为时钟输出
0	<b>DCEN</b>	递减计数允许位 0: 禁止定时器 2 作为递增/递减计数器, 定时器 2 仅作为递增计数器 1: 允许定时器 2 作为递增/递减计数器
6	-	保留

**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	-	ET2	-	ET1	EINT1	ET0	EINT0
读/写	读/写	-	读/写	-	读/写	读/写	读/写	读/写
上电初始值	0	x	0	x	0	0	0	0

位编号	位符号	说明
5	<b>ET2</b>	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	-	-	IPT2	-	IPT1	IPINT1	IPT0	IPINT0
读/写	-	-	读/写	-	读/写	读/写	读/写	读/写
上电初始值	x	x	0	x	0	0	0	0

位编号	位符号	说明
5	<b>IPT2</b>	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”

### 11.3 定时器 Timer3

SC92F637X 单片机内部的 Timer3 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。TRX 是 T3 计数的开关控制，只有在 TRX=1 的时候，T3 才会被打开计数。

定时器模式下，可通过特殊功能寄存器 TXMOD.7(TXFD)来选择 T3 的计数来源是 fsys/12 或 fsys。

TXINX[2:0] = 011, TimerX 寄存器组指向 Timer3, 各寄存器的解释说明如下:

**TXCON (C8H) 定时器 3 控制寄存器(读/写) (TXINX[2:0] = 011)**

位编号	7	6	5	4	3	2	1	0
符号	TFX	EXFX	-	-	EXENX	TRX	C/TX	CP/RLX
读/写	读/写	读/写	-	-	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	<b>TFX</b>	定时器 3 溢出标志位 0: 无溢出(必须由软件清 0) 1: 溢出(由硬件设 1)
6	<b>EXFX</b>	T3EX 引脚外部事件输入(下降沿)被检测到的标志位 0: 无外部事件输入(必须由软件清 0) 1: 检测到外部输入(如果 EXENX = 1, 由硬件设 1)
3	<b>EXENX</b>	T3EX 引脚上的外部事件输入(下降沿)用作重载/捕获触发器允许/禁止控制: 0: 忽略 T3EX 引脚上的事件 1: 检测到 T3EX 引脚上一个下降沿, 产生一个捕获或重载
2	<b>TRX</b>	定时器 3 开始/停止控制位 0: 停止定时器 3/停止 PWM3 计数器 1: 开始定时器 3/开启 PWM3 计数器
1	<b>C/TX</b>	定时器 3 定时器/计数器方式选定位 0: 定时器方式, T3 引脚用作 I/O 端口 1: 计数器方式
0	<b>CP/RLX</b>	捕获/重载方式选定位 0: 16 位带重载功能的定时器/计数器 1: 16 位带捕获功能的定时器/计数器, TXEX 为定时器 3 外部捕获信号输入口
5~4	-	保留

**TXMOD (C9H) 定时器 3 工作模式寄存器(读/写) (TXINX[2:0] = 011)**

位编号	7	6	5	4	3	2	1	0
符号	TXFD	-	EPWM31	EPWM30	INV31	INV30	TXOE	DCXEN
读/写	读/写	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	x	0	0	0	0	0	0

位编号	位符号	说明
7	<b>TXFD</b>	T3 输入频率选择控制 0: T3 频率源自于 f <sub>sys</sub> /12 1: T3 频率源自于 f <sub>sys</sub>
1	<b>TXOE</b>	定时器 3 输出允许位 0: 设置 T3 作为时钟输入或 I/O 端口 1: 设置 T3 作为时钟输出
0	<b>DCXEN</b>	递减计数允许位 0: 禁止定时器 3 作为递增/递减计数器, 定时器 3 仅作为递增计数器 1: 允许定时器 3 作为递增/递减计数器, T3EX 用来选择计数方向。
6	-	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	-	ET3	-	ETK	-	EBTM	-	ESSI
读/写	-	读/写	-	读/写	-	读/写	-	读/写
上电初始值	x	0	x	0	x	0	x	0

位编号	位符号	说明
6	<b>ET3</b>	Timer3 中断使能控制 0: 关闭 Timer3 中断 1: 允许 Timer3 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	-	IPT3	-	IPTK	-	IPBTM	-	IPSSI
读/写	-	读/写	-	读/写	-	读/写	-	读/写
上电初始值	x	0	x	0	x	0	x	0

位编号	位符号	说明
6	<b>IPT3</b>	Timer3 中断优先权选择 0: Timer3 中断优先权为低 1: Timer3 中断优先权为高

## 11.4 T2/3 工作模式

定时器 Timer2/3 的工作模式如下:

- ① 模式 0: 16 位捕获
- ② 模式 1: 16 位自动重载定时器
- ③ 模式 2: 波特率发生器, 仅 Timer2 支持该模式
- ④ 模式 3: 可编程时钟输出
- ⑤ 模式 4: PWM 输出模式, 详见 [12 常规脉冲宽度调制计数器 PWM2/3](#)。

定时器 2/3 工作模式与配置方式如下表:

C/TX	TXOE	DCXEN	TRX	CP/RLX	EXENX	工作模式
X	0	X	1	1	1	模式 0 16 位捕获

X	0	0	1	0	0	模式 1	16 位自动重载定时/计数器，普通自动重载
X	0	0	1	0	1		16 位自动重载定时/计数器，带 TnEX 触发重载
X	0	1	1	0	X		16 位自动重载定时/计数器，递增或递减重载
0	1	X	1	X	X	模式 3	可编程时钟输出
X	X	X	0	X	1	X	定时器停止，TnEX(n=2~3)通路仍旧允许设置捕获/重载产生中断

## 11.5 Timer2/3 工作模式说明

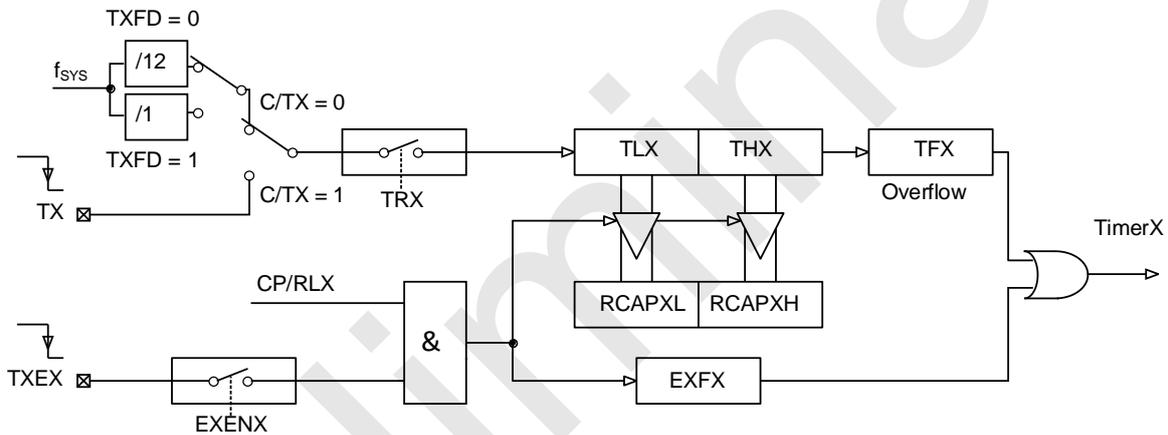
### 11.5.1 工作模式 0: 16 位捕获

配置 CP/RLX = 1，将定时器 n (n=2~3) 设置为 16 位捕获模式。

在捕获方式中，TXCON 的 EXENX 位有两个选项：

如果 EXENX = 0，定时器 n 作为 16 位定时器或计数器，如果 ETn 被允许的话，定时器 n 能设置 TFX 溢出产生一个中断。

如果 EXENX = 1，定时器 n 执行相同操作，但是在外部输入 TnEX 上的下降沿也能引起在 THX 和 TLX 中的当前值分别被捕获到 RCAPXH 和 RCAPXL 中，此外，在 TnEX 上的下降沿也能引起在 TXCON 中的 EXFX 被设置。如果 ETn 被允许，EXFX 位也像 TFX 一样也产生一个中断。



模式 0: 16 位捕获

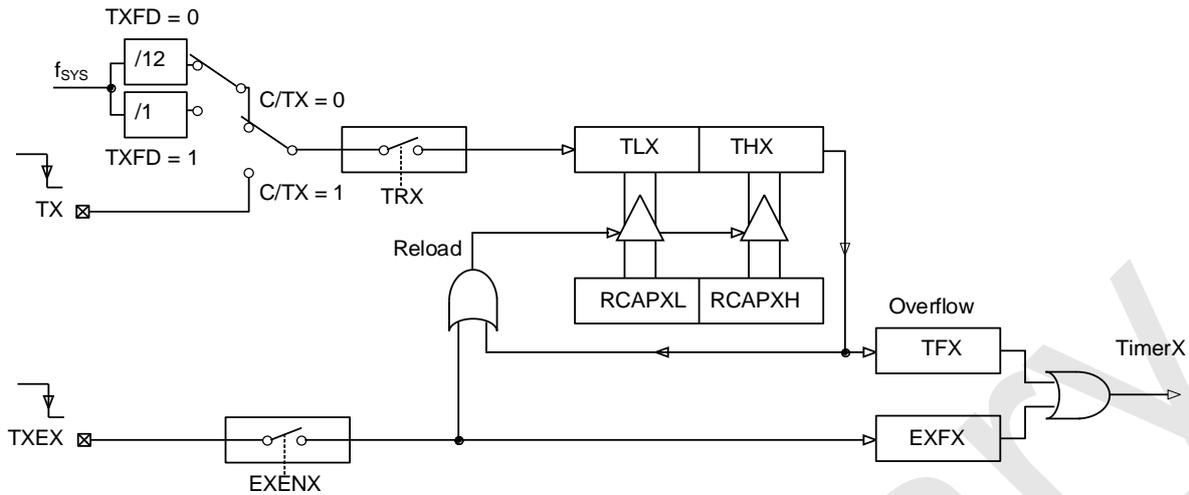
### 11.5.2 工作模式 1: 16 位自动重载定时器

在 16 位自动重载方式下，定时器 n (n=2~3) 可以被选为递增计数或递减计数。这个功能通过 TnMOD 中的 DCEN 位(递减计数允许)选择。系统复位后，DCEN 位复位值为 0，定时器 n 默认递增计数。当 DCEN 置 1 时，定时器 n 递增计数或递减计数取决于 TnEX 引脚上的电平。

当 DCEN = 0，通过在 TXCON 中的 EXENX 位选择两个选项。

如果 EXENX = 0，定时器 n 递增到 0xFFFFH，在溢出后置起 TFX 位，同时定时器自动将用户软件写好的寄存器 RCAPXH 和 RCAPXL 的 16 位值装入 THX 和 TLX 寄存器。

如果 EXENX = 1，溢出或在外输入 TnEX 上的下降沿都能触发一个 16 位重载。TnEX 上有下降沿产生时，EXFX 位置起。如果 ETn 被使能，TFX 和 EXFX 位都能产生一个中断。



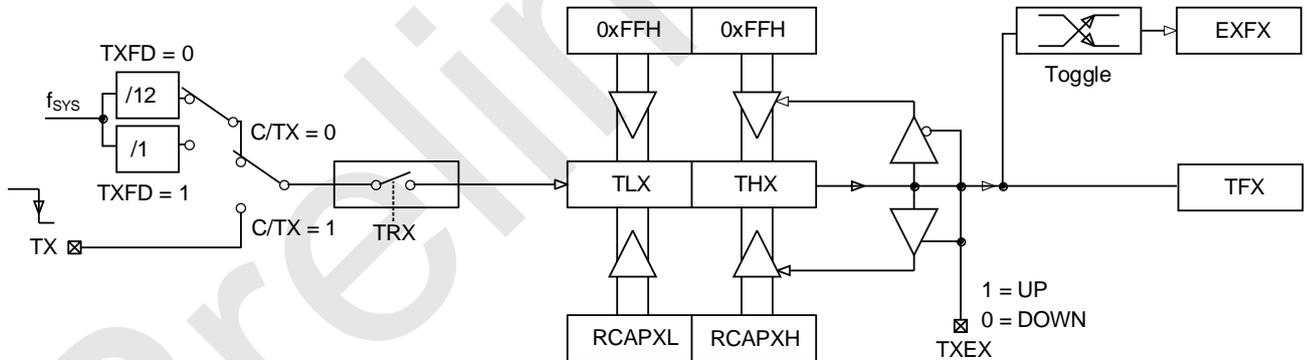
模式 1: 16 位自动重载 DCEN = 0

设置 DCEN 位允许定时器 n 递增计数或递减计数。当 DCEN = 1 时，TnEX 引脚控制计数的方向，而 EXENX 控制无效。

TnEX 置 1 可使定时器 n 递增计数。定时器向 0xFFFFH 溢出，然后设置 TFX 位。溢出也能分别引起 RCAPXH 和 RCAPXL 上的 16 位值重载入定时器寄存器。

TnEX 置 0 可使定时器 n 递减计数。当 THX 和 TLX 的值等于 RCAPXH 和 RCAPXL 的值时，定时器溢出。置起 TFX 位，同时 0xFFFFH 重载入定时器寄存器。

无论定时器 n 溢出与否，EXFX 位都被用作结果的第 17 位。在此工作模式下，EXFX 不作为中断标志。



模式 1: 16 位自动重载 DCEN = 1

### 11.5.3 工作模式 3: 可编程时钟输出

在这种方式中，定时器 n (n=2~3) 可以编程为输出 50% 的占空比时钟周期：当  $C/\overline{Tn} = 0$ ；TnOE = 1，使能定时器 n 作为时钟发生器

在这种方式中，Tn 输出占空比为 50% 的时钟

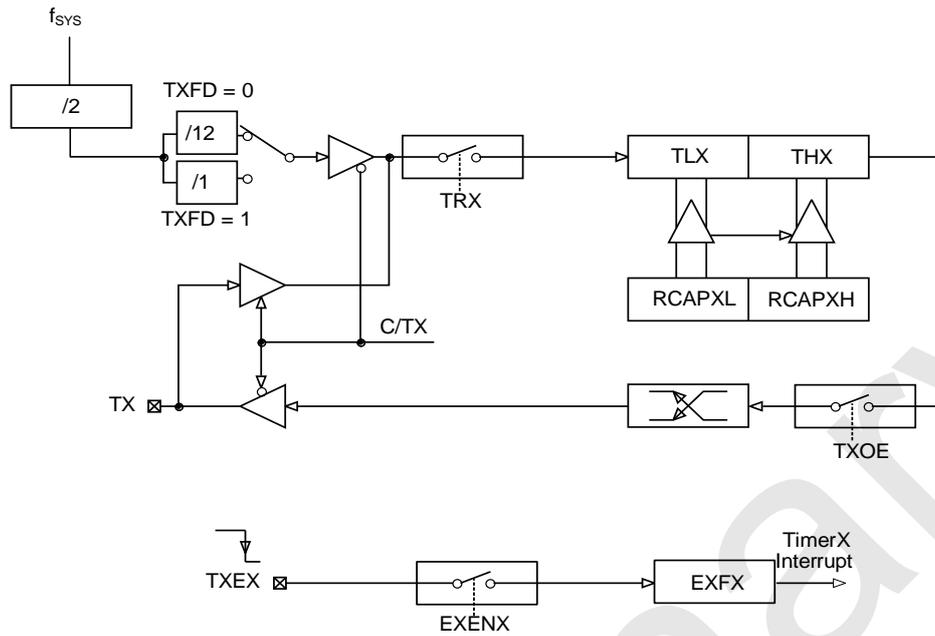
$$\text{Clock Out Frequency} = \frac{f_n}{(65536 - [RCAPXH, RCAPXL]) \times 4};$$

其中， $f_n$  为定时器 n 时钟频率：

$$f_n = \frac{f_{\text{SYS}}}{12}; \quad \text{TXFD} = 0$$

$$f_n = f_{\text{SYS}}; \quad \text{TXFD} = 1$$

定时器 n 溢出不产生中断，Tn 端口作时钟输出。



模式 3: 可编程时钟输出

**注意:**

1. TFX 和 EXFX 都能引起定时器 n (n=2~3)的中断请求，两者有相同的向量地址；
2. 当事件发生时或其它任何时间都能由软件设置 TFX 和 EXFX 为 1，只有软件以及硬件复位才能使之清 0；
3. 当 EA = 1 且 ETn = 1 时，设置 TFX 或 EXFX 为 1 能引起定时器 n 中断；
4. 当定时器 2 作为波特率发生器时，写入 THX/TLX 或 RCAPXH/RCAPXL 会影响波特率的准确性，引起通信出错。

## 12 常规脉冲宽度调制计数器 PWM2/3

SC92F637X 最多提供 4 路常规 PWM，分为两组：PWM2、PWM3。

注意：这两组 PWM 的周期寄存器分别与 Timer2，Timer3 的 TLX 和 THX 共用，因此一旦用户使用了 PWM2、PWM3 资源，就不能再更改 Timer2，Timer3 的定时/计数值，否则会导致 PWM 周期输出异常！

### 12.1 PWM2/3 相关寄存器

PWM2/3 相关寄存器如下：

#### TXINX (CEH) 定时器 2/3 控制寄存器指针(读/写)

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

位编号	位符号	说明
2~0	<b>TXINX[2:0]</b>	定时器 2/3 控制寄存器指针 010: TimerX 寄存器组: TXCON / TXMOD / RCAPXL / RCAPXH / TLX / THX 指向 PWM2 011: TimerX 寄存器组指向 PWM3 其他: 保留
7~3	-	保留

#### TXCON (C8H) 定时器 n 控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	TFX	EXFX	-	-	EXENX	TRX	C/TX	CP/RLX
读/写	读/写	读/写	-	-	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
2	<b>TRX</b>	定时器 n 开始/停止控制位 0: 停止定时器 n/停止 PWMn 计数器 1: 开始定时器 n/开启 PWMn 计数器

当 EPWMn0 或 EPWMn1 置 1 时 Timer 即可开启 PWM 模式，此时 Tn 和 TnEX (n=2~3) 无效，PWMxy (x=2~3, y=0~1) 可输出 PWM 波形。

#### TXMOD (C9H) 定时器 n 工作模式寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	TXFD	-	EPWMn1	EPWMn0	INVn1	INVn0	TXOE	DCXEN
读/写	读/写	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	x	0	0	0	0	0	0

位编号	位符号	说明
5	<b>ENPWMn1</b>	PWMn1 波形输出选择 0: PWMn1 输出被关闭 1: PWMn1 所在的 I/O 作为 PWM 波形输出口
4	<b>ENPWMn0</b>	PWMn0 波形输出选择 0: PWMn0 输出被关闭 1: PWMn0 所在的 I/O 作为 PWM 波形输出口
3	<b>INVn1</b>	PWMn1 波形输出反向控制

位编号	位符号	说明
		1: PWMn1 波形输出反向 0: PWMn1 波形输出不反向
2	<b>INVn0</b>	PWMn0 波形输出反向控制 1: PWMn0 波形输出反向 0: PWMn0 波形输出不反向

THX 和 TLX 计数器从 0 开始向上计数，当计数值与占空比设置项 PDTxy [15:0] 的值匹配时 PWM 输出波形切换高低电平，接着 THX 和 TLX 计数器继续向上计数到自动重载值 PWMPDX，然后重新从 0 开始计数并生成计数上溢事件，一个 PWM 周期结束。如果定时器中断已使能，此时会产生定时中断。

Timer 输出的 PWM 周期  $T_{PWM}$  计算公式如下：

$$T_{pwm} = \frac{PWMPDX[15:0] + 1}{f_{sys}}$$

占空比 duty 计算公式：

$$duty = \frac{PDTxy [15:0]}{PWMPDX[15:0] + 1}$$

PWM 周期通过以下寄存器设定：

#### RCAPXH (CBH)

**PWMn 周期寄存器高 8 位(读/写)**

注意：PWM2/3 的周期寄存器与 Timer2, Timer3 复用，因此，用户一旦使用了 PWM2、PWM3 资源，就不能再更改 Timer2, Timer3 的定时/计数值，否则会导致 PWM 周期输出异常！

位编号	7	6	5	4	3	2	1	0
符号	PWMPDHX[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

#### RCAPXL (CAH)

**PWMn 周期寄存器低 8 位(读/写) (TXINX[2:0] = 010)**

注意：PWM2/3 的周期寄存器与 Timer2, Timer3 复用，因此，用户一旦使用了 PWM2、PWM3 资源，就不能再更改 Timer2, Timer3 的定时/计数值，否则会导致 PWM 周期输出异常！

位编号	7	6	5	4	3	2	1	0
符号	PWMPDLX[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	<b>PWMPDX[15:0]</b>	PWMn 周期设置 此数值代表 PWMn 的输出波形的 (周期 - 1); 也就是说 PWMn 输出的周期值为 (PWMPDX[15:0] + 1) * PWM 时钟;

PWM 的 duty 通过以下寄存器设定：

#### PWM2~3 占空比调节寄存器(读/写)

地址	7	6	5	4	3	2	1	0	上电初始值
100H	PDT20[15:8]								0000000b
101H	PDT20[7:0]								0000000b
102H	PDT21[15:8]								0000000b
103H	PDT21[7:0]								0000000b
104H	PDT30[15:8]								0000000b
105H	PDT30[7:0]								0000000b
106H	PDT31[15:8]								0000000b
107H	PDT31[7:0]								0000000b

位编号	位符号	说明
7~0	PDTxy[15:0] (x=2~3, y=0~1)	PWMxy 波形占空比长度设置 PWMxy 的波形的高电平宽度为: (PDTxy[15:0] + 1)个 PWM 时钟

## 12.2 PWM2/3 占空比变化特性

当 PWM2/3 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 PDTxy (x=2~3, y=0~1) 的值实现。但需要注意:更改 PDTxy 的值，占空比不会立即改变，而是等待本周期结束，在下一个周期改变。

## 12.3 PWM2/3 周期变化特性

当 PWM2/3 输出波形时，若需改变周期，可通过改变周期设置寄存器组 TLX 和 THX 的值实现。更改周期寄存器的值，PWM 输出周期变化情况如下：

定义当前周期计数值为  $T_n$ ，写入周期寄存器时，定时器计到的值为  $T_m$ ，待更新的周期计数值为  $T_x$ ，则：

$T_m \leq T_x$ ：周期按照  $T_x$  实时改变；

$T_m > T_x$ ：此时周期变化会分为两个阶段。第一个阶段，写入周期寄存器之后，周期计数器会从当前计数值累加至溢出清零。第二个阶段，周期按照  $T_x$  改变。

## 13 GP I/O

SC92F637X 提供了最多 18 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 18 个 IO 同其他功能复用，I/O 端口在输入或输出状态下，从端口数据寄存器里读到的都是端口的实际状态值。

**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。**

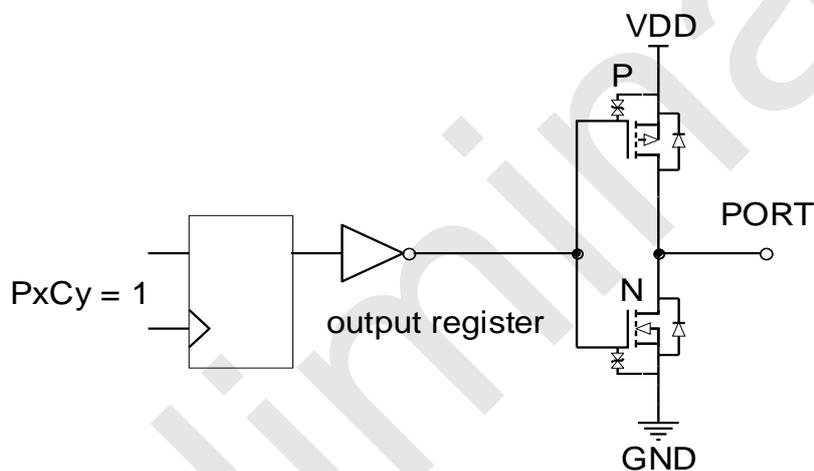
### 13.1 GPIO 结构图

#### 13.1.1 强推挽输出模式

强推挽输出模式下：

- P20 / P21 / P24 / P25 / P26 / P05 六个端口能够提供持续的大电流驱动：大于 20mA@4.3V 的输出高，大于 100mA@0.8V 的输出低。
- 除 P20 / P21 / P24 / P25 / P26 / P05 六个端口之外的所有端口能够提供持续的大电流驱动：大于 20mA@4.3 的输出高，大于 80mA@0.8V 的输出低。

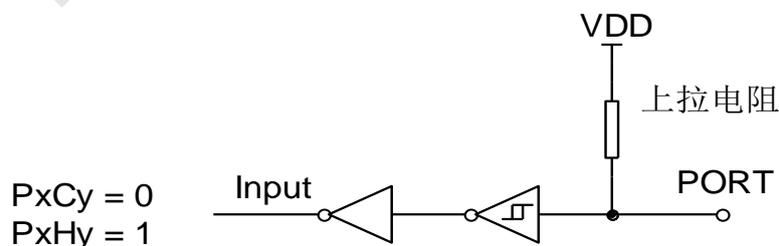
强推挽输出模式的端口结构示意图如下：



强推挽输出模式

#### 13.1.2 带上拉的输入模式

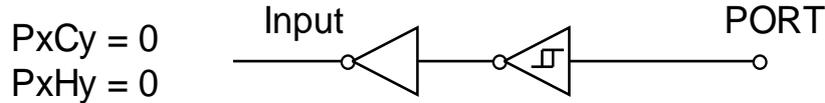
带上拉的输入模式下，输入口上恒定接一个上拉电阻，仅当输入口上电平被拉低时，才会检测到低电平信号。带上拉的输入模式的端口结构示意图如下：



带上拉的输入模式

#### 13.1.3 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	-	-	P0C5	P0C4	P0C3	P0C2	P0C1	P0C0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	x	x	0	0	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	-	-	P0H5	P0H4	P0H3	P0H2	P0H1	P0H0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	x	x	0	0	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	-	-	P1C5	P1C4	P1C3	P1C2	P1C1	P1C0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	x	x	0	0	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	-	-	P1H5	P1H4	P1H3	P1H2	P1H1	P1H0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	x	x	0	0	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	P2C5	P2C4	-	-	P2C1	P2C0
读/写	读/写	读/写	读/写	读/写	-	-	读/写	读/写
上电初始值	0	0	0	0	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	P2H5	P2H4	-	-	P2H1	P2H0
读/写	读/写	读/写	读/写	读/写	-	-	读/写	读/写
上电初始值	0	0	0	0	x	x	0	0

位编号	位符号	说明
7~0	<b>PxCy</b> (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

位编号	位符号	说明
7~0	<b>PxHy</b> (x=0~2, y=0~7)	Px 口上拉电阻设置, 仅在 PxCy=0 时有效: 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	-	-	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	x	x	0	0	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	-	-	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	x	x	0	0	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	P2.5	P2.4	-	-	P2.1	P2.0
读/写	读/写	读/写	读/写	读/写	-	-	读/写	读/写
上电初始值	0	0	0	0	x	x	0	0

位编号	位符号	说明
5~0	<b>P0.x</b> (x=0~5)	P0 口锁存寄存器数据
5~0	<b>P1.x</b> (x=0~5)	P1 口锁存寄存器数据
7~4, 1~0	<b>P2.x</b> (x=0~1, 4~7)	P2 口锁存寄存器数据

## 14 SPI/TWI/UART 三选一串行接口 SSI

SC92F637X 内部集成了三选一串行接口电路（简称 SSI），可方便 MCU 与不同接口的器件或者设备的连接。用户可通过配置寄存器 OTCON 的 SSMOD[1:0] 位将 SSI 接口配置为 SPI、TWI 和 UART 中任意一种通信模式。其特点如下：

1. SPI 模式可配置为主模式或从属模式中的一种
2. TWI 模式通信时只能做从机
3. UART 模式可工作在模式 1（10 位全双工异步通信）和模式 3（11 位全双工异步通信）

具体配置方式如下：

### OTCON (8FH) 输出控制寄存器(读/写)

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

位编号	位符号	说明
7-6	<b>SSMOD[1:0]</b>	<b>SSI 通信模式控制位</b> 00: SSI 关闭 01: SSI 设置为 SPI 通信模式; 10: SSI 设置为 TWI 通信模式; 11: SSI 设置为 UART 通信模式;

### 14.1 SPI

SSMOD[1:0] = 01，三选一串行接口 SSI 配置为 SPI 接口。串行外部设备接口(简称 SPI)是一种高速串行通信接口，允许 MCU 与外围设备(包括其它 MCU)进行全双工，同步串行通信。

#### 14.1.1 SPI 操作相关寄存器

##### SSCON0 (9DH) SPI 控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	SPEN	-	MSTR	CPOL	CPHA	SPR2	SPR1	SPR0
读/写	读/写	-	读/写	读/写	读/写	读/写	读/写	读/写
上电初始值	0	x	0	0	0	0	0	0

位编号	位符号	说明
7	<b>SPEN</b>	<b>SPI 使能控制</b> 0: 关闭 SPI 1: 打开 SPI
5	<b>MSTR</b>	<b>SPI 主从选择</b> 0: SPI 为从设备 1: SPI 为主设备
4	<b>CPOL</b>	<b>时钟极性控制位</b> 0: SCK 在空闲状态下为低电平 1: SCK 在空闲状态下为高电平
3	<b>CPHA</b>	<b>时钟相位控制位</b> 0: SCK 周期的第一沿采集数据 1: SCK 周期的第二沿采集数据
2-0	<b>SPR[2:0]</b>	<b>SPI 时钟速率选择位</b> 000: $f_{sys} / 4$ 001: $f_{sys} / 8$ 010: $f_{sys} / 16$

位编号	位符号	说明
		011: fsys /32 100: fsys /64 101: fsys /128 110: fsys /256 111: fsys /512
6	-	保留

**SSCON1 (9EH) SPI 状态寄存器(读/写)**

位编号	7	6	5	4	3	2	1	0
符号	SPIF	WCOL	-	-	TXE	DORD	-	TBIE
读/写	读/写	读/写	-	-	读/写	读/写	-	读/写
上电初始值	0	0	x	x	0	0	x	0

位编号	位符号	说明
7	<b>SPIF</b>	<b>SPI 数据传送标志位</b> 0: 由软件清 0 1: 表明已完成数据传输, 由硬件置 1
6	<b>WCOL</b>	<b>写入冲突标志位</b> 0: 由软件清 0, 表明已处理写入冲突 1: 由硬件置 1, 表明检测到一个冲突
3	<b>TXE</b>	<b>发送缓存器空标志</b> 0: 发送缓存器不空 1: 发送缓存器空, 必须由软件清零
2	<b>DORD</b>	<b>传送方向选择位</b> 0: MSB 优先发送 1: LSB 优先发送
0	<b>TBIE</b>	<b>发送缓存器中断允许控制位</b> 0: TXE=1 时, 不允许产生中断 1: TXE=1 时, 将产生 SPI 中断
5~4,1	-	保留

**SSDAT (9FH) SPI 数据寄存器(读/写)**

位编号	7	6	5	4	3	2	1	0
符号	SPD[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	<b>SPD[7:0]</b>	<b>SPI 数据缓存寄存器</b> 写入 SSDAT 的数据被放置到发送移位寄存器中。 读取 SSDAT 时将获得接收移位寄存器的数据。

**14.1.2 信号描述**
**主输出从输入(MOSI):**

该路信号连接主设备和一个从设备。数据通过 MOSI 从主设备串行传送到从设备, 主设备输出, 从设备输入。

**主输入从输出(MISO):**

该路信号连接从设备和主设备。数据通过 MISO 从从设备串行传送到主设备, 从设备输出, 主设备输入。当 SPI 配置为从设备并未被选中, 从设备的 MISO 引脚处于高阻状态。

**SPI 串行时钟(SCK):**

SCK 信号用作控制 MOSI 和 MISO 线上输入输出数据的同步移动。每 8 时钟周期线上传送一个字节。如果从设备未被选中，SCK 信号被此从设备忽略。

**14.1.3 工作模式**

SPI 可配置为主模式或从属模式中的一种。SPI 模块的配置和初始化通过设置 SSSCON0 寄存器(SPI 控制寄存器)和 SSSCON1(SPI 状态寄存器)来完成。配置完成后，通过设置 SSSCON0, SSSCON1, SSSDAT(SPI 数据寄存器)来完成数据传送。

在 SPI 通讯期间，数据同步地被串行的移进移出。串行时钟线(SCK)使两条串行数据线(MOSI 和 MISO)上数据的移动和采样保持同步。如果从设备没有被选中，则不能参与 SPI 总线上的活动。

当 SPI 主设备通过 MOSI 线传送数据到从设备时，从设备通过 MISO 线发送数据到主设备作为响应，这就实现了在同一时钟下数据发送和接收的同步全双工传输。发送移位寄存器和接收移位寄存器使用相同的特殊功能器地址，对 SPI 数据寄存器 SSSDAT 进行写操作将写入发送移位寄存器，对 SSSDAT 寄存器进行读操作将获得接收移位寄存器的数据。

有些设备的 SPI 接口会引出 SS 脚（从设备选择引脚，低有效），与 SC92F637X 的 SPI 通信时，SPI 总线上其它设备的 SS 脚的连接方式需根据不同的通信模式进行连接。下表列出了 SC92F637X 的 SPI 不同通信模式下，SPI 总线上其它设备 SS 脚的连接方式：

SC92F637X SPI	SPI 总线上其它设备	模式	从机的 SS（从设备选择引脚）
主模式	从模式	一主一从	拉低
		一主多从	SC92F637X 引出多根 I/O，分别接至从机的 SS 脚。在数据传送之前，从设备的 SS 引脚必须被置低
从模式	主模式	一主一从	拉高

**14.1.3.1 主模式**
**● 模式启动:**

SPI 主设备控制 SPI 总线上所有数据传送的启动。当 SSSCON0 寄存器中的 MSTR 位置 1 时，SPI 在主模式下运行，只有一个主设备可以启动传送。

**● 发送:**

在 SPI 主模式下，写一个字节数据到 SPI 数据寄存器 SSSDAT，数据将会写入发送移位缓冲器。如果发送移位寄存器已经存在一个数据，那么主 SPI 产生一个 WCOL 信号以表明写入太快。但是在发送移位寄存器中的数据不会受到影响，发送也不会中断。另外如果发送移位寄存器不为空，那么主设备立即按照 SCK 上的 SPI 时钟频率串行地移出发送移位寄存器中的数据到 MOSI 线上。当传送完毕，SSSCON1 寄存器中 SPIF 位被置 1。如果 SPI 中断被允许，当 SPIF 位置 1 时，也会产生一个中断。

**● 接收:**

当主设备通过 MOSI 线传送数据给从设备时，相对应的从设备同时也通过 MISO 线将其发送移位寄存器的内容传送给主设备的接收移位寄存器，实现全双工操作。因此，SPIF 标志位置 1 即表示传送完成也表示接收数据完毕。从设备接收的数据按照 MSB 或 LSB 优先的传送方向存入主设备的接收移位寄存器。当一个字节的数据完全被移入接收寄存器时，处理器可以通过读 SSSDAT 寄存器获得该数据。

**14.1.3.2 从模式**
**● 模式启动:**

当 SSSCON0 寄存器中的 MSTR 位清 0，SPI 在从模式下运行。

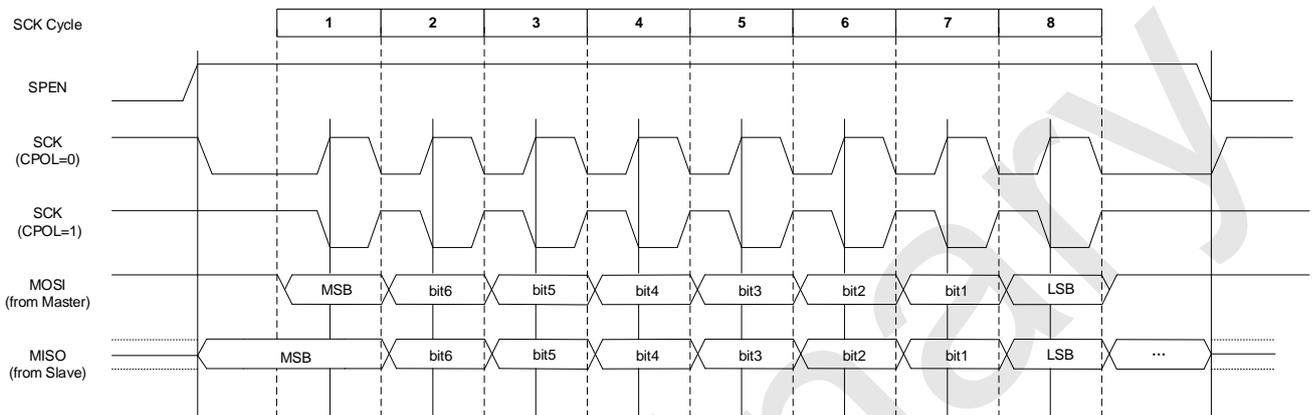
**● 发送与接收:**

从属模式下，按照主设备控制的 SCK 信号，数据通过 MOSI 引脚移入，MISO 引脚移出。一个位计数器记录 SCK 的边沿数，当接收移位寄存器移入 8 位数据(一个字节)同时发送移位寄存器移出 8 位数据(一个字节)，SPIF 标志位被置 1。数据可以通过读取 SSSDAT 寄存器获得。如果 SPI 中断被允许，当 SPIF 置 1 时，也会产生一个中断。此时接收移位寄存器保持原有数据并且 SPIF 位置 1，这样 SPI 从设备将不会接收任何数据直到 SPIF 清 0。SPI 从设备必须在主设备开始一次新的数据传送之前将要传送的数据写入发送移位寄存器。如果在开始发送之前未写入数据，从设备将传送“0x00”字节给主设备。如果写 SSSDAT 操作发生在传送过程中，那么 SPI 从设备的 WCOL 标志位置 1，即如果传送移位寄存器已经含有数据，SPI 从设备的 WCOL 位置 1，表示写 SSSDAT 冲突。但是移位寄存器的数据不受影响，传送也不会被中断。

### 14.1.4 传送形式

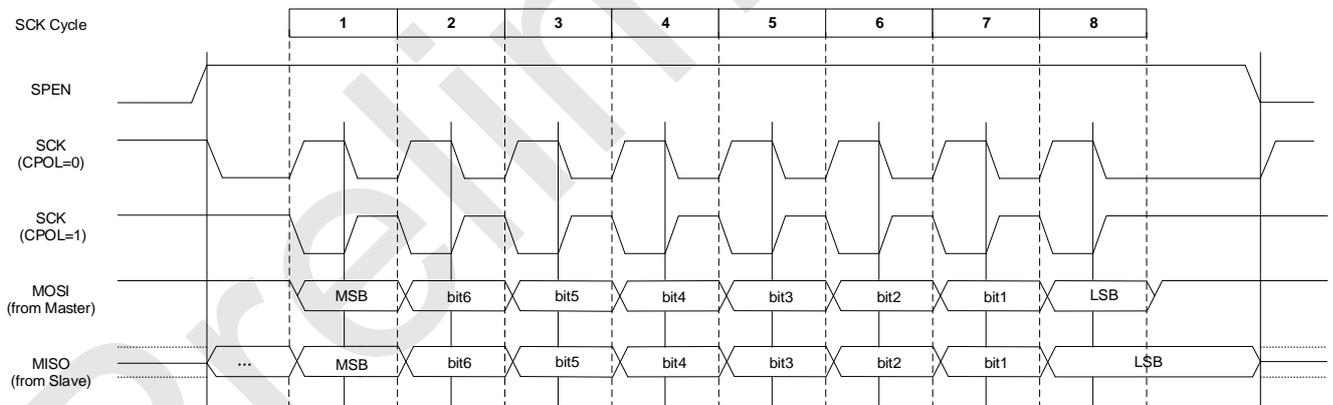
通过软件设置 **SSCON0** 寄存器的 **CPOL** 位和 **CPHA** 位，用户可以选择 **SPI** 时钟极性和相位的四种组合方式。**CPOL** 位定义时钟的极性，即空闲时的电平状态，它对 **SPI** 传输格式影响不大。**CPHA** 位定义时钟的相位，即定义允许数据采样移位的时钟边沿。在主从通讯的两个设备中，时钟极性相位的设置应一致。

当 **CPHA = 0**，**SCK** 的第一个沿捕获数据，从设备必须在 **SCK** 的第一个沿之前将数据准备好。



**CPHA = 0 数据传输图**

当 **CPHA = 1**，主设备在 **SCK** 的第一个沿将数据输出到 **MOSI** 线上，从设备把 **SCK** 的第一个沿作为开始发送信号，**SCK** 的第二沿开始捕获数据，因此用户必须在第一个 **SCK** 的两个沿内完成写 **SSDAT** 的操作。这种数据传输形式是一个主设备一个从设备之间通信的首选形式。



**CPHA = 1 数据传输图**

### 14.1.5 出错检测

在发送数据序列期间写入 SSDAT 寄存器会引起写冲突，SSCON1 寄存器中的 WCOL 位置 1。WCOL 位置 1 不会引起中断，发送也不会中止。WCOL 位需由软件清 0。

## 14.2 TWI

SSMOD[1:0] = 10，三选一串行接口 SSI 配置为 TWI 接口。SC92F637X 在 TWI 通信时只能做从机。

### SSCON0 (9DH) TWI 控制寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	TWEN	TWIF	-	GCA	AA	STATE[2:0]		
读/写	读/写	读/写	-	读	读/写	读/写	读/写	读/写
上电初始值	0	0	x	0	0	0	0	0

位编号	位符号	说明
7	<b>TWEN</b>	TWI 使能控制 0: 关闭 TWI 1: 打开 TWI
6	<b>TWIF</b>	TWI 中断标志位 0: 由软件清零 1: 在下列条件下，中断标志位由硬件置 1 ①第一帧地址匹配成功 ②成功接收或发送 8 位数据 ③重新启动 ④从机收到停止信号
4	<b>GCA</b>	通用地址响应标志位 0: 非响应通用地址 1: 当 GC 置 1，同时通用地址匹配时该位由硬件置 1，并自动清零
3	<b>AA</b>	接收使能位 0: 不允许接收主机发送的信息 1: 允许接收主机发送的信息
2~0	<b>STATE[2:0]</b>	状态机状态标志位 000: 从机处于空闲状态，等待 TWEN 置 1，检测 TWI 启动信号。当从机接收到停止条件后会跳转到此状态 001: 从机正在接收第一帧地址和读写位（第 8 位为读写位，1 为读，0 为写）。从机接收到起始条件后会跳转到此状态 010: 从机接收数据状态 011: 从机发送数据状态 100: 在从机发送数据状态中，当主机回 UACK（应答位为高电平）时跳转到此状态，等待重新启动信号或停止信号。 101: 从机处于发送状态时，将 AA 写 0 会进入此状态，等待重新启动信号或停止信号。
5	-	保留

### SSCON1 (9EH) TWI 地址寄存器(读/写)

位编号	7	6	5	4	3	2	1	0
符号	TWA[6:0]							GC
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
7~1	<b>TWA[6:0]</b>	TWI 地址寄存器
0	<b>GC</b>	TWI 通用地址使能

位编号	位符号	说明
		0: 禁止响应通用地址 1: 允许响应通用地址

**SSDAT (9FH) TWI 数据缓存寄存器(读/写)**

位编号	7	6	5	4	3	2	1	0
符号	TWDAT[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	<b>TWDAT[7:0]</b>	TWI 数据缓存寄存器

**14.2.1 信号描述**
**14.2.1.1 TWI 时钟信号线 (SCL)**

该时钟信号由主机发出，连接到所有的从机。每 9 个时钟周期传送一个字节数据。前 8 个周期作数据的传送，最后一个时钟作为接收方应答时钟。

**14.2.1.2 TWI 数据信号线 (SDA)**

SDA 是双向信号线，空闲时应为高电平，由 SDA 线上的上拉电阻拉高。

**14.2.2 工作模式**

SC92F637X 的 TWI 通信只有从机模式：

- **模式启动：**

当 TWI 使能标志位打开 (TWEN = 1)，同时接收到主机发送的启动信号时，模式启动。

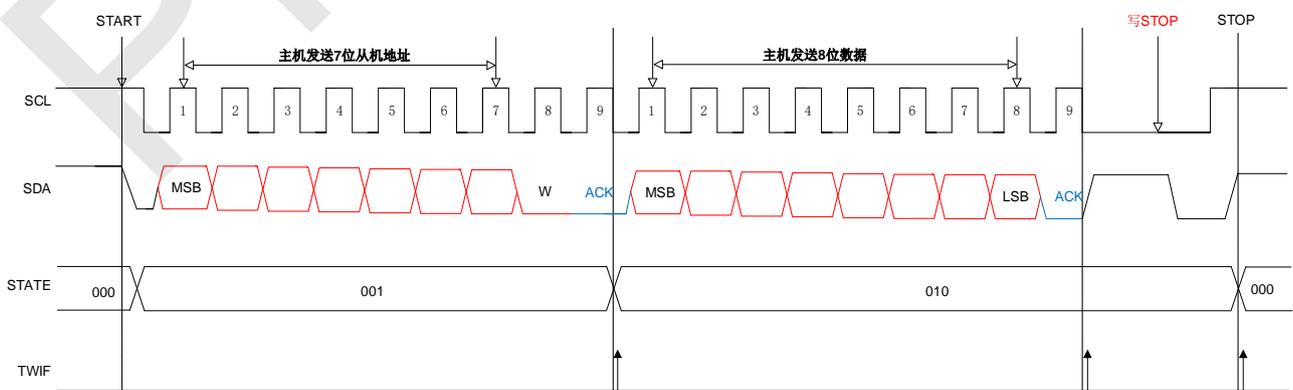
从机从空闲模式 (STATE[2:0] = 000) 进入接收第一帧地址 (STATE[2:0] = 001) 状态，等待主机的第一帧数据。第一帧数据由主机发送，包括了 7 位地址位和 1 位读写位，TWI 总线上所有从机都会收到主机的第一帧数据。主机发送完第一帧数据后释放 SDA 信号线。若主机所发地址与某一从机自身地址寄存器中的值相同，说明该从机被选中，被选中的从机会判断接总线上的第 8 位，即数据读写位 (=1，读命令；=0，写命令)，然后占用 SDA 信号线，在 SCL 的第 9 个时钟周期给主机一个低电平的应答信号，之后会释放总线。从机被选中后，会根据读写位的不同而进入不同的状态：

- **非通用地址响应，从机接收模式：**

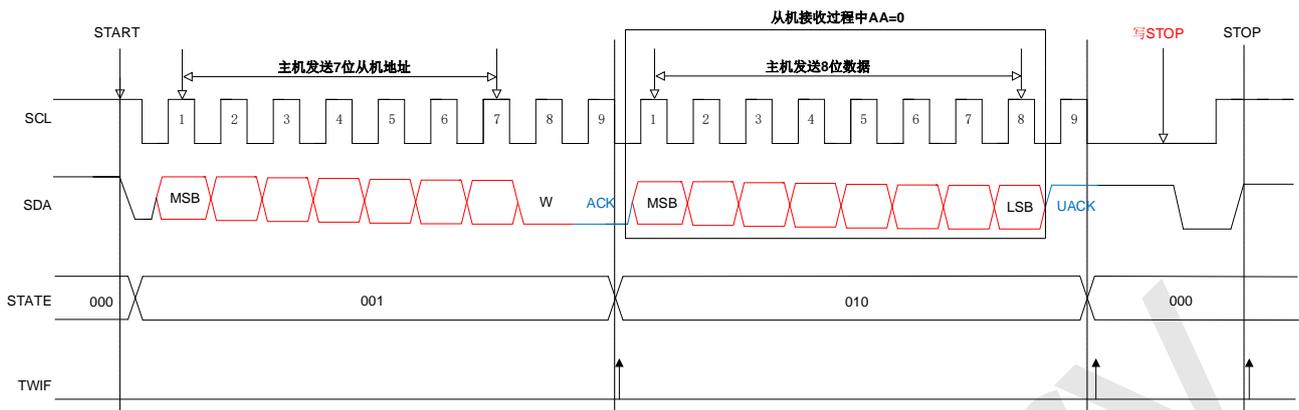
如果第一帧接收到的读写位是写 (0)，则从机进入到从机接收状态 (STATE[2:0] = 010) 等待接收主机发送的数据。主机每发送 8 位，都要释放总线，等待第 9 个周期从机的应答信号。

1. 如果从机的应答信号是低电平，主机的通信可以有以下几种方式：

- 1) 继续发送数据；
- 2) 重新发送启动信号 (start)，此时从机重新进入接收第一帧地址 (STATE[2:0] = 001) 状态；
- 3) 发送停止信号，表示本次传输结束，从机回到空闲状态，等待主机下一次的启动信号。



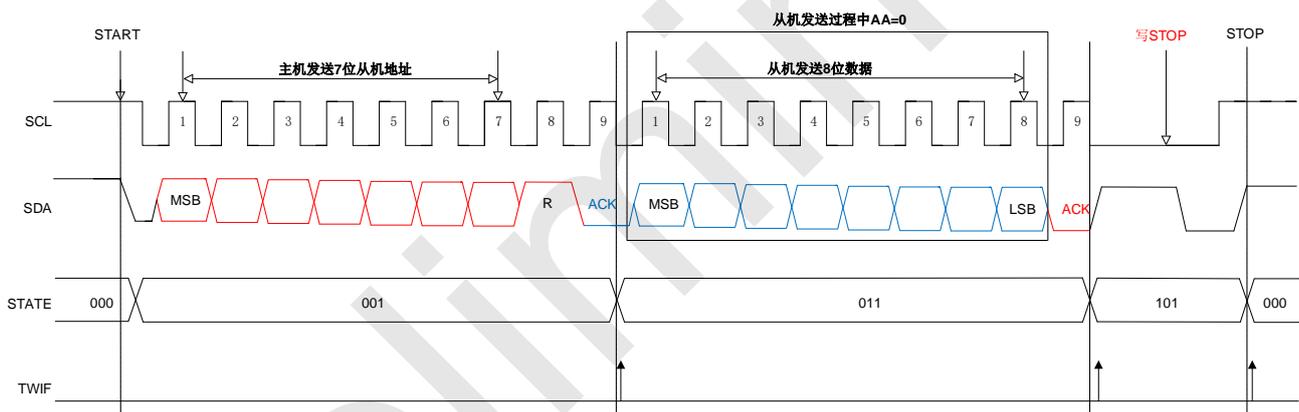
2. 如果从机应答的是高电平 (在接收过程中，从机寄存器中的 AA 值改写为 0)，表示当前字节传输完以后，从机会主动结束本次传输，回到空闲状态 (STATE[2:0] = 000)，不再接收主机发送的数据。



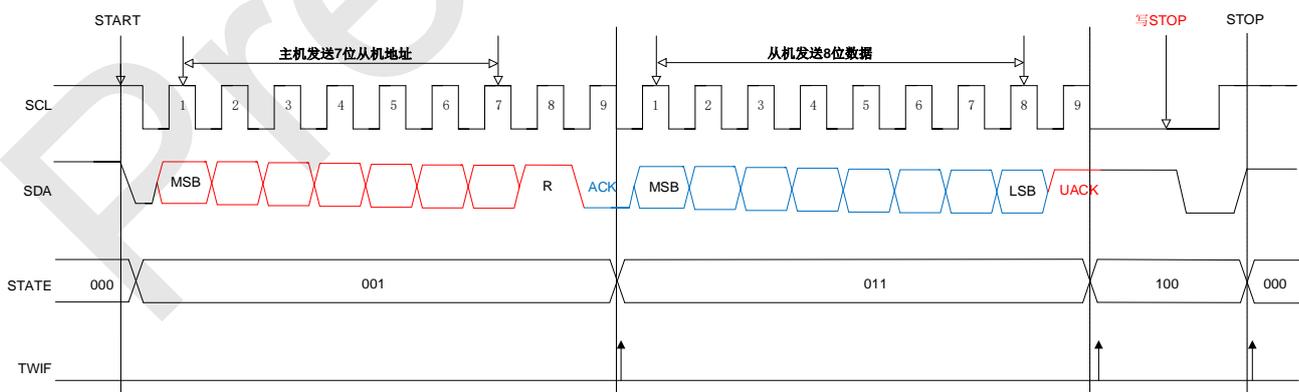
● **非通用地址响应，从机发送模式：**

如果第一帧接收到的读写位是读（1），则从机会占用总线，向主机发送数据。每发送 8 位数据，从机释放总线，等待主机的应答：

1. 如果主机应答的是低电平，则从机继续发送数据。在发送过程中，如果从机寄存器中的 AA 值被改写为 0，则传输完当前字节从机会主动结束传输并释放总线，等待主机的停止信号或重新启动信号（STATE[2:0] = 101）。



2. 如果主机应答的是高电平，则从机 STATE[2:0] = 100，等待主机的停止信号或重新启动信号。

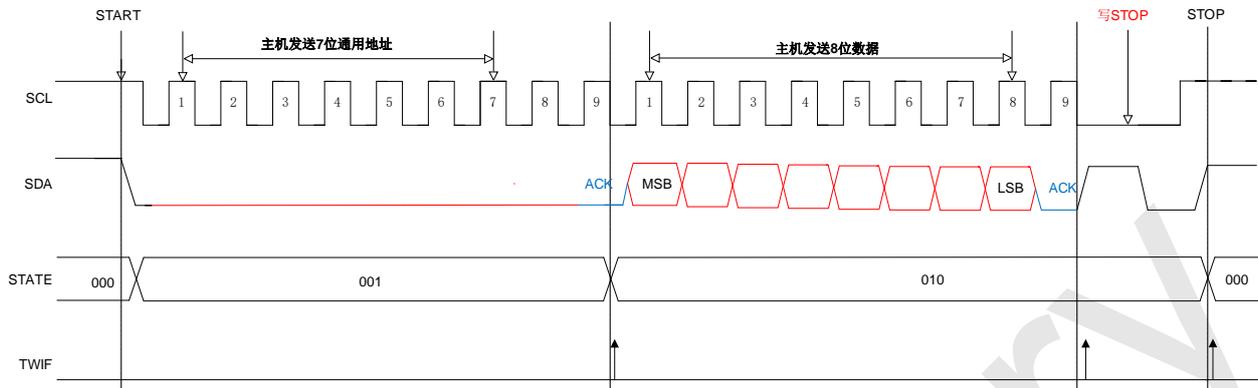


● **通用地址的响应：**

GC=1 时，此时通用地址允许使用。从机进入到接收第一帧地址（STATE[2:0] = 001）状态，接收的第一帧数据中的地址位数据为 0x00，此时所有从机响应主机。主机发送的读写位必须是写（0），所有从机接收后进入接收数据（STATE[2:0] = 010）状态。主机每发送 8 个数据释放一次 SDA 线，并读取 SDA 线上的状态：

1. 如果有从机应答，则主机的通信可以有以下三种方式：
  - 1) 继续发送数据；

- 2) 重新启动;
- 3) 发送停止信号, 结束本次通讯。



2. 若无从机应答, 则 SDA 为空闲状态。

**注意:** 在一主多从模式下使用通用地址时, 主机发送的读写位不能为读 (1) 状态, 否则除发送数据的设备, 总线上其它设备均会响应。

### 14.2.3 操作步骤

三合一串口中 TWI 工的操作步骤如下:

- ① 配置 SSMOD[1:0], 选择 TWI 模式;
- ② 配置 SSSCON0 TWI 控制寄存器;
- ③ 配置 SSSCON1 TWI 地址寄存器;
- ④ 如果从机接收数据, 则等待 SSSCON0 中的中断标志位 TWIF 置 1。从机每接收到 8 位数据, 中断标志位会被置 1。中断标志位需手动清零;
- ⑤ 如果从机发送数据, 则要将待发送的数据写进 TWDAT 中, TWI 会自动将数据发送出去。每发送 8 位, 中断标志位 TWIF 就会被置 1。

## 14.3 UART

SSMOD[1:0] = 11, 三选一串行接口 SSI 配置为 UART 接口。

### SSCON0 (9DH) 串口 1 控制寄存器(读/写)

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

位编号	位符号	说明
7	<b>SM0</b>	串行通信模式控制位 0: 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; 1: 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	<b>SM2</b>	串行通信模式控制位 2, 此控制位只对模式 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	<b>REN</b>	接收允许控制位 0: 不允许接收数据;

位编号	位符号	说明
		1: 允许接收数据。
3	<b>TB8</b>	只对模式 3 有效, 为发送数据的第 9 位
2	<b>RB8</b>	只对模式 3 有效, 为接收数据的第 9 位
1	<b>TI</b>	发送中断标志位 需用户软件写 1 清 0
0	<b>RI</b>	接收中断标志位 需用户软件写 1 清 0
6	-	保留

**SSCON1 (9EH) 串口 1 波特率控制寄存器低位(读/写)**

位编号	7	6	5	4	3	2	1	0
符号	BAUDL [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

**SSCON2 (95H) 串口 1 波特率控制寄存器高位(读/写)**

位编号	7	6	5	4	3	2	1	0
符号	BAUDH [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	<b>BAUD [15:0]</b>	串口波特率控制  $\text{BaudRate} = \frac{f_{\text{SYS}}}{\text{BAUD1H, BAUD1L}}$ 注意: [BAUD1H,BAUD1L] 必须大于 0x0010

**SSDAT (9FH) 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	<b>SBUF[7:0]</b>	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。

## 15 高灵敏度触控电路

SC92F637X 内建一个 17 通道的高灵敏度电容触控电路，其特点如下：

1. 可适应隔空按键触控、接近感应等对灵敏度要求较高的触控应用
2. 可通过 10V 动态 CS 测试
3. 可实现 17 路触控按键及衍生功能
4. CMOD 管脚需对地接入 103 电容
5. 高灵活度开发软件库支持，低开发难度
6. 自动化调试软件支持，智能化开发
7. 触控模块可以在 MCU STOP 模式下进入低功耗模式工作

注意：高灵敏度触控电路的时钟源固定为  $f_{HRC} = 32\text{MHz}$ ，不会随着内外系统时钟的切换而改变。

### 15.1 触控电路的耗电模式

SC92F637X 允许在 STOP Mode 开启触控扫描功能：这样的方式可以降低 MCU 的整体功耗从而满足有低功耗需求的触控应用。

用户可以理解为 SC92F637X 的触控电路具有两种耗电模式：

1. 普通运行模式
2. 低功耗运行模式

两种耗电模式的定义如下：

说明	普通运行模式	低功耗运行模式
CPU	RUN (Normal mode)	Stop (STOP Mode)
触控电路	RUN	RUN

注意：用户通过使用赛元提供的触控按键库文件（可从赛元官网下载获取），可快速简单实现所需的触控功能。

## 16 Check Sum 模块

SC92F637X 内建了 1 个 check sum 模块，可用来实时生成程序代码的 16 位 check sum 值，用户可利用此 check sum 和理论值比较，监测程序区的内容是否正确。

**注意：**check sum 值是整个程序区的数据累加和，即 0000H~3FFDH 地址单元所有的数据。若地址单元中有用户上次操作后的残留值，会导致 check sum 值与理论值不符。因此，建议用户对整片 code 区域进行擦除或写 0 操作后再烧录代码以保证 check sum 值与理论值一致。

### 16.1 Check Sum 校验操作相关寄存器

#### CHKSUML (FCH) Check Sum 结果寄存器低位(读/写)

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

位编号	位符号	说明
7~0	<b>CHKSUML [7:0]</b>	Check Sum 结果寄存器低位

#### CHKSUMH (FDH) Check Sum 结果寄存器高位(读/写)

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

位编号	位符号	说明
7~0	<b>CHKSUMH [7:0]</b>	Check Sum 结果寄存器高位

#### OPERCON (EFH) 运算控制寄存器(读/写)

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

位编号	位符号	说明
0	<b>CHKSUMS</b>	Check sum 运算开始触发控制 (Start) 对此 bit 写“1”，开始做一次 check sum 计算。此位只可写入 1 有效。 <b>注意：</b> checksum 开启后需要加 8 个 NOP 以确保程序正确运行

## 17 电气特性

### 17.1 极限参数

符号	参数	最小值	最大值	UNIT
V <sub>DD</sub>	直流供电电压	-0.3	6	V
V <sub>PIN</sub>	任一管脚输入/输出电压	-0.3	V <sub>DD</sub> +0.3	V
T <sub>A</sub>	工作环境温度	-40	105	°C
T <sub>STG</sub>	储存温度	-55	125	°C
I <sub>VDD</sub>	流过 VDD 的电流值	-	200	mA
I <sub>VSS</sub>	流过 VSS 的电流值	-	200	mA

### 17.2 推荐工作条件

符号	参数	最小值	最大值	UNIT
V <sub>DD</sub>	工作电压	2.4	5.5	V
T <sub>A</sub>	工作环境温度	-40	105	°C

### 17.3 FLASH ROM 参数

符号	参数	最小值	典型值	最大值	UNIT	测试条件
V <sub>Write</sub>	写电压	3	5	5.5	V	
N <sub>END</sub>	擦写次数	1000	-	-	Cycles	V <sub>DD</sub> = 5.0V T <sub>A</sub> = +25°C
T <sub>DR</sub>	数据保存时间	10	-	-	Years	T <sub>A</sub> = +125°C

### 17.4 EEPROM 参数

符号	参数	最小值	典型值	最大值	UNIT	测试条件
V <sub>Read</sub>	读电压	2.4		5.5	V	
V <sub>Write</sub>	写电压	2.5		5.5	V	
N <sub>END</sub>	擦写次数	10,000	-	-	Cycles	
T <sub>DR</sub>	数据保存时间	10	-	-	Years	V <sub>DD</sub> = 5.0V T <sub>A</sub> = +125°C
T <sub>Write</sub>	单个 byte 写入时间	-	4	-	ms	V <sub>DD</sub> = 5.0V T <sub>A</sub> = +25°C PAYTIMES[1:0] = 00
		-	2	-	ms	V <sub>DD</sub> = 5.0V T <sub>A</sub> = +25°C PAYTIMES[1:0] = 01
		-	1	-	ms	V <sub>DD</sub> = 5.0V T <sub>A</sub> = +25°C PAYTIMES[1:0] = 10

## 17.5 直流电气特性

(V <sub>DD</sub> = 5V, T <sub>A</sub> = +25°C, 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
I <sub>op</sub>	Operation Mode 工作电流	-	3.1	-	mA	f <sub>sys</sub> = 32MHz
		-	2.3	-	mA	f <sub>sys</sub> = 16MHz
		-	1.8	-	mA	f <sub>sys</sub> = 8MHz
		-	1.5	-	mA	f <sub>sys</sub> = 2.66MHz
I <sub>IDL</sub>	IDLE Mode 工作电流	-	1.3	-	mA	f <sub>sys</sub> = 32MHz
		-	0.7	-	mA	f <sub>sys</sub> = 2.66MHz
I <sub>pd</sub>	STOP Mode 工作电流	-	1.7	-	μA	
I <sub>BTM</sub>	Base Timer 工作电流	-	0.9	2	μA	BTMFS[3:0]=1000 每 4.0 秒产生一个 中断
I <sub>WDT</sub>	WDT 电流	-	0.9	2	μA	WDTCKS[2:0]=000 WDT 溢出时间 500ms
I <sub>TK</sub>	Touch key 工作电流 (高灵敏度模式)	-	0.6	1	mA	

(V <sub>DD</sub> = 3.3V, T <sub>A</sub> = +25°C, 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
I <sub>op</sub>	Operation Mode 工作电流	-	2.7	-	mA	f <sub>sys</sub> = 32MHz
		-	2.0	-	mA	f <sub>sys</sub> = 16MHz
		-	1.6	-	mA	f <sub>sys</sub> = 8MHz
		-	1.3	-	mA	f <sub>sys</sub> = 2.66MHz
I <sub>IDL</sub>	IDLE Mode 工作电流	-	1.3	-	mA	f <sub>sys</sub> = 32MHz
		-	0.7	-	mA	f <sub>sys</sub> = 2.66MHz
I <sub>pd</sub>	STOP Mode 工作电流	-	1.5	-	μA	
I <sub>BTM</sub>	Base Timer 工作电流	-	0.8	2	μA	BTMFS[3:0]=1000 每 4.0 秒产生一个 中断
I <sub>WDT</sub>	WDT 电流	-	0.8	2	μA	WDTCKS[2:0]=000 WDT 溢出时间 500ms
I <sub>TK</sub>	Touch key 工作电流 (高灵敏度模式)	-	0.6	1	mA	

## 17.6 IO 电气特性

IO 口特性 (V <sub>DD</sub> = 5V, T <sub>A</sub> = +25°C, 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 <sub>IH1</sub>	输入高电压	0.7V <sub>DD</sub>	-	V <sub>DD</sub> +0.3	V	
V <sub>IL1</sub>	输入低电压	-0.3	-	0.3V <sub>DD</sub>	V	
V <sub>IH2</sub>	输入高电压 <sup>注1</sup>	0.8V <sub>DD</sub>	-	V <sub>DD</sub>	V	
V <sub>IL2</sub>	输入低电压 <sup>注1</sup>	-0.2	-	0.2V <sub>DD</sub>	V	
I <sub>OL1</sub>	输出低电流 除 P22~P26 以外的端口	-	45	-	mA	V <sub>Pin</sub> = 0.4V
		-	80	-	mA	V <sub>Pin</sub> = 0.8V

IO 口特性 ( $V_{DD} = 5V$ , $T_A = +25^\circ C$ , 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
I <sub>OL2</sub>	输出低电流 P22~P26	-	60	-	mA	$V_{Pin} = 0.4V$
		-	100	-	mA	$V_{Pin} = 0.8V$
I <sub>OH1</sub>	输出高电流	-	22	-	mA	$V_{Pin} = 4.3V$
I <sub>OH2</sub>	输出高电流	-	9	-	mA	$V_{Pin} = 4.7V$
I <sub>lkg</sub>	输入漏电流	-1	-	1	$\mu A$	IO 为高阻输入模式 $V_{IN} = V_{DD}$ 或 $V_{SS}$
R <sub>PH</sub>	上拉电阻	16.5	33	49.5	k $\Omega$	$V_{IN} = V_{SS}$

注 1: 施密特触发输入端口包含: NRST、T\_CLK / T\_DIO、UART 的 RX 信号口、SPI / TWI 信号输入口、INT0~INT15、PWM 故障检测口 FLT、Timer 时钟输入口 Tn、Timer 捕获口 TnEX。

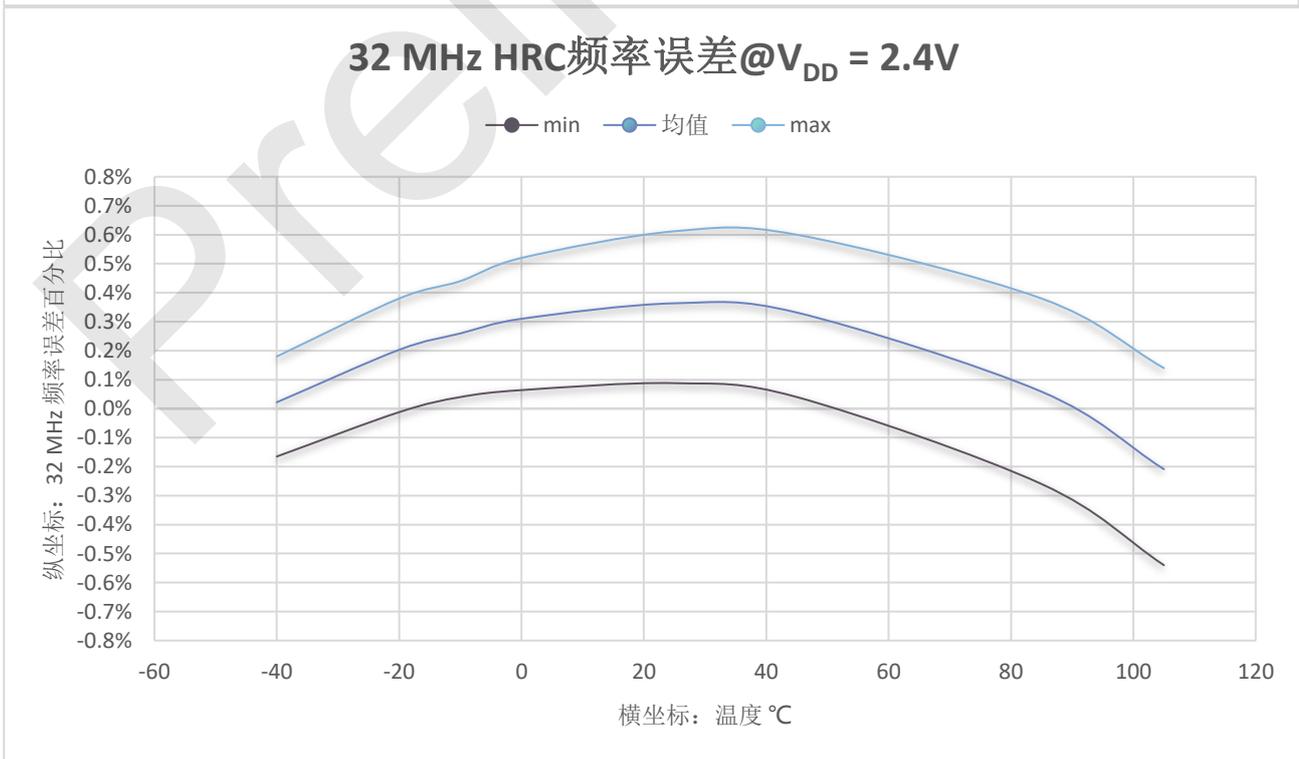
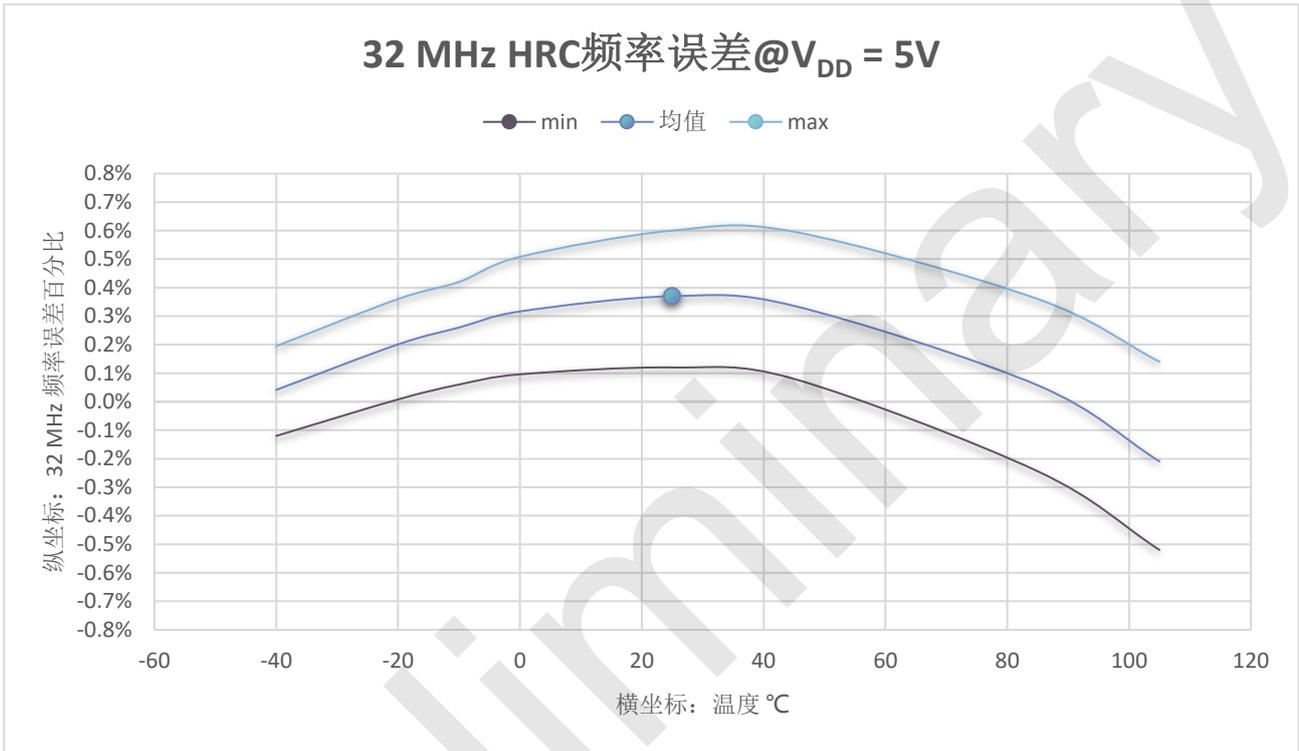
IO 口特性 ( $V_{DD} = 3.3V$ , $T_A = +25^\circ C$ , 除非另有说明)						
符号	参数	最小值	典型值	最大值	单位	测试条件
V <sub>IH3</sub>	输入高电压	$0.7V_{DD}$	-	$V_{DD} + 0.3$	V	
V <sub>IL3</sub>	输入低电压	-0.3	-	$0.3V_{DD}$	V	
V <sub>IH4</sub>	输入高电压 <sup>注1</sup>	$0.8V_{DD}$	-	$V_{DD}$	V	
V <sub>IL4</sub>	输入低电压 <sup>注1</sup>	-0.2	-	$0.2V_{DD}$	V	
I <sub>OL3</sub>	输出低电流 除 P22~P26 以外的端口	-	30	-	mA	$V_{Pin} = 0.4V$
		-	60	-	mA	$V_{Pin} = 0.8V$
I <sub>OL4</sub>	输出低电流 P22~P26	-	40	-	mA	$V_{Pin} = 0.4V$
		-	75	-	mA	$V_{Pin} = 0.8V$
I <sub>OH3</sub>	输出高电流	-	7	-	mA	$V_{Pin} = 3.0V$
I <sub>lkg</sub>	输入漏电流	-1	-	1	$\mu A$	IO 为高阻输入模式 $V_{IN} = V_{DD}$ 或 $V_{SS}$
R <sub>PH</sub>	上拉电阻	28	56	84	k $\Omega$	$V_{IN} = V_{SS}$

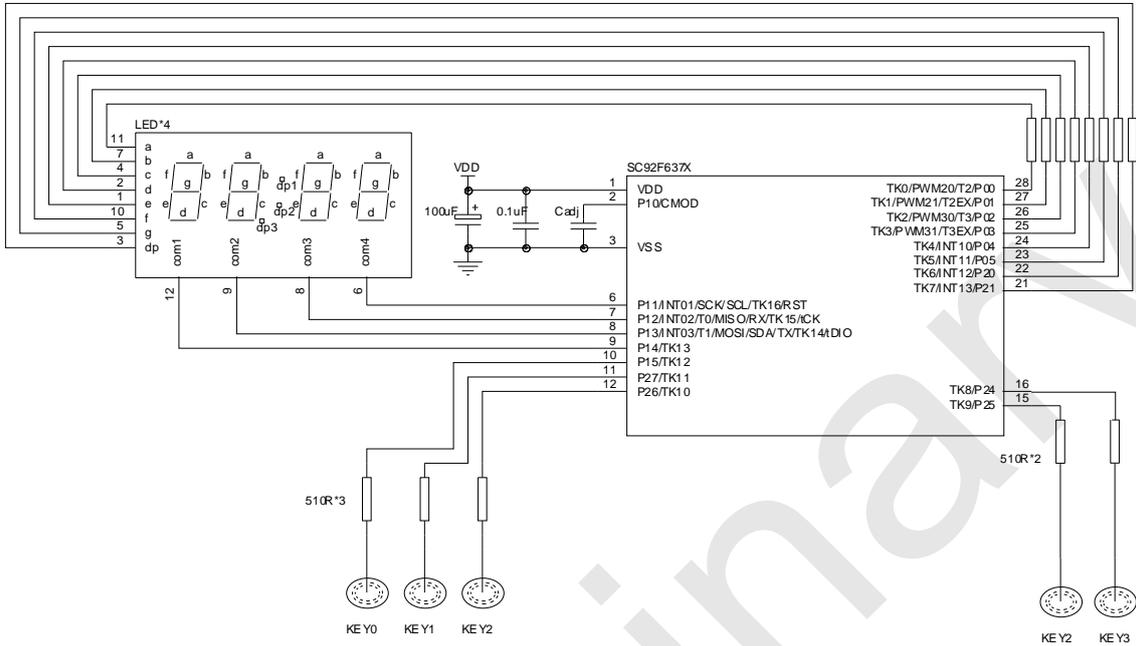
注 1: 施密特触发输入端口包含: NRST、T\_CLK / T\_DIO、UART 的 RX 信号口、SPI / TWI 信号输入口、INT0~INT15、PWM 故障检测口 FLT、Timer 时钟输入口 Tn、Timer 捕获口 TnEX。

## 17.7 交流电气特性

(V <sub>DD</sub> = 2.4V ~ 5.5V, T <sub>A</sub> = 25°C, 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 <sub>POR</sub>	Power On Reset 时间	-	13	-	ms	
T <sub>PDW</sub>	Power Down 模式唤醒时间	-	60	130	$\mu s$	
T <sub>Reset</sub>	复位脉冲宽度	18	-	-	$\mu s$	低电平有效
T <sub>LVR</sub>	LVR 消抖时间	-	30	-	$\mu s$	

(V <sub>DD</sub> = 2.4V ~ 5.5V, T <sub>A</sub> = 25°C, 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
f <sub>HRC</sub>	RC 振荡稳定性	-1	-	+1	%	V <sub>DD</sub> = 2.4~5.5V T <sub>A</sub> = -40°C~85°C
		-1.5	-	+1.5	%	V <sub>DD</sub> = 2.4~5.5V T <sub>A</sub> = -40°C~105°C

**17.7.1 32 MHz HRC 频率误差曲线图**


**18 应用电路**


SC92F637X Touch Key 应用电路图

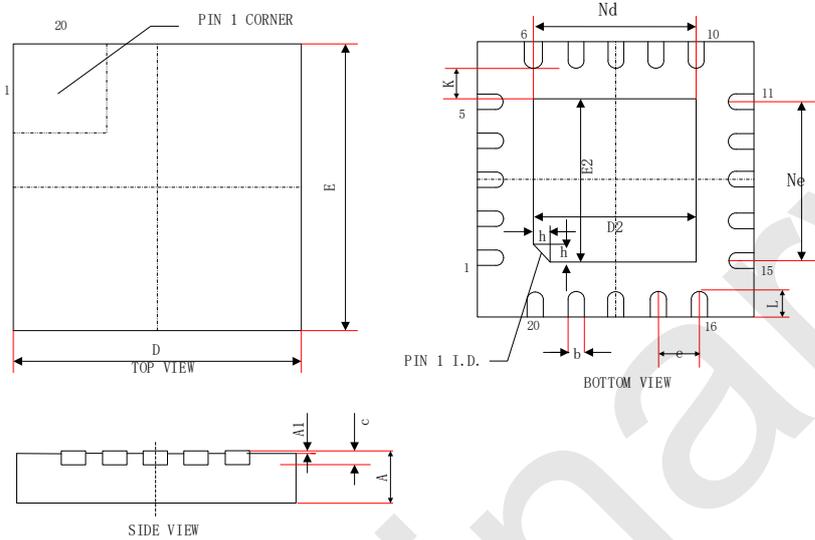
## 19 订购信息

产品编号	封装	包装
SC92F6372Q20R	QFN20(3X3)	盘装
SC92F6372M20U	SOP20	管装
SC92F6372X20U	TSSOP20	管装
SC92F6371M16U	SOP16	管装
SC92F6378S10U	MSOP10	管装
SC92F6370M08U	SOP8	管装

## 20 封装信息

### 20.1 SC92F6372Q20R

QFN20 L(3\*3) 外形尺寸 单位:毫米

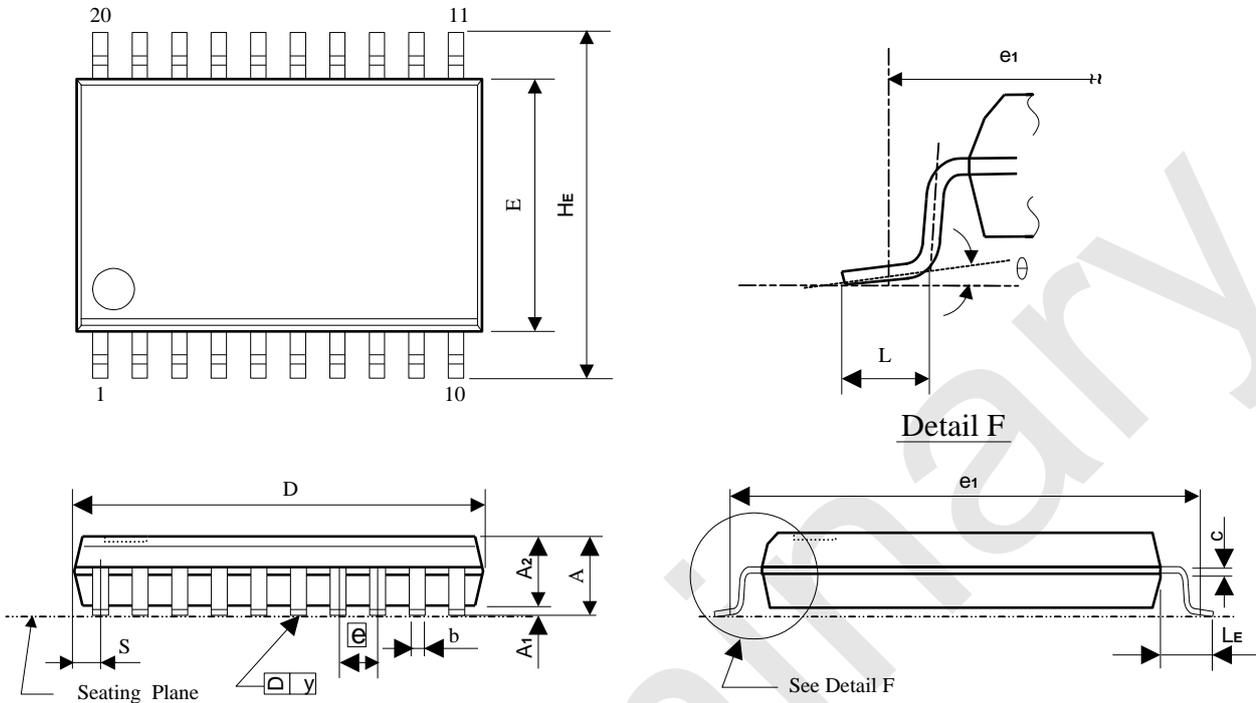


符号	mm(毫米)		
	最小	正常	最大
A	0.50	0.55	0.60
A1	0	0.02	0.05
b	0.15	0.20	0.25
c	0.15REF		
D	2.90	3.00	3.10
D2	1.60	1.70	1.80
e	0.40BSC		
Ne	1.60BSC		
Nd	1.60BSC		
E	2.90	3.00	3.10
E2	1.60	1.70	1.80
L	0.25	0.30	0.35
h	0.20	0.25	0.30
K	0.30	0.35	0.40

**20.2 SC92F6372M20U**

SOP20L(300mil) 外形尺寸

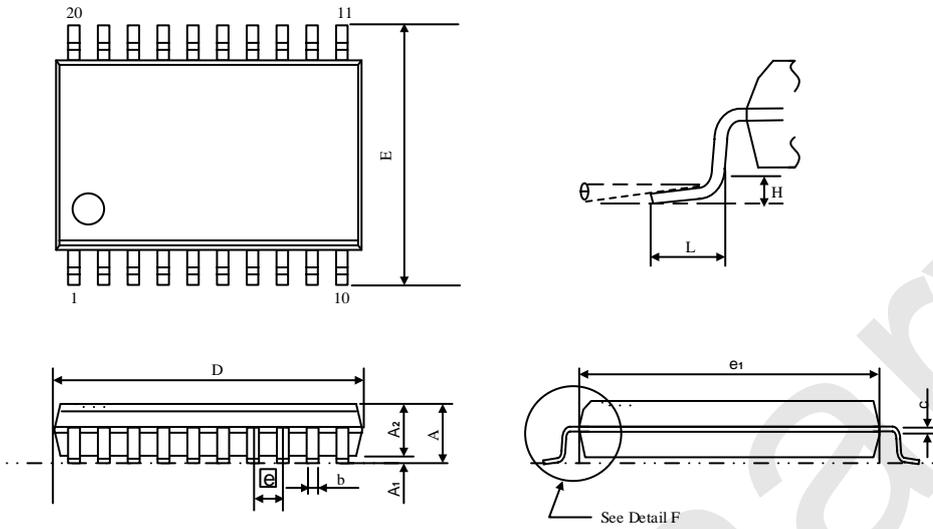
单位:毫米



符号	mm(毫米)		
	最小	正常	最大
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
$\bar{e}$	1.27(BSC)		
L	0.700	0.850	1.000
LE	1.30	1.40	1.50
$\theta$	0°	-	8°

**20.3 SC92F6372X20U**

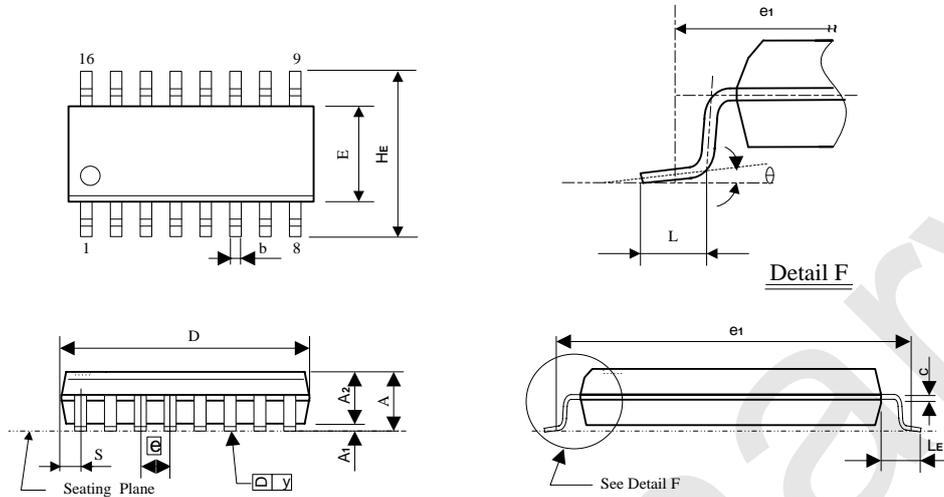
TSSOP20L 外形尺寸      单位：毫米



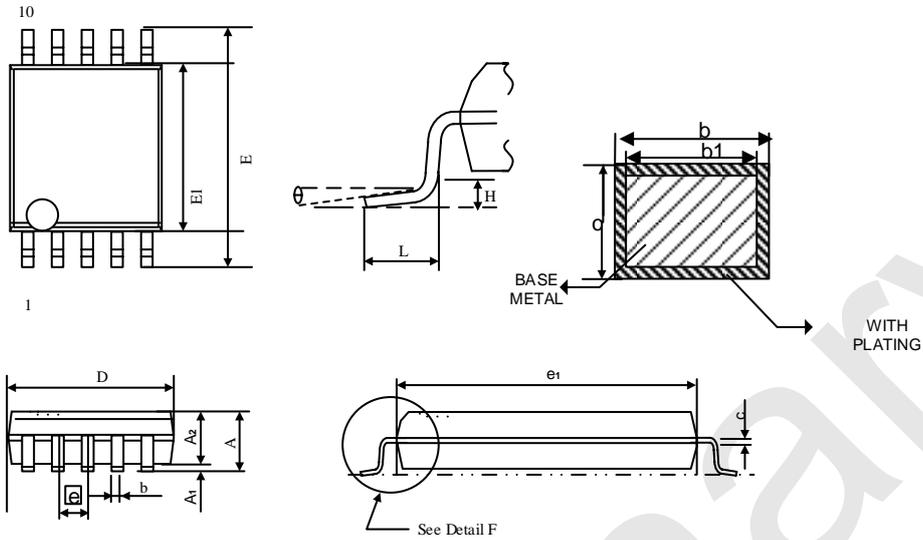
符号	mm(毫米)		
	最小		最大
A	-	-	1.200
A1	0.050	-	0.150
A2	0.800	-	1.050
b	0.190	-	0.300
c	0.090	-	0.200
D	6.400	-	6.600
E	6.20	-	6.60
e1	4.300	-	4.500
	0.65(BSC)		
L	-	-	1.00
$\theta$	0°	-	8°
H	0.05	-	0.15

**20.4 SC92F6371M16U**

SOP16L(150mil) 外形尺寸 单位:毫米



符号	mm(毫米)		
	最小	正常	最大
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
	1.27(BSC)		
L	0.50	0.65	0.80
LE	0.95	1.05	1.15
$\theta$	0°	-	8°

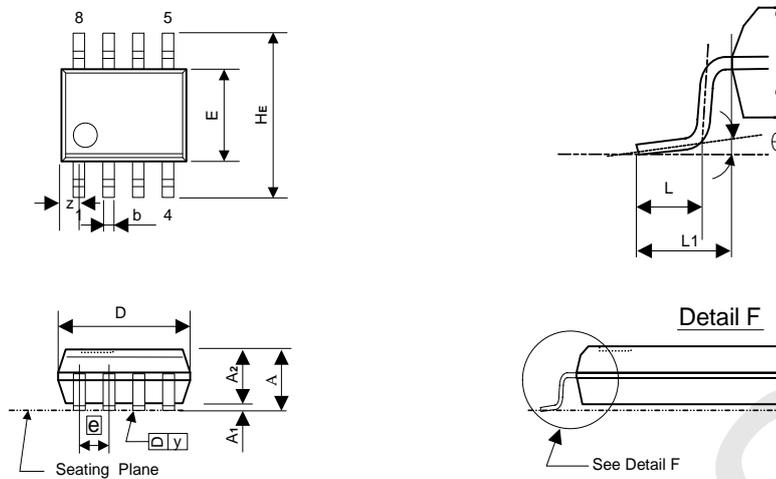
**20.5 SC92F6378S10U**
**MSOP10 外形尺寸**
**单位: 毫米**


符号	mm(毫米)		
	最小	正常	最大
A	--	--	1.10
A1	0.02	--	0.15
A2	0.75	0.85	0.95
b	0.18	--	0.33
c	0.09	-	0.23
D	2.90	3.00	3.10
E	4.70	4.90	5.10
E1	2.90	3.00	3.10
e	0.50(BSC)		
L	0.40	--	0.80
$\theta$	0	--	8°

**20.6 SC92F6370M08U**

SOP 8L(150mil) 外形尺寸

单位:毫米



符号	mm(毫米)		
	最小	正常	最大
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
$\bar{e}$	1.270(BSC)		
L	0.50	0.65	0.80
L1	0.95	1.05	1.15
$\theta$	0°	-	8°

## 21 规格更改记录

版本	记录	日期
V0.1	初版	2025 年 3 月 20 日

Preliminary

## 22 声明

深圳市赛元微电子股份有限公司（以下简称赛元）保留随时对赛元产品、文档或服务进行变更、更正、增强、修改和改进的权利，恕不另行通知。赛元认为提供的信息是准确可信的。本文档信息于 2025 年 3 月开始使用。在实际进行生产设计时，请参阅各产品最新的数据手册等相关资料。

Preliminary