MC51F8124 用户手册 V1.6

SinoMCU 8 位增强型 8051 单片机

1T 8051 内核 FLASH 型 MCU，16KB FLASH ROM，1536B SRAM，1KB 独立 EEPROM，12 位高速 ADC，16 通道高灵敏触摸电路，6 通道 16 位 PWM，8 位 PWM，4 个 16 位定时器，2 路 UART，SPI，CRC，双两线调试

主要特性

CORE
- ✧ 1T 高速增强型 8051 内核
- ✧ 双 DPTR

ROM
- ✧ 片上 16K 字节 FLASH，擦写 10 万次以上
- ✧ 片上 1024 字节 E2(EEPROM)，擦写 10 万次以上
- ✧ 支持代码分区保护功能（有效防止非法读/写/擦）
- ✧ 支持 FLASH 和 E2 在电路编程（ICP），支持 E2 在应用编程（IAP）

RAM
- ✧ 片上 1536（256+1280）字节 SRAM

时钟源
- ✧ 片上高精度 32MHz 高速振荡器（常温 5.0V 电压下±1%精度；全工作条件下±5%精度）
- ✧ 片上 32KHz 超低功耗低速振荡器，供看门狗定时器和 T3 使用
- ✧ 外部可选接 32768Hz 晶体振荡器

电源管理模式
- ✧ 4 种工作模式：高速/低速/停止/休眠
- ✧ 增加高级能耗控制功能，满足用户低功耗的需求

复位
- ✧ 上电复位（POR）/外部复位/低电压复位（LVR）/看门狗复位/软件复位
- ✧ LVR 电压 4 级可选：2.1V、2.5V、3.5V、4.1V
- ✧ 看门狗复位可选 8 种溢出时间

I/O
- ✧ 最多 26 个双向通用 I/O 口（28PIN 封装下）
- ✧ 支持 3 种输入/输出模式，支持输入上拉电阻配置
- ✧ 14 个 IO 具备独立大电流驱动能力
- ✧ 24 个 I/O 可软件模拟成 1/2 BIAS 的 LCD COM 口

定时器/计数器
- ✧ 2 个 16 位 T0/T1 定时器，兼容标准 8051
- ✧ 1 个 16 位增强型 T2 定时器，兼容 8052 的 T2，带输入捕获和输出比较功能
- ✧ 1 个 16 位 T3 时基定时器，可连接外部 32768Hz 晶振，在停止/休眠模式下可定时唤醒

PWM 定时器
- ✧ 1 路独立 8 位 PWM0，可作通用定时器
- ✧ 3 组共 6 路 16 位 PWM1 阵列，可互补输出且死区时间可调，并具有故障保护中断功能
- ✧ PWM1 可在选片上高速时钟或其分频下独立工作

12 位高精度 ADC
- ✧ 12 位高精度逐次逼近型 ADC
- ✧ 14 通道：外部 12 通道+内部 2 通道
- ✧ 参考电压可选：内部 2.0V、VDD、Vref 引脚输入

2 路 UART
- ✧ 2 路 UART 模块，可兼容 8051 标准
- ✧ 增强 UART0 支持“帧出错”检测及自动地址识别
- ✧ 支持 8 位同步半双工、8 位/9 位异步全双工等 4 种工作方式

SPI
- ✧ 支持全双工，3 线/4 线同步模式，主/从机可选
- ✧ 支持主机模式错误用以防止主机冲突

TK 触摸按键
- ✧ 最大支持 16 通道的高灵敏触摸按键
- ✧ 支持 4MHz/2MHz/1MHz/500KHz 四种工作频率
- ✧ 支持可选基准电压，支持触摸按键扫描中断，支持触摸按键唤醒

中断
- ✧ INT0X、T0、INT1X、T1、UART0、TK、ADC、T2、SPI、PWM1、PWM0、PWM1FB 、UART1、CRC、T3 共 15 个中断源
- ✧ 2 级中断优先级可设
- ✧ 其中 INTnx（n=0～1，x=0～4）支持多重映射输入，5 选 1 分别对应两个中断源（INT0x/INT1x）；支持上升沿触发方式和高电平脉宽测量功能

循环冗余校验算法模块（符合 CRC-16 标准）

双两线调试与编程接口
- ✧ 两组调试和编程接口任意二选一，支持自动识别

开发工具兼容 KEIL™ 集成开发环境

工作电压
- ✧ 2.0V ~5.5V

工作环境温度
- ✧ -40℃~85℃

封装形式
- ✧ SOP28、SOP24、SSOP24、SOP20、TSSOP20

本公司保留对以下所有产品在可靠性、功能和设计方面的改进作进一步说明的权利。
同时保留在未通知的情况下，对本文档做更改的权利。

1 产品简介

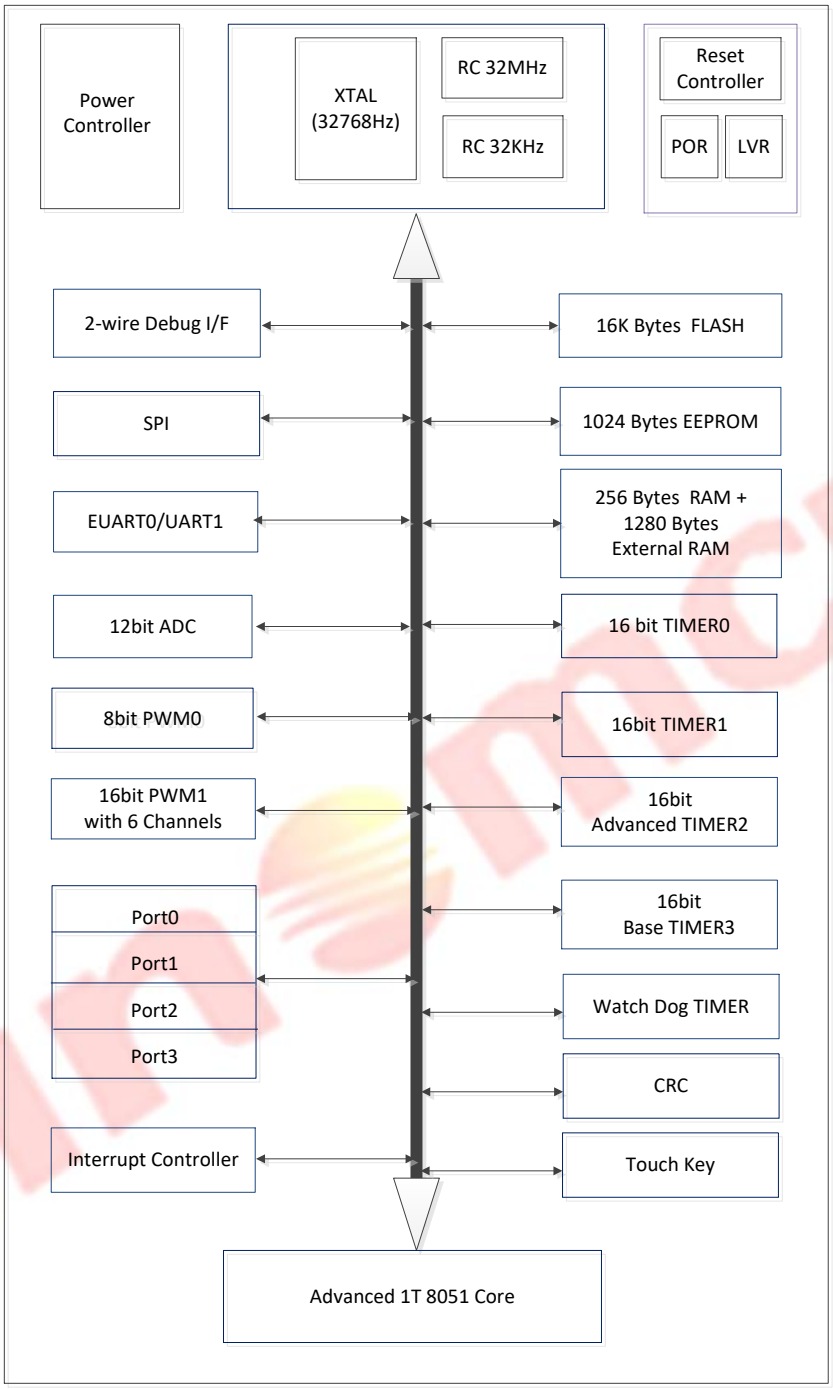
1.1 概述

本产品是一款高速低功耗 1T 周期 8051 内核 8 位增强型 FLASH 微控制器芯片，较传统 8051 相比，运行效率更高。

MC51F8124 的最高运行频率为 16MHz，片上集成了 16K 字节 FLASH ROM、1024 字节 EEPROM 和 1536 字节 SRAM（256 字节内部 RAM 加 1280 字节外部 RAM），内置 1 个高频 RC 振荡器和 1 个低频 RC 振荡器以及预留了 1 个外部 32768Hz 晶体振荡器接口，最大包含 26 个双向 GPIO 口，集成灵活的二选一调试和编程接口，具有 4 个 16 位定时器、3 组 6 通道 16 位共周期且占空比可调的 PWM 定时器（带有死区互补和刹车保护功能）、1 路 8 位独立 PWM 定时器（也可做通用定时器用），集成了 1 个 14 通道高精度 12 位 ADC 模块，以及 2 路 UART、1 路 SPI 通信接口，支持软件 LCD 功能，同时内部集成 4 级可选的 LVR、看门狗定时器、CRC 校验等容错功能，极大地提高了芯片的可靠性。

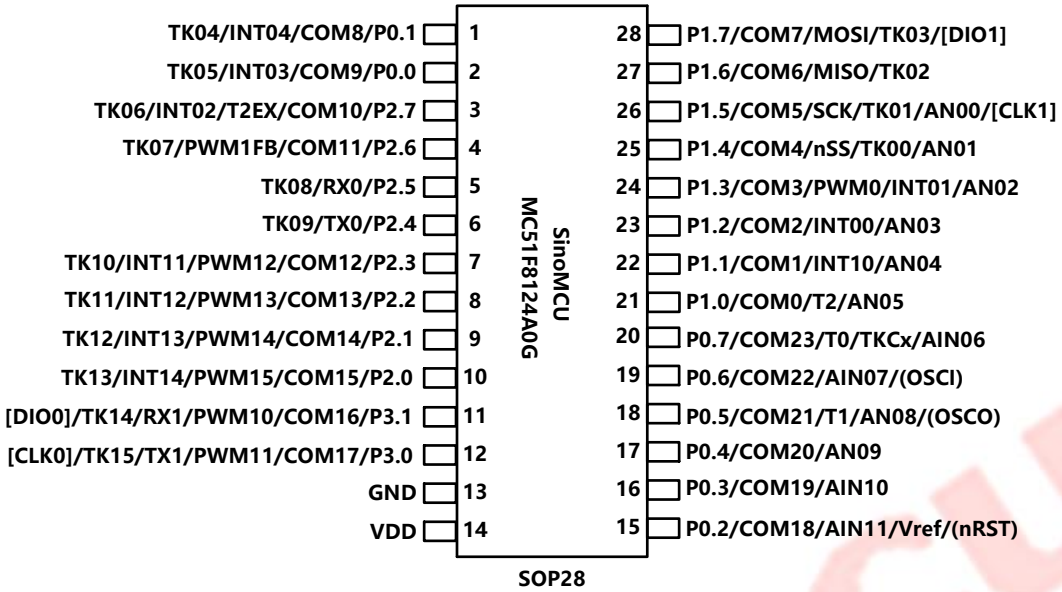
本产品为工业级 IC，具有较强的抗干扰性能，又内部集成了触摸单元，简化了系统设计，降低了整体成本，可广泛用于各种带触摸按键的场合、以及智能小家电领域。

1.2 系统框图

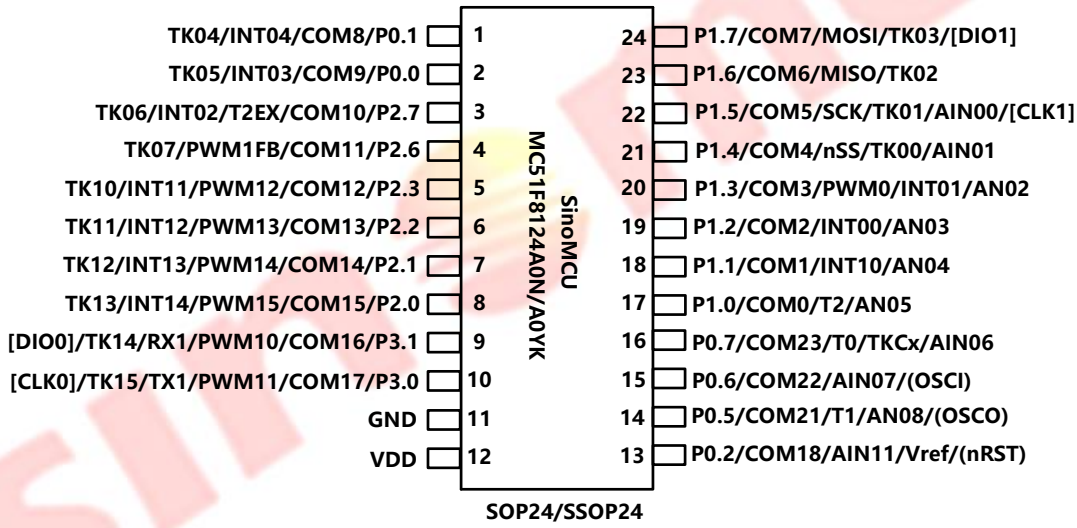

Power Controller
XTAL (32768Hz)
RC 32MHz
RC 32KHz
Reset Controller
POR
LVR
2-wire Debug I/F
SPI
EUART0/UART1
12bit ADC
8bit PWM0
16bit PWM1 with 6 Channels
Port0
Port1
Port2
Port3
Interrupt Controller
16K Bytes FLASH
1024 Bytes EEPROM
256 Bytes RAM + 1280 Bytes External RAM
16 bit TIMER0
16bit TIMER1
16bit Advanced TIMER2
16bit Base TIMER3
Watch Dog TIMER
CRC
Touch Key
Advanced 1T 8051 Core

1.3 引脚排列

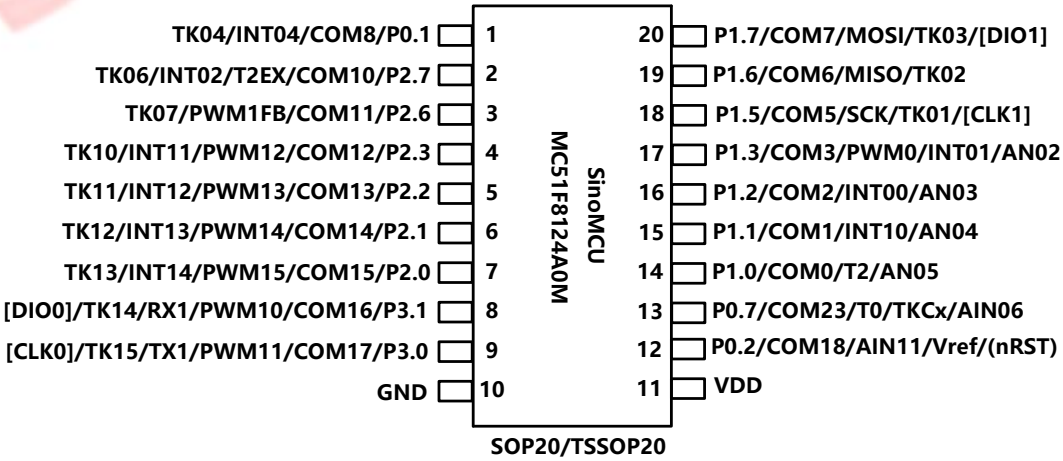
1.3.1 MC51F8124A0G



1.3.2 MC51F8124A0N/A0YK



1.3.3 MC51F8124A0M/A0Y



1.4 引脚说明

引脚汇总表

引脚编号			端口名称 （最左边优先级最高）	属性	端口功能 （最上边优先级最低）
SOP28	SOP24/ SSOP24	SOP20			
1	1	1	TK04/INT04/COM8/P0.1	I/O	P0.1：GPIO 组 0 的引脚 1 端口
					COM8：LCD 驱动公共端 COM8 端口
					INT04：外部中断 0 输入通道 4 端口
					TK04: TK 通道 4 输入端口
2	2	-	TK05/INT03/COM9/P0.0	I/O	P0.0：GPIO 组 0 的引脚 0 端口
					COM9：LCD 驱动公共端 COM9 端口
					INT03：外部中断 0 输入通道 3 端口
					TK05: TK 通道 5 输入端口
3	3	2	TK06/INT02/T2EX/COM10/P2.7	I/O	P2.7：GPIO 组 2 的引脚 7 端口
					COM10：LCD 驱动公共端 COM10 端口
					T2EX：定时器 2 的重载触发/输入捕捉输入端口
					INT02：外部中断 0 输入通道 2 端口
					TK06: TK 通道 6 输入端口
4	4	3	TK07/PWM1FB/COM11/P2.6	I/O	P2.6：GPIO 组 2 的引脚 6 端口
					COM11：LCD 驱动公共端 COM11 端口
					PWM1FB：PWM1 刹车控制输入端口
					TK07: TK 通道 7 输入端口
5	-	-	TK08/RX0/P2.5	I/O	P2.5：GPIO 组 2 的引脚 5 端口
					RX0: UART0 的 RXD 端口
					TK08: TK 通道 8 输入端口
6	-	-	TK09/TX0/P2.4	I/O	P2.4：GPIO 组 2 的引脚 4 端口
					TX0: UART0 的 TXD 端口
					TK09: TK 通道 9 输入端口
7	5	4	TK10/INT11/PWM12/COM12/ P2.3	I/O	P2.3：GPIO 组 2 的引脚 3 端口
					COM12：LCD 驱动公共端 COM12 端口
					PWM12：PWM1 通道 2 输出端口
					INT11：外部中断 1 输入通道 1 端口
					TK10: TK 通道 10 输入端口
8	6	5	TK11/INT12/PWM13/COM13/ P2.2	I/O	P2.2：GPIO 组 2 的引脚 2 端口
					COM13：LCD 驱动公共端 COM13 端口
					PWM13：PWM1 通道 3 输出端口
					INT12：外部中断 1 输入通道 2 端口
					TK11: TK 通道 11 输入端口
9	7	6	TK12/INT13/PWM14/COM14/ P2.1	I/O	P2.1：GPIO 组 2 的引脚 1 端口
					COM14：LCD 驱动公共端 COM14 端口
					PWM14：PWM1 通道 4 输出端口
					INT13：外部中断 1 输入通道 3 端口
					TK12: TK 通道 12 输入端口

引脚编号			端口名称 (最左边优先级最高)	属性	端口功能 (最上边优先级最低)
SOP28	SOP24/ SSOP24	SOP20			
10	8	7	TK13/INT14/PWM15/COM15/ P2.0	I/O	P2.0：GPIO 组 2 的引脚 0 端口
					COM15：LCD 驱动公共端 COM15 端口
					PWM15：PWM1 通道 5 输出端口
					INT14：外部中断 1 输入通道 4 端口
					TK13: TK 通道 13 输入端口
11	9	8	[DIO0]/TK14/RX1/PWM10/ COM16/P3.1	I/O	P3.1：GPIO 组 3 的引脚 1 端口
					COM16：LCD 驱动公共端 COM16 端口
					PWM10：PWM1 通道 0 输出端口
					RX1: UART1 的 RXD 端口
					TK14: TK 通道 14 输入端口
					DIO0：组 0 调试和编程的数据端口
12	10	9	[CLK0]/TK15/TX1/PMW11/ COM17/P3.0	I/O	P3.0：GPIO 组 3 的引脚 0 端口
					COM17：LCD 驱动公共端 COM17 端口
					PWM11：PWM1 通道 1 输出端口
					TX1: UART1 的 TXD 端口
					TK15: TK 通道 15 输入端口
					CLK0：组 0 调试和编程的时钟端口
13	11	10	GND	P	GND：芯片接地端口
14	12	11	VDD	P	VDD：芯片接电源端口
15	13	12	(nRST)/Vref/AIN11/COM18/P0.2	I/O	P0.2：GPIO 组 0 的引脚 2 端口
					COM18：LCD 驱动公共端 COM18 端口
					AIN11：ADC 输入通道 11 端口
					Vref：ADC 外部参考源输入端口
					nRST：外部 RESET 输入端口(低电平有效)
16	-	-	AIN10/COM19/P0.3	I/O	P0.3：GPIO 组 0 的引脚 3 端口
					COM19：LCD 驱动公共端 COM19 端口
					AIN10：ADC 输入通道 10 端口
17	-	-	AIN09/COM20/P0.4	I/O	P0.4：GPIO 组 0 的引脚 4 端口
					COM20：LCD 驱动公共端 COM20 端口
					AIN09：ADC 输入通道 9 端口
18	14	-	(OSCO)/AIN08/T1/COM21/P0.5	I/O	P0.5：GPIO 组 0 的引脚 5 端口
					COM21：LCD 驱动公共端 COM21 端口
					T1：定时器 1 的输入/输出端口
					AIN08：ADC 输入通道 8 端口
					OSCO：外部 RTC 晶振输出端口
19	15	-	(OSCI)/AIN07/COM22/P0.6	I/O	P0.6：GPIO 组 0 的引脚 6 端口
					COM22：LCD 驱动公共端 COM22 端口
					AIN07：ADC 输入通道 7 端口
					OSCI：外部 RTC 晶振输入端口
20	16	13	AIN06/TKCx/T0/COM23/P0.7	I/O	P0.7：GPIO 组 0 的引脚 7 端口

引脚编号			端口名称 (最左边优先级最高)	属性	端口功能 (最上边优先级最低)
SOP28	SOP24/ SSOP24	SOP20			
					COM23：LCD 驱动公共端 COM23 端口
					T0：定时器 0 的输入/输出端口
					TKCx: TK 通道的外接 Cx 端口
					AIN06：ADC 输入通道 6 端口
21	17	14	AN05/T2/COM0/P1.0	I/O	P1.0：GPIO 组 1 的引脚 0 端口
					COM0：LCD 驱动公共端 COM0 端口
					T2：定时器 2 的输入/输出端口（含输出比较）
					AIN05：ADC 输入通道 5 端口
22	18	15	AN04/INT10/COM1/P1.1	I/O	P1.1：GPIO 组 1 的引脚 1 端口
					COM1：LCD 驱动公共端 COM1 端口
					INT10：外部中断 1 输入通道 0 端口
					AIN04：ADC 输入通道 4 端口
23	19	16	AN03/INT00/COM2/P1.2	I/O	P1.2：GPIO 组 1 的引脚 2 端口
					COM2：LCD 驱动公共端 COM2 端口
					INT00：外部中断 0 输入通道 0 端口
					AIN03：ADC 输入通道 3 端口
24	20	17	AN02/INT01/PWM0/COM3/P1.3	I/O	P1.3：GPIO 组 1 的引脚 3 端口
					COM3：LCD 驱动公共端 COM3 端口
					PWM0：独立 PWM0 输出端口
					INT01：外部中断 0 输入通道 1 端口
					AIN02：ADC 输入通道 2 端口
24	21	-	AN01/TK00/nSS/COM4/P1.4	I/O	P1.4：GPIO 组 1 的引脚 4 端口
					COM4：LCD 驱动公共端 COM4 端口
					nSS：SPI 的 SS 端口(低电平有效)
					TK00: TK 通道 0 输入端口
					AIN01：ADC 输入通道 1 端口
26	22	18	[CLK1]/AN00/TK01/SCK/COM5/ P1.5	I/O	P1.5：GPIO 组 1 的引脚 5 端口
					COM5：LCD 驱动公共端 COM5 端口
					SCK：SPI 的 CLK 端口
					TK01: TK 通道 1 输入端口
					AIN00：ADC 输入通道 0 端口
					CLK1：组 1 调试和编程的时钟端口
27	23	19	TK02/MISO/COM6/P1.6	I/O	P1.6：GPIO 组 1 的引脚 6 端口
					COM6：LCD 驱动公共端 COM6 端口
					MISO：SPI 的 MISO 端口
					TK02:TK 通道 2 输入端口
28	24	20	[DIO1]/TK03/MOSI/COM7/P1.7	I/O	P1.7：GPIO 组 1 的引脚 7 端口
					COM7：LCD 驱动公共端 COM7 端口
					MOSI：SPI 的 MOSI 端口
					TK03: TK 通道 3 输入端口

引脚编号			端口名称 （最左边优先级最高）	属性	端口功能 （最上边优先级最低）
SOP28	SOP24/ SSOP24	SOP20			
					DIO1：组 1 调试和编程的数据端口

1.5 订购信息

产品名称	封装形式	备注
MC51F8124A0G	SOP28	-
MC51F8124A0N	SOP24	-
MC51F8124A0YK	SSOP24	
MC51F8124A0M	SOP20	-
MC51F8124A0Y	TSSOP20	

注：产品名称中的倒数第 2 位 x 代表本产品的出货包装信息，最终的具体类型请与我司销售人员确认。

1.6 基本概念和术语

- **R/W**：代表 Read/Write 的含义，即可读可写。当仅有 R 时，代表只读；当仅有 W 时，代表只写。
- **R*/W,R/W***：R*代表仅可读到 0 或 1；W*代表仅可写 0 或 1。
- **0/1/X/-**：其中 0=数字逻辑 0（低电平）；1=数字逻辑 1（高电平）；X=不确定（高低电平皆可）；-=保留，有点类似于 X（高低电平皆可）。
- **B/b**：B 代表 Byte 的含义，即字节，表示了 8 位数据长度；b 代表 bit 的含义，即位，表示 1 位数据长度。
- **双两线调试接口（Two-wire）**：是一种类似于 SWD 的调试接口形式，是晟矽微电子特有的调试/编程接口。为用户精简了调试/编程接口电路，方便了硬件工程师的 PCB 设计；通过两组调试接口交替仿真，让工程师可以做到“所有端口的全覆盖仿真”。
- **ICP**：代表在电路编程，即 In Circuit Programming 的首字母缩写。用户可通过电路板上预留的两线调试和编程接口（Two-wire），在印刷电路板组装以后（PCBA），直接将程序下载到芯片中。
- **IAP**：代表在应用编程，即 In Application Programming 的首字母缩写。只要芯片可以运行，用户就可以通过程序对芯片本身进行编程。
- **FLASH 的分区保护功能**：用户可以部分或者全部保护 FLASH 区。可以防范用 Flash 区中的代码在不可信的情况下被读出，也可以防范程序跑飞对 Flash 的意外擦除或者内容改写，还可以为用户的二次开发提供有效手段，这是一种高可靠性的安全防护措施。

2 电气参数

下述无特殊说明的情况时，Ta=25℃，VDD-GND=5V。

2.1 极限参数

参数	符号	值	单位
直流供电电压	VDD/GND	-0.3～5.5	V
IO 输入/输出电压	Vin/Vout	GND-0.3～VDD+0.3	V
工作环境温度	Ta	-40～85	℃
储存环境温度	Tstg	-55～150	℃
片上 FLASH/EEPROM 操作温度	Tflash	-40～85	℃

*注：流过 VDD、GND 电流总和的最大值在 5.0V/25℃ 下必须小于 **200mA**。本规格仅采用本芯片在上述条件内测试获得，对超出上述条件的“特殊应用”不做保证。若芯片长期处于极限参数的最大或者最小条件下工作，可能会直接影响器件的可靠性并加速器件的老化。*

2.2 推荐工作条件

参数	符号	最小值	最大值	单位	备注
工作电压 1	VDD-GND	3.5	5.5	V	CORE 可运行于 HIRC@8～16MHz 建议开启 LVR=3.5V 或者 4.1V
工作电压 2	VDD-GND	2.1	5.5	V	CORE 可运行于 HIRC@1～8MHz 或 LIRC@32KHz，建议按需开启 LVR
工作环境温度	Ta	-40	85	℃	-

2.3 直流特性参数

特性	符号	引脚	条件	最小	典型	最大	单位
输入低电压	Vil	所有输入脚	IO 输入模式下	GND-0.3	-	0.2VDD	V
输入高电压	Vih	所有输入脚	IO 输入模式下	0.8VDD	-	VDD+0.3	V
输入漏电流	Iilc	所有输入脚	IO 输入模式下，上拉电阻关闭，Vin=VDD 或者 GND	-1	-	1	uA
输出高电流（推挽输出方式）	Ioh01	P0.0,P0.1,P2,P3	IO 输出模式下，Voh=0.9VDD，档位配置参见<6.6 章节>PXODM 寄存器	-	10	-	mA
	Ioh02			-	20	-	
	Ioh11	P0.2-P0.7,P1（可控大电流端口）		-	10	-	
	Ioh12			-	20	-	
	Ioh13			-	30	-	
	Ioh14			-	40	-	
输出低电流（推挽输出方式）	Iol01	P0.0,P0.1,P2,P3	IO 输出模式下，Vol=0.1VDD，档位配置参见<6.6 章节> PXODM 寄存器	-	20	-	mA
	Iol02			-	40	-	
	Iol11	P0.2-P0.7,P1（可控大电流端口）		-	20	-	
	Iol12			-	40	-	
	Iol13			-	60	-	
	Iol14			-	80	-	
内部上拉电阻	Rpu	P0,P1,P2,P3	VDD=5V，上拉电阻使能	-	20	-	KΩ

特性	符号	引脚	条件	最小	典型	最大	单位
RESET 输入低电压	Vilrst	P0.2	P0.2 做 RESET 使用时（option 配置）	GND	-	0.2VDD	V
RESET 输入高电压	Vihrst	P0.2	P0.2 做 RESET 使用时（option 配置）	0.8VDD	-	VDD	V
RESET 引脚的内部上拉电阻	Rrst	P0.2	P0.2 做 RESET 使用时（option 配置）	20	-	600	KΩ
工作电流（所有未说明的功能模块关闭；所有引脚无负载；CORE 执行 NOP 指令）	Iddc	VDD	VDD=5V, F_{CORE}=16MHz@HIRC	-	12	-	mA
			VDD=5V, F_{CORE}=8MHz@HIRC	-	6.5	-	mA
			VDD=5V, F_{CORE}=2MHz@HIRC	-	2.5	-	mA
			VDD=5V, F_{CORE}=1MHz@HIRC	-	1.7	-	mA
			VDD=5V, F_{CORE}=32KHz@LIRC, HIRC 关, XTAL 关	-	50	-	uA
待机电流（停止模式）	Istop	VDD	VDD=5V,F_{CORE}=16MHz@HIRC; CORE 停止, LIRC 关, XTAL 关; WDT 关, LVR 开；所有引脚无负载, INTn 唤醒开; 其他功能模块关	-	6.5	-	mA
			VDD=5V,F_{CORE}=32KHz@LIRC；CORE 停止，HIRC 关，XTAL 关；WDT 关，LVR 开；所有引脚无负载，INTn 唤醒开；其他功能模块关	-	50	-	uA
待机电流（休眠模式）	Isleep	VDD	VDD=5V，CORE 休眠，HIRC/LIRC 关，XTAL 关；WDT 关，LVR 关；所有引脚无负载，INTn 唤醒开；其他功能模块关	-	5	-	uA
待机后定时唤醒的平均电流（休眠模式）	Isa	VDD	VDD=5V, CORE 休眠, HIRC 关, XTAL 关， LIRC 开, T3 的 1S 定时唤醒开; WDT 关, LVR 开；所有引脚无负载；其他功能模块关	-	12	-	uA
WDT 电流	Iwdt	VDD	VDD=5V, CORE 休眠, HIRC/LIRC 关, XTAL 关; WDT 工作；其他功能模块关	-	3.5	-	uA
LVR 电流	Ilvr	VDD	VDD=5V, CORE 休眠, HIRC/LIRC 关, XTAL 关; LVR 工作；其他功能模块关	-	2.5	-	uA
RAM 保持电压	V_{RAM}	VDD	-	-	0.7	-	V

2.4 LVR 特性参数

特性	符号	条件	最小	典型	最大	单位
LVR 设定电压 1	Vlvr1	LVR 使能，VDD=1.8～5.5V，偏差在±5%之内	-	2.1	-	V
LVR 设定电压 2	Vlvr2		-	2.5	-	V
LVR 设定电压 3	Vlvr3		-	3.5	-	V
LVR 设定电压 4	Vlvr4		-	4.1	-	V

2.5 ADC 特性参数

特性	符号	条件	最小	典型	最大	单位
ADC 有效工作电压	Vadc	-	2.5	-	5.5	V
分辨率	Nr	-	-	12	-	bit
积分线性误差	INL	VDD=5V，Fadc=1-8MHz，采样时间=8Clk，12bit 精度	-	-	±2	LSB
微分线性误差	DNL		-	-	±1	LSB
上限偏置误差	Eot		-	±1	±4	LSB
下限偏置误差	Eob		-	±1	±4	LSB
零点偏移误差	Ez		-	±0.5	±4	LSB
总绝对误差	Ead		-	±4	-	LSB
转换时钟	Fadc	VDD=5V	-	-	8	MHz
		VDD=3V	-	-	4	MHz
转换时间	Tcon	VDD=5V，Fadc=8MHz，采样时间=4Clk，12bit 精度	16	-	-	Tadc
ADC 输入电压	Vain	-	GND	-	VREF	V
ADC 输入阻抗	Rain	-	2	-	-	MΩ
ADC 输入电流	Iain	-	-	-	10	uA
ADC 动态电流	Iadd	AD 转换中	-	1	1.5	mA
ADC 静态电流	Iads	ADEN 位=0	-	0.1	1	uA
模拟信号源推荐阻抗	Zain	-	-	-	10	KΩ
内部参考电压	Vref	@VDD≥2.5V	1.96	2.0	2.04	V
内部基准源建立时间	Tvref	VDD=2.5~5.5V	-	-	100	us

注：上表数据为批量测试所得，芯片出厂不做精度测试。

2.6 片上 FLASH 和 EEPROM 特性

特性	符号	条件	最小	典型	最大	单位
FLASH 读写循环次数	N_{FNDUR}	-	100000	-	-	cycle
FLASH 数据保存时间	T_{FRET}	-	10	-	-	year
EEPROM 操作电压	V_{EEWR}	T=-40℃~85℃，Fcore = 8~16MHz	2.8	-	5.5	V
		T=-40℃~85℃ Fcore = 32KHz~8MHz	2.0		5.5	V
EEPROM 平均读取 1 字节时间	T_{EERD}	T=-40℃~85℃, VDD=2.0V~5.5V(连续擦除 1KB 后取平均值), Fcore=16MHz	-	100	-	ns
EEPROM 平均读取 1 字节电流	I_{EERD}	T=-40℃~85℃, VDD=2.0V~5.5V(连续擦除 1KB 后取平均值), Fcore=16MHz	-	-	80	uA/MHz
EEPROM 平均擦除 1 字节时间	T_{EEES}	T=-40℃~85℃, VDD=2.0V~5.5V(连续擦除 1KB 后取平均值), Fcore=16MHz	-	2.8	-	ms
EEPROM 平均擦除 1 字节电流	I_{EEES}	T=-40℃~85℃, VDD=2.0V~5.5V(连续擦除 1KB 后取平均值), Fcore=16MHz	-	2	-	mA
EEPROM 平均编程 1 字节时间	T_{EEWR}	T=-40℃~85℃, VDD=2.0V~5.5V(连续编程 1KB 后取平均值), Fcore=16MHz	-	1.7	-	ms
EEPROM 平均编程 1 字节电流	I_{EEWR}	T=-40℃~85℃, VDD=2.0V~5.5V(连续编程 1KB 后取平均值), Fcore=16MHz	-	2	-	mA

特性	符号	条件	最小	典型	最大	单位
EEPROM 待机电流	I_{EESB1}	-	-	1	-	uA
	I_{EESB2}	T=85℃, VDD=5V	-	5	-	uA
EEPROM 读写循环次数	N_{ENDUR}	-	100000	-	-	cycle
EEPROM 数据保存时间	T_{ERET}	-	10	-	-	year

***注：**片上 FLASH 不支持 IAP，但可以通过 ICP 进行操作；片上 EEPROM 既支持 IAP 又支持 ICP，但当 IAP 操作时芯片的 F_{SYS} 必须处于 HIRC 且工作频率要大于等于 2MHz 时钟的条件下，F_{SYS} 在 LIRC 时钟或者外部 OSC 时钟下禁止 IAP 操作。*

2.7 交流电气参数

特性	符号	条件	最小	典型	最大	单位
HIRC 振荡频率	F_{HIRC}	VDD=5V，Ta= 25℃	-1%	32	+1%	MHz
		T=-40℃～85℃, VDD=2.4～5.5V	-3%	32	+3%	MHz
HIRC 起振时间	T_{HIRC}	VDD=2.4～5.5V	-	-	50	us
LIRC 振荡频率	F_{LIRC}	VDD=5V	-50%	32	+50%	KHz
		VDD=3.3V	-50%	24	+50%	KHz
LIRC 起振时间	T_{LIRC}	VDD=5V	-	-	150	us
32768Hz 振荡器起振时间	T_{XTAL}	VDD=5V	-	-	1	s
32768Hz 振荡器匹配电容	C_{XTAL}	VDD=5V	-	10	-	pF
外部复位脉冲宽度	T_{RST}	VDD=5V	10	-	-	us

3 特性曲线

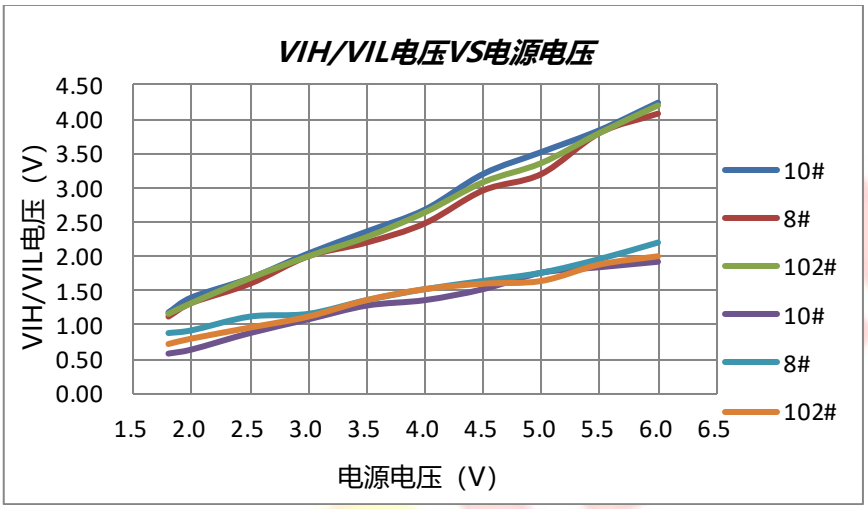
注：

1、特性曲线图中数据均来自抽样实测，仅作为应用参考，部分数据因生产工艺偏差，可能与实际芯片不符；为保证芯片能正常工作，请确保其工作条件符合电气特性参数说明；

2、若图文中无特别说明，则电压特性曲线的温度条件为 T=25℃，温度特性曲线的电压条件为 VDD=5V；

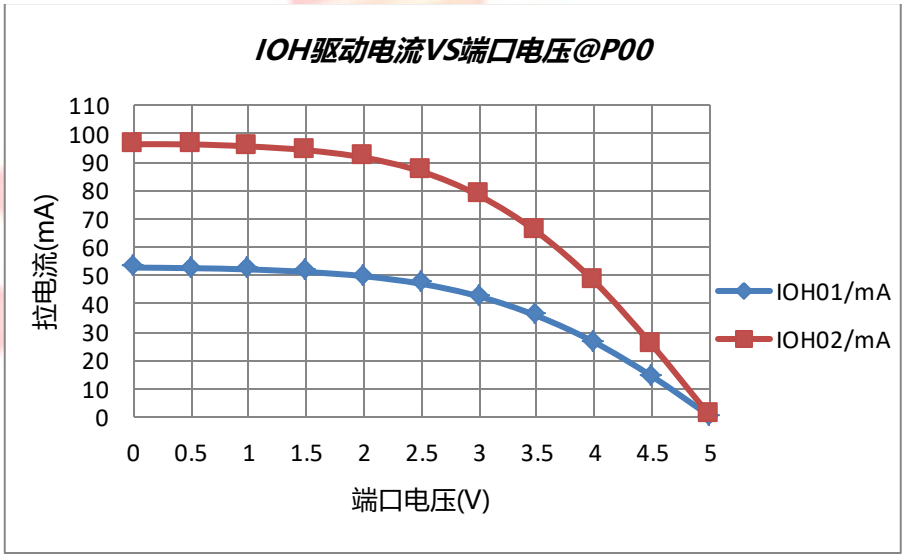
3.1 I/O 特性

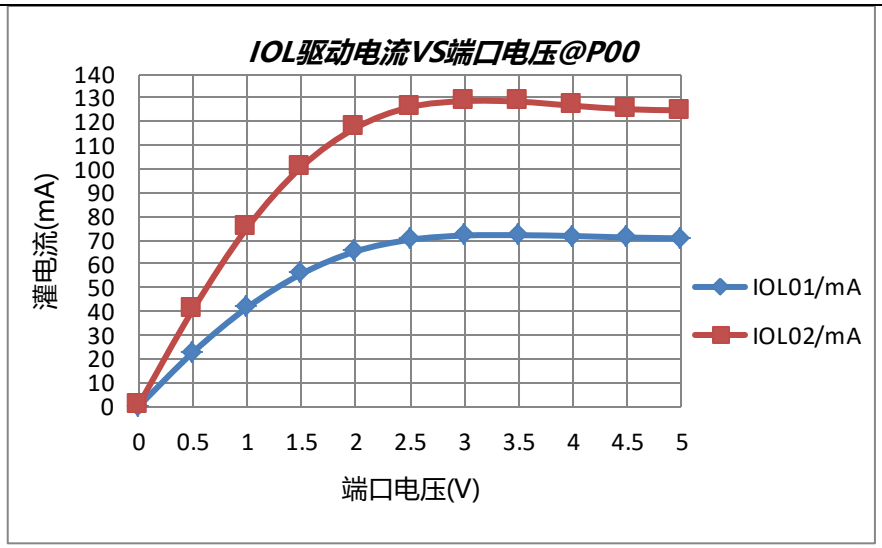
3.1.1 输入电压 VS 电源电压



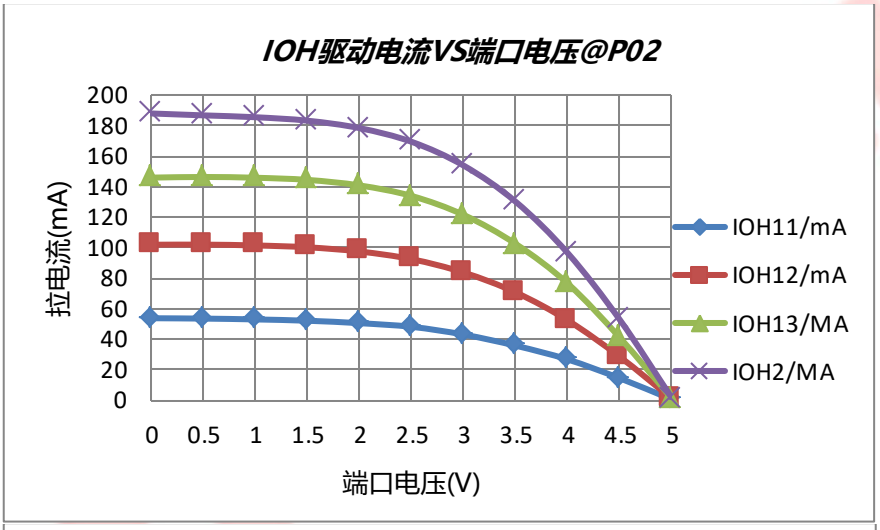
3.1.2 输出驱动电流 VS 输出端口电压

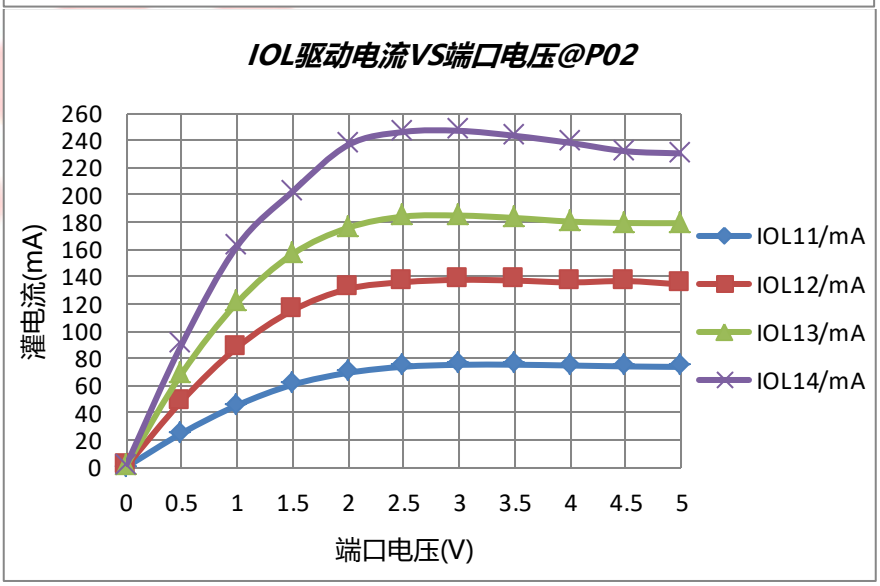
普通 IO 口:



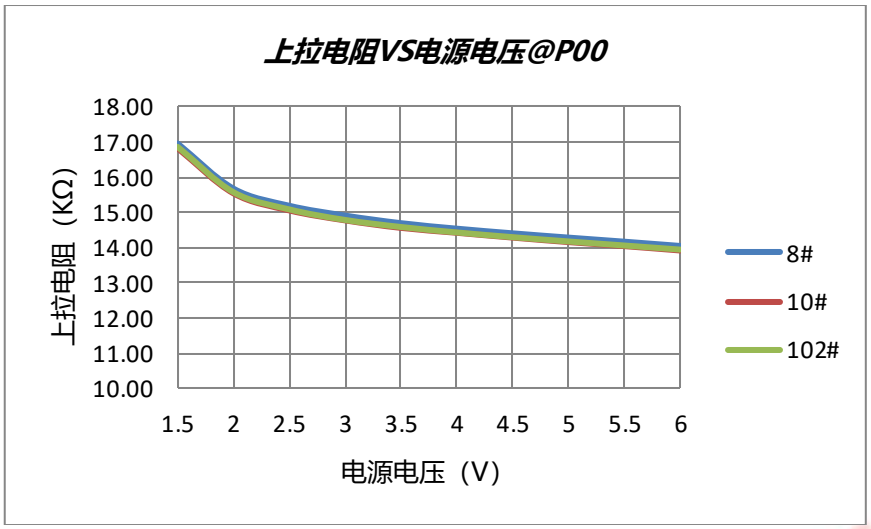

IOL驱动电流VS端口电压@P00
灌电流(mA)
140
130
120
110
100
90
80
70
60
50
40
30
20
10
0
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5
端口电压(V)
IOL01/mA
IOL02/mA

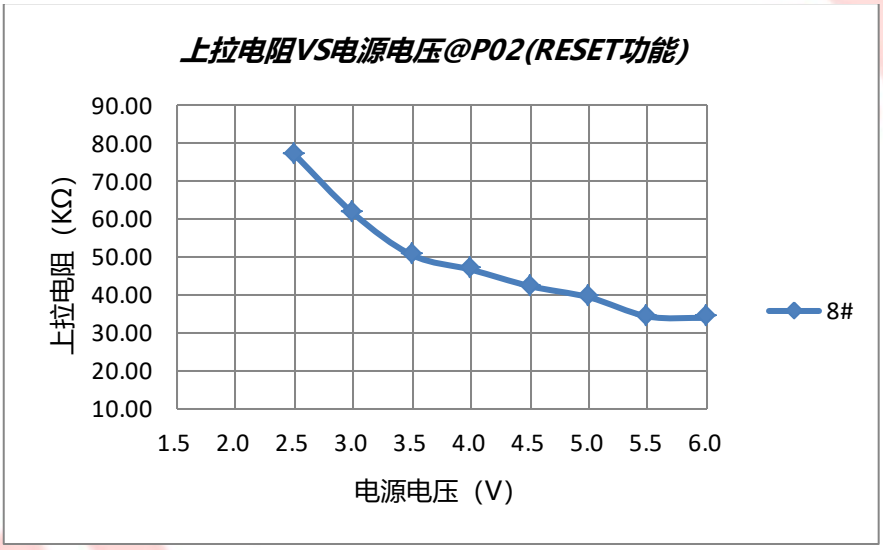
大电流驱动 IO:

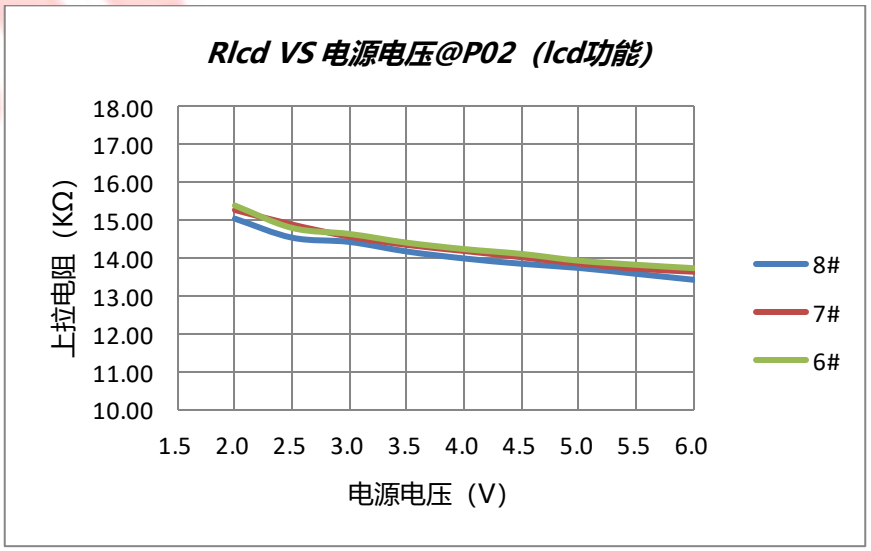

IOH驱动电流VS端口电压@P02
拉电流(mA)
200
180
160
140
120
100
80
60
40
20
0
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5
端口电压(V)
IOH11/mA
IOH12/mA
IOH13/MA
IOH2/MA


IOL驱动电流VS端口电压@P02
灌电流(mA)
260
240
220
200
180
160
140
120
100
80
60
40
20
0
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5
端口电压(V)
IOL11/mA
IOL12/mA
IOL13/mA
IOL14/mA

3.1.3 上拉电阻 VS 电源电压

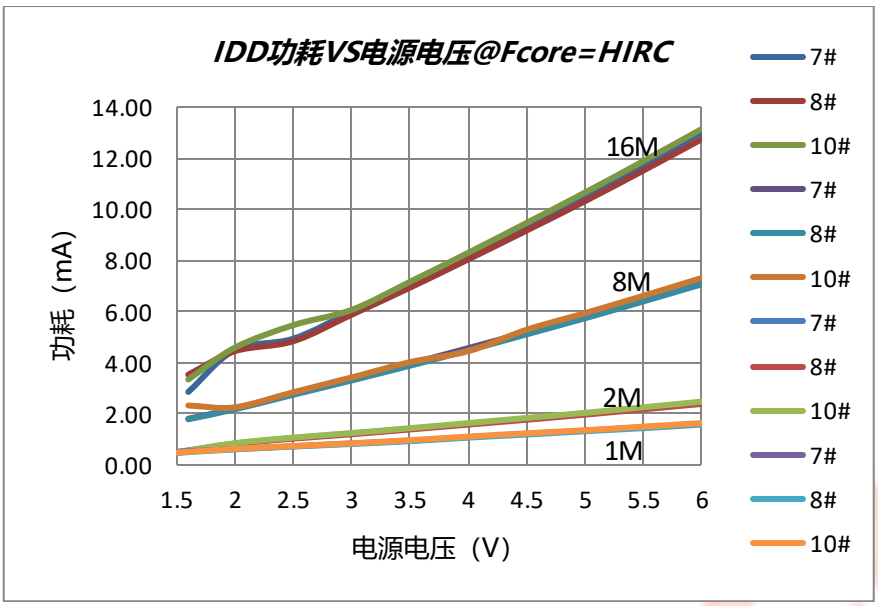


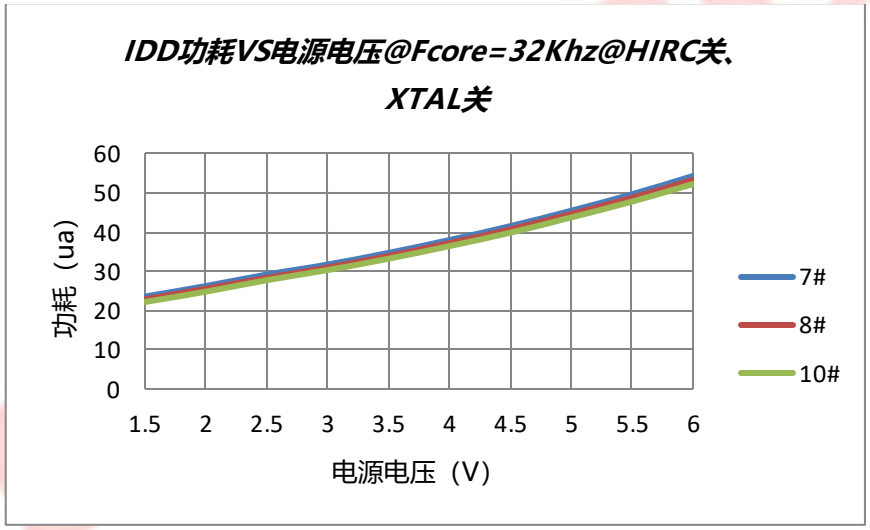




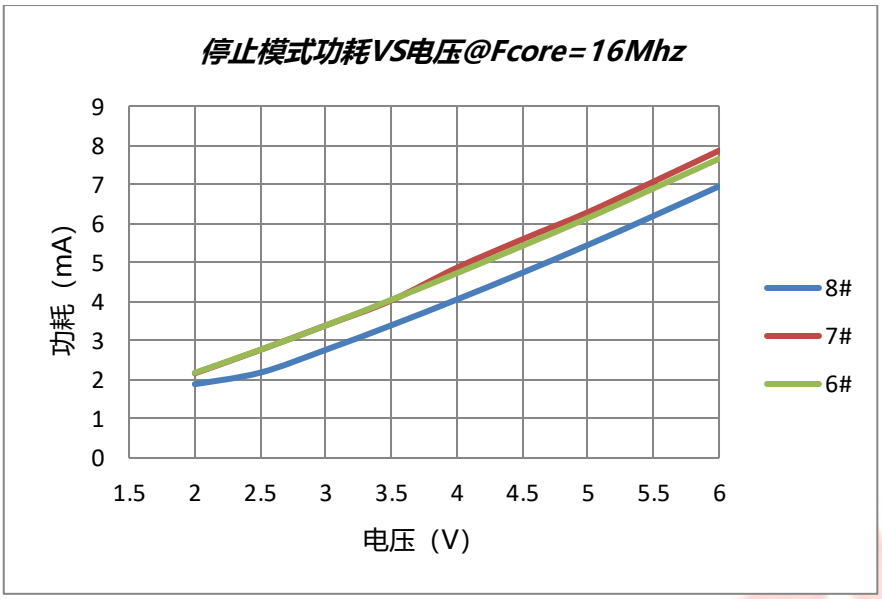
3.2 功耗特性

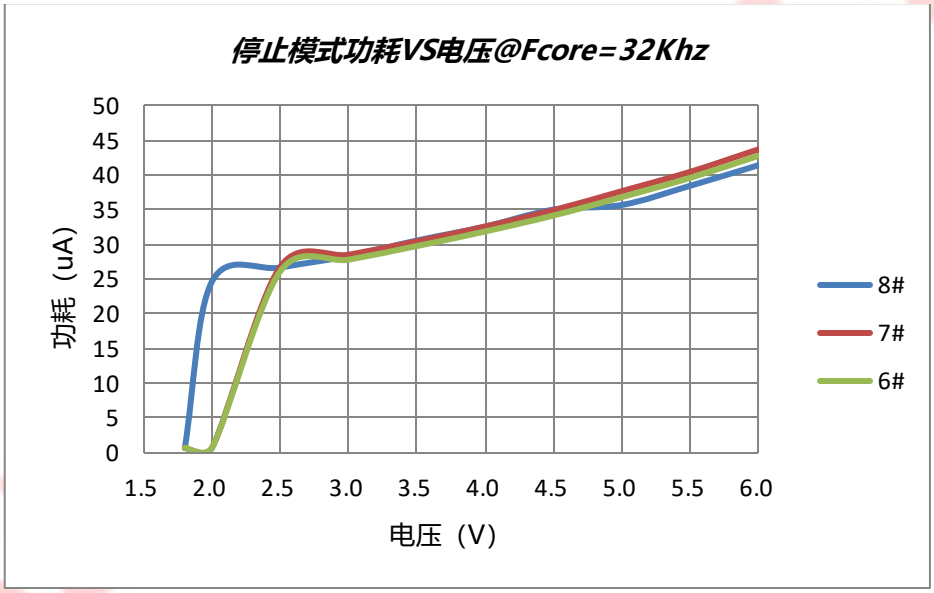
3.2.1 运行功耗 VS 电源电压



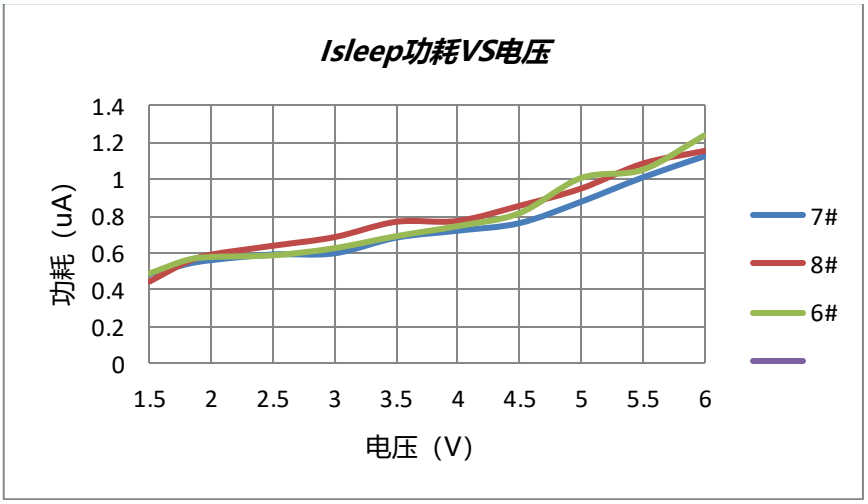


3.2.2 停止模式功耗 VS 电源电压



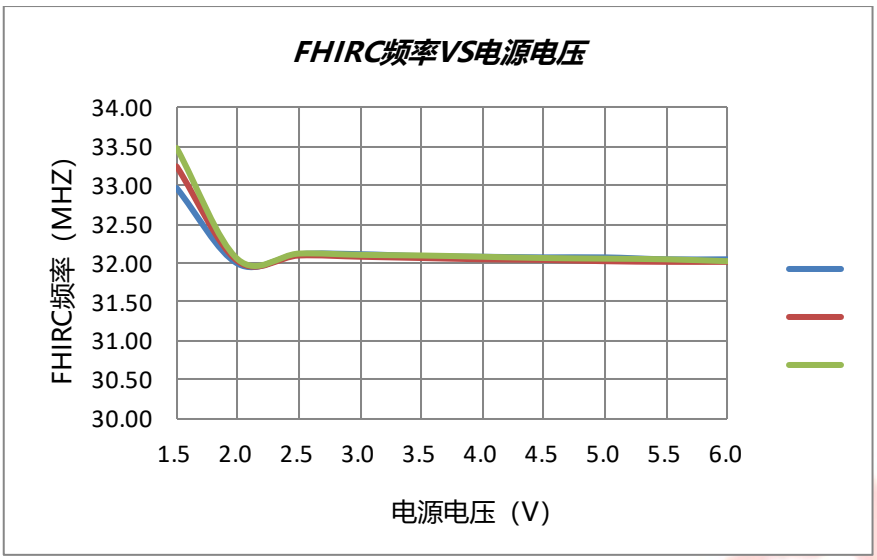


3.2.3 睡眠模式功耗 VS 电源电压

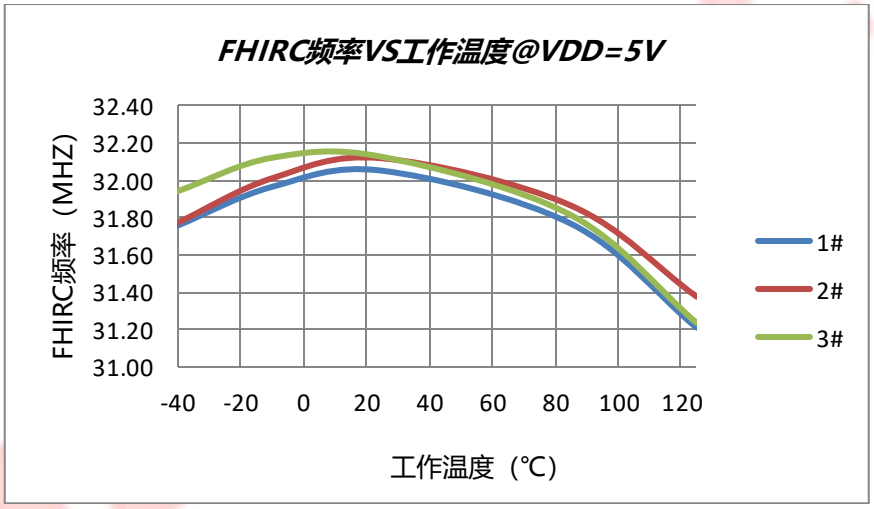


3.3 时钟特性

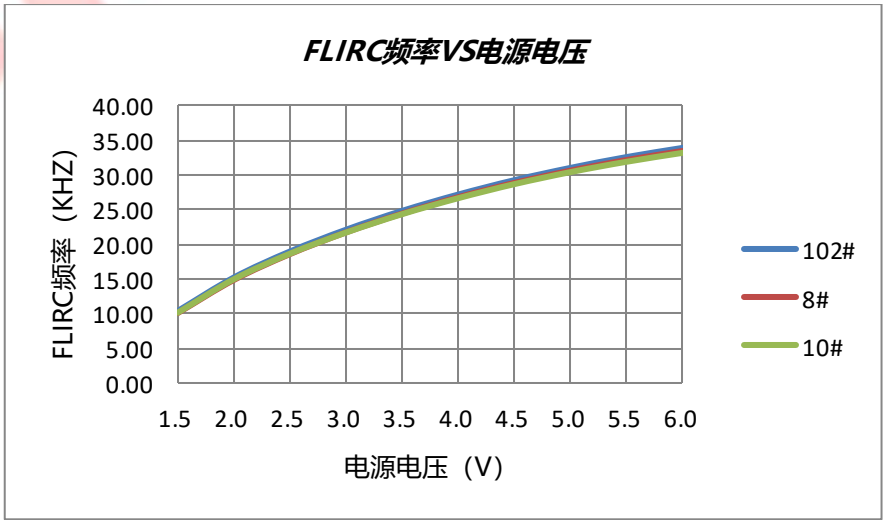
3.3.1 FHIRC VS 电源电压



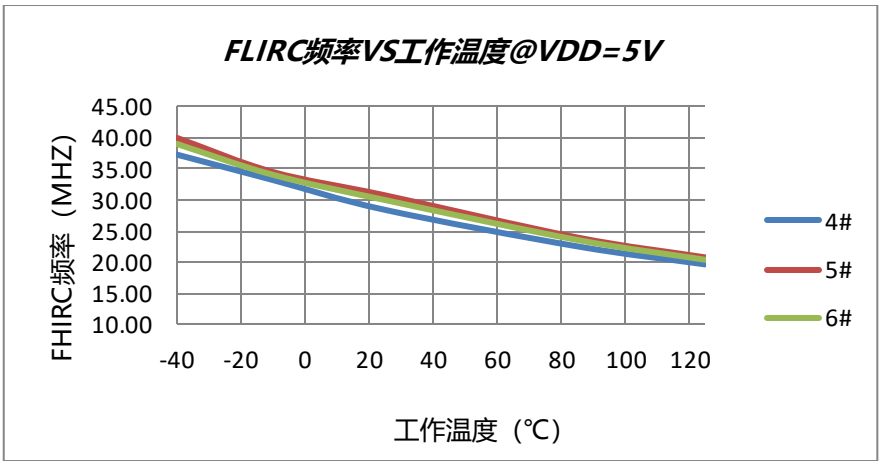
3.3.2 FHIRC VS 工作温度



3.3.3 FLIRC VS 电源电压



3.3.4 FLIRC VS 工作温度



4 中央处理器

4.1 CPU 概述

CORE 为 1T 标准的高速 8051 内核，其指令完全兼容传统 8051 单片机。

CORE 包含复位初值为 0000H 的程序计数器 PC，包含 1 个专用累加器 A、辅助寄存器 B、堆栈指针 SP 和数据指针 DPTR/DPTR1，以及程序状态寄存器 PSW。

堆栈指针 SP 是 1 个 8 位专用寄存器，保存堆栈栈顶的地址信息。在执行 PUSH、子程序调用、中断响应等指令时，SP 先加 1 再将数据压栈；执行 POP、RET、RETI 等指令时，数据退出堆栈后 SP 再减 1。堆栈栈顶可以是片上内部 RAM（00H~FFH）的任意地址，系统复位后，SP 初始化为 07H，堆栈实际从 08H 地址开始，用户可自行调整堆栈栈顶地址。

程序状态寄存器 PSW 用来保存指令执行中的各种状态。

数据指针 DPTR/DPTR1 是两组 16 位专用寄存器，其高字节寄存器为 DPH/DPH1 表示，低字节寄存器为 DPL/DPL1 表示。在一组 DPTR/DPTR1 中，它们既可以作为一个 16 位寄存器来处理，也可作为两个独立 8 寄存器来处理。兼容模式下仅使用 DPTR，但用户可通过设定 DPS 寄存器同时使用这两组，关于 DPS 的详细内容，请参考相关章节内容。

程序状态寄存器（0xD0，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PSW	CY	AC	F0	RS1	RS0	OV	F1	P
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] **CY** – 进/借位标志位

0：算术或逻辑运算中，最高位（Bit7）没有进位或借位发生；

1：算术或逻辑运算中，最高位（Bit7）有进位或借位发生。

BIT[6] **AC** – 进/借位辅助标志位（可在 BCD 码加减法运算时方便调整）

0：算数逻辑运算中，半字节（Bit3）没有辅助进位或借位发生；

1：算数逻辑运算中，半字节（Bit3）有辅助进位或借位发生。

BIT[5] **F0** – 用户自定义标志位

BIT[4:3] **RS[1:0]** – 工作寄存器组选择位

RS[1:0]	当前使用的工作寄存器组 0~3
00	工作寄存器组 0（00H~07H）
01	工作寄存器组 1（08H~0FH）
10	工作寄存器组 2（10H~17H）
11	工作寄存器组 3（18H~1FH）

BIT[2] **OV** – 溢出标志位

0：表示累加器 A 中的结果没有超出 8 位（0xFF）的范围，未发生溢出；

1：表示累加器 A 中的结果超出了 8 位（0xFF）的范围，发生溢出。

BIT[1] **F1** – 用户自定义标志位

BIT[0] **P** – 奇偶校验位

0：表示累加器 A 中的二进制结果中有 1 的个数为偶数个；

1：表示累加器 A 中的二进制结果中有 1 的个数为奇数个。

4.2 程序存储器（FLASH ROM）

程序存储器 FLASH ROM 总的大小为 16K 字节，地址范围为 0000H~3FFFH，该 FLASH ROM 可反复擦写 10 万次，数据保存年限至少 10 年。

FLASH ROM 通常需要先擦除后写入。本芯片的 FLASH ROM 仅支持在电路编程（ICP）。

注：ICP 相关内容，请参考相关章节的内容。

4.3 非易失性数据存储器（EEPROM）

非易失性存储器 EEPROM 总的大小为 1K 字节，独立编址，地址范围为 000H~3FFH，该 EEPROM 可反复擦写 10 万次，数据保存年限至少 10 年。

本芯片的 EEPROM 支持在电路编程（ICP）和在应用编程（IAP），也可通过 MOVC 指令进行寻址。

注：ICP 和 IAP 相关内容请参考相关章节的内容。

4.4 随机存储器（RAM）

随机数据存储器 RAM 包括内部 256 字节和外部 1280 字节共 1536 字节的 SRAM。

内部 RAM 的地址为 00H~FFH，其中高 128 字节（80H~FFH）与特殊功能寄存器 SFR 重叠，且只能间接寻址，低 128 字节（00H~7FH）可直接寻址也可以间接寻址。其中最低的 32 字节（00H~17H）作为通用寄存器分成四组 8 个寄存器，程序视这些寄存器为 R0 到 R7，具体通过程序状态字 PSW 的 RS[1:0]位来选择哪一个寄存器组会被使用；接下来的 16 个字节（20H~2FH）还支持位寻址；剩下的字节（30H~7FH）作为通用 RAM，可用来设置栈指针，用作“栈”使用。

外部 RAM 的地址为 0000H~04FFH，这部分空间必须通过 MOVX 指令来寻址。

另外，从 **04E0H～04FFH 被 TK 所占用**，这部 RAM 只读，也是通过 MOVX 指令寻址。

特殊功能寄存器 SFR 地址为 80H~FFH，只能直接寻址。CORE 自动通过指令区分是访问 SFR 还是内部 RAM 的高 128 字节；SFR 通过 DPS 的 SFRSL0 位切换，拓展了 SFR 的范围。

***注 1：**本芯片不支持 P0 和 P2 口的外部 RAM 扩展功能。*

***注 2：**PSW 的 RS[1:0]相关内容请参考相关章节的内容。*

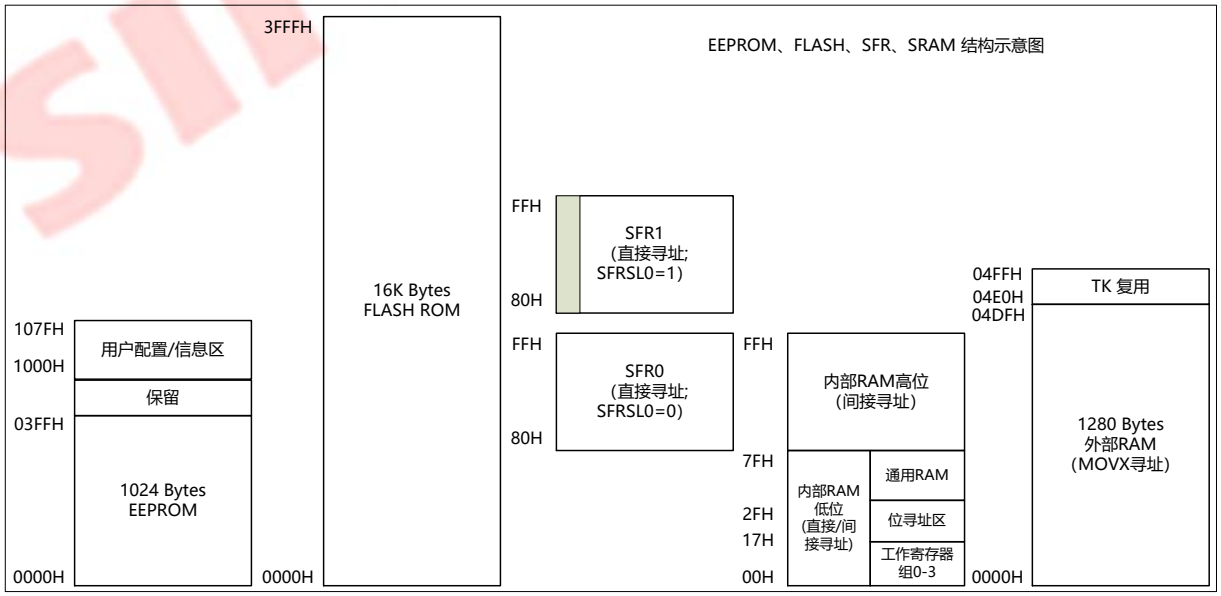
***注 3：**DPS 的 SFRSL0 相关内容请参考相关章节的内容。*

4.5 存储器结构

本芯片的程序存储器 FLASH、随机存储器 RAM 以及非易失性数据存储器 EEPROM 分别是独立编址的。由于没有提供外部存储器扩展的总线接口，所以并不能像传统的 8051 通过 P2 端口进行外扩。

本芯片片上存储器的结构，如下图所示：

存储器 MAP 示意图



该芯片的 FLASH 空间由 8 位宽的存储单元组成，即可以保存代码又可以保存数据。当前 FLASH 有 16 页组成（每页 1KB）或 4 扇区（每个扇区 4KB，即 4 页）组成。在 FLASH“分区保护”的时候，以扇区为操作单位，具体在“用户

配置字”中设定，请参考相关章节的内容。

ROM 空间操作汇总表

操作形式	ICP	IAP
FLASH 代码保护	支持	不支持
FLASH 页擦除	支持	不支持
FLASH 扇区擦除	支持（不保护的情况下）	不支持
FLASH 整体擦除	支持	不支持
FLASH 读/写	支持（不保护的情况下）	不支持
EEPROM 擦除	支持	支持
EEPROM 读/写	支持	支持

4.6 寻址方式

CORE 指令的寻址方式有以下 7 种：立即寻址、直接寻址、间接寻址、寄存器寻址、相对寻址、变址寻址、位寻址。

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。

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。

间接寻址

间接寻址采用 R0 或 R1 前加“@”符号来表示。

寄存器寻址

寄存器寻址寻址时对选定的工作寄存器 R7~R0、累加器 A、通用寄存器 B、地址寄存器和进位标志 CY 中的数进行操作。其中寄存器 R7~R0 由指令码的低 3 位表示，A、B、DPTR 及进位标志 CY 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寄存器 PSW 中的 RS[1:0]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。

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，所以这种寻址方式称为相对寻址。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。

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。

语句“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。

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，借助于进位标志 CY 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。

4.7 特殊功能寄存器映射

特殊功能寄存器 SFR 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。所有的特殊功能寄存器只能通过直接寻址方式进行寻址。

特殊功能寄存器的名称及地址列表

起始地址	可位寻址	不可位寻址							SFR 页
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8H	PWRCR	ADCCR0	ADCCR1	ADRIOS0	ADRIOS1	ADRL	ADRH	OSADJCR	SFR0
		PWRCR1	-	-	-	-	-	PWM145F	SFR1
F0H	B	AUXC	TKCHEN0	TKCHEN1	TKSSN0	TKSSN1	TKCMPH	TKCMPL	SFR0
		PWM15DH	PWM15DL	PWM14DH	PWM14DL	PWM123F	PWM13DH	PWM13DL	SFR1
E8H	RSTFLAG	IAPCR	IAPADL	IAPADH	IAPDB	-	IAPMD	IAPKEY	SFR0
		PWM12DH	PWM12DL	PWM101F	PWM11DH	PWM11DL	PWM10DH	PWM10DL	SFR1
E0H	A	TKCON0	TKCON1	TKCON2	TKCON3	TKVREF	TKANA	TKCHERR	SFR0
		PWM1PH	PWM1PL	PWM1CR3	PWM1CR2	PWM1CR1	PWM1CR0	PWM1IER	SFR1
D8H	DPS	LCDCR0	LCDSW	LCDSW1	LCDSW2	PWM0CR	PWM0P	PWM0D	SFR0
		-	-	-	-	-	-	-	SFR1
D0H	PSW	CRCCR	CRCL	CRCH	-	-	-	-	SFR0
		-	-	-	-	-	-	-	SFR1
C8H	T2CON	T2MOD	RCAPL2	RCAPH2	TL2	TH2	T2CON1	WDTCR	SFR0
		-	-	-	-	-	-	-	SFR1
C0H	OSCM	P0MOD0	P0MOD1	P0PU	TKAMP0	TKAMP1	TKAMP2	TKAMP3	SFR0
		-	-	-	-	-	-	-	SFR1
B8H	IP	IPH	IP1	IPH1	-	-	-	-	SFR0
		-	-	-	-	-	-	-	SFR1
B0H	P3	P3MOD0	-	P3PU	-	-	-	-	SFR0
		-	-	-	-	-	-	-	SFR1
A8H	IE	IE1	-	EXTIS1	EXTIS	SPCON	SPSTA	SPDAT	SFR0
		-	-	EXTEN1	EXTEN	-	-	-	SFR1
A0H	P2	P2MOD0	P2MOD1	P2PU	-	-	-	-	SFR0
		-	-	-	-	-	-	-	SFR1
98H	SCON	SBUF	SADDR	SADEN	SCON_1	SBUF_1	-	PXODM	SFR0
		-	-	-	-	-	-	-	SFR1
90H	P1	P1MOD0	P1MOD1	P1PU	-	-	TL3	TH3	SFR0
		-	-	-	-	-	-	-	SFR1
88H	TCON	TMOD	TL0	TL1	TH0	TH1	TCON1	T3CR	SFR0
		-	-	-	-	-	-	-	SFR1
80H	P0	SP	DPL	DPH	DPL1	DPH1	SLPCR	PCON	SFR0
		-	-	-	-	-	-	-	SFR1

注 1：“-”表示该字节未被使用，这些未使用的 SFR 区域，读出数据不确定，写入数据可能会影响其他 SFR 的值。

注 2：SFR 的页切换相关内容请参考相关章节的内容。

注 3：在 SFR 页切换中“0/8 可位寻址”的这 16 个字节地址固定，无切换。

注 4：灰色底纹位置为系统内部寄存器，禁止访问。

4.8 SFR 的页切换与硬件乘除法器

芯片内部扩展了一页 SFR，即新增的 SFR1 页，用于存放芯片上新增外设的寄存器。

除可位寻址的 16 个 SFR（0/8 地址结尾），其余的 SFR，在 SFR1 页与 SFR0 页上位置是重叠的，通过 DPS 中的 SFRSL0 位进行区分。上电复位后，SFR0 页有效。可位寻址的 16 个 SFR，不需要分页处理，2 种分页情况下都可访问。

芯片内部的扩展了一个新的 AUXC 寄存器，满足了 16 位乘法和 16 位除法。当用户需要完成 16 位*8 位或者 16 位/8 位运算时，通过增强型指令“MUL”和“DIV”可以一次完成这个运算。上电复位后，该功能关闭，仅兼容 51 系列的 8 位*8 位或者 8 位/8 位运算。

数据指针选择寄存器（0xD8，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPS	-	SFRSL0	-	-	DIV16	MUL16	-	SEL
R/W	-	R/W	-	-	R/W	R/W	-	R/W
复位值	-	0	-	-	0	0	-	0

BIT[6] **SFRSL0** – SFR 页切换低位

SFRSL0	对应操作的 SFR 页面	SFR 页号
0	标准 SFR（默认）	SFR0
1	扩展第一页 SFR	SFR1

SFR 页切换，C 语言代码应用举例：

```
sfr DPS = 0xD8;
(假设当前为 SFR0 页的状态)
    DPS |= 0x40;          //由 SFR0 切换至 SFR1
    ......
    DPS &= (~0x40);       //由 SFR1 切换至 SFR0
    ......
```

BIT[3] **DIV16** – 16 位硬件除法器使能位

0：使用 8 位硬件除法器；

1：使用 16 位硬件除法器。

BIT[2] **MUL16** – 16 位硬件乘法器使能位

0：使用 8 位硬件乘法器；

1：使用 16 位硬件乘法器。

寄存器位	对应操作		计算结果		
			AUXC	B	A
MUL16 位	MUL16=0：8 位乘法	(A) * (B)	-	积高位字节	积低位字节
	MUL16=1：16 位乘法	(AUXC A) * (B)	积高位字节	积中位字节	积低位字节
DIV16 位	DIV16=0：8 位除法	(A) / (B)	-	余数	商低位字节
	DIV16=1：16 位除法	(AUXC A) / (B)	商高位字节	余数	商低位字节

4.9 用户配置字

芯片内部包含一块额外的 Info Block 空间用于保存芯片的配置信息。芯片为保证系统正常工作，会将关键模块的配置预先存储于此区域，在上电或其他特定复位发生后将配置信息载入系统内部寄存器中，进而确定关键模块的初始工作状态。这部分配置信息，即为用户配置字。

***注：** “用户配置字”在芯片正常操作时不允许修改，否则芯片会出现不可预知的情况。*

用户配置字详细说明

符号	功能说明
FHS[1:0]	**内部高频振荡器 HIRC 频率选择位：** 00：16MHz； 01：8MHz； 10：2MHz； 11：1MHz； *注 1：本位与 FMOD 位联动，当 FMOD 为“（单时钟）低速工作模式”时，则本控制位无效；* *注 2：本位与 VLVRS 位联动，HIRC 需要匹配对应的 LVR 协同工作。若主频选择 16MHz,LVR 必须选择 3.5/4.1V 档；若选择其他主频，LVR 自由选择。*
FMOD	**芯片运行模式选择位：** 0：（双时钟）运行模式：HIRC（内置 32MHz 高频振荡器）+ LIRC（内置 32KHz 低频振荡器）同时开启，Core 可根据软件的设置工作在高速运行模式或者低速运行模式，CLKS 可调； 1：（单时钟）低速运行模式：仅 LIRC（内置 32KHz 低频振荡器）开启，HIRC（内置高频振荡器）将被常关闭，且 Core 仅可工作在低速运行模式下，CLKS 恒=1。 *注：当本位选择了“（单时钟）低速运行模式”时，Core 无法切换到高速运行模式，且芯片上与 HIRC 时钟源及其相关的功能均不可用。*
LEXTEN	**外部低频晶振（32768Hz）接入使能位：** 0：外部低频晶体谐振器接入 PIN 无效：OSCI/OSCO 口可用于 GPIO 或者其他复用功能，不可用于外接“低频晶体振荡器”； 1：外部低频晶体谐振器接入 PIN 有效：OSCI/OSCO 口仅可用于接入外部的“低频晶体振荡器”，其他功能无效； *注：当本位选择了“外部低频晶体谐振器接入 PIN 有效”时，请务必外接 32768Hz 的晶体振荡器及其振荡电路，否则影响芯片的稳定性。*
WDTM	**WDT 工作模式选择位：** 00：始终关闭 WDT：“看门狗”功能无效； 01：自动关闭 WDT：在停止模式/休眠模式下，自动关闭“看门狗”； 10：始终开启 WDT：在停止模式/休眠模式下，“看门狗”继续工作，故需要手动喂狗。 11：保留，禁止配置 *注：当本位选择了“始终开启‘看门狗’”时，请务必按时手动喂狗，否则将会产生“WDT 复位”。*
RSTEN	**允许外部复位操作位：休眠模式下 LVR 功能的选择位：** 0：外部复位功能无效：对应的 nRST 口可用于 GPIO 或者其他复用功能，该 PIN 不可用于外部复位； 1：外部复位功能有效：nRST 口自动内部带上拉，低电平时复位有效，该 PIN 仅可用于外部复位，其他功能无效。 *注：当本位选择了“外部复位功能无效”时，本芯片将无外部复位功能可用。*
LVRSLP	**休眠模式下 LVR 功能的选择位：** 0：休眠模式下 LVR 无效：LVR 功能将在 Core 进入休眠模式后自动关闭； 1：休眠模式下 LVR 有效：LVR 功能始终开启； *注：“停止模式”下，LVR 不可控，将始终有效。*
VLVRS[2:0]	**LVR 电压等级选择位：** 010：2.1V； 011：2.5V；

符号	功能说明
	100：3.5V； 111：4.1V； 其他：保留，禁止配置 *注：本位与 LVRSLP 位联动，当 LVRSLP 为“停止模式/休眠模式下 LVR 无效”时，则 LVR 功能将在停止模式/休眠模式下，自动关闭。*
SCnP(n=0～3)	**FLASH 分区保护控制位：** 00：n 分区加密块不加密：对应的 n 分区加密块无限制，FLASH 内容可正常读出； 01：n 分区加密块一级加密：对应的 n 分区一级加密限制，FLASH 内容禁止外部调试器/编程器读出，但允许内部 MOVC 指令在“脱机”情况下的读出、“调试”情况下无效； 1x：n 分区加密块二级加密：对应的 n 分区二级加密限制，FLASH 内容禁止外部调试器/编程器读出，同时禁止内部 MOVC 指令任何情况下的读出； *注：该芯片中，每 4K 字节大小设定为一个分区加密块，根据 FLASH ROM 空间地址从低字节到高依次递增排列，即 SC0P 对应 0x0000～0x0FFF 区域，SC1P 对应 0x1000～0x1FFF 区域，SC2P 对应 0x2000～0x2FFF 区域，SC3P 对应 0x3000～0x3FFF 区域。*

5 系统时钟与工作模式

芯片支持 3 种时钟源：

- ✧ 内部高频 RC 时钟 FHIRC 32MHz
- ✧ 内部低频 RC 时钟 FLIRC 32kHz
- ✧ 外部低频晶体时钟 FLOSC 32.768KHz

支持 2 种时钟被选择为系统时钟 FSYS:

- ✧ 高频时钟 FHCLK，由 FHIRC/2 时钟再分频产生
- ✧ 低频时钟 FLCLK，由 FLIRC 时钟产生

用户配置字中 FMOD 位，设置为“（双时钟）运行模式”时，CORE 可根据需要通过软件设置，切换高频时钟（HIRC）或低频时钟(LIRC)作为系统时钟。

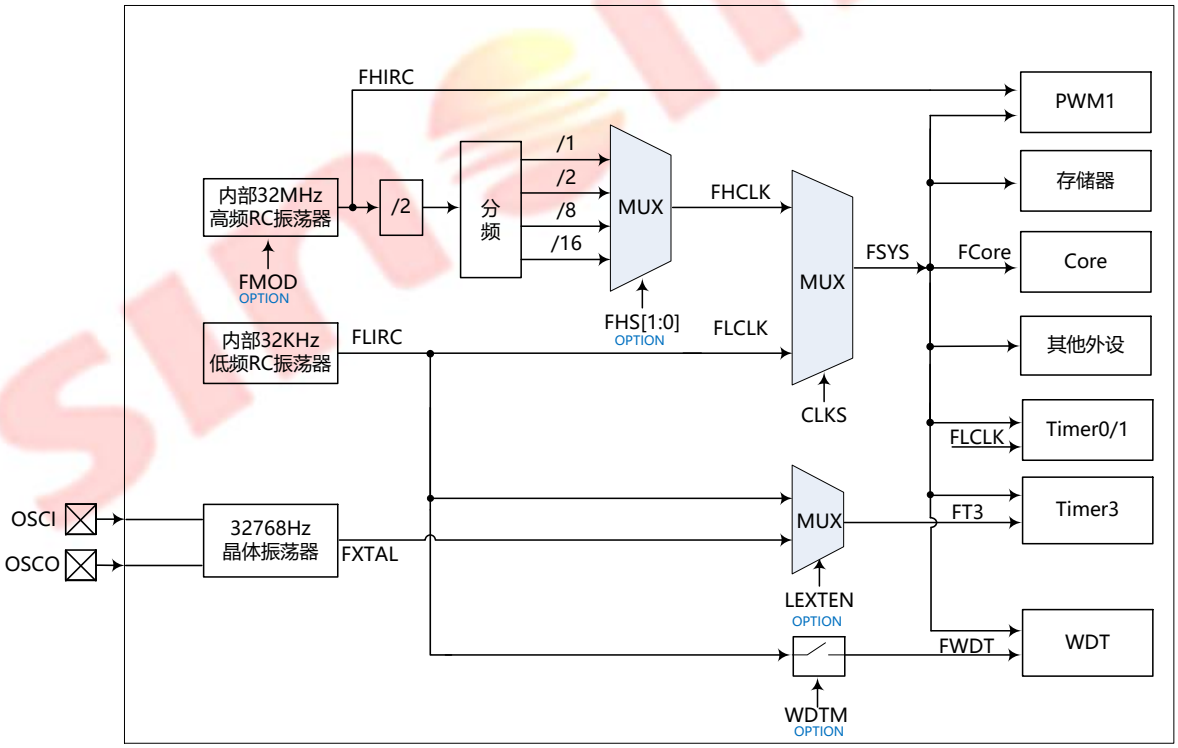
用户配置字中 FMOD 位，设置为“（单时钟）低速运行模式”时，片上高频时钟 HIRC 硬件关闭，内置低频时钟 LIRC 振荡器工作，系统只可工作在低速运行模式，无法工作在高速运行模式。

无论 CORE 工作于 HIRC 或 LIRC 时钟下，CORE 的时钟 F_{CORE} 始终都等于系统时钟 Fsys。

用户配置字中 FHS 位，设置 HIRC/2 时钟再分频，1/2/8/16 四种分频档位可选。

用户配置字中 LEXTEN 位，设置外部低频晶体振荡器接入使能，若使能控制位需外接低频 32768Hz 晶体振荡器，此时芯片内部的某些外设（T3 定时器）可选择时钟源为外部 32768Hz 晶体振荡器。该时钟源不可作为系统时钟 Fsys 使用。

5.1 系统时钟框图



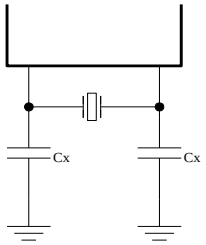
5.2 系统时钟相关寄存器

系统时钟寄存器汇总表

地址(SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)

SFR1 /F9	**PWRCR1**	高级能耗控制寄存器 1	TK_CLKEN	-	CRC_CLKEN	-	-	T3_CLKEN	T2_CLKEN	T01_CLKEN	1111 1111
SFR0 /F8	**PWRCR**	高级能耗控制寄存器	-	-	PWM1_CLKEN	PWM0_CLKEN	SPI_CLKEN	ADC_CLKEN	LCD_CLKEN	OCD_CLKEN	1111 1111
SFR0 /C0	**OSCM**	振荡模式寄存器	XTSPD	-	STBH	STBXT	-	CLKS	LFEN	HFEN	1-00 -x1x
SFR0 /87	**PCON**	电源控制寄存器	UART1EN	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP	0--0 0000
SFR0 /86	**SLPCR**	模式保护控制寄存器	SLPCR7	SLPCR6	SLPCR5	SLPCR4	SLPCR3	SLPCR2	SLPCR1	SLPCR0	0000 0000

5.3 外接晶体振荡器



使用外部晶体作为内部模块的低频时钟源时，连接方式见左图。低频一般是接 32768Hz 晶体，通常 Cx 是必须的。

在实际使用中，用户应使晶体离 XTAL1、XTAL2 引脚的距离尽可能短，这样有助于振荡器的起振和振荡的稳定性。

下表列出典型频率晶振选用电容 Cx 的推荐值和相应最低起振电压参考值。

晶体频率（Hz）	电容 Cx（pF）	最低起振电压（V）
32768	10	2.0

注：因为晶体的品牌很多，电容值仅为推荐值，起振电压仅供参考，具体参数请根据实际使用的晶振性能而定。

5.4 内置 32MHz 高频 RC 振荡器

芯片内置高精度 32MHz RC 振荡器，用户可通过用户配置字配置成 16/8/2/1MHz 使用，该振荡器可用于系统高频时钟。

HIRC 也可以直接提供 PWM1 时钟，用户可配置相关控制位选择 FSYS 或 FHIRC。

内置的高频 RC 振荡器，出厂时即已调校至 32MHz@5V/25℃，用户使用时无需重新调校。

5.5 内置 32KHz 低频 RC 振荡器

芯片内置一个低频 RC 振荡器，该振荡器可用于系统低频时钟。该振荡器频率典型值为 32KHz。

FLIRC 作为 WDT 固定工作时钟。

通过用户配置字 FMOD，可配置 FLIRC 为 timer3 方式 1（时基模式）的低频时钟。若 FMOD 选择时钟包含 FLOSC，则选择 FLOSC 作为 Timer3 时钟源，否则选择 FLIRC 作为时钟源。

5.6 工作模式

CORE 支持高速运行模式、低速运行模式、停止模式、休眠模式共 4 种工作模式。

工作模式	模式描述	功耗
高速运行模式	CORE 时钟运行在高频时钟（HIRC）上，所有外设均可正常工作。	高
低速运行模式	CORE 时钟运行在低频时钟（LIRC）上，大部分外设外设可正常工作。	中
停止模式	CORE 时钟停止（HIRC 时钟源可选关闭），程序计数器（PC）挂起，但所有外设可继续正常工作，用户可随时通过各种外设产生的中断快速唤醒 CORE。	低
休眠模式	CORE 时钟停止（HIRC 时钟源关闭、LIRC 时钟源可选关闭），程序计数器（PC）挂起，仅有限的外设可以继续工作，用户仅可通过这部分外设产生的中断唤醒 CORE。	最低

工作模式与时钟切换

通过用户配置字可配置运行模式为下面列表中 2 种之一，通过软件可配置进入停止模式和休眠模式。

芯片上电或复位默认时钟为 HIRC，执行配置字加载后切换为用户预设时钟。

振荡器	（双时钟）模式 HIRC+LIRC (CLKS=0/1)	（单时钟）低速模式 LIRC (CLKS 恒=1)	停止模式 (当 STOP=1)	休眠模式 (当 SLEEP=1)
内部 32MHz 高频振荡器	工作	停止注 1	HFEN 决定注 2	停止注 1
内部 32KHz 低频振荡器	工作	工作	工作	LFEN 决定注 3

注 1：为了尽量降低功耗，内部高频振荡器（HIRC）在某些场合下必须“停止”工作。（单时钟）低速运行模式在“用户配置字”中的“时钟振荡模式选择位”中选择；当进入“（单时钟）低速运行模式”后，即使用户强行切高速时钟 CLKS=0，HIRC 也不会工作，切换不成功；在进入“休眠模式”前，即使用户之前选择了“双时钟（高频）振荡器模式”，且用户选择了不关闭 HIRC（设置 HFEN=1），一旦进入“休眠模式”后，HIRC 也将自动停止工作。

注 2：当进入“停止模式”后，若使用户之前选择了“（双时钟）运行模式”，这里可以通过保留 LIRC，然后关闭 HIRC（设置 HFEN=0），进一步降低功耗。

注 3：当进入“休眠模式”后，用户可以通过关闭 LIRC（设置 LFEN=0），进一步降低功耗。特别注意，当用户进入休眠模式时关闭 LIRC，T3 不可用来唤醒“休眠模式”下的芯片；T3 需要使用外部 32768 时钟源，才可以唤醒“休眠模式”下的芯片。但 INT0/1 任何情况下，都可以直接用来唤醒“休眠模式”下的芯片，并再次启动 LIRC。只不过这样的唤醒时间会比开启 LIRC 进入“休眠模式”的时间长，因为芯片多了一个 LIRC 起振和稳定的时间。

5.6.1 工作模式设置

振荡模式寄存器（0xC0，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSCM	XTSPD	-	STBH	STBXT	-	CLKS	LFEN	HFEN
R/W	R/W	-	R	R	-	R/W	R/W	R/W
复位值	1	-	0	0	-	X	1	X

BIT[7] **XTSPD** – 外部 32768Hz 振荡器加速模式控制位

0：32768Hz 振荡器常规模式，由软件清 0；

1：32768Hz 振荡器加速模式，由软件或硬件置 1。

注：当使用 32768Hz 振荡器时，此位在系统发生复位后，由硬件自动置 1，以加速 32768Hz 振荡器的起振时间；如有需要，此位也可由软件置 1 以便在模式切换时加速外部振荡器的起振，或软件清 0 以降低功耗。

BIT[5] **STBH** – 高频振荡器稳定标志位

0：高频振荡器停振或未稳定，由硬件清 0，可读但写入无效；

1：高频振荡器已稳定运行，由硬件置 1，可读但写入无效。

BIT[4] **STBXT** – 外部 32768Hz 振荡器稳定标志位

0：低频振荡器停振或未稳定，由硬件清 0，可读但写入无效；

1：低频振荡器已稳定运行，由硬件置 1，可读但写入无效。

注：当 T3 需要使用该振荡器时，必须等待 STBXT 置 1，置 1 说明振荡器已经稳定，方可切换；否则 T3 将出现不可预知的情况。

BIT[2] **CLKS** – CORE 工作时钟选择位（复位初值取决于时钟振荡模式）

0：高频时钟 FHCLK 作为系统时钟 FSYS；

1：低频时钟 FLCLK 作为系统时钟 FSYS。

注 1：当用户设定了“用户配置字”中设为“（单时钟）低速运行模式”方式工作，该位设为 0 无效，系统固定为 1。

注 2：当用户需要切换振荡器时，必须等待 STBH 置 1，置 1 说明振荡器已经稳定，方可切换；否则芯片将出现不

可预知的情况。

BIT[1] **LFEN** – 低频振荡器控制位

0：任意模式下，当芯片进入“休眠模式”后，低频振荡器（LIRC）自动停止；

1：任意模式下，当芯片在进入“休眠模式”后，低频振荡器（LIRC）始终工作。

注 1：这里的“任意模式”指，“（单时钟）低速运行模式”和“（双时钟）运行模式”，用户通过设定“用户配置字”中的 FMOD 位决定。

注 2：若用户配置字中的“WDT 工作模式选择位”处于“始终开启看门狗”状态，LFEN 控制的 LIRC 是不能被直接关闭的，设置 LFEN=0 无效。

注 3：进入停止模式，低频振荡器（LIRC）始终工作。

BIT[0] **HFEN** – 高频振荡器控制位（复位初值取决于时钟振荡模式）

0：（双时钟）运行模式下，在低频运行模式（CLKS=1）下进入“停止模式”，高频振荡器（HIRC）停止；

1：（双时钟）运行模式下，在低频运行模式下（CLKS=1）下进入“停止模式”，高频振荡器（HIRC）工作。

注 1：“（单时钟）低速运行模式’和”（双时钟）运行模式”，用户通过设定“用户配置字”中的 FMOD 位决定。

注 2：当用户设定了“用户配置字”的 FMOD 为“（单时钟）低速运行模式”，当芯片进入“停止模式”，该位的设置无效，即使设为 1，高频振荡器不会工作。

注 3：当用户设定了“用户配置字”的 FMOD 为“（双时钟）运行模式”，且当芯片进入“休眠模式”，该位的设置无效，即使设定为 1，高频振荡器也将自动停止；仅当芯片进入“停止模式”，该位的设置才有效；当 Core 工作于高频运行模式（CLKS=0）下，用户再去设置 HFEN=1，芯片有自我保护机制，设置 HFEN=1 无效。

5.6.2 停止模式

在高速或低速运行模式时，通过先对寄存器 SLPCR 写入“55H”、再将寄存器 PCON 中 STOP 位置“1”，则系统进入停止模式，芯片进行相应处理：

- ✧ CORE 停止工作；
- ✧ 高频振荡器根据 HFEN 位，低频振荡器根据 LFEN 位决定停止或工作；
- ✧ RAM 内容保持不变；
- ✧ 所有的输入输出端口保持不变；
- ✧ 全部外设可以根据用户的设定，停止或保持继续工作；

以下情况可退出停止模式：

- ✧ 发生一个有效的中断请求（可以参考中断章节的内容）；
- ✧ 发生一个有效的复位（可以参考中断章节的内容）；

芯片退出停止模式后，经过振荡等待后，会将 STOP 清 0，以高速/低速运行模式恢复工作。

5.6.3 休眠模式

在高速或低速运行模式时，通过先对寄存器 SLPCR 写入“55H”、再将寄存器 PCON 中 SLEEP 位置“1”，则系统进入休眠模式，芯片进行相应处理：

- ✧ CORE 停止工作；
- ✧ 高频振荡器停止，低频振荡器根据 LFEN 位决定停止或工作；
- ✧ RAM 内容保持不变；
- ✧ 所有的输入输出端口保持不变；
- ✧ 部分外设可以根据用户的设定，停止或保持继续工作；

以下情况可退出休眠模式：

- ✧ 发生外部中断请求（可以参考中断章节的内容）；
- ✧ 发生定时器 T3 中断（可以参考中断章节的内容）；

- ✧ 发生一个有效的复位（可以参考中断章节的内容）；

芯片退出休眠模式后，经过振荡等待后，会将 SLEEP 清 0，以高速/低速运行模式恢复工作。

休眠唤醒时钟预热：

- ✧ 若 HIRC 时钟进休眠模式，HIRC 硬件关闭，唤醒后 6400clk@HIRC16M 预热计时，CPU 才会切换 HIRC 启动；
- ✧ 若 LIRC 时钟进休眠模式，进模式前 LFEN 关闭 LIRC，HIRC 也会硬件关闭，唤醒后 64clk@LIRC 预热计时，需重新使能 HFEN，才能启动 HIRC，预热时间也是 6400clk@HIRC16M；

电源控制寄存器（0x87，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	UART1EN	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP
R/W	R/W	-	-	R/W	R/W	R/W	R/W	R/W
复位值	0	-	-	0	0	0	0	0

BIT[1] **SLEEP** – 休眠模式控制位

0：未进入休眠模式，退出休眠模式后硬件清 0；

1：进入休眠模式（必须紧跟解锁保护后），用户是无法读取到（1）这个值的。

BIT[0] **STOP** – 停止模式控 制位

0：未进入停止模式，退出停止模式后硬件清 0；

1：进入停止模式（必须紧跟解锁保护后），用户是无法读取到（1）这个值的。

注 1：以上两个模式控制位，停止/休眠模式都是 CORE 的节能模式， CORE 同一时刻只会进入其中一种模式，并不会嵌套进入；若这两位同时置 1，CORE 优先进入休眠 SLEEP 模式。

注 2：特别注意，对上述两位（PCON[1:0]）操作时，必须先将寄存器 SLPCR 写“55H”，然后立刻将 STOP/SLEEP 位置 1，才能使系统进入停止/休眠模式，否则下个时钟周期到来后，本次操作失效。

*注 3：**特别注意，强烈建议在进入 STOP/SLEEP 指令后，插入两条 NOP 指令，防止系统误动作。***

模式保护控制寄存器（0x86，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SLPCR	SLPCR7	SLPCR6	SLPCR5	SLPCR4	SLPCR3	SLPCR2	SLPCR1	SLPCR0
W	W	W	W	W	W	W	W	W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **SLPCR[7:0]** – 模式切换保护位

SLPCR[7:0]	模式保护允许位
55H	允许进入停止/休眠模式
其他	禁止进入停止/休眠模式

注：该寄存器为只写寄存器，只有先将寄存器 SLPCR 写“55H”再写 STOP/SLEEP 位置 1 的连续 2 条指令，才能使系统进入停止/休眠模式，否则下个时钟周期到来后，寄存器 SLPCR 和 STOP/SLEEP 位中先前写入的值将被硬件清 0。

CORE 进入 STOP 停止模式，C 语言代码应用举例：

```
#include <intrins.h>
……
sfr SLPCR = 0x86;
sfr PCON = 0x87;

    SLPCR = 0x55;          //解锁保护寄存器
    PCON |= 0x01;          //CORE 进入停止模式
    _nop_();               //NOP 指令，不可少
    _nop_();               //NOP 指令，不可少
```

CORE 进入 SLEEP 休眠模式，C 语言代码应用举例：

```
#include <intrins.h>
……
sfr SLPCR = 0x86;
sfr PCON = 0x87;

    SLPCR = 0x55;          //解锁保护寄存器
    PCON |= 0x02;          //CORE 进入休眠模式
    _nop_();               //NOP 指令，不可少
    _nop_();               //NOP 指令，不可少
```

5.7 高级能耗控制

芯片内置了一个高级的能耗控制寄存器，在低功耗需求下，用户可以通过配置该寄存器，进一步降低芯片本身的功耗。

高级能耗控制寄存器（0xF8，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWRCR	-	-	PWM1_ CLKEN	PWM0_ CLKEN	SPI_ CLKEN	ADC_ CLKEN	-	OCD_ CLKEN
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	-	1	1	1	1	1	1

BIT[5] **PWM1_CLKEN** –PWM1 时钟控制位
0：关闭 PWM1 模块的时钟；
1：打开 PWM1 模块的时钟。

BIT[4] **PWM0_CLKEN** – PWM0 时钟控制位
0：关闭 PWM0 模块的时钟；
1：打开 PWM0 模块的时钟。

BIT[3] **SPI_CLKEN** – SPI 时钟控制位
0：关闭 SPI 模块的时钟；
1：打开 SPI 模块的时钟。

BIT[2] **ADC_CLKEN** – ADC 时钟控制位
0：关闭 ADC 模块的时钟；
1：打开 ADC 模块的时钟。

BIT[0] **OCD_CLKEN** – OCD 时钟控制位

0：关闭 OCD（片上调试）模块的时钟；

1：打开 OCD（片上调试）模块的时钟。

注：该寄存器中的位需要在 OCD 模块不工作的情况下，才能被关闭；否则芯片有保护机制，“关闭”动作会失败。

高级能耗控制寄存器 1（0xF9，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWRCR1	TK_CLKEN	-	CRC_ CLKEN	-	-	T3_CLKEN	T2_CLKEN	T01_ CLKEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7] **TK_CLKEN** –TK 时钟控制位

0：关闭 TK 模块的时钟；

1：打开 TK 模块的时钟。

BIT[6] **保留位**

BIT[5] **CRC_CLKEN** –CRC 时钟控制位

0：关闭 CRC 模块的时钟；

1：打开 CRC 模块的时钟。

BIT[2] **T3_CLKEN** – T3 时钟控制位

0：关闭 T3 模块的时钟；

1：打开 T3 模块的时钟。

BIT[1] **T2_CLKEN** – T2 时钟控制位

0：关闭 T2 模块的时钟；

1：打开 T2 模块的时钟。

BIT[0] **T01_CLKEN** – T01 时钟控制位

0：关闭 T01 模块的时钟；

1：打开 T01 模块的时钟。

注：该寄存器中的位需要在模块不使用的情况下，才能被关闭；否则芯片有保护机制，“关闭”动作会失败。

6 复位

片上有五种复位方式

- ✧ 上电复位 POR；
- ✧ 外部复位；
- ✧ 低电压复位 LVR；
- ✧ 看门狗复位；
- ✧ 软件复位。

任何一种复位发生后，系统将会重新从 0000H 地址处开始执行指令；另外系统还会将大部分的特殊功能寄存器重置为默认复位值。并且可用通过寄存器标志位可以确定复位源。

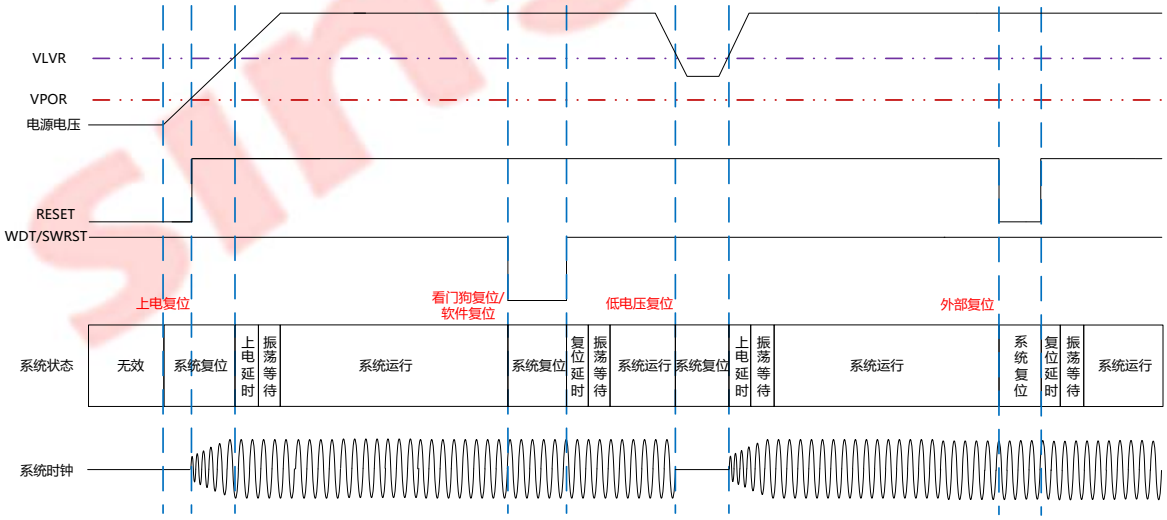
复位寄存器汇总表

地址(SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0/E8	**RSTFLAG**	复位标志寄存器	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST	00000--0
SFR0/CF	**WDTCR**	看门狗定时器控制寄存器	-	-	-	-	-	WDTOT2	WDTOT1	WDTOT0	00000111

6.1 复位条件

CORE 根据复位方式的不同，执行不同的复位动作，根据配置信息完成相应的模块配置后，在外部条件满足时才结束复位，系统重新开始工作。

系统状态与五种复位的关系示意图



6.2 上电复位

在上电过程中，当 VDD 低于参考电压门限值，上电复位将保持芯片为复位模式。这种设计使芯片在 VDD 不满足执行读取存储器时，不访问程序存储器空间。如果从程序存储器读取并执行一个不确定的操作码，可能会使芯片甚至是整个系统进入错误状态。VDD 上升到参考门限电压以上，系统工作，所选的振荡器起振，程序从 0000H 开始执行，同时 PORF 上电标志位将置 1。若发生上电复位，上电标志 PORF 置 1，且上电后内部 RAM 的值为不确定，强烈建议用户初始化 RAM 再使用。

6.3 外部复位

外部复位功能是否开启可以通过芯片配置信息进行设置，选择外部复位功能后复位引脚的内部上拉电阻自动有效。外部复位引脚 RST 是施密特结构的，低电平有效。当外复位引脚为高电平时，系统正常运行；为低电平时，系统产生复位。若发生外部复位，程序从 0000H 开始执行。同时外部复位 RSTF 标志位将置 1。

注：特别注意，当“用户配置字”中使能 P1.7 作为外部复位使用后，但“调试”情况下会自动禁止，仅“脱机”情况下有效。

6.4 低电压复位

芯片的 LVR 电压有 4 级可选：2.1V、2.5V、3.5V、4.1V。LVR 的电压检测电路有一定的回滞特性，通常回滞电压为 0.05V 左右，则当电源电压下降到 LVR 电压时 LVR 复位有效，而电压需要上升到 LVR 电压+0.05V 时 LVR 复位才会解除。若发生低电压复位，程序从 0000H 开始执行。同时低电压 LVRF 标志位将置 1。

LVR 通过芯片的“用户配置字”进行设置，软件无法调整复位电压等级。

6.5 看门狗复位

看门狗复位是一种对程序正常运行的保护机制。正常情况下，用户软件需要按时对 WDT 定时器进行清零操作，保证 WDT 不溢出。若出现异常状况，程序未按时对 WDT 定时器清零，WDT 会溢出从而产生看门狗复位，系统重新初始化，返回受控状态。若发生看门狗复位，程序从 0000H 地址开始执行。同时看门狗复位 WDTRF 标志位将被置 1。

WDT 开启与关闭通过“用户配置字”进行设置，软件无法调整； WDT 只能通过用户配置字的 WDTM 选择“始终开启 WDT”、“停止/休眠模式关闭 WDT”、“始终关闭 WDT”等 3 种工作模式。

(1) 若选择始终开启 WDT 模式，在停止/休眠模式下 WDT 依然运行，WDT 溢出后将复位芯片。

(2) 若选择停止/休眠模式下关闭 WDT 模式，则 WDT 在停止/休眠模式下被硬件自动关闭，并在芯片恢复运行后自动继续。

(3) 若选择始终关闭 WDT 模式，则整个 WDT 模块功能将失效，“用户配置字”中默认为此项。

当 WDT 开启时，默认为最大溢出时间，用户可以通过改变 WDTOT[2:0]的值，来调整的 WDT 溢出时间。因为 WDT 的时钟源来自 LIRC，故一旦在“用户配置字”中选择“始终开启 WDT”后，LIRC 将不能在软件中被关闭，将始终为 WDT 提供时钟源。

WDT 设计为一个向上计数的计数器，计数到设定值以后产生溢出，触发对应的看门狗复位事件。该计数器的值对用户不可见，但用户可以通过读/写一次 WDTCR 寄存器，实现“喂狗操作”，该动作发生的同时，硬件上自动完成一次对 WDT 计数器的重载。

看门狗定时器控制寄存器（0xCF，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTCR	-	-	-	-	-	WDTOT2	WDTOT1	WDTOT0
R/W	-	-	-	-	-	R/W	R/W	R/W
复位值	-	-	-	-	-	1	1	1

BIT[2:0] **WDTOT[2:0]** – WDT 溢出时间控制位

WDT[2:0]	WDT 溢出时间@5V 注 1
000	8ms
001	16ms
010	32ms
011	64ms
100	128ms
101	256ms
110	512ms
111	1024ms（复位值）

注 1:上表中的“WDT 溢出时间”是一个理论的溢出时间,实际溢出时间和时钟源 LIRC 有关,请以 LIRC 规格为准;一般情况下，随电压的降低下降 WDT 溢出的时间延长。

注 2：寄存器的配置（用户软件上）仅可控制看门狗的溢出时间；但是看门狗定时器开启与否，必须在“用户配置字”中设定。

注 3：在 WDT 的溢出时间之内，读/写一次 WDTCR 寄存器，即完成一次“喂狗操作“；此时硬件会自动重载一次 WDT 计数器的值，并继续开始向下递减计数。

喂狗操作（清 WDT），C 语言代码应用举例：

```
sfr WDTCR = 0xCF;

    WDTCR = 0x03;          //WDT 复位之前，喂狗操作（清 WDT），再从 64ms 开始重新倒数计时
```

6.6 软件复位

软件复位是一种对可控的程序自发的复位行为,有点类似于外部复位。正常情况下,用户软件需要连续两次对 SWRST 位写两次，启动软件复位。若发生软件复位，程序从 0000H 开始执行。同时软件复位 SWRF 标志位将置 1。

复位标志寄存器（0xE8，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RSTFLAG	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST
R/W	R/W	R/W	R/W	R/W	R/W	-	-	R/W
复位值	0	0	0	0	0	-	-	0

BIT[0] **SWRST** – 软件复位

0：无软件复位要求；

1：启动软件复位（连续写两遍有效），系统软件复位后硬件清 0，并将 SWRF 标志位置 1。

注 1：只有对该位连续写 2 条相同的指令，才能使系统进行软件复位，否则下个时钟周期到来后，寄存器中先前写入的值将被硬件自动清掉，保持为 0，这是一种简单的保护机制。

注 2：写入 1 成功后，芯片直接“软件复位”，用户查询不到该值为 1 的状态。

软件复位，C 语言代码应用举例：

```
sfr RSTFLAG = 0xE8;

    RSTFLAG |= 0x01;
    RSTFLAG |= 0x01;          //软件复位生效，以下代码将不会执行
```

6.7 复位标志寄存器

系统在热复位的情况下，用户可以通过查看复位标志寄存器，确认系统复位的原因。

复位标志寄存器（0xE8，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RSTFLAG	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST
R/W	R/W	R/W	R/W	R/W	R/W	-	-	R/W
复位值	0	0	0	0	0	-	-	0

BIT[7] **PORF** – 上电复位标志位

0：未发生上电复位；

1：曾经发生过上电复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[6] **LVRF** – 低电压复位标志位

0：未发生低电压复位；

1：曾经发生过低电压复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[5] **RSTF** – 外部复位标志位

0：未发生外部复位；

1：曾经发生过外部复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[4] **WDTRF** – 看门狗复位标志位

0：没有发生过看门狗定时器溢出复位；

1：曾经发生过看门狗定时器溢出复位，硬件置 1 但不会清 0，需要用户手动清。

BIT[3] **SWRF** – 软件复位标志位

0：没有发生过软件复位；

1：曾经发生过软件复位，硬件置 1 但不会清 0，需要用户手动清。

注 1：以上 5 个复位标志在置位（=1）后，需要手动清除，否则硬件不会自动清零（=0）；建议用户在每次使用这些标志位之前，将以上 5 个“复位标志位“全部清零。

注 2：若不清零，不同复位原因标志位之间可以累积记录；但对于同一复位原因标志位，前后复位之间，对已经置位（=1）的标志位将无影响，所以芯片可能遗漏因“重复”复位而产生的标志位。

6.8 各种复位与低功耗模式的关系

复位与停止模式和休眠模式的关系

复位源	停止模式中（STOP 位置 1）复位有效	休眠模式中（SLEEP 位置 1）复位有效
POR 复位	Yes	Yes
外部复位	Yes	Yes
低电压复位	Yes	Yes 注 2
看门狗复位	Yes	Yes 注 2
软件复位	No	No

注 1：通过上表中的部分复位操作，可以实现芯片从停止模式和休眠模式以复位方式退出，程序将从 0x0000 地址开始执行，与中断请求退出不同，因此可能造成实际运行结果与用户的预期不一致。

注 2：通过在“用户配置字”中，关闭休眠模式下的 LVR 功能，则不会产生 LVR 复位；始终关闭 WDT 功能，则不会产生 WDT 复位。

7 I/O端口

7.1 端口特性

芯片共有 3 组 8 位端口的 P0、P1 和 P2，以及一组 2 位端口的 P3，总共 4 组 IO 端口。

其中 P0.2~P0.7 和 P1 用作输出端口时，支持较大的驱动电流能力；4 档可选，通过寄存器控制。

每个 IO 都有 3 种输入/输出模式可选：推挽输出模式、输入模式和高阻态模式，且在“输入模式”中，可根据需要控制片内上拉电阻是否有效。

推挽输出模式：此模式下能输出很强的驱动能力。若用户需要更强的驱动能力，请使用大驱动口的推挽输出模式。

高阻输入模式：此模式下可实现输入，而根据用户的配置实现不带上拉电阻、带上拉电阻 2 种输入方式。

高阻态模式：此模式下既不输入也不输出，既不是高电平也不是低电平，“高阻态模式”可以用来隔绝芯片和外围电路之间的相互影响，近似于断开。

7.2 端口相关寄存器

端口寄存器汇总表

地址(SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0/C3	**P0PU**	P0 端口上拉电阻控制寄存器	P07PU	P06PU	P05PU	P04PU	P03PU	P02PU	P01PU	P00PU	0000 0000
SFR0/C2	**P0MOD1**	P0 端口模式寄存器 1	P07M1	P07M0	P06M1	P06M0	P05M1	P05M0	P04M1	P04M0	1010 1010
SFR0/C1	**P0MOD0**	P0 端口模式寄存器 0	P03M1	P03M0	P02M1	P02M0	P01M1	P01M0	P00M1	P00M0	1010 1010
SFR0/B3	**P3PU**	P3 端口上拉电阻控制寄存器	-	-	-	-	-	-	P31PU	P30PU	------00
SFR0/B1	**P3MOD0**	P3 端口模式寄存器 0	-	-	-	-	P31M1	P31M0	P30M1	P30M0	----1010
SFR0/B0	**P3**	P3 端口数据寄存器	-	-	-	-	-	-	P31D	P30D	------xx
SFR0/A3	**P2PU**	P2 端口上拉电阻控制寄存器	P27PU	P26PU	P25PU	P24PU	P23PU	P22PU	P21PU	P20PU	0000 0000
SFR0/A2	**P2MOD1**	P2 端口模式寄存器 1	P27M1	P27M0	P26M1	P26M0	P25M1	P25M0	P24M1	P24M0	1010 1010
SFR0/A1	**P2MOD0**	P2 端口模式寄存器 0	P23M1	P23M0	P22M1	P22M0	P21M1	P21M0	P20M1	P20M0	1010 1010
SFR0/A0	**P2**	P2 端口数据寄存器	P27D	P26D	P25D	P24D	P23D	P22D	P21D	P20D	xxxxx xxx
SFR0/9F	**PXODM**	端口输出驱动控制寄存器	-	-	-	-	P1ODM1	P1ODM0	P0ODM1	P0ODM0	----0000
SFR0/93	**P1PU**	P1 端口上拉电阻控制寄存器	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU	0000 0000
SFR0/92	**P1MOD1**	P1 端口模式寄存器 1	P17M1	P17M0	P16M1	P16M0	P15M1	P15M0	P14M1	P14M0	1010 1010
SFR0/91	**P1MOD0**	P1 端口模式寄存器 0	P13M1	P13M0	P12M1	P12M0	P11M1	P11M0	P10M1	P10M0	1010 1010
SFR0/90	**P1**	P1 端口数据寄存器	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D	xxxxx xxx

SFR0 /80	**P0**	P0 端口数据寄存器	P07D	P06D	P05D	P04D	P03D	P02D	P01D	P00D	xxxx xxxx

7.3 端口模式控制

P0 端口模式寄存器 0（0xC1，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0MOD0	P03M1	P03M0	P02M1	P02M0	P01M1	P01M0	P00M1	P00M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

P0 端口模式寄存器 1（0xC2，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0MOD1	P07M1	P07M0	P06M1	P06M0	P05M1	P05M0	P04M1	P04M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

BIT[7:0] **P0nM[1:0] (n = 1 - 0)** – P0 端口模式选择位，初始状态为高阻输入模式

P1 端口模式寄存器 0（0x91，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1MOD0	P13M1	P13M0	P12M1	P12M0	P11M1	P11M0	P10M1	P10M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

P1 端口模式寄存器 1（0x92，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1MOD1	P17M1	P17M0	P16M1	P16M0	P15M1	P15M0	P14M1	P14M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

BIT[7:0] **P1nM[1:0] (n = 7 - 0)** – P1 端口模式选择位，初始状态为高阻输入模式

P2 端口模式寄存器 0（0xA1，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2MOD0	P23M1	P23M0	P22M1	P22M0	P21M1	P21M0	P20M1	P20M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

P2 端口模式寄存器 1（0xA2，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2MOD1	P27M1	P27M0	P26M1	P26M0	P25M1	P25M0	P24M1	P24M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	0	1	0	1	0	1	0

BIT[7:0] **P2nM[1:0] (n = 7 - 0)** – P2 端口模式选择位，初始状态为高阻输入模式

P3 端口模式寄存器 0（0xB1，SFR0）

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

P3MOD0	-	-	-	-	P31M1	P31M0	P30M1	P30M0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
复位值	-	-	-	-	1	0	1	0

BIT[3:0] **P3nM[1:0] (n = 1 - 0)** – P3 端口模式选择位，初始状态为高阻态模式

注：因引脚数量关系，P3 口只有一个模式寄存器。

端口输入/输出模式对应表

PxnM1 (x = 0-3; n = 7-0)	PxnM0 (x = 0-3; n = 7-0)	IO 工作模式
0	0	输入模式
0	1	推挽输出模式
1	0	高阻态模式（复位值）
1	1	高阻态模式

7.4 端口数据控制

P0 端口数据寄存器（0x80，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0	P07D	P06D	P05D	P04D	P03D	P02D	P01D	P00D
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] **P0nD (n = 1 - 0)** – P0 端口数据位

P1 端口数据寄存器（0x90，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] **P1nD (n = 7 - 0)** – P1 端口数据位

P2 端口数据寄存器（0xA0，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2	P27D	P26D	P25D	P24D	P23D	P22D	P21D	P20D
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] **P2nD (n = 7 - 0)** – P2 端口数据位

P3 端口数据寄存器（0xB0，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P3	-	-	-	-	-	-	P31D	P30D
R/W	-	-	-	-	-	-	R/W	R/W
复位值	-	-	-	-	-	-	X	X

BIT[1:0] **P3nD (n = 1 - 0)** – P3 端口数据位

端口输出电平对应表

PxnD[1:0] (x = 0 - 3; n = 7 - 0)	IO 电平

0	低电平
1	高电平
X	不确定（复位值）

注：P0、P1、P2 口的数据值在复位之后为（X）不确定状态，建议用户使用之前先写入一个确定的初始电平值。

7.5 输入上拉电阻控制

P0、P1、P2、P3 端口的每位都内置有独立的上拉电阻输入控制。仅当端口处于输入状态下，可单独使能或禁止上拉电阻输入功能。

P0 端口上拉电阻控制寄存器（0xC3，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0PU	P07PU	P06PU	P05PU	P04PU	P03PU	P02PU	P01PU	P00PU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[1:0] **P0nPU（n = 7 - 0）** – P0 端口上拉电阻选择位

P1 端口上拉电阻控制寄存器（0x93，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1PU	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **P1nPU（n = 7 - 0）** – P1 端口上拉电阻选择位

P2 端口上拉电阻控制寄存器（0xA3，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2PU	P27PU	P26PU	P25PU	P24PU	P23PU	P22PU	P21PU	P20PU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **P2nPU（n = 7 - 0）** – P2 端口上拉电阻选择位

P3端口上拉电阻控制寄存器（0xB3，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P3PU	-	-	-	-	-	-	P31PU	P30PU
R/W	-	-	-	-	-	-	R/W	R/W
复位值	-	-	-	-	-	-	0	0

BIT[1:0] **P3nPU0（n = 1 - 0）** – P3 端口上拉电阻选择位 0

端口上拉电阻选择对应表

PxnPU（x = 0-3；n = 7-0）	**上拉电阻选择**
0	无上拉电阻控制（复位值）
1	输入带内部上拉

7.6 大电流方式控制

端口输出驱动控制寄存器（0x9F，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PXODM	-	-	-	PXODM0	P1ODM1	P1ODM0	P0ODM1	P0ODM0
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
复位值	-	-	-	0	0	0	0	0

BIT[4] **PXODM0** – P3&P2&P00&P01 端口输出驱动控制选择

PnODM0（n = 3 - 2）	输出模式下，GPIO 的驱动电流选择
0	输出电流大小参考 Ioh01 /Iol01 的值（复位值）
1	输出电流大小参考 Ioh02/Iol02 的值

BIT[3:2] **P1ODM[1:0]** – P1 端口输出驱动控制选择

BIT[1:0] **P0ODM[1:0]** – P02~P07 端口输出驱动控制选择

PnODM[1:0]（n = 1 - 0）	输出模式下，GPIO 的驱动电流选择
00	输出电流大小参考 Ioh11/Iol11 的值（复位值）
01	输出电流大小参考 Ioh12/Iol12 的值
10	输出电流大小参考 Ioh13/Iol13 的值
11	输出电流大小参考 Ioh14/Iol14 的值

注：本芯片同一时刻 P0 口、P1 口、P2 口、P3 口只能为一种输出驱动电流方式，且一组端口上的一个 PIN 不能单独配置。

7.7 端口复用设置

P0、P1、P2、P3 的部分端口可以复用为其他功能模块的输入/输出端口，芯片默认的优先级按引脚排列中所示外部最高内部最低的原则，用户可根据需要通过寄存器配置成需要功能。

具体内容，请参考相关章节的内容。

8 定时器/计数器

8.1 定时器 T0/T1

芯片包含 2 个兼容 8051 标准的 16 位定时器 T0 和 T1，定时器的两个 8 位数据寄存器（THx\TLx，x=0,1）可作为一个 16 位寄存器来访问。

定时器具有 4 种工作方式，通过定时器模式寄存器的方式选择位（TxM[1:0]）来选择定时器 Tx 的工作方式。

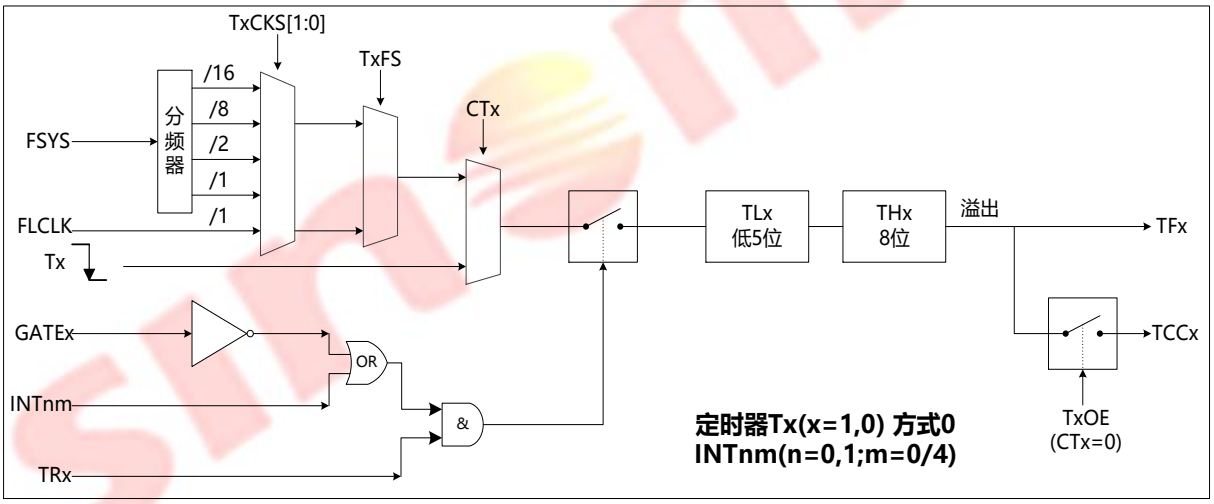
8.1.1 方式 0：13 位计数器/定时器

在方式 0 中，定时器 Tx（x=0,1）为 13 位计数器/定时器。递增计数寄存器 THx[7:0]存放 13 位计数器/定时器的高 8 位，TLx[4:0]存放低 5 位，而 TLx[7:5]的值不确定，在读取时应被忽略。当 13 位计数寄存器溢出时，溢出标志 TFx 将被置 1，如果 Tx 中断被允许，将会产生定时器 Tx 中断。

CTx 位选择计数器/定时器 Tx 的时钟源。如果 CTx = 1，Tx 用作计数器，Tx 引脚将被设置成输入端口（需其他复用功能未复用该端口），Tx 的下降沿将使 Tx 计数寄存器加 1；如果 CTx = 0，Tx 用作定时器，可选择低频时钟 FLCLK 或系统时钟 FSYS（或其分频）作为 Tx 的时钟源。

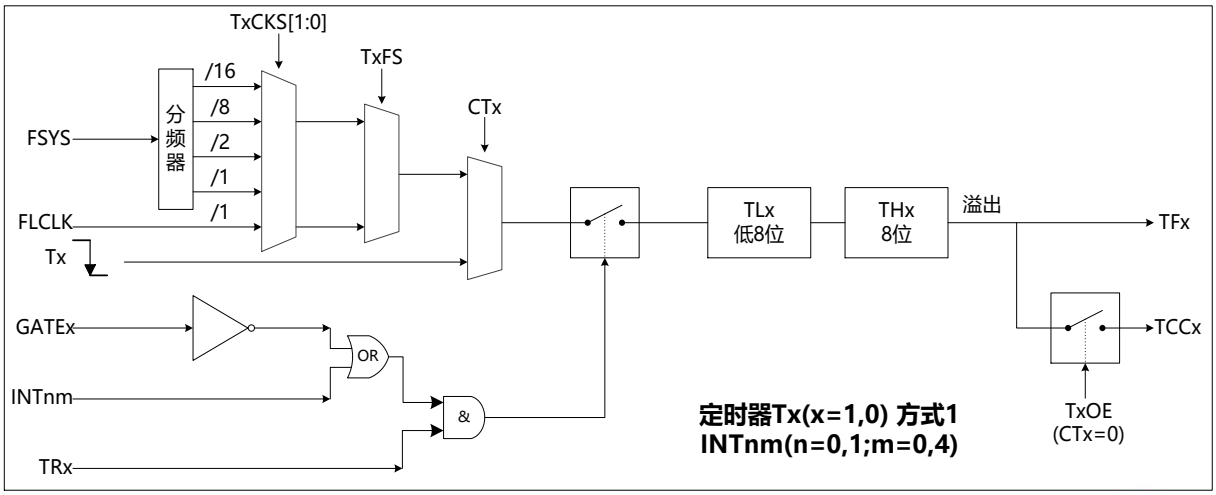
当 GATEx = 0，或者 GATEx = 1 且 INTnm 有效时，TRx 置 1 将启动定时器。GATEx 置 1，允许定时器 Tx 由外部输入信号 INTnm 控制，用于测量 INTnm 引脚上输入高电平的脉冲宽度。TRx 位置 1 不强行复位定时器，即 TRx 置 1 时计数寄存器将从上次 TRx 清 0 时的值恢复计数。所以在启动定时器之前应设定计数寄存器的复位值。

当 Tx 用作定时器时，可设置寄存器 TCON1 中的 TxOE 位使 Tx 计数值溢出时可以对 Tx 引脚电平自动翻转。仅当 TxOE 被置 1，Tx 引脚自动设置为输出有效（需其他复用功能未复用该端口）。



8.1.2 方式 1：16 位计数器/定时器

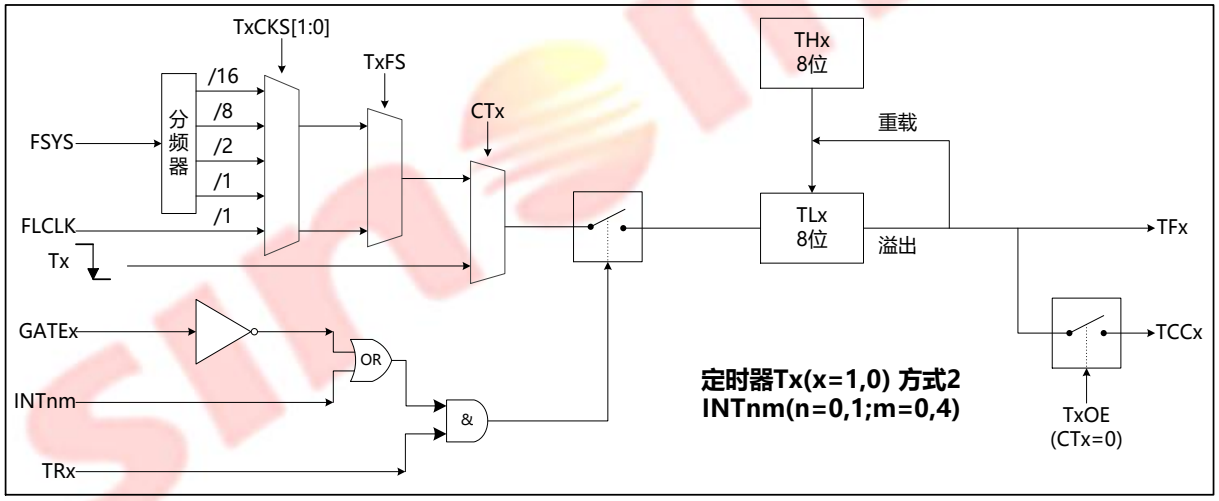
除了使用 16 位计数寄存器 THx/TLx 外，方式 1 的运行与方式 0 相同，打开和设置计数器/定时器也与方式 0 相同。



8.1.3 方式 2：8 位自动重载计数器/定时器

方式 2 中，定时器 Tx 用作 8 位自动重载计数器/定时器。TLx 存放计数值，THx 存放重载值。当递增计数器 TLx 溢出至 0x00 时，溢出标志 TFx 将被置 1，寄存器 THx 的值重新载入计数器 TLx 中，而 THx 的重载值不变。如果 Tx 中断被允许，将会产生定时器 Tx 中断。在此方式下，首次打开定时器开始计数之前，应将 TLx 初始化为重载值，确保首次溢出时间与后续一致。

仅 T1 可以用作 UART0 的波特率发生器，T0 不可作为波特率发生器。除自动重载功能以外，方式 2 中的计数器/定时器的使能和设置与方式 1 和 0 基本相同。



在 UART 方式 1 和方式 3 中波特率可由该方式提供，波特率的计算公式如下：

$$\textbf{T1产生的波特率} = \frac{1}{16} \times \frac{Ft1}{[256-(TH1)]}$$

其中 Ft1 为 T1 实际的工作频率，TH1 为实际设定值。

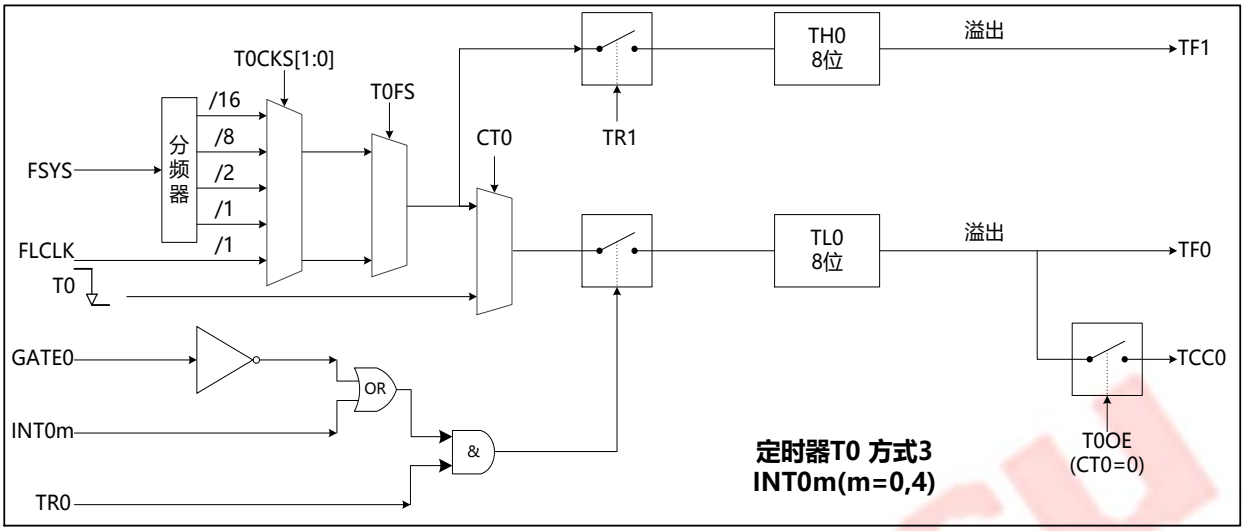
注：当用定时器 T1 作为波特率发生器时，T1 的 TH1 值不可设置为 0xFF，并应该同时关闭相应 T1 的中断使能位。

8.1.4 方式 3：两个 8 位计数器/定时器（只限于定时器 T0）

在方式 3 中，定时器 T0 用作两个独立的 8 位计数器/定时器，分别由 TL0 和 TH0 控制。TL0 使用定时器 T0 的控制（TCON 中）和状态（TMOD 中）位：TR0、CT0、GATE0 和 TF0。TL0 可采用振荡器时钟信号用作定时器，也可采用外部输入信号用作计数器。

TH0 只能用作定时器功能，时钟源来自振荡器时钟信号。TH0 借用定时器 T1 的控制位 TR1 控制使能，溢出时借用定时器 T1 溢出标志 TF1 置 1，产生 T1 中断。

当 T0 工作在方式 3 时，定时器 T1 可以工作在方式 0、1 或 2，但是无法将 TF1 标志置 1，无法产生中断。TH1 和 TL1 只能用作定时器功能，时钟源来自振荡时钟信号，GATE1 无效。因为 TR1 被定时器 0 占用，所以定时器 T1 改由方式设置来控制使能/关闭。设为方式 0/1/2 时 T1 使能，设为方式 3 时 T1 被关闭。



8.2 T0/T1 相关寄存器

T0/T1 寄存器汇总表

地址(SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0 /8E	**TCON1**	定时器控制寄存器 1	T1FS	T1OE	T1CKS 1	T1CKS 0	T0FS	T0OE	T0CKS 1	T0CKS 0	0000 0000
SFR0 /8D	**TH1**	定时器计数高位寄存器 1	TH17	TH16	TH15	TH14	TH13	TH12	TH11	TH10	0000 0000
SFR0 /8C	**TH0**	定时器计数高位寄存器 0	TH07	TH06	TH05	TH04	TH03	TH02	TH01	TH00	0000 0000
SFR0 /8B	**TL1**	定时器计数低位寄存器 1	TL17	TL16	TL15	TL14	TL13	TL12	TL11	TL10	0000 0000
SFR0 /8A	**TL0**	定时器计数低位寄存器 0	TL07	TL06	TL05	TL04	TL03	TL02	TL01	TL00	0000 0000
SFR0 /89	**TMOD**	定时器模式寄存器	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0	0000 0000
SFR0 /88	**TCON**	定时器控制寄存器	TF1	TR1	TF0	TR0	-	-	-	ADCIF	0000 ---1

定时器模式寄存器（0x89，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMOD	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7、3]　**GATEx** – 定时器/计数器 Tx 门控制位（x=1,0）

0：TRx 置 1 时，定时器 Tx 立刻开始计数；

1：TRx 置 1 且外部 INTxn（x=1,0 与 Tx 保持一致；n=4）信号同时有效（具体可参考中断章节内容），

定时器 Tx 才开始计数。

注：当 GATEx=1 时，T0/T1 只能通过 INT04 和 INT14 测量外部信号的脉宽，且仅测量高电平的脉宽。

BIT[6、2] **CTx** – Tx 的计数器/定时器方式选择位（x=1,0）

0：定时器方式，振荡器时钟信号触发，即内部有一个 clock，Tx 的计数器递增一次；

1：计数器方式，外部输入下降沿触发，即 Tx 端口上一个下降沿的有效信号，Tx 的计数器递增一次，此时 Tx 口作为 Tx 定时器/计数器的专用输入口，且不能做其它复用功能。

BIT[5:4] **T1M[1:0]** – 定时器/计数器 T1 工作方式选择位

T1M[1:0]	定时器 T1 工作方式
00	方式 0
01	方式 1
10	方式 2
11	禁止（T1 将停止工作）

注：T1 没有工作方式 3，用户强制使用 T1 的方式 3，T1 将停止工作。

BIT[1:0] **T0M[1:0]** – 定时器/计数器 T0 工作方式选择位

T0M[1:0]	定时器 T0 工作方式
00	方式 0
01	方式 1
10	方式 2
11	方式 3

定时器控制寄存器（0x88，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON	TF1	TR1	TF0	TR0	-	-	-	ADCIF
R/W	R/W	R/W	R/W	R/W	-	-	-	R/W
复位值	0	0	0	0	-	-	-	1

BIT[7、5] **TFx** – 定时器/计数器 Tx 溢出标志位（x=1,0）

0：定时器/计数器 Tx 无溢出，当中断响应后硬件清 0，可由软件清 0；

1：定时器/计数器 Tx 溢出时，由硬件置 1。

BIT[6、4] **TRx** – 定时器/计数器 Tx 使能位（x=1,0）

0：定时器/计数器 Tx 中止递增，THx 和 TLx 寄存器的数值将停在“中止”时候的数值；

1：定时器/计数器 Tx 开始递增，或者（THx 和 TLx 寄存器不重新赋值）继续递增。

定时器控制寄存器 1（0x8E，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON1	T1FS	T1OE	T1CKS1	T1CKS0	T0FS	T0OE	T0CKS1	T0CKS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7、3] **TxFS** – 定时器/计数器方式时，Tx 时钟源选择位（x=1,0）

0：Tx 时钟源为系统时钟 Fsys；

1：Tx 时钟源为低频时钟 F_{LCLK}；

注：若系统时钟选择 FLCLK，则 Tx（x=1,0）时钟源必须选择 Fsys（即 TxFS = 0）,否则 Tx 工作异常；若系统时钟选择 FHCLK，则 Tx（x=1,0）时钟源可以任意选择。

BIT[6、2] **TxOE** – 定时器/计数器 Tx 输出使能位（x=1,0）（CTx=0 时有效）

0：禁止 Tx 溢出时翻转 Tx 端口电平，端口用于通用 IO 或其他复用模块；

1：使能 Tx 端口复用于定时器 Tx，Tx 溢出时翻转端口电平，并且第一次反转输出的是高电平（逻辑 1），

第二次反转输出的是低电平（逻辑 0），之后依次交替反转。

BIT[5:4,1:0] **TxCKS[1:0]** – 定时器/计数器 Tx（x=1,0）时钟分频选择位

TxCKS[1:0]	Tx 频率（TxFS=0）	Tx 频率（TxFS=1）
00	F_{SYS}	F_{LCLK}
01	$F_{SYS}/2$	禁止（T0/T1 将停止工作）
10	$F_{SYS}/8$	
11	$F_{SYS}/16$	

注：Tx 的工作频率 F_{Tx}，与用户选择的实际频率和分频比有关；在低频（TxFS=1）的情况下，用户仅有一挡（TxCKS[1:0]=00）可选择，若强行选择了“禁止”状态，否则 Tx 将出现不可预知的情况。

定时器计数高位寄存器 0（0x8C，SFR0）和定时器计数高位寄存器 1（0x8D，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
THx	THx7	THx6	THx5	THx4	THx3	THx2	THx1	THx0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **THx[7:0]** – 定时器/计数器 Tx 高 8 位计数器（x=1,0）

定时器计数低位寄存器 0（0x8A，SFR0）和定时器计数低位寄存器 1（0x8B，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TLx	TLx7	TLx6	TLx5	TLx4	TLx3	TLx2	TLx1	TLx0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **TLx[7:0]** – 定时器/计数器 Tx 低 8 位计数器（x=1,0）

注 1：无论是读或者写 TLx/THx 的值，必须先停止（TRx=0），否则 Tx 将产生不可预知的情况。

注 2：写操作时：必须先写 TLx，再写 THx；否则 TLx 的值不能立即反映到芯片上；读操作时：必须先读 TLx，再读 THx，否则将读取到错误的值。

8.3 定时器 T2

芯片包含 1 个 16 位增强型定时器 T2，兼容 52 系列中标准型 T2，但功能更强。T2 是一个向上递增型定时器/计数器。T2 具有 4 种工作在模式：自动重载模式、捕获模式、波特率发生器模式、比较模式。

8.3.1 方式 0：16 位重载方式定时器/计数器方式

T2 的这种工作方式和 T0/T1 的工作方式 2 类似，只不过 T0/T1 是 8 位的，而 T2 则是 16 位的。

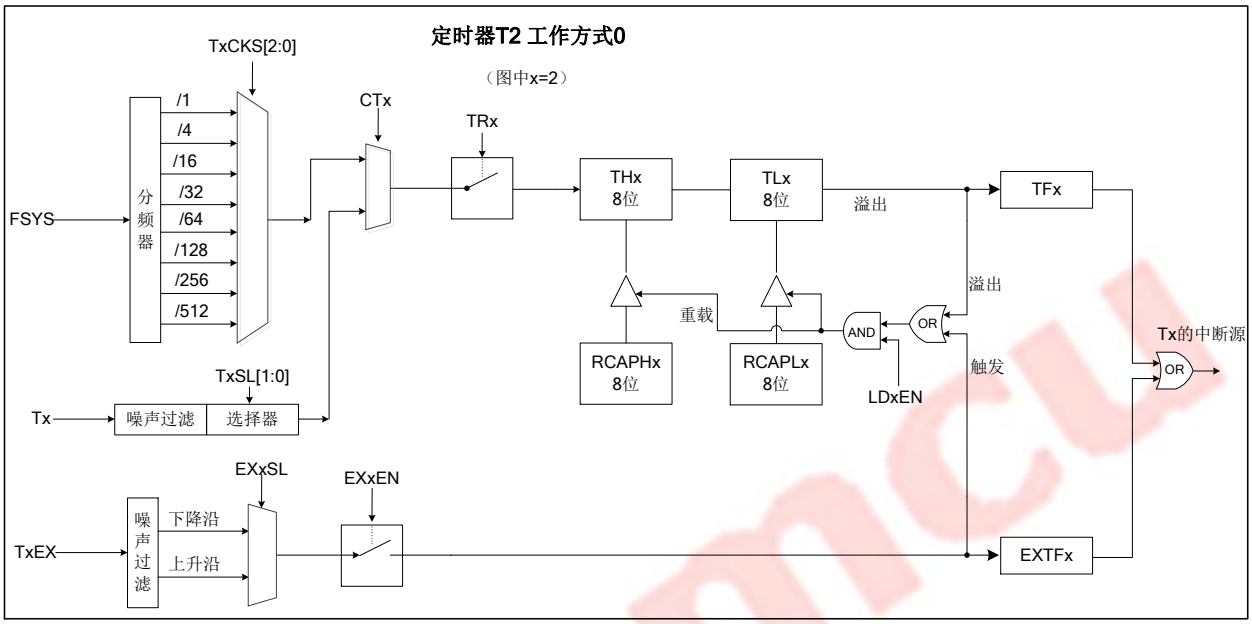
T2 的自动重载模式中，时钟源由 CT2 位决定是对 F_{SYS} 计数，还是对 T2 端口输入信号计数。RCAPH2 和 RCAPL2 保存着重复装载的数值。一旦产生重载触发事件，硬件会自动把 RCAPH2 和 RCAPL2 中的值分别装入 TH2 和 TL2 中。重载触发事件指，T2 定时器溢出（TF2=1）或者是外部捕获信号有效（T2EX 端口出现有效信号，EXTF2=1），两者都可产生 T2 中断，并重新装载 TH2/TL2，使其开始新一轮的计数。

标准情况：使能 T2，将 T2 设置为方式 0，设置时钟频率、定时器模式（CT2=0）等等参数。启动 T2（TR2=1），打开重载功能 LD2EN=1，且禁止外部捕获信号 T2EX 输入（EX2EN=0），该情况不产生外部触发事件（T2EX 端口无有效信号，始终是 EXTF2=0），仅通过 T2 定时器产生溢出（TF2=1）产生触发事件的情况。向上递增计数中，TH2/TL2（从 RCAPH2/RCAPL2 装载）的值，开始递增计数直到 0x0 产生溢出，最终产生 TF2 标志位置位（TF2=1）和 T2 中断。

特殊情况一：用户觉得内部时钟源不好，可以改变时钟源。当 CT2=1 时，时钟源切换到外部，通过 T2 口送入。用户还可通过 T2SL 位，决定有效信号的方式，当 T2SL=01B 时上升沿有效，当 T2SL=10B 时下降有效，当 T2SL=11B 时双沿有效。

特殊情况二：不重载或者 T2EX 加入触发重载控制。当 LD2EN 位=0 关闭重载功能，则 T2 仅就做一次性定时任务，T2 的溢出和外部 T2EX 的有效信号都不再重载 TH2 和 TL2。若用户需要再次开启 T2 的方式 0，则先让 TR2=0 停止 T2，

对 RCAPH2/RCAPL2 重新赋值后（对 TH2 和 TL2 赋值无效），让 TR2=1 再次启动 T2，T2 方可正常工作。当 T2EX=1，T2EX 端口的信号可以参与触发重载，外部触发信号与 T2 溢出的信号同一优先级。若 T2EX 有效信号和 T2 溢出信号同时来到，两者都会产生中断，且 2 个标志位的产生，就意味着会有 2 次中断，用户需要根据自己的应用编写不同的应用程序，以满足自己的实际需要。EX2SL 为决定有效信号的类型，0 为下降沿有效，1 为上升沿有效。当通过外部触发重载了 TH2/TL2 并产生中断，其中断标志位是 EXTF2，而不是 TF2，用户可以通过查询 EXTF2 标志位/TF2 标志位，具体识别本次 T2 中断事件的类型。



8.3.2 方式 1：16 位输入捕获方式

输入捕获功能，主要用于测量脉冲宽度或周期。在 T2 的捕获方式中，时钟源由 CT2 位决定是对 F_{SYS} 或其分频计数，还是对 T2 端口输入信号计数。T2 的 T2EX 端口的输入由 EX2EN 位控制，当在该方式下打开了 T2EX 端口后，芯片始终通过监视并捕获 T2EX 端口上的有效信号。RCAPH2 和 RCAPL2 保存最近一次捕获的结果。一旦产生了输入捕获事件，硬件自动把 TH2 和 TL2 中的当前值分别送到 RCAPH2 和 RCAPL2 中保存。输入捕获事件指，T2EX 上产生一个有效的输入捕获信号有效，并被芯片识别（T2 硬件对 EXTF2 标志位置 1）的过程。注意若 T2 的参数设置不当，导致捕获事件发生前的时间间隔超过最大计数，同样会造成 T2 定时器的溢出（TF2 标志位置 1）引起 T2 中断，但用户可以通过标志位的不同进行区分。另外，当 RCAPH2 和 RCAPL2 中保存了 TH2 和 TL2 的溢出值（0x0），也是一种异常情况，用户应该做特别处理。

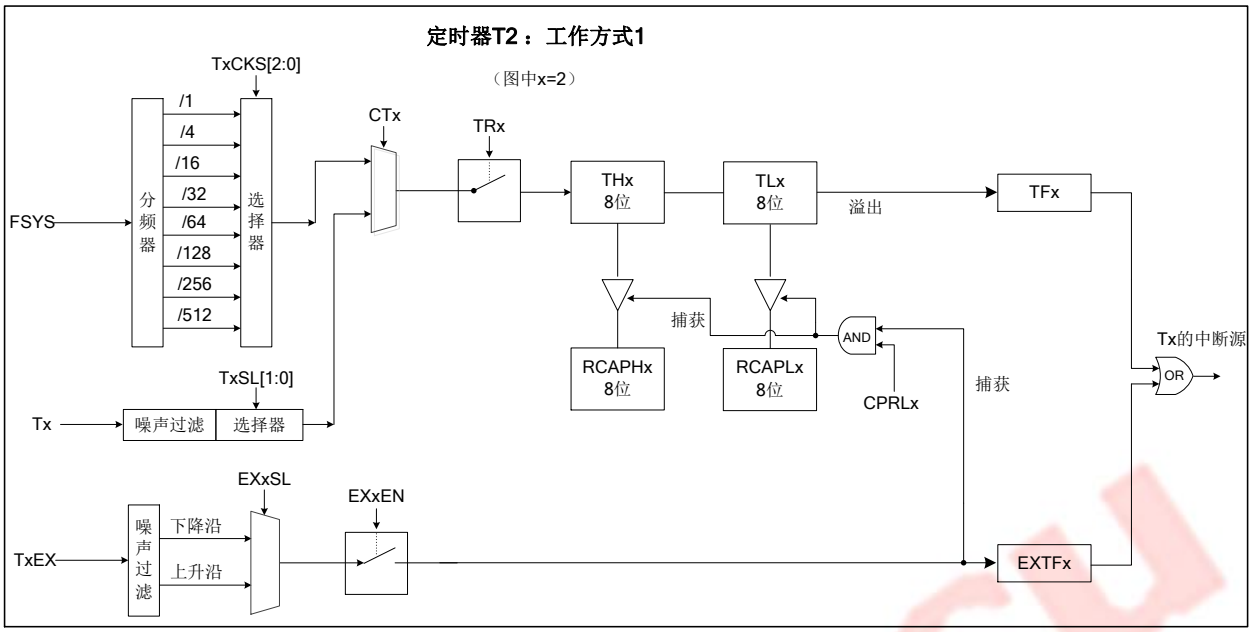
标准情况：使能 T2，将 T2 设置为方式 1，设置时钟频率、定时器模式（CT2=0）等参数。开启 T2EX 端口（EX2EN=1），并选择捕获极性为上升沿 EX2SL=1（下降沿 EX2SL=0）有效，并持续对其进行监视。启动 T2（TR2=1），TH2/TL2 从 0x0 向上开始递增。当 T2EX 端口上产生有效的触发信号（EXTF2=1）时，硬件立即将 TH2/TL2 的值保存到 RCAPH2/RCAPL2 中，同时产生 EXTF2 标志位和 T2 中断。每一次的 T2EX 有效信号都会触发一次捕获，并覆盖上一次的 RCAPH2/PCAPL2 值。

特殊情况一：用户觉得内部时钟源不好，可以改变时钟源。当 CT2=1 时，时钟源切换到外部，通过 T2 口送入。用户还可通过 T2SL 位，决定有效信号的方式，当 T2SL=01B 时上升沿有效，当 T2SL=10B 时下降有效，当 T2SL=11B 时双沿有效。当选择外部时钟时，由于内部系统时钟采样同步，外部时钟最大不能超过 $F_{SYS}/2$。

特殊情况二：假设测量 T2EX 端口上的一个低电平，但由于测量的波形时间特别长，且超过了 T2 的最大计数时间。T2 启动（TR2=1）开始计数，T2EX 端口下降沿执行第一次捕获，上升沿未到达之前，将产生 T2 溢出，发生 TF2 中断。用户可以在进入中断后通过标志位判定中断原因，并记录溢出次数，等待 T2EX 上的触发信号到来。这样，用户的中断处理函数增加一定溢出处理，便可以可以实现长时间间隔的有效沿捕获。假设，首次触发捕获后，经过了 3 个溢出时间中断后，有效的上升沿到来，那么本次测量 2 次有效沿宽度的时间计数值如下：

有效沿宽度=RCAP1(最后一次中断取得的 RCAPH2 和 RCAPL2)+0xFFFF*3（前 3 次 T2 的溢出时间）- RCAP0(首

次捕获值 RCAPH2 和 RCAPL2)。



8.3.3 方式 2：16 位波特率发生器方式

通过设置 RCLKUn 位和 TCLKUn 位（n=0-1）来选择 T2 定时器作为 UARTn（n=0-1）的波特率发生器。UARTn 的接收器和发送器可以同时使用 T1 或者 T2；UARTn 的接收器和发送器也可以设置不同的波特率，但 T2 同一时刻只可以提供一种波特率，若用户同时需要两种不同的波特率，可以共同使用 T2 与 T1 配合实现。T1 和 T2 物理连接 UARTn 的控制位，分别通过 RCLKU0、TCLKU0、RCLKU1、TCLKU1 位控制，因为兼容性的关系这些控制位都分布在 T2 的控制寄存器中。对 UART 的配置，请用户参考相关章节的内容。

T2 作为波特率发生器模式，LD2EN 控制无效，保持重载方式，溢出将不会发生溢出中断，T2EX 可以作为独立的外部中断使用。

在此方式下，若 EX2EN 置位（外部信号有效），则当 T2EX 端口上出现有效的信号，仍会产生 EXTF2 标志位和 T2 中断，但不会引起 TH2/TL2 的重载和重新计数。此模式下，T2EX 可以作为独立的外部中断功能使用。

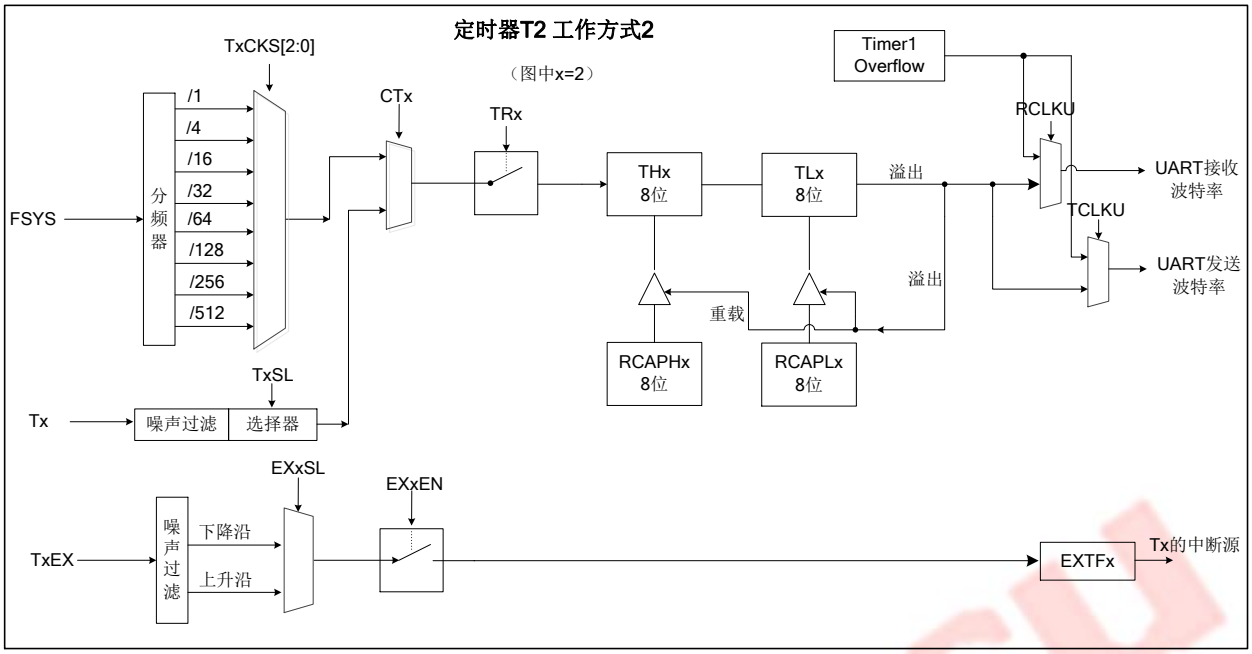
特殊情况一：用户觉得内部时钟源不好，可以改变时钟源。当 CT2=1 时，时钟源切换到外部，通过 T2 口送入。用户还可通过 T2SL 位，决定有效信号的方式，当 T2SL=01B 时上升沿有效，当 T2SL=10B 时下降有效，当 T2SL=11B 时双沿有效。当选择外部时钟时，由于内部系统时钟采样同步，外部最大不能超过 Fsys/2。

在 UART 方式 1 和方式 3 中波特率可由该方式提供，波特率的计算公式如下：

$$\textbf{T2产生的波特率} = \frac{1}{16} \times \frac{\textbf{Fsys}}{\{\textbf{Pre_scale} \times [65536 - ((\textbf{RCAPH2}) \times 256 + (\textbf{RCAPL2}))]\}}$$

其中 n 为 0 或者 1，Fsys 为系统时钟频率，预分频值 Pre_scale 为 T2 时钟选择位 T2CKS[2:0]的分频系数，RCAPH2/RCAPL2 为实际设定值。

注：当用定时器 T2 作为波特率发生器时，T2 的 TH2/TL2 和 RCAPH2/RCAPL2 值不可设置为 0xFFFF；同时，应该关闭 T2 的中断使能位，不让 T2 进入中断。



8.3.4 方式 3：16 位输出比较方式

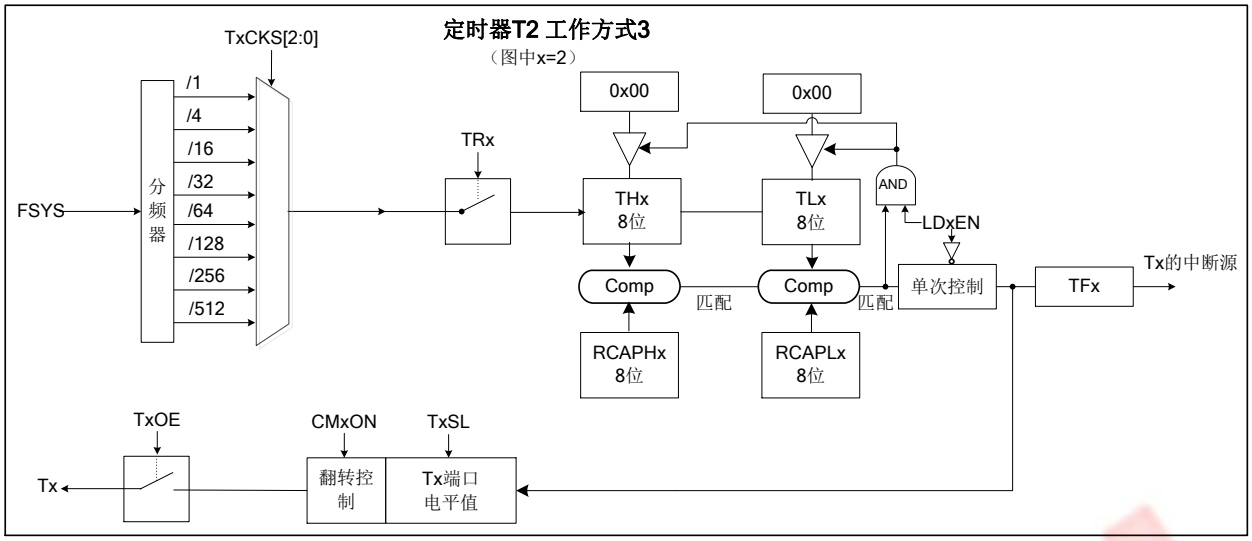
在 T2 的比较模式中，时钟源仅来源于对 F_{SYS} 或其分频，T2EX 端口作输入无效。T2 端口仅作输出使用（若用户不想输出也可以不输出），不可作为时钟输入，但 T2 端口最终是否输出，可由 T2OE 位控制，且该端口的输出初始电平值由 T2SL[1:0]位控制。在 T2 端口允许输出的前提下，每次比较事件后 T2 端口上可以指定输出一个高/低电平（CM2ON=0，初始电平由 T2SL 决定），或者控制其翻转一次电平（CM2ON=1）。

比较模式下，T2 从 0 开始计数，TH2 和 TL2 计数计到与 RCAPH2 和 RCAPL2 的值相同后，发生输出比较匹配事件，硬件产生一次 TF2 置位的过程，若 LD2EN=1，则 TH2 和 TL2 重载 0x0 继续计数。若 LD2EN=0，TH2/TL2 计数至比较匹配时不自动重载。当中断打开，每次比较匹配（TF2 置位）将触发中断。输出比较功能，可用于产生类方波。

标准情况：使能 T2，将 T2 设置为方式 1，设置时钟频率、定时器模式（CT2=0）等等参数。重载有效（LD2EN=1），T2 启动（TR2=1），向上递增计数时，TH2/TL2 从 0x0 开始计到 RCAPH2/RCAPL2 中的值产生比较匹配事件，则 TF2 标志位置位（TF2=1）和产生 T2 中断，并将 0x0 载入 TH2/TL2 重新开始计数；允许外部输出 T2OE=1，且当 T2SL0=0 时，CM2ON=1，第一次比较匹配事件发生后 T2 端口上的电平由初始低电平翻转为高电平，第二次比较匹配事件后则由高电平翻转为低电平，之后交替翻转。

特殊情况：重载无效（LD2EN=0），T2 启动（TR2=1），基本同“标准情况”，每次比较匹配事件发生后，T2 端口上输出的电平翻转一次后保持不变。当 T2OE=0 时，将没有输出，T2 端口可用作其它复用功能，这样的话仅实现了一个定时中断功能。

注：初始电平的配置（T2SL0 和 CM2ON）会立即生效，T2OE 开启，对应的电平状态即会传递至 GPIO，若要实现更准确的单脉冲输出，请配置 GPIO 电平为预期脉冲电平的相反电平，T2SL0 和 CM2ON 配置脉冲初始电平，然后尽可能的同时启动 T2OE 和 TR2。



8.4 T2 相关寄存器

T2 寄存器汇总表

地址(SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0 /CE	**T2CON1**	T2 控制寄存器 1	CM2ON	CM2	BR2EN	RCLKU1	TCLKU1	EX2SL	-	T2EN	0000 0010
SFR0 /CD	**TH2**	T2 计数高位寄存器	TH27	TH26	TH25	TH24	TH23	TH22	TH21	TH20	0000 0000
SFR0 /CC	**TL2**	T2 计数低位寄存器	TL27	TL26	TL25	TL24	TL23	TL22	TL21	TL20	0000 0000
SFR0 /CB	**RCAPH2**	T2 重载/捕获高位寄存器	RCAPH27	RCAPH26	RCAPH25	RCAPH24	RCAPH23	RCAPH22	RCAPH21	RCAPH20	0000 0000
SFR0 /CA	**RCAPL2**	T2 重载/捕获低位寄存器	RCAPL27	RCAPL26	RCAPL25	RCAPL24	RCAPL23	RCAPL22	RCAPL21	RCAPL20	0000 0000
SFR0 /C9	**T2MOD**	工作模式寄存器	LD2EN	T2CKS2	T2CKS1	T2CKS0	T2OE	T2SL1	T2SL0	-	0000 000-
SFR0 /C8	**T2CON**	T2 控制寄存器	TF2	EXTF2	RCLKU0	TCLKU0	EX2EN	TR2	CT2	CPRL2	0000 0000

T2 控制寄存器（0xC8，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2CON	TF2	EXTF2	RCLKU0	TCLKU0	EX2EN	TR2	CT2	CPRL2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] **TF2** – 定时器/计数器 T2 溢出标志位

0：T2 无溢出，中断响应后仅可由软件清 0；

1：T2 溢出时由硬件置 1。

BIT[6] **EXTF2** – 定时器/计数器 T2 重载触发/输入捕获标志位

0：T2EX 上无外部信号输入/对外输出信号，必须由软件清 0；

1：T2EX 上检测到有效的外部信号输入/产生有效的对外输出信号，T2 做重载触发/输入捕获的同时，由硬件置 1。

BIT[5] **RCLKU0** – UART0 波特率接收时钟选择位

0：UART0 用 T1 溢出率作为发送波特率；

1：UART0 用 T2 溢出率作为发送波特率。

***注：** 当 UART 的波特率选择用 T1 或者 T2 作为发送用时，用户应当避免 T1 或者 T2 再做他用。*

BIT[4] **TCLKU0** – UART0 波特率发送时钟选择位

0：UART0 用 T1 溢出率作为接收波特率；

1：UART0 用 T2 溢出率作为接收波特率。

***注：** 当 UART 的波特率选择用 T1 或者 T2 作为接收用时，用户应当避免 T1 或者 T2 再做他用。*

BIT[3] **EX2EN** – T2EX 端口作为重载触发/输入捕获使能位

0：禁止 T2EX 端口作为重载触发/输入捕获功能，或者用于通用 IO 或其他复用模块（由 T2CON 的 C/T2 位决定）；

1：使能 T2EX 端口作为重载触发/输入捕获功能，具体是重载触发还是输入捕获功能，由此时 T2 的具体工作方式决定（工作方式 0=重载触发；工作方式 1=输入捕获），激活 T2 执行一次重载触发/输入捕获的事件。

***注：** 有效触发的形式，由 EX2SL 位决定，即 EX2SL=0 为下降沿触发，EX2SL=1 为上升沿触发。*

BIT[2] **TR2** – 定时器/计数器 T2 使能位

0：停止定时器/计数器 T2；

1：开始定时器/计数器 T2，或者再次写入 1 后复位 T2，让 T2 重新开始计数，即定时器将从 0x0/0xFFFF 开始向上/向下计数，具体计数方向视 DC2EN 位而定。

BIT[1] **CT2** – T2 的计数器/ 定时器方式选择位

0：T2 工作于定时器方式，T2 的时钟源来自内部；

1：T2 工作于计数器方式，T2 端口作为 T2 的外部时钟源专用输入口，且不能做其它复用功能。

BIT[0] **CPRL2** – 定时器/计数器 T2 重载触发/输入捕获方式选择位

0：T2 工作于带重载触发功能的定时器/计数器方式；

1：T2 工作于带输入捕获功能的定时器/计数器方式。

T2 工作方式整理表

T2 工作方式	T2CON1		T2CON
	Bit 6	Bit 5	Bit 0
	CM2	BR2EN	CPRL2
方式 0：16 位重载方式	0	0	0
方式 1：16 位捕获方式	0	0	1
方式 2：16 位波特率发生器方式	0	1	0
方式 3：16 位比较方式	1	0	0
无效状态	其他值		

注：当 CM2、BR2EN、CPRL2 同时置位时，比较模式优先于波特率模式优先于捕获模式。

T2 工作模式寄存器（0xC9，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2MOD	LD2EN	T2CKS2	T2CKS1	T2CKS0	T2OE	T2SL1	T2SL0	-
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	-
复位值	0	0	0	0	0	0	0	0

BIT[7] **LD2EN** – 定时器/计数器 T2 溢出重载控制位

0：在重载模式（方式 0/方式 1/方式 3）中，T2 溢出后不自动重载 TH2/TL2；

1：在重载模式（方式 0/方式 1/方式 3）中，T2 溢出后自动重载 TH2/TL2，使能后将连续重载。

BIT[6:4] **T2CKS[2:0]** – 定时器/计数器 T2 时钟分频选择位

T2CKS[2:0]	T2 频率（CT2M=0）
000	Fsys
001	Fsys/4
010	Fsys/16
011	Fsys/32
100	Fsys/64
101	Fsys/128
110	Fsys/256
111	Fsys/512

BIT[3] **T2OE** – 定时器/计数器 T2 输出使能位

0：禁止 T2 溢出时翻转 T2 端口电平，端口用于通用 IO 或其他复用模块；

1：使能 T2 端口复用于定时器 T2，T2 溢出时翻转一次端口电平，初始电平可由 T2SL[1:0]控制。

BIT[2:1] **T2SL[1:0]** – T2 端口输入/输出信号选择位

T2SL[1:0]	T2 端口有效输入信号选择注 1	T2 端口初始输出信号选择注 2
00	无效	禁止
01	上升沿有效	高电平
10	下降沿有效	低电平
11	双边沿同时有效	禁止

注 1：*仅当 C/T2 位置 1 后，T2 端口上的输入信号才受控于该控制位。*

注 2：*仅在输出比较的情况下，即“工作方式 3”，且 T2OE=1 时，该位决定了 T2 端口的初始电平值，若输出比较模式中，用户需要了解当前 T2 端口的实际电平值，可以通过查询对应的（P2.0）端口的数据值获得。*

T2 控制寄存器 1（0xCE，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2CON1	CM2ON	CM2	BR2EN	RCLKU1	TCLKU1	EX2SL	-	T2EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	-	R/W
复位值	0	0	0	0	0	0	-	0

BIT[7] **CM2ON** – 比较方式下 T2 端口输出电平翻转控制位

0：T2 端口在比较模式（方式 3）下，每次触发后 T2 端口自动翻转一次输出电平；

1：T2 端口在比较模式（方式 3）下，每次触发后 T2 端口输出（初始）电平保持不变。

注：*T2 端口上的初始电平由 T2SL 决定。*

BIT[6] **CM2** – 比较方式选择位

0：T2 工作于（方式 0、方式 1、方式 2）非 16 位输出比较模式；

1：T2 工作于（方式 3）16 位输出比较模式。

BIT[5] **BR2EN** – UART 波特率发生器模式使能位

0：T2 工作于（方式 0、方式 1、方式 3）非波特率发生器模式；

1：T2 工作于（方式 2）波特率发生器模式。

BIT[4] **RCLKU1** – UART1 波特率接收时钟选择位

0：UART1 用 T1 溢出率作为发送波特率；

1：UART1 用 T2 溢出率作为发送波特率。

注：当 UART 的波特率选择用 T1 或者 T2 作为发送用时，用户应当避免 T1 或者 T2 再做他用。

BIT[3] **TCLKU1** – UART1 波特率发送时钟选择位

0：UART1 用 T1 溢出率作为接收波特率；

1：UART1 用 T2 溢出率作为接收波特率。

***注：**当 UART 的波特率选择用 T1 或者 T2 作为接收用时，用户应当避免 T1 或者 T2 再做他用。*

BIT[2] **EX2SL** – T2EX 端口输入捕获/重载触发信号有效沿/电平选择位

0：T2EX 端口作为输入捕获或者重载触发用时，下降沿为有效输入信号；

1：T2EX 端口作为输入捕获或者重载触发用时，上升沿为有效输入信号。

EX2SL	（当 EX2EN=1）T2EX 端口有效输入信号选择
1	上升沿有效
0	下降沿有效

BIT[0] **T2EN** – 定时器/计数器 T2 模块使能位

0：关闭 T2 模块；

1：打开 T2 模块。

T2 计数高位寄存器（0xCD，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH2	TH27	TH26	TH25	TH24	TH23	TH22	TH21	TH20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **TH2[7:0]** – 定时器/计数器 T2 高 8 位计数器设定值

T2 计数低位寄存器（0xCC，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TL2	TL27	TL26	TL25	TL24	TL23	TL22	TL21	TL20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **TL2[7:0]** – 定时器/计数器 T2 低 8 位计数器设定值

注 1：无论是读或者写 TL2/TH2 的值，必须先停止（TR2=0），否则 T2 将产生不可预知的情况。

注 2：写操作时：必须先写 TL2，再写 TH2；否则 TL2 的值不能立即反映到内部 counter 上；读操作时：必须先读 TL2，再读 TH2，否则将读取到错误的值。

T2 重载/捕获高位寄存器（0xCB，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCAPH2	RCAPH27	RCAPH26	RCAPH25	RCAPH24	RCAPH23	RCAPH22	RCAPH21	RCAPH20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **RCAPH2[7:0]** – 定时器/计数器 T2 重载/捕获高 8 位寄存器

T2 重载/捕获低位寄存器（0xCA，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCAPL2	RCAPL27	RCAPL26	RCAPL25	RCAPL24	RCAPL23	RCAPL22	RCAPL21	RCAPL20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **RCAPL2[7:0]** – 定时器/计数器 T2 重载/捕获低 8 位寄存器

注 1：无论是读或者写 RCAPL2/RCAPH2 的值，必须先停止（TR2=0），否则 T2 将产生不可预知的情况。

注 2：写操作时：必须先写 RCAPL2，再写 RCAPH2；否则 RCAPH2 的值不能立即反映到内部 counter 上；读操作时：必须先读 RCAPL2，再读 RCAPH2，否则将读取到错误的值。

8.5 定时器 T3

芯片包含 1 个 16 位基础定时器 T3，T3 是一个向上递增型定时器/计数器，具有 2 种工作在模式：通用定时器模式和时基计数器模式。

8.5.1 方式 0：16 位通用定时器模式

当 T3 作定时使用时，可通过 CT3 位（=0）设定，T3 可作为 16 位通用定时模式使用。此时，其时钟源为系统时钟或其分频，通过 T3CKS 设置，可选择多种溢出时间间隔（具体值参考 T3 控制寄存器说明），溢出后产生 T3 中断并重新从 0000H 开始递增计数。在停止/休眠模式下，若定时器选定的时钟源在该模式下继续工作，则定时器也将继续工作并溢出可唤醒。

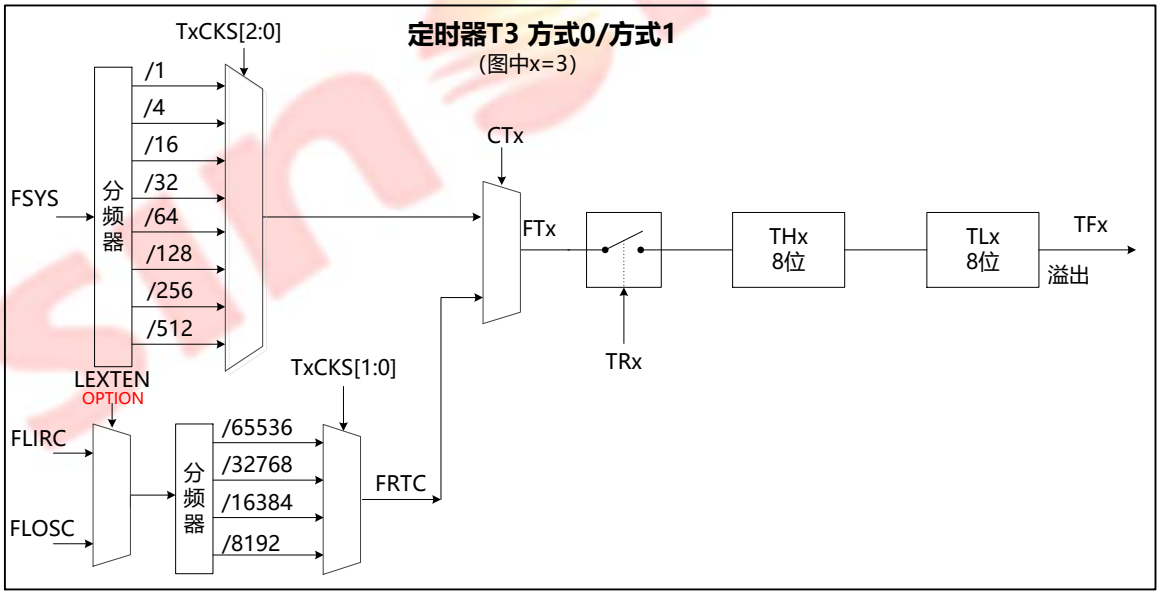
8.5.2 方式 1：16 位时基计数器模式

当 T3 作计数器使用时，可通过 CT3 位（=1）设定，T3 可作为 16 位时基计数模式。此时，通过用户配置字的 LEXTEN 位设定其时钟源为外部 32768Hz 振荡器或其分频，还是选择内部 32KHz 低频振荡器或其分频，通过 T3CKS 设置，可选择多种溢出时间间隔（具体值参考 T3 控制寄存器说明）。方式 1 类似于方式 0，它们的本质类似，只是计数的来源不同。

若选择内置 32KHz 低频振荡器作为时钟源，通过设定 OSCM 的 LFEN 位确定 T3 是否在停止模式/休眠模式下具备唤醒功能。当 LFEN=1，停止模式/休眠模式下 32KHz 时钟源继续工作，当寄存器 TH3/TL3 溢出后会产生 T3 中断，并通过中断从而实现在停止模式/休眠模式下唤醒芯片的功能。当 LFEN=0，停止模式下 32KHz 时钟将继续工作，则 T3 在停止模式具备唤醒芯片的功能；休眠模式下 32KHz 时钟源停止，T3 不具备唤醒芯片的功能。

若选择外部 32768Hz 振荡器作为时钟源时（需在用户配置字中使能 LEXTEN），并连接外部 32768Hz 振荡器到对应的 OSCI/OSCO 端口。这种模式下可以实现对时间的精准计时，且 32768Hz 振荡器不会因芯片进入停止模式/休眠模式而停止工作。所以在停止模式/休眠模式下，外部 32768Hz 振荡器将始终工作，T3 始终具备唤醒芯片的功能。

用户应充分考虑在停止模式/休眠模式下是否需要使用 T3 作为唤醒源，并做相应的设定，否则无论是内置 32KHz 还是外部 32768Hz 振荡器，始终“开启”为 T3 提供时钟源的情况下，都会增加芯片的功耗。



8.6 T3 相关寄存器

T3 寄存器汇总表

地址(SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0	**TH3**	T3 计数高位寄	TH37	TH36	TH35	TH34	TH33	TH32	TH31	TH30	0000

/97		存器									0000
SFR0 /96	**TL3**	T3 计数低位寄存器	TL37	TL36	TL35	TL34	TL33	TL32	TL31	TL30	0000 0000
SFR0 /8F	**T3CR**	T3 控制寄存器	TR3	CT3	TF3	-	T3EN	T3CKS 2	T3CKS 1	T3CKS 0	000-0 000

T3 控制寄存器（0x8F，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3CR	TR3	CT3	TF3	-	T3EN	T3CKS2	T3CKS1	T3CKS0
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
复位值	0	0	0	-	0	0	0	0

BIT[7] **TR3** – 定时器/计数器 T3 使能位

0：停止定时器/计数器 T3；

1：打开定时器/计数器 T3，每次写入 1，作定时器情况，将从 TH3 和 TL3 的起始值开始递增计数；计数器情况下，将从 0 开始递增计数。

BIT[6] **CT3** – 定时器/计数器 T3 模式选择位

0：T3 为通用定时器模式，时钟源为系统时钟或其分频；

1：T3 为时基计数器模式，时钟源为内置 32KHz 振荡器或者外部 32768Hz 振荡器或其分频，具体由“用户配置字”决定。

注 1：当该位=1 时，时钟源由“用户配置字”决定，若用户配置字中打开外部 32768Hz 振荡器（相应的端口 OSCI 和 OSCO 上已经接了振荡器），那么选 32768Hz 振荡器或其分频作为时钟源，否则为选 32KHz 内部振荡器或其分频作为时钟源。

注 2：当该位=1 时，由于外部没有 327868Hz 振荡器，（LFEN=0）32KHz 内部振荡器也停止时，T3 将停止工作。

BIT[5] **TF3** – 定时器/计数器 T3 溢出标志位

0：定时器/计数器 T3 无溢出，中断响应后硬件清 0，也可由软件清 0；

1：定时器/计数器 T3 溢出时，由硬件置 1。

BIT[3] **T3EN** – 定时器/计数器 T3 模块使能位

0：关闭定时器/计数器 T3 模块；

1：打开定时器/计数器 T3 模块。

BIT[2:0] **T3CKS[2:0]** – T3 时基计数器时间选择位 / T3 通用定时器时钟分频选择位

作计数器时，对应时间选择列表：

T3CKS[2:0]	选择外部 32768Hz (CT3=1，LETXEN 有效)	选择内部低速 Flclk（32KHz) (CT3=1，LETXEN 无效)
000	0.25S	约 0.25s
001	0.5S	约 0.5s
010	1S	约 1s
011	2S	约 2s
100	禁止	禁止
101	禁止	禁止
110	禁止	禁止
111	禁止	禁止

注：上表中的“禁止”状态，用户请勿设置，否则 T3 的工作状态将不可预知。

作定时器时，对应频率选择列表：

T3CKS[2:0]	选择内部高速 Fsys (CT3=0，T3M=X)

000	Fsys
001	Fsys/4
010	Fsys/16
011	Fsys/32
100	Fsys/64
101	Fsys/128
110	Fsys/256
111	Fsys/512

T3 计数高位寄存器（0x97，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH3	TH37	TH36	TH35	TH34	TH33	TH32	TH31	TH30
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **TH3[7:0]** – 定时器/计数器 T3 高 8 位计数器设定值

T3 计数低位寄存器（0x96，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TL3	TL37	TL36	TL35	TL34	TL33	TL32	TL31	TL30
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **TL3[7:0]** – 定时器/计数器 T3 低 8 位计数器设定值

注 1：无论是读或者写 TL3/TH3 的值，必须先停止（TR3=0），否则 T3 将产生不可预知的情况。

注 2：写操作时：必须先写 TL3，再写 TH3；否则 TL3 的值不能立即反映到内部 counter 上；特别注意，读操作时：必须先读 TH3，再读 TL3，否则将读取到错误的值。

9 8位单路PWM0

9.1 单路 PWM0 概述

片上提供 1 路 8 位 PWM0 模块，可设置独立中断向量入口，输出极性可选，该 PWM 也可用作具有重载功能的通用定时器使用。输出引脚为 PWM01（P1.4）。

PWM0 周期的计算公式：

$$\text{PWM0周期} = \frac{\text{PWM0P} + 1}{\text{Fsys}}$$

注： *PWM0 周期的时钟设定，固定为 F_{SYS} 或 F_{SYS} 的分频，最大不应该超过 16MHz。*

PWM0 占空比（默认为高电平的时间）的计算公式：

$$\text{PWM0占空比} = \frac{\text{PWM0D} + 1}{\text{PWM0P} + 1}$$

注 1：假设为（默认）高电平有效，当 PWM0D=0 时，占空比为最小 0%，输出恒为 0，但每个周期之间存在一个 clock(由 1 到 0)切换的时间;当 PWM0D<PWM0P,占空比为 PWM0D+1 的高电平时间;当 PWM0D≥PWM0P 时，占空比为最大 100%，输出恒为 1。

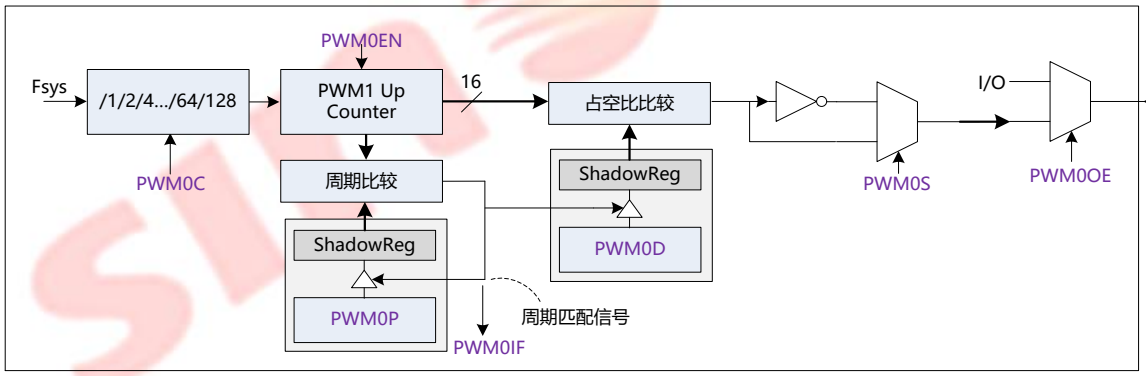
注 2：在 PWM0 输出波形时，若当前周期或占空比被修改，不会立即生效，需要等到下个周期来到才会相应改变。

注 3：占空比所代表的电平，由 PWM0S 位决定，用户可以切换。当 PWM0S 设为 1 时，这里的 PWM0 占空比指"低电平"所占的时间。

注 4：当用户需要 PWM0 输出恒 0/1 时，推荐下面两种做法：A. 在 PWM 功能下反向控制极限位，并以 100%占空比输出，此时将获得需要的电平值；B. 给 Px 端口指定一个 0/1 电平，并通过改变 PWM0OE 位退出 PWM 功能，切换至 GPIO 的输出功能，直接输出指定的电平值。

9.2 PWM0 结构图

PWM0 结构示意图



9.3 单路 PWM0 相关寄存器

PWM0 寄存器汇总表

地址(SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0 /DF	**PWM0D**	PWM0 占空比设置寄存器	PWM0D7	PWM0D6	PWM0D5	PWM0D4	PWM0D3	PWM0D2	PWM0D1	PWM0D0	0000 0000
SFR0 /DE	**PWM0P**	PWM0 周期设置寄存器	PWM0P7	PWM0P6	PWM0P5	PWM0P4	PWM0P3	PWM0P2	PWM0P1	PWM0P0	0000 0000
SFR0 /DD	**PWM0CR**	PWM0 控制寄存器	PWM0EN	-	PWM0IF	PWM0OE	PWM0S	PWM0C2	PWM0C1	PWM0C0	0-00 0000

PWM0 控制寄存器（0xDD，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0CR	PWM0EN	-	PWM0IF	PWM0OE	PWM0S	PWM0C2	PWM0C1	PWM0C0
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	-	0	0	0	0	0	0

BIT[7] **PWM0EN** – PWM0 使能位

0：关闭 PWM0 模块，关闭后 PWM 停止计数，输出也将停止；

1：打开 PWM0 模块，PWM0 计数器从 0 开始向上计数，PWM0 对应的占空比比较器不断比较 PWM0D 与 PWM0CNT 的值，当两者相同时反转一次输出波形，比较 PWM0P 与 PWM0CNT 的值，当周期结束时再反转一次，并继续新一轮周期的比较反转。

注 1： PWM0EN 即是计数的控制位，配置 PWM0 时须先配置周期寄存器和占空比寄存器，再设置 PWM0EN 位为 1。

注 2：PWM0 的输出与否，由 PWM0OE 位决定；

注 3：PWM0CNT 是一个内部寄存器，对用户不可见，向上自增，将在 PWM0EN 启动后开始计数或者重新计数。

BIT[5] **PWM0IF** – PWM0 中断标志位

0：PWM0 定时器未溢出（计数未超过周期寄存器值），由软件清 0；

1：PWM0 定时器已溢出（计数超过周期寄存器值），由硬件置 1。

BIT[4] **PWM0OE** – PWM0 输出使能位

0：禁止 PWM0 复用功能，端口用于通用 IO 或其他复用模块；

1：打开 PWM0 复用功能，作为 PWM0 输出波形用。

注 1：当 PWM0EN 设置为 0 时，硬件会同时将此 PWM0OE 位清 0，再次开启 PWM 时，需要重新配置 PWM0EN=1 和 PWM0OE=1。

注 2：当禁止 PWM0 复用功能时，PWM0 可用作为“具有重载功能的通用定时器”使用。

BIT[3] **PWM0S** – PWM0 输出极性选择位

0：PWM0 输出占空比有效期间为高电平；

1：PWM0 输出占空比有效期间为低电平。

BIT[2:0] **PWM0C[2:0]** – PWM0 工作时钟频率选择位

PWM0C[2:0]	PWM0 工作频率
000	Fsys
001	Fsys/2
010	Fsys/4
011	Fsys/8
100	Fsys/16
101	Fsys/32
110	Fsys/64
111	Fsys/128

注：在 PWM0 输出波形时，请勿随意修改频率，否则会影响 PWM0 的输出结果；建议按先关闭 PWM0，修改后再打开 PWM0 的顺序操作。

PWM0 周期设置寄存器（0xDE，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0P	PWM0P7	PWM0P6	PWM0P5	PWM0P4	PWM0P3	PWM0P2	PWM0P1	PWM0P0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM0P[7:0]** – 设置 PWM0 的周期。

PWM0 占空比设置寄存器（0xDF，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0D	PWM0D7	PWM0D6	PWM0D5	PWM0D4	PWM0D3	PWM0D2	PWM0D1	PWM0D0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM0D [7:0]** – 设置 PWM0 的占空比，即 PWM0 周期内先输出有效电平的时钟个数。

注：在 PWM0 配置时，建议先设置周期和占空比寄存器，再配置控制寄存器。

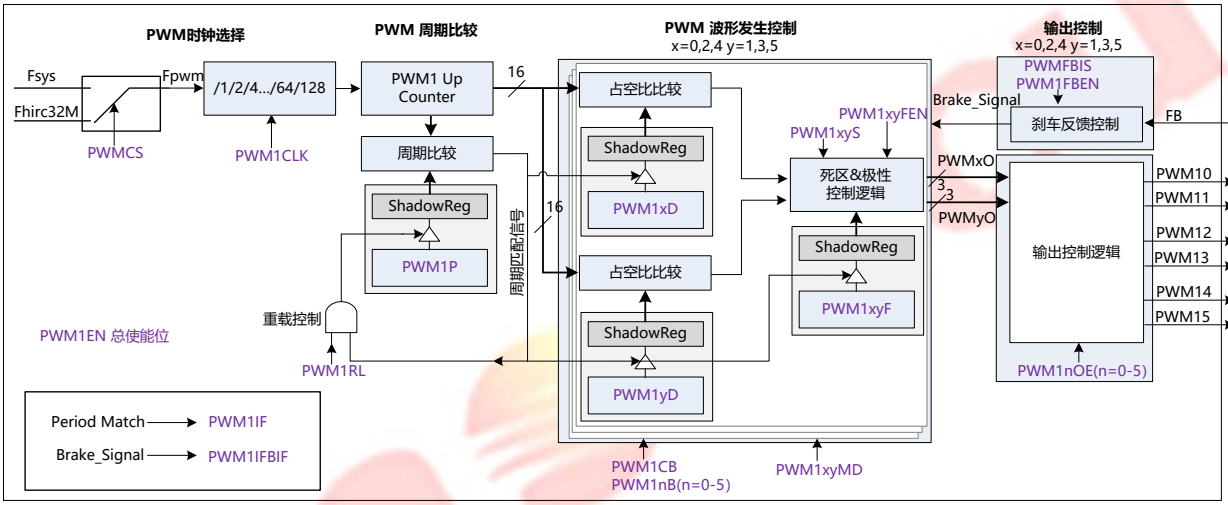
10 16位增强型PWM1阵列

10.1 概述

片上提供一个 16 位共周期增强型 PWM1 阵列模块，可以作为 3 组/对带死区互补输出的 PWM 单元或者 6 路独立输出的 PWM 单元。整个 PWM 阵列共用一个周期产生电路，溢出后也共用 1 个中断向量入口，最大工作时钟为片上 HIRC（32MHz），输出极性可选，自动重载可选，并多提供 1 个紧急刹车保护控制用的中断向量 PWM1FB，可作为外部保护信号的输入切断 PWM1 的输出。作互补输出时，提供最多 3 组/对互补输出 PWM 波形，且每组/对占空比可调、死区时间可调，方便用户应对各种实际驱动电路，适用于全桥/半桥等相关控制；做独立输出时，提供最多 6 路 PWM，且每路的占空比可调，适用于调光等控制。

增强型 PWM1 阵列对应 6 路 PWM 的输出通道分别为 PWM10/PWM11 为一组/对，PWM12/PWM13 为一组/对，PWM14/PWM15 为一组/对，它们两两分别共用 PWM1 的 X/Y/Z 单元。

PWM1 模块方框图



10.2 PWM1 时钟源与周期比较

PWM1 时钟源可以选择系统时钟 Fsys 或 Hirc32M 时钟，选择后的时钟 Fpwm 经过预分频提供给 PWM 计数器。PWMCS 位控制时钟源选择，PWM1CLK 位控制分频系数 1/1、1/2、1/4...1/64、1/128 分频。

PWM 计数器为向上计数器，工作在边沿计数模式；

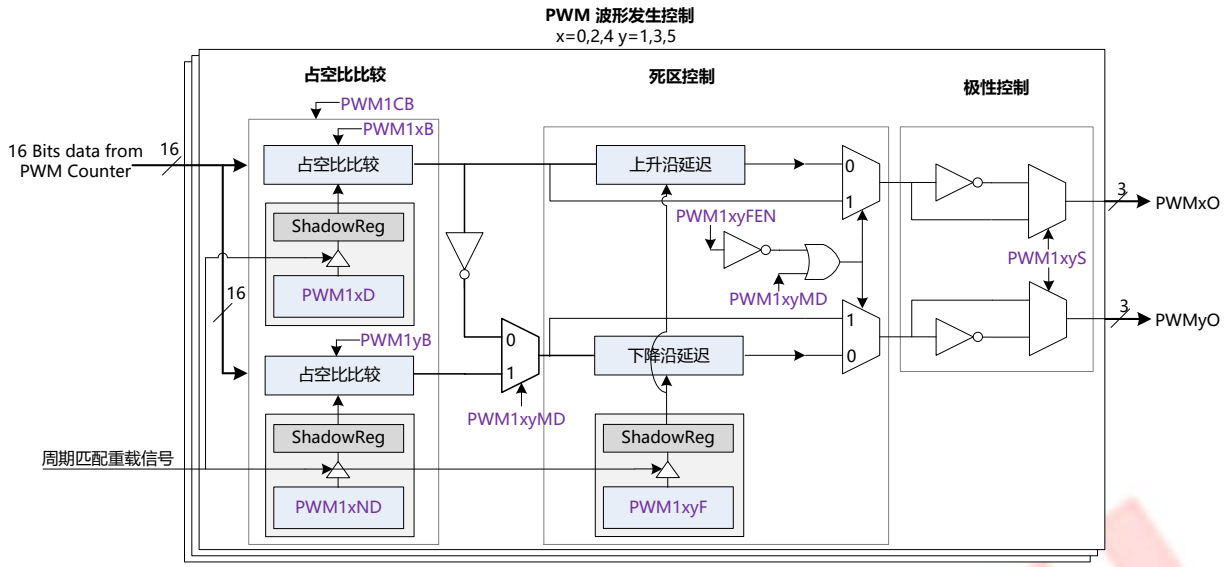
若重载控制位 PWM1RL=1，设置 PWM1EN 位置 1，PWM 计数器将从 0 开始向上计数，直到与 PWM 周期寄存器 PWM1PL/H 匹配，PWM 计数器复位为 0 并继续向上计数，如此往复。

若重载控制位 PWM1RL=0，设置 PWM1EN 位置 1， PWM 计数器将从 0 开始向上计数，直到与 PWM 周期寄存器 PWM1PL/H 匹配，计数器停止计数，PWM 仅启动一次周期计数。

PWM 计数器与周期寄存器值相等，产生周期匹配信号，PWM1IF 置位，若中断使能开启（PWM1IE=1），则发生 PWM1 中断。

10.3 PWM1 波形发生控制

PWM 波形发生控制，使用 PWM 计数器的值与 6 个 16 位占空比缓存寄存器值进行比较，从而产生原始的 6 路 PWM 波形。该原始 6 路 PWM 波形信号经过模式选择、死区控制、极性控制和输出控制，最终在 6 个 PWM 引脚输出。



PWM1 具有 3 个独立的波形发生模块，通过寄存器配置可设置为 3 路互补 PWM 输出或 6 路独立 PWM 输出。在作为互补输出时，可以控制死区时间。

PWM1 周期的计算公式：

$$\text{PWM1周期} = \frac{(\text{PWM1PH} \times 256 + \text{PWM1PL})}{\text{Fpwm1}}$$

注： PWM1P 的取值范围为 0～0xFFFF。PWM1 的时钟 F_{PWM1}，其可选为 F_{HIRC} 和 F_{SYS} 以及它们两个当中某一个的分频，最大可为 32MHz。

PWM1n 通道（n = 0～5）占空比（默认取高电平的时间）的计算公式：

$$\text{PWM1通道占空比} = \frac{(\text{PWM1nDH} \times 256 + \text{PWM1nDL})}{(\text{PWM1PH} \times 256 + \text{PWM1PL})}$$

注 1： PWM1n 通道的占空比 PWM1nD 的设定值，其取值范围为 0～0xFFFF。假设取高电平的有效时间为占空比时间(默认情况)，当 PWM1D=0 时，占空比为最小 0%，输出恒为 0；当 PWM1D<PWM1P，占空比为 PWM1D 的高电平时间；当 PWM1D≥PWM1P 时，占空比为最大 100%，输出恒为 1。

注 2：在 PWM1 输出波形时，若当前周期或占空比被修改，不会立即生效，需要等到下个周期来到才会相应改变。

注 3：占空比所代表的电平，由 PWM1xyS（xy = 01，23，45）位决定，用户可以切换。当 PWM1xyS 位设为 1 时，这里的 PWM1n 通道的占空比时间，指“低电平”所占的时间。

PWM1xy 组（xy = 01，23，45）死区时间的计算公式：

$$\text{PWM1组死区时间} = \frac{\text{PWM1xyF}}{\text{Fpwm1}}$$

注 1：PWM1xy 组的死区时间数 PWM1xyF 的设定值，其取值范围为 0～0xFF，用户可以自行调整。默认情况下，为 0xFF。特别注意，为了保护设备，其默认值为 0xFF，即开启死区后默认“死区时间”最大。

注 2：PWM1xy 组的死区时间可以分开设置，3 个组可以分别使用 3 个不同的死区时间设定值；在一组上的两个通道使用同一个死区时间；死区时间的原理是先补偿偶数通道上的低电平时间，再补偿奇数通道上的；在独立输出模式下，即使用户设定了死区时间值，该功能也将无视。

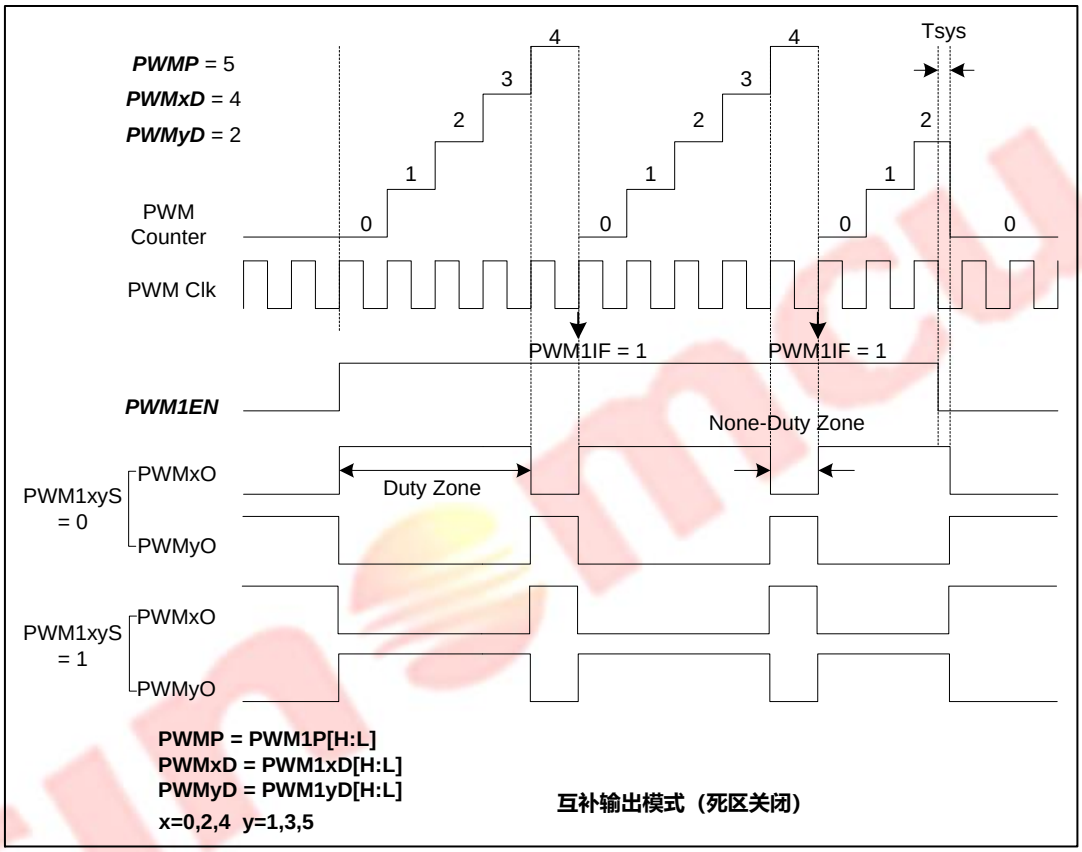
关于 PWM1xy 组（xy = 01，23，45，含死区时间）产生中断时间点的说明：

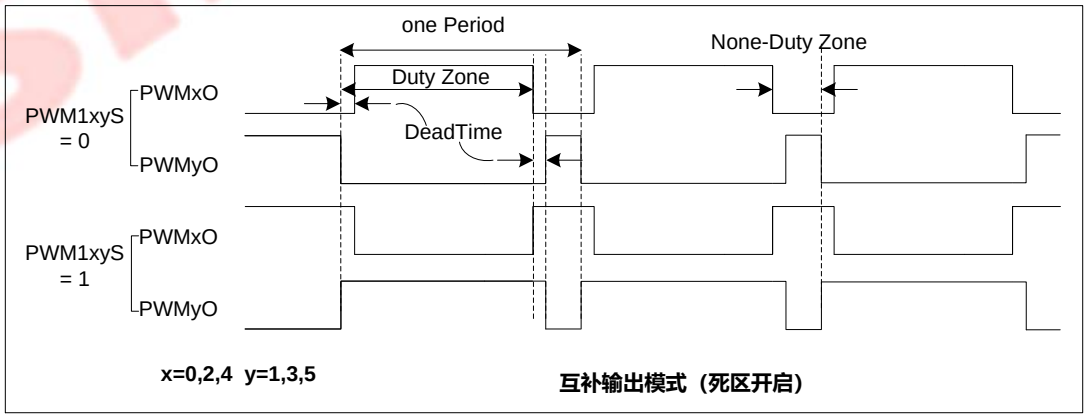
Core 时钟频率（MHz）	当一个周期结束后的（us）
32	0.8
16	1.6
8	3.2
4	6.5
2	13
1	26

注：32MHz 时钟有偏差，故上表的时间点也会存在一定的偏差。

10.3.1 PWM1 互补输出模式

首先打开 PWM1EN 位，使能 PWM1 模块。其次，选择时钟源 PWM1CS 和分频 PWM1CLK[2:0]，设置 PWMxyMD（xy = 01，23，45）置 0（互补输出模式），让 PWM1 阵列工作于互补输出模式，同时对应通道的输出使能位 PWMnOE 置 1（n = 0～5）。在互补输出模式下，需要首先设定 PWM1 的周期 PWM1P，选择是否自动重载 PWM1RL，选择其输出极性 PWM1xyS。接着根据实际应用，设置 PWM1 的对应组中的偶数通道的占空比 PWM1nD（互补输出模式中以偶数通道为准，奇数通道的占空比设置无效），然后按需设置对应组的死区时间 PWM1xyF（一组一个设置值，默认为最大值）。最后将 PWM1nB（n = 0～5）置 1 打开对应的 PWM1 组的比较功能，将 PWM1CB 置 1 启动 PWM1CNT 计数器，开始输出 PWM 波形。用户可以根据实际需求设定是否让 PWM1 的各个通道在一个周期输出后产生中断。

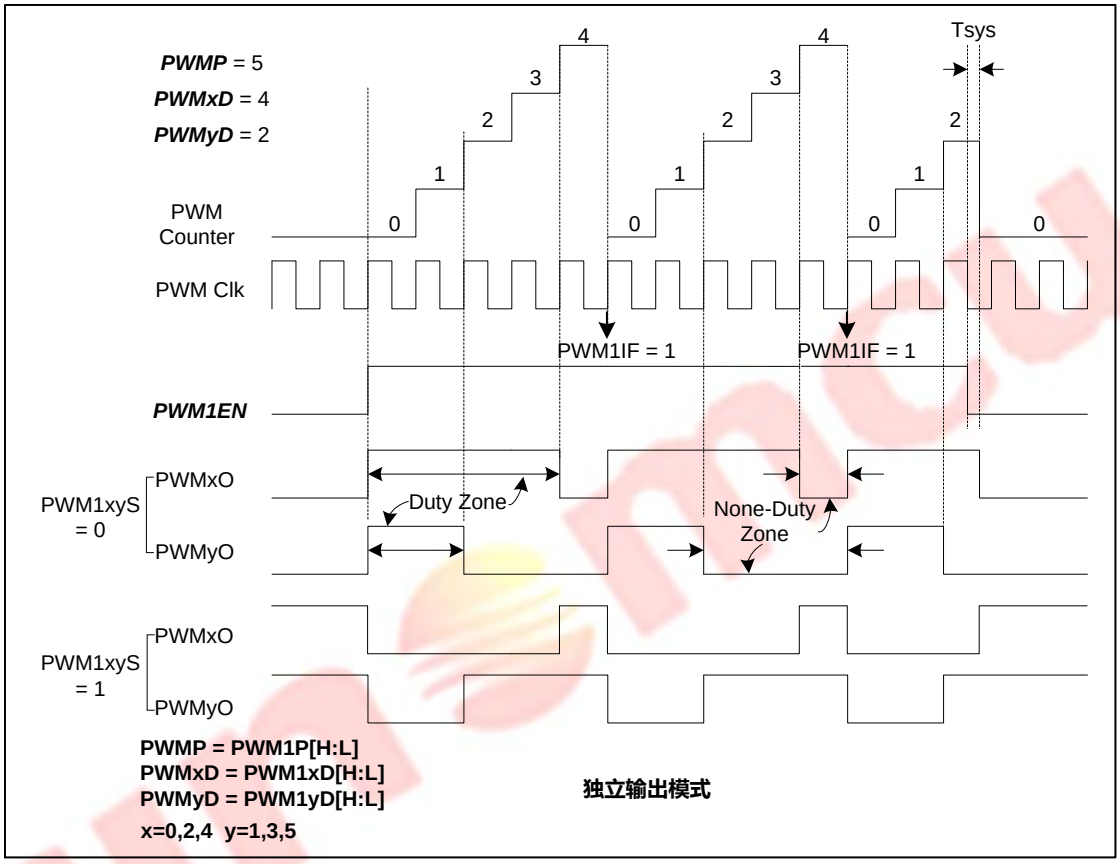




死区时间指是 PWM 输出时，为了使 H 桥或半 H 桥的上/下管不会因为开关速度问题发生同时导通而损坏设备的现象，此时设置的一个保护时段，在这个时间内上/下管都不应打开，即一对互补的 PWM 波形中存在两个的输出停止时间。用户使能了“死区时间”后，PWM 的整个“占空比时间”将减小，即“实际占空比时间”等于“占空比设定时间”减去“死区设定时间”。

10.3.2 PWM1 独立输出模式

首先打开 PWM1EN 位，使能 PWM1 模块。其次，选择时钟源 PWM1CS 和分频 PWM1CLK[2:0]，设置 PWMxyMD（xy = 01，23，45）置 1（独立输出模式），让 PWM1 的工作于独立输出模式，同时对应通道的输出使能位 PWMnOE 置 1（n = 0～5）。在独立输出模式下，需要首先设定 PWM1 的周期 PWM1P，选择是否自动重载 PWM1RL，选择其输出极性 PWM1xyS。接着根据实际应用，设置 PWM1 的对应 x 通道的占空比 PWM1nD（6 路通道的占空比的设置，可设置的各不相同）。最后将 PWM1nB（n = 0～5）置 1 打开对应的 PWM1n 通道的比较功能，将 PWM1CB 置 1 启动 PWM1CNT 计数器，开始输出 PWM 波形。用户可以根据实际需求设定是否让 PWM1 中的各个通道在一个周期输出后产生中断。

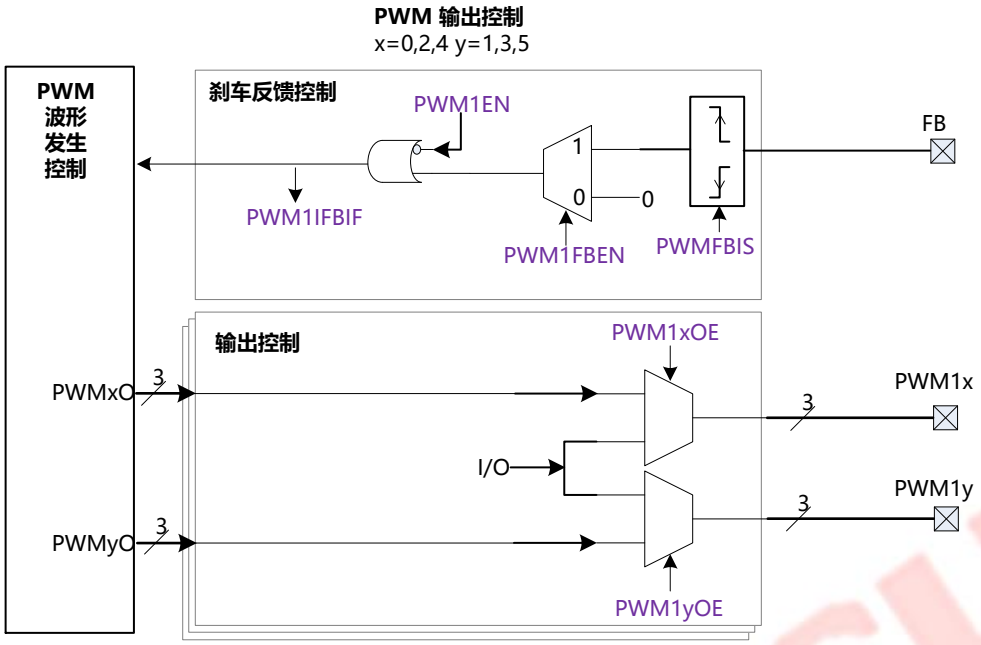


10.3.3 PWM1 混合模式

打开 PWM1EN 位，使能 PWM1 模块。设置 PWMxyMD（xy = 01，23，45）为 0/1，让 PWM1 的某个组工作于互补输出模式，某个组工作于独立输出模式（这里必须是 2 的整数倍通道数）。3 个 PWM 工作组可以灵活调配，工作于以上两种不同的模式下，甚至关闭某个组（互补输出模式下）或者关闭某个通道（独立输出模式下）。设置步骤同上，这里不再赘述。

10.4 PWM1 输出控制

PWM 输出控制与刹车反馈控制用于决定最终的 PWM 波形 PWMxO/PWMyO 是否在 PWM1x/PWM1y 引脚上输出。



刹车反馈控制，PWMFBEN 置 1，使能刹车反馈控制，对应 IO 端口用作 FB 功能；PWM1FBIS 配置 FB 输入有效沿选择；当 FB 输入有效沿信号，刹车保护生效，PWM1IFBIF 置 1，整个 PWM1 停止工作；所有 PWM 端口根据不同的极性设置，**输出高/低电平**。若对应的中断使能 PWM1FBIE 置 1，则发生 PWM1 刹车中断。

PWM1xyS = 0（占空比区间高电平有效），输出低电平；

PWM1xyS = 1（占空比区间低电平有效），输出高电平。

用户在重新启动 PWM1 前必须手动将 PWM1IFBIF 清 0，清 0 后 PWM1 继续输出；用户在清 0 该位前，请确保刹车保护信号无效，否则将再次进入保护状态。。

PWM 输出引脚与 IO 功能复用，PWM 引脚 PWM1x/PWM1y（x=0,2,4 y=1,3,5）与 IO 功能复用，由控制位 PWM1xOE/PWM1yOE（x=0,2,4 y=1,3,5）控制。对应的 PWMOE 位为 1，可以将 PWM 引脚设置为 PWM 输出功能，反之为 IO 功能。

10.5 特殊情况说明

10.5.1 占空比寄存器值为 0 或周期值（不为 0）

若占空比寄存器为 0 或周期值，则根据不同的极性设置，PWM1x/PWM1y（x=0,2,4 y=1,3,5）应输出占空比为 0%或 100%的波形。计数器值与周期匹配时产生中断标志和重载信号。

PWM1xyS = 0（占空比区间高电平有效）

占空比为 0，互补输出模式下， PWM1x 引脚输出低电平，PWM1y 引脚输出高电平。

占空比为 0，独立输出模式下， PWM1x 和 PWM1y 引脚都输出低电平。

占空比为周期值，互补输出模式下， PWM1x 引脚输出高电平，PWM1y 引脚输出低电平。

占空比为周期值，独立输出模式下， PWM1x 和 PWM1y 引脚都输出高电平。

PWM1xyS = 1（占空比区间低电平有效），输出极性与上述相反。

10.5.2 周期寄存器值为 0

若周期寄存器为 0，则根据不同的极性设置，PWM1x/PWM1y（x=0,2,4 y=1,3,5）应输出波形极性如下。此时，周期中断标志不会产生。

PWM1xyS = 0（占空比区间高电平有效）

占空比为 0，互补输出模式下， PWM1x 引脚输出低电平，PWM1y 引脚输出高电平。

占空比为 0，独立输出模式下， PWM1x 和 PWM1y 引脚都输出低电平。

占空比不为 0，互补输出模式下， PWM1x 引脚输出高电平，PWM1y 引脚输出低电平。

占空比不为 0，独立输出模式下， PWM1x 和 PWM1y 引脚都输出高电平。

PWM1xyS = 1（占空比区间低电平有效），输出极性与上述相反。

10.5.3 进入 STOP 模式，PWM1A/B/C 的波形输出状态

进入 STOP 模式后，PWM 输出口不受影响，比如 PWM1 口原先输出 1K 的方波，进入 STOP 模式后会继续输出 1K 的方波，但是如果设置了 PWM1 中断，会唤醒 STOP 模式，继续执行 IDLE 后面的程序。

10.5.4 进入 SLEEP 模式，PWM1A/B/C 的波形输出状态

进入 SLEEP 模式后，PWM 输出口，根据不同的极性设置，PWM1x/PWM1y（x=0,2,4 y=1,3,5）会输出高/低电平，即使设置了 PWM1 中断，也不能唤醒 SLEEP 模式，SLEEP 模式必须通过其他方式唤醒，具体参考工作模式章节。

PWM1xyS = 0（占空比区间高电平有效），PWM1x 引脚输出低电平，PWM1y 引脚输出高电平。

PWM1xyS = 1（占空比区间低电平有效），输出极性相反。

10.6 PWM1 相关寄存器

PWM1 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR1 /FF	**PWM14 5F**	PWM1 的 45 组死区时间设定寄存器	PWM1 45F7	PWM1 45F6	PWM1 45F5	PWM1 45F4	PWM1 45F3	PWM1 45F2	PWM1 45F1	PWM 145F0	1111 1111
SFR1 /F7	**PWM13 DL**	PWM1 通道 3 占空比设定低位寄存器	PWM1 3D7	PWM1 3D6	PWM1 3D5	PWM1 3D4	PWM1 3D3	PWM1 3D2	PWM1 3D1	PWM 13D0	0000 0000
SFR1 /F6	**PWM13 DH**	PWM1 通道 3 占空比设定高位寄存器	PWM1 3D15	PWM1 3D14	PWM1 3D13	PWM1 3D12	PWM1 3D13	PWM1 3D13	PWM1 3D9	PWM 13D8	0000 0000
SFR1 /F5	**PWM12 3F**	PWM1 的 23 组死区时间设定寄存器	PWM1 23F7	PWM1 23F6	PWM1 23F5	PWM1 23F4	PWM1 23F3	PWM1 23F2	PWM1 23F1	PWM 123F0	1111 1111
SFR1 /F4	**PWM14 DL**	PWM1 通道 4 占空比设定低位寄存器	PWM1 4D7	PWM1 4D6	PWM1 4D5	PWM1 4D4	PWM1 4D3	PWM1 4D2	PWM1 4D1	PWM 14D0	0000 0000
SFR1 /F3	**PWM14 DH**	PWM1 通道 4 占空比设定高位寄存器	PWM1 4D15	PWM1 4D14	PWM1 4D13	PWM1 4D12	PWM1 4D11	PWM1 4D14	PWM1 4D9	PWM 14D8	0000 0000
SFR1 /F2	**PWM15 DL**	PWM1 通道 5 占空比设定低位寄存器	PWM1 5D7	PWM1 5D6	PWM1 5D5	PWM1 5D4	PWM1 5D3	PWM1 5D2	PWM1 5D1	PWM 15D0	0000 0000
SFR1 /F1	**PWM15 DH**	PWM1 通道 5 占空比设定高位寄存器	PWM1 5D15	PWM1 5D14	PWM1 5D13	PWM1 5D12	PWM1 5D15	PWM1 5D15	PWM1 5D9	PWM 15D8	0000 0000
SFR1 /EF	**PWM10 DL**	PWM1 通道 0 占空比设定低位寄存器	PWM1 0D7	PWM1 0D6	PWM1 0D5	PWM1 0D4	PWM1 0D3	PWM1 0D2	PWM1 0D1	PWM 10D0	0000 0000
SFR1 /EE	**PWM10 DH**	PWM1 通道 0 占空比设定高位寄存器	PWM1 0D15	PWM1 0D14	PWM1 0D13	PWM1 0D12	PWM1 0D11	PWM1 0D10	PWM1 0D9	PWM 10D8	0000 0000
SFR1 /ED	**PWM11 DL**	PWM1 通道 1 占空比设定低位寄存器	PWM1 1D7	PWM1 1D6	PWM1 1D5	PWM1 1D4	PWM1 1D3	PWM1 1D2	PWM1 1D1	PWM 11D0	0000 0000

SFR1 /EC	**PWM11 DH**	PWM1 通道 1 占空比设定高位寄存器	PWM1 1D15	PWM1 1D14	PWM1 1D13	PWM1 1D12	PWM1 1D11	PWM1 1D11	PWM1 1D9	PWM 11D8	0000 0000
SFR1 /EB	**PWM10 1F**	PWM1 的 01 组死区时间设定寄存器	PWM1 01F7	PWM1 01F6	PWM1 01F5	PWM1 01F4	PWM1 01F3	PWM1 01F2	PWM1 01F1	PWM 101F0	1111 1111
SFR1 /EA	**PWM12 DL**	PWM1 通道 2 占空比设定低位寄存器	PWM1 2D7	PWM1 2D6	PWM1 2D5	PWM1 2D4	PWM1 2D3	PWM1 2D2	PWM1 2D1	PWM 12D0	0000 0000
SFR1 /E9	**PWM12 DH**	PWM1 通道 2 占空比设定高位寄存器	PWM1 2D15	PWM1 2D14	PWM1 2D13	PWM1 2D12	PWM1 2D11	PWM1 2D12	PWM1 2D9	PWM 12D8	0000 0000
SFR1 /E7	**PWM1I ER**	PWM1 中断控制寄存器	PWM1I F	PWM1 FBIF	-	-	-	-	PWM1 FBIS	PWM 1FBEN	00-- --00
SFR1 /E6	**PWM1 CR0**	PWM1 控制寄存器 0	PWM1 EN	-	PWM1 5OE	PWM1 4OE	PWM1 3OE	PWM1 2OE	PWM1 1OE	PWM 10OE	0-00 0000
SFR1 /E5	**PWM1 CR1**	PWM1 控制寄存器 1	PWM1 CS	PWM1 45MD	PWM1 23MD	PWM1 01MD	-	PWM1 CLK2	PWM1 CLK1	PWM 1CLK0	0000 -000
SFR1 /E4	**PWM1 CR2**	PWM1 控制寄存器 2	PWM1 RL	PWM1 45FEN	PWM1 23FEN	PWM1 01FEN	-	PWM1 45S	PWM1 23S	PWM 101S	0000 -000
SFR1 /E3	**PWM1 CR3**	PWM1 控制寄存器 3	PWM1 CB	-	PWM1 5B	PWM1 4B	PWM1 3B	PWM1 2B	PWM1 1B	PWM 10B	0-00 0000
SFR1 /E2	**PWM1P L**	PWM1 周期设定低位寄存器	PWM1 01P7	PWM1 01P6	PWM1 01P5	PWM1 01P4	PWM1 01P3	PWM1 01P2	PWM1 01P1	PWM 101P0	0000 0000
SFR1 /E1	**PWM1P H**	PWM1 周期设定高位寄存器	PWM1 01P15	PWM1 01P14	PWM1 01P13	PWM1 01P12	PWM1 01P11	PWM1 01P10	PWM1 01P9	PWM 101P8	0000 0000

PWM1 中断控制寄存器（0xE7，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1IER	PWM1IF	PWM1FBIF	-	-	-	-	PWM1FBIS	PWM1FBEN
R/W	R/W	R/W	-	-	-	-	R/W	R/W
复位值	0	0	-	-	-	-	0	0

BIT[7] **PWM1IF** – PWM1 阵列中断标志位

0：PWM1 计数器没有产生溢出，由软件清 0；

1：PWM1 计数器产生溢出后，由硬件置 1。

BIT[6] **PWM1FBIF** – PWM1 阵列刹车保护中断标志位

0：无刹车保护中断产生，由软件清 0；

1：当刹车保护端口 PWM1FB 上一个有效的信号来到后，PWM1 阵列刹车保护功能启动，由硬件置 1，整个 PWM1 阵列将停止工作。

注：当刹车保护生效时，即该位置 1，整个 PWM1 停止工作；该位清 0 则 PWM1 继续输出，用户在清 0 该位前，请确保刹车保护信号无效，否则将再次进入保护状态。

BIT[1] **PWM1FBIS** – PWM1 阵列刹车保护输入信号选择位

0：PWM1FB 端口上，上升沿有效；

1：PWM1FB 端口上，下降沿有效。

BIT[0] **PWM1FBEN** – PWM1 阵列刹车控制输入端口 PWM1FB 使能位

0：禁止 PWM1FB（刹车保护）输入功能，端口用于通用 IO 或其他复用模块；

1：端口用作 PWM1FB（刹车保护）输入功能。

注：若 PWM1 整个模块关闭（即 PWM1EN=0），即 PWM1FB 端口使能等以上设定为有效，也不会产生 PWM1FB（刹车保护）中断。

PWM1 控制寄存器 0（0xE6，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR0	PWM1EN	-	PWM15OE	PWM14OE	PWM13OE	PWM12OE	PWM11OE	PWM10OE
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	-	0	0	0	0	0	0

BIT[7] **PWM1EN** – PWM1 总使能位

0：关闭整个 PWM1 模块（含刹车保护功能），PWM1CNT 停止计数；

1：打开整个 PWM1 模块（含刹车保护功能），PWM1CNT 开始计数。

注 1： PWM1EN 即是计数的控制位，又是整个模块的控制位，建议先配置其他设置寄存器，再设置 PWM1EN 位为 1。

注 2：当 PWM1RL 位=0 时，仅发生一个周期的 PWM 波，进入中断的同时 PWM1CNT 停止计数，用户需要先关闭 PWM1EN=0，然后使能 PWM1EN=1，重新再开 PWM1（若设置相同，其它寄存器可以不配），才可以发生 PWM 波；当 PWM1RL 位=1 时，完成一轮周期后硬件自动清 0 后继续计数，将连续不断发生 PWM 波。

注 3：PWM1 计数器（PWM1CNT）是一个内部寄存器，对用户不可见，向上递增方式计数。

BIT[5～0] **PWM1xOE（x = 0～5）** – PWM1x 通道使能位（x = 0～5）

0：关闭对应的 PWM1x 通道，端口用于通用 IO 或其他复用模块；

1：打开对应的 PWM1x 通道，用于 PWM1 的一个通道。

PWM1 控制寄存器 1（0xE5，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR1	PWM1CS	PWM145 MD	PWM123 MD	PWM101 MD	-	PWM1CLK 2	PWM1CLK 1	PWM1CLK 0
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
复位值	0	0	0	0	-	0	0	0

BIT[7] **PWM1CS** – PWM1 时钟源选择位

0：PWM1 阵列的时钟源 F_{PWM1} 选择为 F_{SYS}；

1：PWM1 阵列的时钟源 F_{PWM1} 选择为 F_{HIRC}。

BIT[6,5,4] **PWM1xyMD（xy = 01、23、45）** – PWM1xy 组（xy = 01、23、45）工作模式选择位

0：对应的 PWM1xy 工作于互补输出模式（xy 通道输出为反相）；

1：对应的 PWM1xy 工作独立输出模式。

BIT[2:0] **PWM1CLK[2:0]** – PWM1 时钟选择位

PWM1CLK[2:0]	工作频率
000	F_{PWM1}
001	$F_{PWM1}/2$
010	$F_{PWM1}/4$
011	$F_{PWM1}/8$
100	$F_{PWM1}/16$
101	$F_{PWM1}/32$
110	$F_{PWM1}/64$
111	$F_{PWM1}/128$

注：在 PWM1 输出中，请勿随意修改频率，否则会影响 PWM1 的输出结果；建议按为先关闭 PWM1，修改后再打开 PWM1 的顺序操作。

PWM1 控制寄存器 2（0xE4，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR2	PWM1RL	PWM145FEN	PWM123FEN	PWM101FEN	-	PWM145S	PWM123S	PWM101S
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
复位值	0	0	0	0	-	0	0	0

BIT[7] **PWM1RL** – PWM1 自动重载使能位

0：PWM1CNT 溢出后，自动重载功能无效，PWM1 启动仅一次；

1：PWM1CNT 溢出后，自动重载功能有效。

***注 1：**该位同时对 PWM1xy（xy = 01、23、45）3 组 6 路通道产生影响，即 6 路通道必须同时设置为有效或者无效，不能单独设置；用户可以通过 PWM1nB（n = 0～5）控制一路通道的启动或者停止。*

***注 2：**PWM1CNT 是一个内部寄存器，对用户不可见，向上自增，将在 PWM1CB 启动后开始计数或者重新计数。*

BIT[6,5,4] **PWM1xyFEN（xy = 01、23、45）** – PWM1xy 组（xy = 01、23、45）死区时间使能位

0：在互补模式下，PWM1 对应 xy 组的死区时间无效；

1：在互补模式下，PWM1 对应 xy 组的死区时间有效，硬件会自动在每 1 路通道的上升沿按 PWM1xyF 的设定值（xy = 01、23、45）插入一个延时时间。

***注：**在独立输出模式下，即使该位为 1，死区时间计数器也不会工作。*

BIT[2,1,0] **PWM1xyS（xy = 01、23、45）** – PWM1xy 组（xy = 01、23、45）输出极性选择位

0：PWM1xy 输出占空比有效时间为高电平，即对应端口输出为高电平的时间代表占空比；

1：PWM1xy 输出占空比有效期间为低电平，即对应端口输出为低电平的时间代表占空比。

***注 1：**在独立输出模式下，PWM1xy 两个通道的极性是相同的，即 xy 通道同时为高电平或者低电平有效，但不同组之间极性的设置可以不同；在互补输出模式下，因为同一组的 xy 通道是互补（相反）的，所以该位的设定值代表 PWM1x（x = 0、2、4）的偶数通道极性为准，互补的通道则与之相反。*

***注 2：**极性也决定了占空比在 0%或周期为 0%情况下的输出电平， 详情请参考<9.5 特殊情况说明>章节。*

***注 3：**若用户将 PWM1nOE 置 0 时，则 PWM1n 对应的端口上将无任何的输出。*

PWM1 控制寄存器 3（0xE3，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR3	PWM1CB	-	PWM15B	PWM14B	PWM13B	PWM12B	PWM11B	PWM10B
R/W	R/W	-	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	-	0	0	0	0	0	0

BIT[7] **PWM1CB** – PWM1 比较总使能位

0：PWM1 模块比较无效；

1：PWM1 模块比较有效。

BIT[5～0] **PWM1nB（n = 0～5）** – PWM1x 通道（x = 0～5）比较使能位

0：关闭 PWM1x 通道比较功能，关闭后 PWM1x 通道的比较器不工作；

1：打开 PWM1x 通道比较功能，PWM1 对应的 n 通道的占空比比较器不断比较 PWM1xyD 与 PWM1CNT 的值，当两者相同时反转一次 n 通道的输出波形，比较 PWM1P 与 PWM1CNT 的值，当周期结束时再反转一次，若重载 PWM1RL 位为 0，则继续新一轮周期的比较反转。

***注 1：**设置时，推荐用户依次使能，通道比较器控制 PWM1nB 位，计数器控制 PWM1CB 位，模块总控 PWM1EN 位，保证 PWM1x 通道的同步，以输出完整的波形；否则输出波形可能产生异常；若用户需要打开/关闭某个通道，请先全部关闭后，再依照上述顺序开启。*

***注 2：**在互补输出模式下，xy 一组（xy = 01、23、45）中的两个通道由其所在组中，偶数通道的比较使能位控制，即不论奇数通道打开还是关闭，只要偶数通道打开了，则 xy 组的两个通道都将工作。*

***注 3：**PWM1x 通道的输出与否，由 PWM1nOE 位（n = 0～5）决定，PWM1xy 组 2 个通道中的 1 个通道可单独*

输出。

PWM1 周期设定低位寄存器（0xE2，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PL	PWM101P 7	PWM101P 6	PWM101P 5	PWM101P 4	PWM101P 3	PWM101P 2	PWM101P 1	PWM101P 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM1PL** – PWM1x 周期设定寄存器低 8 位

PWM1 周期设定高位寄存器（0xE1，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PH	PWM101P 15	PWM101P 14	PWM101P 13	PWM101P 12	PWM101P 11	PWM101P 10	PWM101P 9	PWM101P 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM1PH** – PWM1x 周期设定寄存器高 8 位

注：当修改 PWM1x 的周期时，应先修改高位部分，再修改地位部分；读取时不受此限制。

PWM1 通道 0 占空比设定低位寄存器（0xEF，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM10DL	PWM10D 7	PWM10D 6	PWM10D 5	PWM10D 4	PWM10D 3	PWM10D 2	PWM10D 1	PWM10D 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM10DL** – PWM1 通道 0 占空比设定寄存器低 8 位

PWM1 通道 0 占空比设定高位寄存器（0xEE，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM10DH	PWM10D 15	PWM10D 14	PWM10D 13	PWM10D 12	PWM10D 11	PWM10D 10	PWM10D 9	PWM10D 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM10DH** – PWM1 通道 0 占空比设定寄存器高 8 位

注：当修改 PWM1x 的占空比时，应先修改高位部分，再修改地位部分；读取时不受此限制。

PWM1 通道 1 占空比设定低位寄存器（0xED，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM11DL	PWM11D 7	PWM11D 6	PWM11D 5	PWM11D 4	PWM11D 3	PWM11D 2	PWM11D 1	PWM11D 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM11DL** – PWM1 通道 1 占空比设定寄存器低 8 位

PWM1 通道 1 占空比设定高位寄存器（0xEC，SFR1）

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

PWM11DH	PWM11D 15	PWM11D 14	PWM11D 13	PWM11D 12	PWM11D 11	PWM11D 11	PWM11D 9	PWM11D 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM11DH** – PWM1 通道 1 占空比设定寄存器高 8 位

注 1：当修改 PWM1x 的占空比时，应先修改高位部分，再修改地位部分；读取时不受此限制。

注 2：对于 PWM1 的奇数通道 PWM1x（x = 1、3、5），在互补模式下，其占空比设置寄存器无效。

PWM1 的 01 组死区时间设定寄存器（0xEB，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM101F	PWM101F 7	PWM101F 6	PWM101F 5	PWM101F 4	PWM101F 3	PWM101F 2	PWM101F 1	PWM101F 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7:0] **PWM101F** – PWM1 的 01 组死区时间设定寄存器

注 1：该寄存器控制同一组的两路 PWM1xy（xy = 01、23、45）的死区时间；死区时间设定值范围 0～0xFF，默认值为 0xFF，保证默认情况下开启死区保护，硬件不发生故障。

注 2：PWM1xy（xy = 01、23、45）的死区时间，在互补模式下，同时作用于 xy 两个通道，3 组之间的设定值可以不同；在独立模式下，死区时间设置寄存器无效。

PWM1 通道 2 占空比设定低位寄存器（0xEA，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM12DL	PWM12D 7	PWM12D 6	PWM12D 5	PWM12D 4	PWM12D 3	PWM12D 2	PWM12D 1	PWM12D 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM12DL** – PWM1 通道 2 占空比设定寄存器低 8 位

PWM1 通道 2 占空比设定高位寄存器（0xE9，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM12DH	PWM12D 15	PWM12D 14	PWM12D 13	PWM12D 12	PWM12D 11	PWM12D 12	PWM12D 9	PWM12D 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM12DH** – PWM1 通道 2 占空比设定寄存器高 8 位

注：当修改 PWM1x 的占空比时，应先修改高位部分，再修改地位部分；读取时不受此限制。

PWM1 通道 3 占空比设定低位寄存器（0xF7，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM13DL	PWM13D 7	PWM13D 6	PWM13D 5	PWM13D 4	PWM13D 3	PWM13D 2	PWM13D 1	PWM13D 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM13DL** – PWM1 通道 3 占空比设定寄存器低 8 位

PWM1 通道 3 占空比设定高位寄存器（0xF6，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM13DH	PWM13D15	PWM13D14	PWM13D13	PWM13D12	PWM13D13	PWM13D13	PWM13D9	PWM13D8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM13DH** – PWM1 通道 3 占空比设定寄存器高 8 位

注 1：当修改 PWM1x 的占空比时，应先修改高位部分，再修改地位部分；读取时不受此限制。

注 2：对于 PWM1 的奇数通道 PWM1x（x = 1、3、5），在互补模式下，其占空比设置寄存器无效。

PWM1 的 23 组死区时间设定寄存器（0xF5，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM123F	PWM123F7	PWM123F6	PWM123F5	PWM123F4	PWM123F3	PWM123F2	PWM123F1	PWM123F0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7:0] **PWM123F** – PWM1 的 23 组死区时间设定寄存器

注 1：该寄存器控制同一组的两路 PWM1xy（xy = 01、23、45）的死区时间；死区时间设定值范围 0～0xFF，默认值为 0xFF，保证默认情况下开启死区保护，硬件不发生故障。

注 2：PWM1xy（xy = 01、23、45）的死区时间，在互补模式下，同时作用于 xy 两个通道，3 组之间的设定值可以不同；在独立模式下，死区时间设置寄存器无效。

PWM1 通道 4 占空比设定低位寄存器（0xF4，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM14DL	PWM14D7	PWM14D6	PWM14D5	PWM14D4	PWM14D3	PWM14D2	PWM14D1	PWM14D0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM14DL** – PWM1 通道 4 占空比设定寄存器低 8 位

PWM1 通道 4 占空比设定高位寄存器（0xF3，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM14DH	PWM14D15	PWM14D14	PWM14D13	PWM14D12	PWM14D11	PWM14D14	PWM14D9	PWM14D8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM14DH** – PWM1 通道 4 占空比设定寄存器高 8 位

注：当修改 PWM1x 的占空比时，应先修改高位部分，再修改地位部分；读取时不受此限制。

PWM1 通道 5 占空比设定低位寄存器（0xF2，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM15DL	PWM15D7	PWM15D6	PWM15D5	PWM15D4	PWM15D3	PWM15D2	PWM15D1	PWM15D0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM15DL** – PWM1 通道 5 占空比设定寄存器低 8 位

PWM1 通道 5 占空比设定高位寄存器（0xF1，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM15DH	PWM15D15	PWM15D14	PWM15D13	PWM15D12	PWM15D15	PWM15D15	PWM15D9	PWM15D8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **PWM15DH** – PWM1 通道 5 占空比设定寄存器高 8 位

注 1：当修改 PWM1x 的占空比时，应先修改高位部分，再修改地位部分；读取时不受此限制。

注 2：对于 PWM1 的奇数通道 PWM1x（x = 1、3、5），在互补模式下，其占空比设置寄存器无效。

PWM1 的 45 组死区时间设定寄存器（0xFF，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM145F	PWM145F7	PWM145F6	PWM145F5	PWM145F4	PWM145F3	PWM145F2	PWM145F1	PWM145F0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7:0] **PWM145F** – PWM1 的 45 组死区时间设定寄存器

注 1：该寄存器控制同一组的两路 PWM1xy（xy = 01、23、45）的死区时间；死区时间设定值范围 0～0xFF，默认值为 0xFF，保证默认情况下开启死区保护，硬件不发生故障。

注 2：PWM1xy（xy = 01、23、45）的死区时间，在互补模式下，同时作用于 xy 两个通道，3 组之间的设定值可以不同；在独立模式下，死区时间设置寄存器无效。

11 模数转换器ADC

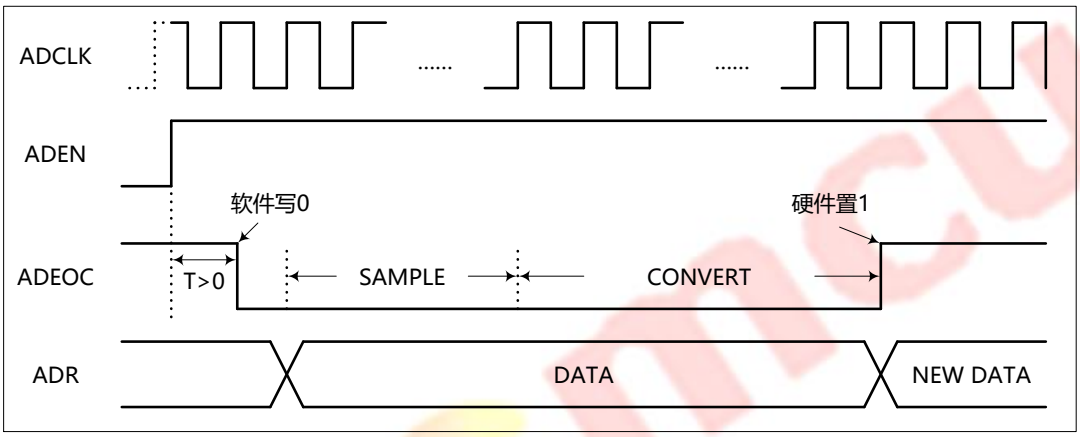
11.1 ADC 概述

ADC 模块为 12 位高精度逐次逼近型 ADC，包含外部 12 通道和内部 2 通道（VDD/4、GND）。ADC 时钟源为系统时钟 Fsys，可选择 1/2/4/8/16/32/64/128 分频；ADC 参考电压可配置：VDD、内部 2.0V、外部 Vref 输入电压。

ADC 模块通过 ADEN 使能，通过 ADCHS 选择转换的模拟通道，ADCKS 选择转换速度，ADEOC 为 ADC 启动位及转换结束标志位。当 ADEOC 标志为 1 时，对该位写 0 将启动模数转换，转换时间需要 13~27 个 AD 转换时钟周期，转换结果放在 ADRH/ADRL 中，ADEOC 将自动置 1，同时中断标志 ADCIF 置 1，产生中断请求。

11.2 ADC 转换时序

ADC 转换时序如下图所示：



注：SAMPLE 时间可选 1~15 个 ADCLK，CONVERT 时间为 12 个 ADCLK，一次 ADC 转换为 13~27 个 ADCLK。

11.3 ADC 常规操作步骤

AD 转换常规操作步骤：

(1) 设置相应端口为输入端口，关闭上/下拉电阻；
(2) 通过端口数模控制寄存器，关闭相应端口的数字 I/O 功能；
(3) 若转换时钟可选，则设置 ADCKS，选择适当的 ADC 转换时钟；
(4) 若采样时间可选，则设置 ADSPS，选择适当的 ADC 采样时间；
(5) 若参考电压可选，则设置 ADVRS，选择适当的参考电压；
(6) 若数据格式可选，则设置 ADRSEL，选择 ADC 转换结果的数据格式；
(7) ADEN 置 1，使能 ADC 模块；
(8) 设置 ADCHS，选择 ADC 转换通道；
(9) 延时等待电路稳定后，ADEOC 写 0，启动 AD 转换；
(10) 等待 ADEOC 硬件置 1（或利用 ADC 中断）；
(11) 读取 ADC 转换结果（ADRH、ADRL）；
(12) 重复执行（8）~（11），对不同的通道进行转换或对同一通道进行多次转换。

注 1：AD 转换过程中或者 ADEN 未使能时，ADRH/ADRL 中的数据未知，应在 ADEN 置 1 使能且 AD 转换结束 ADEOC 硬件置 1 的情况下读取 AD 转换数据。

注 2：若 ADC 模块已经使能（ADEN 置 1），参考电压由 ADC 切换至内部参考电压 2.0V（设置 VRS[1:0]=0x00），需要等待一定时间（> 100us），参考电压稳定后，再能启动 ADC 转换（ADEOC = 0）。

注 3：若 ADC 模块关闭（ADEN 清 0），则 ADC 端口配置寄存器 ADRIOS0/ADRIOS1 中的使能位均自动失效，相应端口自动恢复为通用 IO 或其他复用端口，此时可通过端口模式寄存器将相应端口设为高阻态模式，以消除端口上的模拟信号引发的漏电流。

11.4 ADC 相关寄存器

ADC 寄存器汇总表

地址 (SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0/FF	**OSADJCR**	零点偏移修正寄存器	OSADJEN	-	-	OSADJT4	OSADJT3	OSADJT2	OSADJT1	OSADJT0	0--xxxxx
SFR0/FE	**ADRH**	(ADRSEL = 0)ADC 转换结果高位寄存器	ADR11	ADR10	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4	xxxxxxxx
		(ADRSEL = 1)ADC 转换结果高位寄存器	-	-	-	-	ADR11	ADR10	ADR9	ADR8	
SFR0/FD	**ADRL**	(ADRSEL = 0)ADC 转换结果低位寄存器	-	-	-	-	ADR3	ADR2	ADR1	ADR0	xxxxxxxx
		(ADRSEL = 1)ADC 转换结果低位寄存器	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	
SFR0/FC	**ADRIOS1**	ADC 端口配置寄存器 1	-	-	EX_VREF_EN	-	-	-	AN9EN	AN8EN	--0-----00
SFR0/FB	**ADRIOS0**	ADC 端口配置寄存器 0	AN7EN	AN6EN	AN5EN	AN4EN	AN3EN	AN2EN	AN1EN	AN0EN	00000000
SFR0/FA	**ADCCR1**	ADC 控制寄存器 1	ADRSEL	ADCKS2	ADCKS1	ADCKS0	ADSPS3	ADSPS2	ADSPS1	ADSPS0	00001111
SFR0/F9	**ADCCR0**	ADC 控制寄存器 0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	VRS1	VRS0	ADEOC	ADEN	11111110
SFR0/88	**TCON**	定时器控制寄存器	TF1	TR1	TF0	TR0	-	-	-	ADCIF	0000---1

ADC 控制寄存器 0（0xF9，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCCR0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	VRS1	VRS0	ADEOC	ADEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	0

BIT[7:4] **ADCHS[3:0]** – ADC 模拟通道选择

ADCHS[3:0]	ADC 模拟通道选择
0000	AIN0
0001	AIN1
0010	AIN2
0011	AIN3
0100	AIN4
0101	AIN5
0110	AIN6
0111	AIN7
1000	AIN8

1001	AIN9
1010	AIN10
1011	AIN11
1100	禁止
1101	禁止
1110	AN14（内部接 VDD/4）
1111	AN15（内部接地）

BIT[3:2] **VRS[1:0]** – ADC 参考电压选择

VRS[1:0]	ADC 参考电压
00	VDD
01	禁止
10	禁止
11	内部 2.0V

BIT[1] **ADEOC** – ADC 启动位及转换结束标志位

0：AD 启动或者 AD 正在转换中，转换结束后硬件自动置 1；

1：AD 转换已结束，同时作为 AD 转换完成的查询标志。

注：该位写入 0 时，启动一次 ADC 转换；当未转换完成时，再写入 0，则上次转换中止，开始一次新的转换；1 可以被写入，但其对 AD 模块无效，即不会影响 AD 转换结果。

BIT[0] **ADEN** – ADC 功能使能位

0：关闭 ADC 模块；

1：使能 ADC 模块。

ADC 控制寄存器 1（0xFA，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCCR1	ADRSEL	ADCKS2	ADCKS1	ADCKS0	ADSPS3	ADSPS2	ADSPS1	ADSPS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	1	1	1	1

BIT[7] **ADRSEL** – ADC 转换数据格式选择

0：ADC 转换结果 12 位数据，高 8 位存入 ADRH[7:0]、低 4 位存入 ADRL[3:0]；

1：ADC 转换结果 12 位数据，高 4 位存入 ADRH[3:0]、低 8 位存入 ADRL[7:0]。

BIT[6:4] **ADCKS[2:0]** – AD 转换时钟选择

ADCKS[2:0]	AD 转换时钟 F_{ADC}
000	Fsys
001	Fsys/2
010	Fsys/4
011	Fsys/8
100	Fsys/16
101	Fsys/32
110	Fsys/64
111	Fsys/128

BIT[3:0] **ADSPS[3:0]** – ADC 采样时间选择

ADSPS[3:0]	SAMPLE 时间
0000	禁止
0001	1 个 ADCLK

ADSPS[3:0]	SAMPLE 时间
0010~1110	2~14 个 ADCLK
1111	15 个 ADCLK

ADC 端口配置寄存器 0（0xFB，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRIOS0	AN7EN	AN6EN	AN5EN	AN4EN	AN3EN	AN2EN	AN1EN	AN0EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **ANnEN** – ADC 输入 AINn 使能位（n = 7～0）

0：禁止 ADC 输入功能，端口用于通用 IO 或其他复用模块；

1：端口用作 ADC 输入。

ADC 端口配置寄存器 1（0xFC，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRIOS1	-	-	EX_VREF_EN	-	AN11EN	AN10EN	AN9EN	AN8EN
R/W	-	-	R/W	-	R/W	R/W	R/W	R/W
复位值	-	-	0	-	0	0	0	0

BIT[5] **EX_VREF_EN** – ADC 外部 Vref 输入使能位

0：禁止 ADC 外部 Vref 输入功能，端口用于通用 IO 或其他复用模块；

1：端口用作 ADC 外部 Vref 输入。

注：当该位置 1 时，Vref 引脚输入有效，且 ADC 模块强制将 Vref 电压作为 AD 的参考电压使用，这时无视 ADC 参考电压选择位（VRS[1:0]）的值。

BIT[1:0] **ANnEN** – ADC 输入 AINn 使能位（n = 11~8）

0：禁止 ADC 输入功能，端口用于通用 IO 或其他复用模块；

1：端口用作 ADC 输入。

定时器控制寄存器（0x88，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON	TF1	TR1	TF0	TR0	-	-	-	ADCIF
R/W	R/W	R/W	R/W	R/W	-	-	-	R/W
复位值	0	0	0	0	-	-	-	1

BIT[0] **ADCIF** – ADC 中断标志位

0：未发生 ADC 中断，由软件清 0；

1：发生 ADC 中断，由硬件置 1。

注：此位默认值 1，首次使用请先清零此位。

当 ADRSEL = 0 时:

ADC 转换结果高位寄存器（0xFE，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRH	ADR11	ADR10	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[7:0] **ADR[11:4]** – ADC 转换结果高 8 位

ADC 转换结果低位寄存器（0xFD，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRL	-	-	-	-	ADR3	ADR2	ADR1	ADR0
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[3:0] **ADR[3:0]** – ADC 转换结果低 4 位

当 ADRSEL = 1 时:

ADC 转换结果高位寄存器（0xFE，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRH	-	-	-	-	ADR11	ADR10	ADR9	ADR8
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[3:0] **ADR[11:8]** – ADC 转换结果高 4 位

ADC 转换结果低位寄存器（0xFD，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRL	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0
R/W	R	R	R	R	R	R	R	R
复位值	X	X	X	X	X	X	X	X

BIT[7:0] **ADR[7:0]** – ADC 转换结果低 8 位

注：芯片复位后，ADC 结果寄存器中的值为随机值。

零点偏移修正寄存器（0xFF，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSADJCR	OSADJEN	OSADJT6	OSADJT5	OSADJT4	OSADJT3	OSADJT2	OSADJT1	OSADJT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	X	X	X	X	X	X	X

BIT[7] **OSADJEN** – ADC 零点偏移修调模式使能位，复位值为出厂配置

0：禁止 ADC 零点偏移修调模式（默认）；

1：使能 ADC 零点偏移修调模式，此模式下 OSADJT[6:0]的值将对 AD 的结果进行自动纠偏。

***注：**建议在使能 AD 零点偏移修调模式前，“用户”通过软件方式进行一次修调，以保证 AD 转换得到的结果最佳，具体步骤可以参考下面的说明。*

BIT[6] **OSADJT6** – ADC 零点偏移修调方向选择位

0：负向修调，即根据修调电压减小转换值（转换结果大于理论值时应选择负向修调）；

1：正向修调，即根据修调电压增加转换值（转换结果小于理论值时应选择正向修调）。

BIT[3:0] **OSADJT[5:0]** – ADC 零点偏移修调电压选择位

OSADJT[5:0]	修调电压（典型值）
00 0000	0
00 0001	1 × VREF/4096
00 0010	2 × VREF/4096
...	...
00 1111	15 × VREF/4096

01 0000	16 × VREF/4096
…	…
11 1110	62 × VREF/4096
11 1111	63 × VREF/4096

***注：** 在 ADC 的修调结果中 OSADJT[5:0]=00 0000B,OSADJT6=0 或 1，两者效果是一样的，它们的偏移电压都是 0mV;*

***注：** 这里的 V_{REF} 可以是 VDD、内部 2.0V、外部 Vref 输入电压三选一，与用户的实际选用的 AD 参考电压有关。*

11.5 ADC 零点偏移修调步骤

AD 模块零点偏移修调步骤：

(1) 将 ADC 的输入通道选择为内部接地（**AN15**），设置 **OSADJEN=1** 打开修调功能；设置一个基本的 ADC 时钟(建议 ADC 工作频率在 **1MHz** 为宜)、采样时间（建议 ADC 采样时间在 **8 个 Clk** 为宜）等；

(2) 设置 **OSADJT6 =0，OSADJT[5:0]**=**00 0000B**，进行一次 ADC 转换；

若 ADC 转换结果为 0（实际值≤0），则跳至步骤（4），进行正向修调；

若 ADC 转换结果不为 0，则执行步骤（3）；

(3) 将 **OSADJT[5:0]**的值，加 1 后，再次进行 ADC 转换；

若 ADC 转换结果为 0，则跳至步骤（6）；

若 ADC 转换结果不为 0,则循环执行步骤(3),直到 ADC 转换结果为 0 或 **OSADJT[5:0]**递增到最大值 **11 1111B**，跳至步骤（6）；

(4) 设置 **OSADJT6 =1，**OSADJT[5:0]=**11 1111B**，进行一次 ADC 转换；

若 ADC 转换结果为 0，则跳至步骤（6）；

若 ADC 转换结果不为 0，则执行步骤（5）；

(5) 将 **OSADJT[5:0]**的值，减 1 后，再次进行 ADC 转换；

若 ADC 转换结果为 0，则跳至步骤（6）；

若 ADC 转换结果不为 0，则循环执行步骤（5），直到 ADC 转换结果为 0 或 **OSADJT[5:0]**递减到 **00 0000B**，跳至步骤（6）；

(6) **OSADJT6 和 OSADJT[5:0]**的值，即为零点偏移最佳修调结果，修调流程结束，后续 ADC 工作时可直接使用该值，不需要再次进行修调（除非该值丢失，如断电等情况）。

注：修调过程工作条件变化以及 ADC 有效位数，可能影响修调的最佳效果，用户可以通过增加单次步骤中的 ADC 采样次数或多次修调择优方式处理。

12 增强型异步通讯UART

12.1 概述

片上包含 2 路兼容 8051 标准的增强型通用异步收发器 UART，波特率可选择为系统时钟分频或者定时器 T1/T2 的时钟溢出。

UART0 的增强功能包括帧出错检测以及自动地址识别，UART0 支持 8 位同步半双工（方式 0）、8 位异步全双工（方式 1）、9 位异步固定波特率（方式 2）、9 位异步可变波特率（方式 3）4 种工作方式。

UART1 没有错误检测和自动硬件地址识别，且 UART1 仅支持 8 位异步全双工（方式 1）、9 位异步可变波特率（方式 3）2 种工作方式，其寄存器功能和 UART0 相同。

12.2 工作方式

UART 有 4 种工作方式。在通讯之前必须先初始化串口控制寄存器 SCON，选择 UART 的工作方式和波特率。如果使用方式 1 或方式 3 还应先初始化定时器 T1/T2。

在所有四种方式中，任何将串口缓冲寄存器 SBUF 作为目标寄存器的写操作都会启动发送。在方式 0 中由条件 RI=0 和 REN=1 初始化接收，将在 TX 引脚上产生 1 个时钟信号，然后在 RX 引脚上串行移入/移出 8 位数据。在其他方式中则利用外部输入的起始位来初始化接收（如果 REN=1），外部发送器通过发送起始位开始通信。

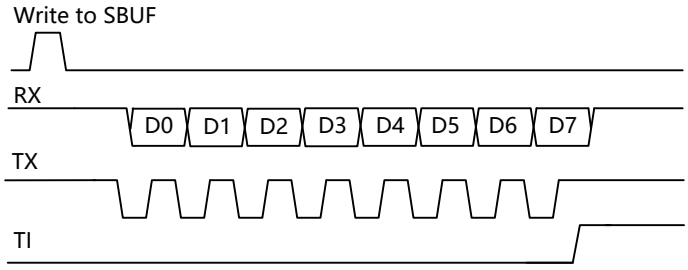
UART 方式列表

SM[0:1]	方式	类型	波特率	帧长度	起始位	停止位	第 9 位
00	0	同步	Fsys/4 或 Fsys/12	8 位	无	无	无
01	1	异步	T1/T2 溢出率/16	10 位	1	1	无
10	2	异步	Fsys/32 或 Fsys/64	11 位	1	1	0/1
11	3	异步	T1/T2 溢出率/16	11 位	1	1	0/1

12.2.1 方式 0：8 位同步半双工

方式 0 支持与外部设备的同步通信。UART 通过 TX 引脚发送移位时钟，在 RX 引脚上收发串行数据。因此这个方式是串行通信的半双工方式，在此方式中，每帧收发 8 位，低位先接收/发送。

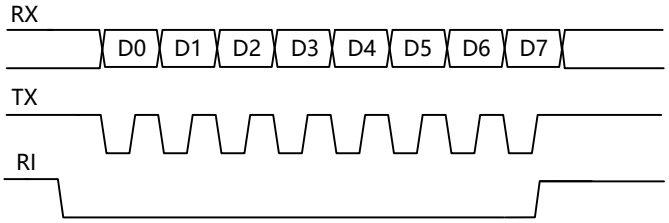
通过设置寄存器 SCON 中的 SM2 位为 0 或 1，波特率固定为系统时钟的 1/12 或 1/4。当 SM2 位为 0 时，串行端口以系统时钟的 1/12 运行。当 SM2 位置 1 时，串行端口以系统时钟的 1/4 运行。与标准 8051 唯一不同的是，芯片在方式 0 中有可变波特率。



Send Timing of Mode 0

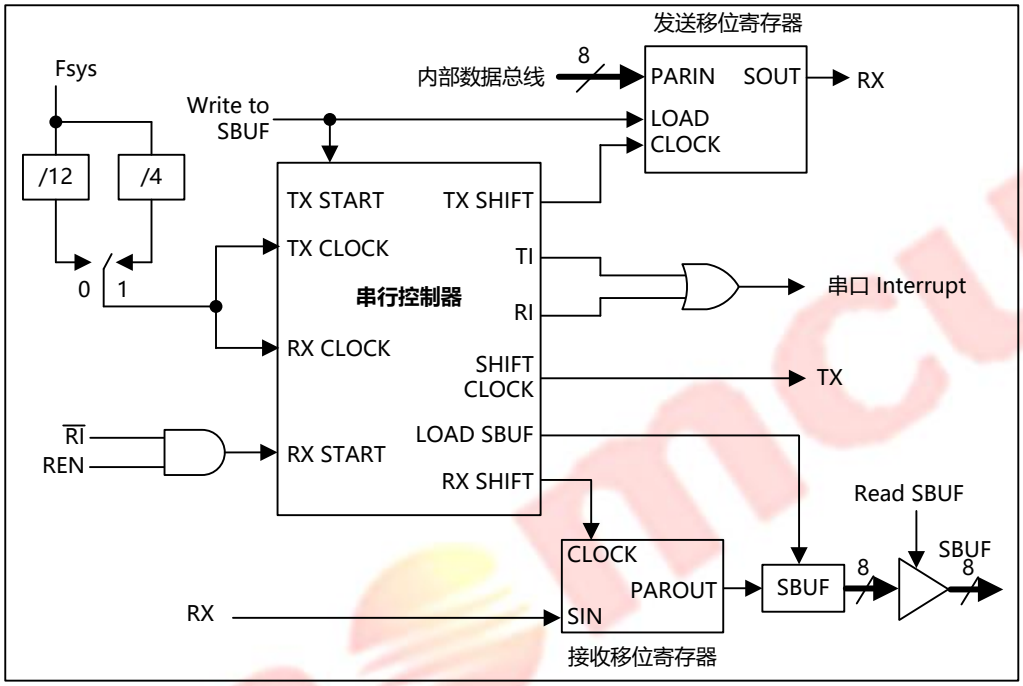
UART 模块通过 TX 引脚输出同步时钟，通过 RX 引脚将数据读入或移出串行端口。任何将 SBUF 作为目标寄存器的写操作都会启动发送。下一个系统时钟 TX 控制块开始发送。数据转换发生在移位时钟的上升沿，移位寄存器的内容逐次从左往右移位，空位置 0。当移位寄存器中的所有 8 位都发送后，TX 控制模块停止发送操作，然后在下一个系统时钟的上升沿将 TI 置 1。

REN 位置 1 和 RI 位清 0 将初始化接收。下一个系统时钟启动接收，在移位时钟的上升沿锁存数据，接收转换寄存器的内容逐次向左移位。当所有 8 位都接收到接收移位寄存器中后，RX 控制模块停止接收，然后在下一个系统时钟的上升沿上 RI 置 1，直到被软件清 0 才允许接收。



Receive Timing of Mode 0

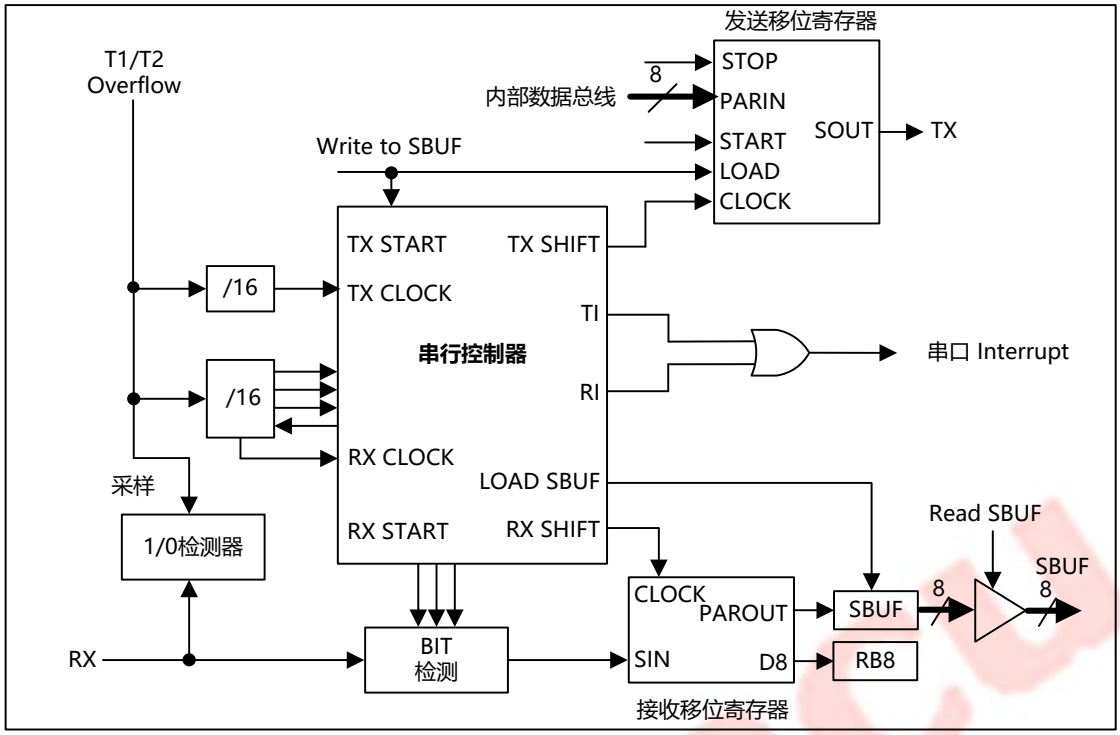
方式 0 功能块框图如下图所示：



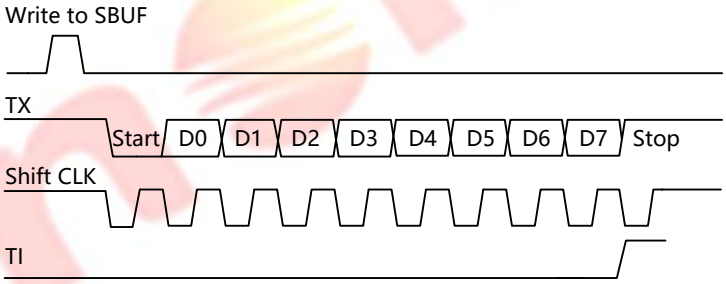
12.2.2 方式 1：8 位异步全双工，可变波特率

方式 1 提供 10 位全双工异步通信，10 位由一个起始位（逻辑 0）、8 个数据位（低位在前）和一个停止位（逻辑 1）组成。在接收时，这 8 个数据位存储在 SBUF 中而停止位储存在 RB8 位中。方式 1 中的波特率是可变的，串行收发波特率为定时器 T1/T2 溢出率的 1/16。

方式 1 功能块框图如下图所示：



任何将 SBUF 作为目标寄存器的写操作都会启动发送，实际上发送是从 16 分频计数器中的下一次跳变之后的系统时钟开始的，因此位时间与 16 分频计数器是同步的，与对 SBUF 的写操作不同步。起始位首先在 TX 引脚上移出，然后是 8 位数据位。在发送移位寄存器中的所有 8 位数据都发送完后，停止位在 TX 引脚上移出，在停止位发出的同时 TI 标志置 1。



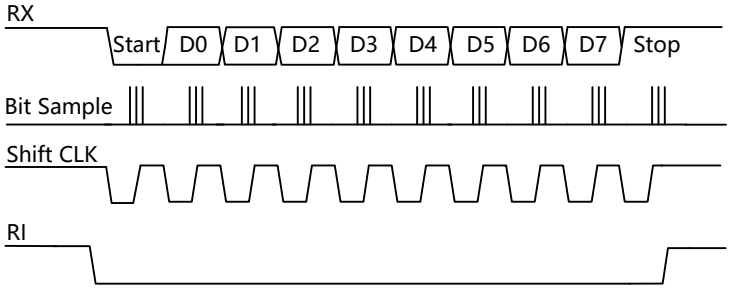
Send Timing of Mode 1

只有 REN 位置 1 时才允许接收。当 RX 引脚检测到下降沿时串行口开始接收串行数据。为此，芯片对 RX 不断采样，采样速率为波特率的 16 倍。当检测下降沿时，16 分频计数器立即复位，这有助于 16 分频计数器与 RX 引脚上的串行数据位同步。16 分频计数器把每一位的时间分为 16 个状态，在第 7、8、9 状态时，位检测器对 RX 端的电平进行采样。为抑制噪声，在这 3 个状态采样中至少有 2 次采样值一致数据才被接收。如果所接收的第一位不是 0，说明这位不是一帧数据的起始位，该位被忽略，接收电路被复位，等待 RX 引脚上另一个下降沿的到来。若起始位有效，则移入移位寄存器，并接着移入其它位到移位寄存器。8 个数据位和 1 个停止位移入之后，移位寄存器的内容被分别装入 SBUF 和 RB8 中，RI 置 1，但必须满足下列条件：

（1）RI=0；

（2）SM2=0，或者接收的停止位=1；

如果这些条件被满足，那么停止位装入 RB8，8 个数据位装入 SBUF，RI 被置 1。否则接收的帧会丢失。这时，接收器将重新去探测 RX 端是否另一个下降沿。用户必须用软件清除 RI，然后才能再次接收。

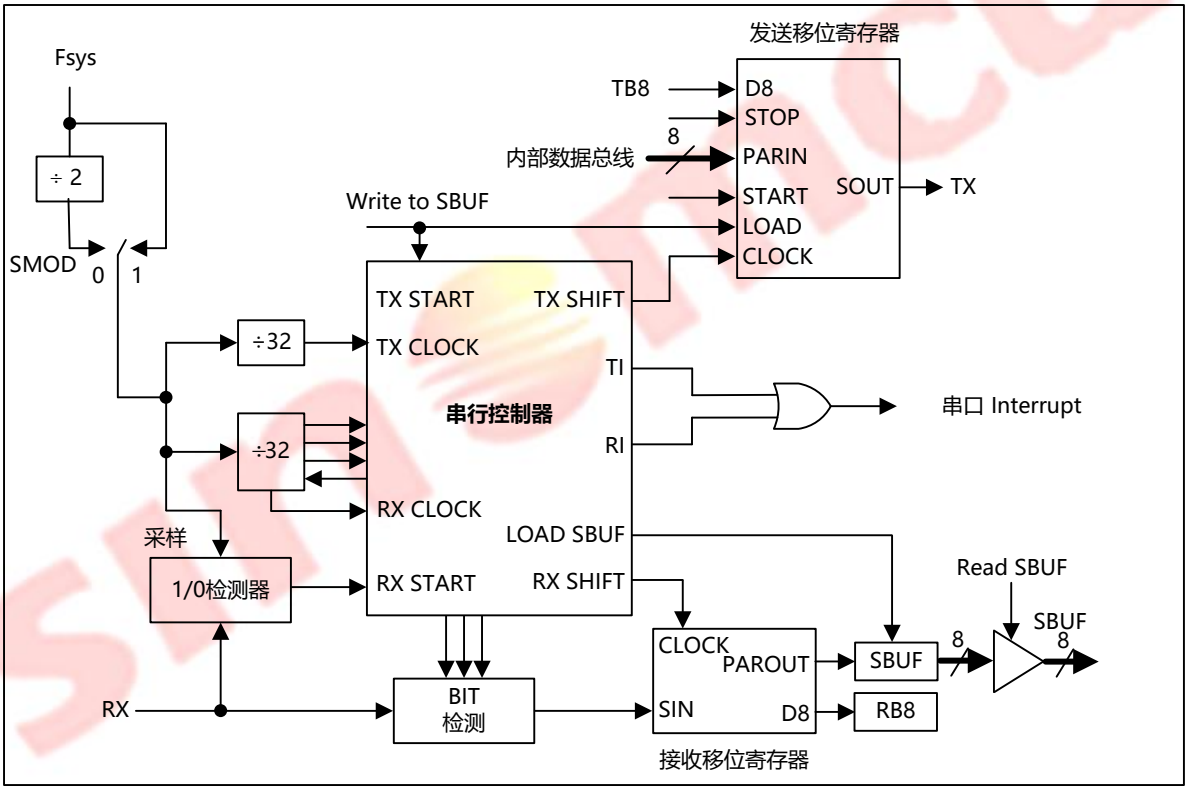


Receive Timing of Mode 1

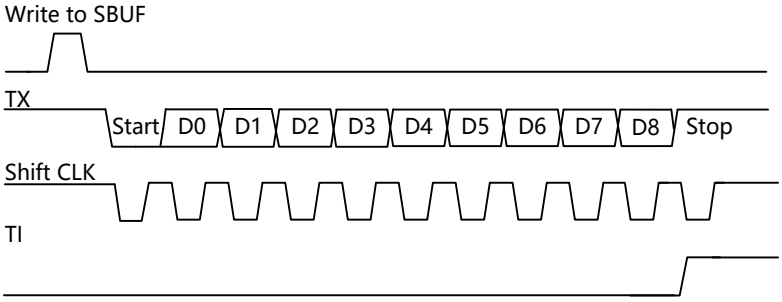
12.2.3 方式 2：9 位异步全双工，固定波特率

这个方式使用异步全双工通信中的 11 位。一帧由一个起始位（逻辑 0），8 个数据位（低位在前），一个可编程的第 9 数据位和一个停止位（逻辑 1）组成。方式 2 支持多机通信和硬件地址识别（详见多机通讯章节）。在数据传送时，第 9 数据位（SCON 中的 TB8）可以写 0 或 1，例如，可写入 PSW 中的奇偶位 P，或用作多机通信中的数据/地址标志位。当接收到数据时，第 9 数据位进入 RB8 而停止位不保存。

方式 2 功能块框图如下所示：



任何将 SBUF 作为目标寄存器的写操作都会启动发送，同时也将 TB8 载入到发送移位寄存器的第 9 位中。实际上发送是从 16 分频计数器中的下一次跳变之后的系统时钟开始的，因此位时间与 16 分频计数器是同步的，与对 SBUF 的写操作不同步。起始位首先在 TX 引脚上移出，然后是第 9 位数据。在发送转换寄存器中的所有 9 位数据都发送完后，停止位在 TX 引脚上移出，在停止位开始发送时 TI 标志置 1。



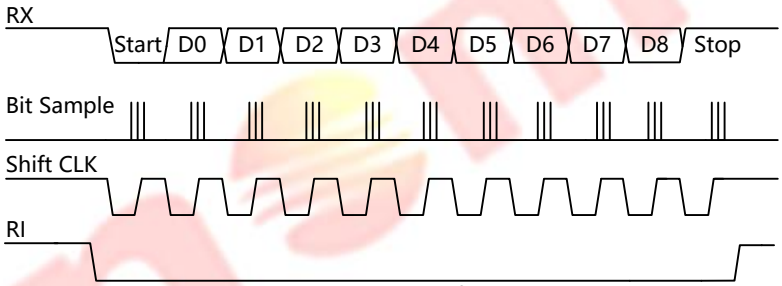
Send Timing of Mode 2

只有 REN 位置 1 时才允许接收。当 RX 引脚检测到下降沿时串行口开始接收串行数据。为此，芯片对 RX 不断采样，采样速率为波特率的 16 倍。当检测下降沿时，16 分频计数器立即复位。这有助于 16 分频计数器与 RX 引脚上的串行数据位同步。16 分频计数器把每一位的时间分为 16 个状态，在第 7、8、9 状态时，位检测器对 RX 端的电平进行采样。为抑制噪声，在这 3 个状态采样中至少有 2 次采样值一致数据才被接收。如果所接收的第一位不是 0，说明这位不是一帧数据的起始位，该位被忽略，接收电路被复位，等待 RX 引脚上另一个下降沿的到来。若起始位有效，则移入移位寄存器，并接着移入其它位到移位寄存器。9 个数据位和 1 个停止位移入之后，移位寄存器的内容被分别装入 SBUF 和 RB8 中，RI 置 1，但必须满足下列条件：

（1）RI=0；

（2）SM2=0 或者接收的第 9 位=1，且接收的字节符合实际从机地址；

如果这些条件被满足，那么第 9 位移入 RB8，8 位数据移入 SBUF，RI 被置 1。否则接收的数据帧会丢失。在停止位的当中，接收器回到寻找 RXD 引脚上的另一个下降沿。用户必须用软件清除 RI，然后才能再次接收。

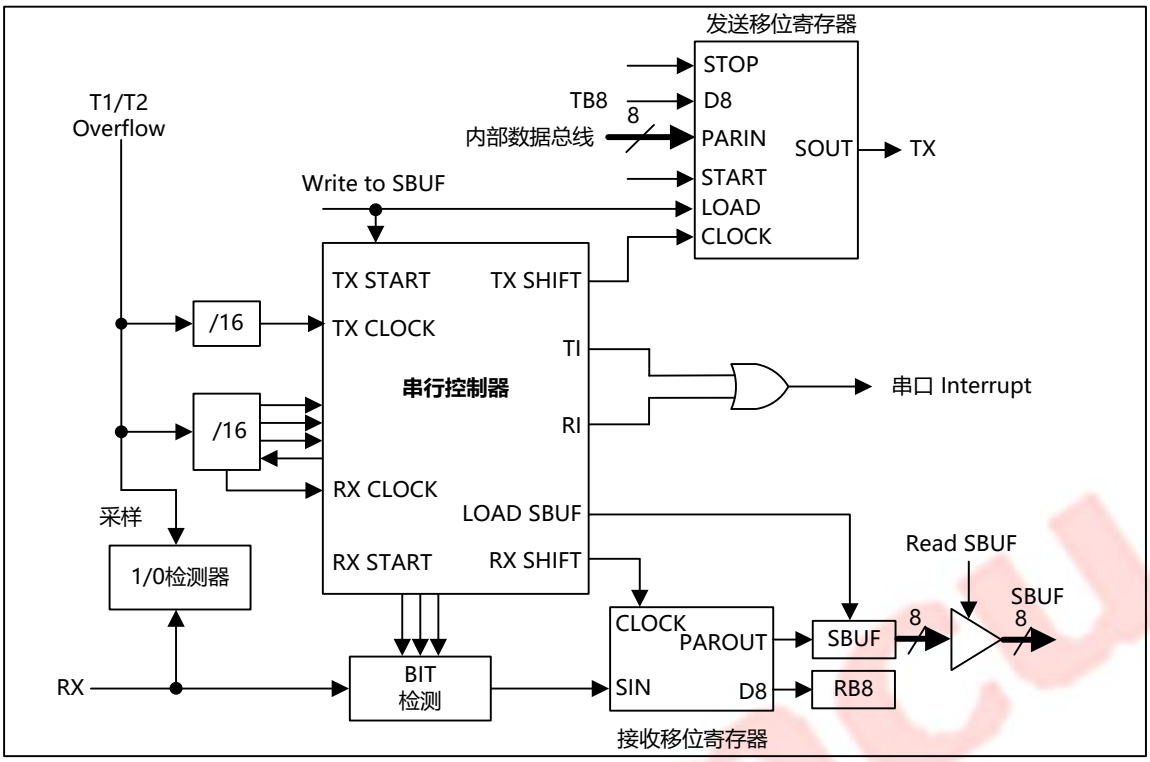


Receive Timing of Mode 2

12.2.4 方式 3：9 位异步全双工，可变波特率

方式 3 使用方式 2 的传输协议以及方式 1 的波特率产生方式。

方式 3 功能块框图如下所示：



12.3 波特率计算

12.3.1 UART 方式 0

波特率可编程为系统时钟的 1/12 或 1/4，由 SM2 位决定。当 SM2 为 0 时，串行端口在系统时钟的 1/12 下运行；当 SM2 为 1 时，串行端口在系统时钟的 1/4 下运行。

$$\text{UART0波特率} = \frac{\text{Fsys}}{4}$$

或者

$$\text{UART0波特率} = \frac{\text{Fsys}}{12}$$

***注：** UART1 没有方式 0，故 SM2_1 位在方式 0 中无效。*

12.3.2 UART 方式 1 和方式 3

工作方式 1 下，T1 作为波特率发生器，在 UART0/UART1 方式 1 和方式 3 中的波特率计算公式为：

$$\text{T1产生的波特率} = \frac{1}{16} \times \frac{\text{Ft1}}{[256 - (\text{TH1})]}$$

其中 Ft1 为 T1 实际的工作频率，TH1 为实际设定值。

注：当用定时器 T1 作为波特率发生器时，T1 的 TH1 值不可设置为 0xFF，并应该同时关闭相应 T1 的中断使能位。

工作方式 3 下，T2 作为波特率发生器，在 UART0/UART1 方式 1 和方式 3 中波特率计算公式：

$$\text{T2产生的波特率} = \frac{1}{16} \times \frac{\text{Fsys}}{\{\text{Pre_scale} \times [65536 - (\text{RCAPH2}) \times 256 + (\text{RCAPL2})]\}}$$

Fsys 为系统时钟频率，预分频值 Pre_scale 为 T2 时钟选择位 T2CKS[2:0]的分频系数，RCAPH2/RCAPL2 为实际设定值。

***注：** 当用定时器 T2 作为波特率发生器时，T2 的 RCAP2 值不可设置为 0xFFFF，并应该同时关闭相应 T2 的中断使能位。*

12.3.3 UART 方式 2

UART0 的波特率为：

$$\text{UART0波特率} = \frac{\text{Fsys}}{64} \times 2^{\text{SMOD0}}$$

其中 SMOD0 位根据用户设定值可以取 0 或者 1，起到倍频的作用。

***注：** UART1 没有方式 2，故无对应的 SMOD0_1 位。*

12.4 多机通讯

12.4.1 软件地址识别

方式 2 和方式 3 有一个专门的适用于多机通讯的功能。在这两个方式下，接收的是 9 位数据，第 9 位移入 RB8 中，然后再来一位停止位。UART 可以这样设定：当接收到停止位时，只有在 RB8=1 的条件下，串行口中断才会有效（请求标志 RI 置 1）。可通过将 SCON 寄存器的 SM2 位置 1 使 UART 具有这个功能。

在多机通讯系统中，以如下所述来利用这一功能。当主机要发送一数据块给几个从机中的一个时，先送出一地址字节，以辨认目标从机。地址字节与数据字节可用第 9 数据位来区别，地址字节的第 9 位为 1，数据字节的第 9 位为 0。

如果从机 SM2 为 1，则不会响应数据字节中断。地址字节可以中断所有从机，这样，每一个从机都检查所接收到的地址字节，以判别自己是不是目标从机。被寻到的从机将 SM2 位清 0，并准备接收即将到来的数据字节，当接收完毕时，从机再一次将 SM2 置 1。没有被寻址的从机，则维持其 SM2 位为 1，忽略到来的数据字节，继续做自己的事情。

注：在方式 0 中，SM2 用来选择波特率加倍。在方式 1 中，SM2 用来检测停止位是否有效，如果 SM2 = 1，接收中断不会响应直到接收到一个有效的停止位。

12.4.2 自动（硬件）地址识别

在方式 2 和方式 3 中，SM2 置 1 将使 UART 在如下状态运行：当 1 个停止位被接收时，如果载入 RB8 的第 9 数据位为 1（地址字节）并且接收到的数据字节符合 UART 的从机地址，UART 产生一个中断。接着，从机应将 SM2 清零，以接收后续的数据字节。

在 9 位方式下要求第 9 位为 1 以表明该字节是地址而非数据。当主机要发送一组数据给几个从机中的一个时，必须先发送目标从机的地址。所有从机在等待接收地址字节时，为了确保仅在接收地址字节时产生中断，SM2 位必须置 1。自动地址识别的特点是只有地址匹配的从机才能产生中断，地址比较通过硬件完成而不是软件。

中断产生后，地址相匹配的从机清零 SM2，继续接收数据字节。地址不匹配的从机不受影响，将继续等待接收和它匹配的地址字节。一旦全部信息接收完毕，地址匹配的从机应该再次把 SM2 置 1，忽略所有传送的非地址字节，直到接收到下一个地址字节。

使用自动地址识别功能时，主机可以通过调用给定的从机地址选择与一个或多个从机通信。使用广播地址可以联系所有的从机。有两个特殊功能寄存器用来定义从机地址（SADDR）和地址屏蔽（SADEN）。从机地址是一个 8 位的字节，存于 SADDR 寄存器中。SADEN 用于定义 SADDR 内位的有效与否，如果 SADEN 中某一位为 0，则 SADDR 中相应位的被忽略，如果 SADEN 中某一位置 1，则 SADDR 中相应位的将用于得到给定的从机地址。这可以使用户在不改变 SADDR 寄存器中的从机地址的情况下灵活地寻址多个从机。使用给定地址可以识别多个从机而排除其他的从机。

	从机 1	从机 2
SADDR	10100100	10100111
SADEN（为 0 的位被忽略）	11111010	11111001
实际从机地址	10100x0x	10100xx1
广播地址（SADDR 或 SADEN）	1111111x	11111111

从机 1 和从机 2 给定地址的最低位是不同的。从机 1 忽略了最低位，而从机 2 的最低位是 1。因此只与从机 1 通讯时，主机必须发送最低位为 0 的地址（10100000）。类似地，从机 1 的第 1 位为 0，从机 2 的第 1 位被忽略。因此，只与从机 2 通讯时，主机必须发送第 1 位为 1 的地址（10100011）。如果主机希望同时与两从机通讯，则第 0 位为 1，第 1 位为 0，第 2 位被两从机都忽略，此时有两个不同的地址用于选定两个从机（1010 0001 和 1010 0101）。

主机可以通过广播地址与所有从机同时通讯。这个地址等于 SADDR 和 SADEN 的逻辑或，结果中的 0 表示该位被忽略。多数情况下，广播地址为 0xFFh，该地址可被所有从机应答。

系统复位后，SADDR 和 SADEN 两个寄存器初始化为 0，这两个结果设定了给定地址和广播地址为 XXXXXXXX（所有位都被忽略）。这有效地去除了多处机通讯的特性，禁止了自动寻址方式。这样的 UART 将对任何地址都产生应答，兼容了不支持自动地址识别的 8051 控制器。用户可以按照上面提到的方法实现软件识别地址的多机通讯。

12.5 端口与 IO 功能

RX 端口: UARTxEN 使能后，REN（接收使能）disable 的情况下，RX 状态由此引脚的其他复用（需开启）或 IO 控制决定；一旦 REN 使能，RX 为输入模式，上拉受 IO 上拉寄存器控制；

TX 端口: UARTxEN 使能后，因为没有发送使能，写 SBUF 前，都是空闲状态，TX 状态由此引脚的其他复用（需开启）或 IO 控制；当对 SBUF 写操作，TX 由 UART 发送控制，发送完成恢复空闲状态，继续由此引脚的其他复用（需开启）或 IO 控制。

注：建议将相应 IO 的功能设为输入上拉，以免 UART 进入空闲状态释放端口后误输入或输出一个低电平起始位 。

12.6 帧出错检测

当寄存器 PCON 中的 SSTAT 位为逻辑 1 时，帧出错检测功能才有效。3 个错误标志位被置 1 后，只能通过软件清零，尽管后续接收的帧没有任何错误也不会自动清零。

注：SSTAT=1 时是访问状态位（FE，RXOV 和 TXCOL），SSTAT=0 时是访问方式选择位（SM0，SM1 和 SM2）。

发送冲突

当一个发送正在进行，而此时软件写数据到 SBUF 寄存器，则发送冲突位 TXCOL 置 1。如果发生了冲突，新数据会被忽略，不会被写入发送缓冲器。

接收溢出

在接收缓冲器中数据未被读取之前，RI 被清 0，此时如果又有新的数据存入接收缓冲器，则接收溢出位 RXOV 位置 1。如果发生了接收溢出，接收缓冲器中原来的数据将丢失。

帧出错

如果检测到一个无效（低）停止位，则帧出错位 FE 置 1。

暂停检测

当连续检测到 11 个位都为低电平位时，则认为检测到一个暂停。由于暂停条件同样满足帧错误条件，因此检测到暂停时也会报告帧错误。一旦检测到暂停条件，UART 将进入空闲状态并一直保持，直至接收到有效停止位（RXD 引脚上出现上升沿）。

12.7 UART 相关寄存器

UART 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0 /9D	**SBUF_1**	串口 1 缓存寄存器	SBUF7_1	SBUF6_1	SBUF5_1	SBUF4_1	SBUF3_1	SBUF2_1	SBUF1_1	SBUF0_1	xxxx xxxx
SFR0 /9C	**SCON_1**	串口 1 控制寄存器	SM0_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1	0000 0000
SFR0 /9B	**SADEN**	串口 0 地址掩码寄存器	SADEN 7	SADEN 6	SADEN 5	SADEN 4	SADEN 3	SADEN 2	SADEN 1	SADEN0	0000 0000
SFR0	**SADDR**	串口 0 从机地址	SADDR	SADDR	SADDR	SADDR	SADDR	SADDR	SADDR	SADD	0000

/9A		寄存器	7	6	5	4	3	2	1	R0	0000
SFR0 /99	**SBUF**	串口 0 缓存寄存器	SBUF7	SBUF6	SBUF5	SBUF4	SBUF3	SBUF2	SBUF1	SBUF0	xxxx xxxx
SFR0 /98	**SCON**	串口 0 控制寄存器	SM0/FE	SM1/RXOV	SM2/TXCOL	REN	TB8	RB8	TI	RI	0000 0000
SFR0 /87	**PCON**	电源控制寄存器	UART1EN	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP	0—0 0100

电源控制寄存器（0x87，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	UART1EN	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP
R/W	R/W	-	-	R/W	R/W	R/W	R/W	R/W
复位值	0	-	-	0	1	0	0	0

BIT[7:4] **UARTnEN（n = 0、1）** – UARTx（x = 0、1）模块使能位

0：关闭 UARTx 模块，端口用于通用 IO 或其他复用模块；

1：打开 UARTx 模块。

BIT[3] **SMOD0** – UART0 波特率倍增位

0：UART0 波特率保持现状；

1：UART0 波特率增加一倍。

注：该控制位，因模块本身的差异，仅在 UART0 中存在，UART1 中并不存在；且仅在 UART0 的方式 2 中有效，其他方式无效。

BIT[2] **SSTAT0** – UART0 寄存器功能选择位

0：寄存器 SCON[7:5]功能为 SM[0:2]；

1：寄存器 SCON[7:5]功能为 FE，RXOV，TXCOL。

注：该控制位，因模块本身的差异，仅在 UART0 中存在，UART1 中并不存在。

特别注意：上述控制位（PCON[7:2]）皆不受 SLPCR 保护寄存器的影响，可以直接读/写。

串口 0 控制寄存器（0x98，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON	SM0/FE	SM1/RXOV	SM2/TXCOL	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

串口 1 控制寄存器（0x9C，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON_1	SM0_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

UART0 在 SSTAT=0 时；UART1 始终为：

BIT[7:6] **SM[0:1]** 或者 **SM[0:1]_1** – UART 方式控制位（SSTAT=0）

SM[0:1]	UART 工作方式	适用
00	方式 0：同步方式，固定波特率	UART0
01	方式 1：8 位异步方式，可变波特率	UART0、UART1

10	方式 2：9 位异步方式，固定波特率	UART0
11	方式 3：9 位异步方式，可变波特率	UART0、UART1

注：当 UART1 的 SM[0:1]_1 设置为 00 或者 10 时，UART1 不能正常工作。

BIT[5] **SM2** 或者 **SM2_1** – UART 功能设定位（SSTAT=0）

SM2	方式 0	方式 1	方式 2/3
0	波特率=Fsys/12	禁止停止位确认检验，停止位将置 RI 为 1 产生中断	任何字节均会置 RI 为 1 产生中断
1	波特率=Fsys/4	允许停止位确认检验，只有有效的停止位(1)才能置 RI 为 1 产生中断	只有寻址字节(第 9 位=1)能置 RI 为 1 产生中断

注：在 UART1 中不支持方式 0 和方式 2。

UART0 在 SSTAT=1 时：

BIT[7] **FE** – 帧出错标志位（SSTAT=1）

0：无帧出错，由软件清 0；

1：发生帧出错，由硬件置 1。

BIT[6] **RXOV** – 接收溢出标志位（SSTAT=1）

0：无接收溢出，由软件清 0；

1：接收溢出，由硬件置 1。

BIT[5] **TXCOL** – 发送冲突标志位（SSTAT=1）

0：无发送冲突，由软件清 0；

1：有发送冲突，由硬件置 1。

注：以上 3 个控制位（Bit[7:5]），因模块本身的差异，仅在 UART0 中存在，UART1 中并不存在。

BIT[4] **REN** 或者 **REN_1** – 接收器允许位

0：禁止接收；

1：允许接收。

BIT[3] **TB8** 或者 **TB8_1** – 方式 2/3 时发送数据的第 9 位

0：方式 2/3 发送数据的第 9 位为 0；

1：方式 2/3 发送数据的第 9 位为 1。

BIT[2] **RB8** 或者 **RB8_1** – 方式 1/2/3 时接收数据的第 9 位（停止位或数据位）

0：方式 1/2/3 时接收数据的第 9 位为 0；

1：方式 1/2/3 时接收数据的第 9 位为 1。

BIT[1] **TI** 或者 **TI_1** – 发送中断标志位

0：由软件清 0；

1：在方式 0 的第 8 位最后，或在其他方式的停止位开始，由硬件置 1。

BIT[0] **RI** 或者 **RI_1** – 接收中断标志位

0：由软件清 0；

1：在方式 0 的第 8 位最后，或在其他方式的停止位开始，由硬件置 1。

串口 0 缓存寄存器（0x99，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SBUF	SBUF7	SBUF6	SBUF5	SBUF4	SBUF3	SBUF2	SBUF1	SBUF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

串口 1 缓存寄存器（0x9D，SFR0）

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

SBUF_1	SBUF7_1	SBUF6_1	SBUF5_1	SBUF4_1	SBUF3_1	SBUF2_1	SBUF1_1	SBUF0_1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] **SBUF[7:0]** 或者 **SBUF[7:0]_1** – UART 数据缓存，SBUF 访问两个寄存器：1 个移位寄存器和 1 个接收锁存寄存器。SBUF 的写入将发送字节到移位寄存器中，然后开始从端口发送；SBUF 的读取将返回接收锁存寄存器中的内容

注：UART0 和 UART1 的发送和接收用缓存，使用了同一个缓存寄存器，写入时是一个通道，读取时是另一个通道，故用户编写软件中需要注意。

串口 0 从机地址寄存器（0x9A，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SADDR	SADDR7	SADDR6	SADDR5	SADDR4	SADDR3	SADDR2	SADDR1	SADDR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] SADDR[7:0] – UART 的从机地址

注：UART1 没有该寄存器，也不支持该功能。

串口 0 地址掩码寄存器（0x9B，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SADEN	SADEN7	SADEN6	SADEN5	SADEN4	SADEN3	SADEN2	SADEN1	SADEN0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] SADENn – SADDRn 检验控制位（n=7-0）

0：忽略 SADDRn 位；

1：检验 SADDRn 位是否对应接收地址。

注：UART1 没有该寄存器，也不支持该功能。

13 串行外设接口SPI

13.1 SPI 概述

片上 SPI 支持全双工，3 线/4 线同步传输，主从机操作模式可选，LSB/MSB 优先可选，带主模式故障出错标志，带写入冲突标志保护等功能。

串行外部设备接口（SPI）是一种高速串行通信接口，允许芯片与外部设备（包括芯片）进行全双工同步串行信号通信。

13.2 SPI 信号描述

主输出从输入（**MOSI**）：该信号表示主设备、从设备之间，数据流经主设备串行发送到从设备，主设备输出，从设备输入。

主输入从输出（**MISO**）：该信号表示主设备、从设备之间，数据流经从设备串行发送到主设备，从设备输出，主设备输入。注意：在 4 线 SPI 通信中从设备必须被选中，方可发送数据；若未被选中 MISO 引脚将处于高阻状态。

串行时钟（**SCLK**）：该信号用作控制 MOSI 和 MISO 线上输入输出数据的同步移动，每 8 个时钟周期 MOSI 和 MISO 线上传送一个字节， 如果从设备未被选中 SCLK 信号将被此设备忽略。注意：只有主设备才能产生 SCLK 信号。

从设备选择信号（n**SS**）：当做从设备使用时，每一个 SPI 从设备都带有一个片选信号，该信号低电平有效，当引脚信号由高到低反转时表明该从设备被选中，开始与 SPI 主设备进行通信。为了防止 SPI 通信冲突，同一时刻仅允许一个主设备与从设备通信。当做主设备使用时，该引脚可另作他用，从设备时交给 SPI 主设备控制。

13.3 工作模式

片上的 SPI 在工作中，仅可配置为主/从模式中的一种，对应于 3 线/4 线 SPI 模式，参考做法如下：

<table>
<tr><th>通信模式</th><th>本机 SPI 设备</th><th>本机 SPI 的 SS 脚</th><th>其他 SPI 设备</th><th>其他 SPI 设备 SS 脚</th></tr>
<tr><td>一主一从</td><td rowspan="2">主模式</td><td rowspan="2">可另作他用</td><td rowspan="2">从模式</td><td>拉低（片选信号始终有效）</td></tr>
<tr><td>一主多从</td><td>通过 GPIO 控制，通信时分别拉低（片选有效），切换不同从设备</td></tr>
<tr><td>一从一主</td><td>从模式</td><td>由其他 SPI 设备控制是否拉低（片选有效）</td><td>主模式</td><td>可另作他用</td></tr>
</table>

13.3.1 主机模式

当 SPCON 寄存器的 MSTR 位置 1 时，SPI 模块工作在主机模式。整个 SPI 系统中只允许一个主机启动传输，每次传输总是由主机发起。

在 SPI 作为主机模式中，向 SPI 的数据寄存器 SPDAT 写入数据后，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为空，那么主设备立即按照 SCLK 的时钟频率串行地移出发送寄存器中的数据到 MOSI 上。当传送完毕，SPSTA 寄存器中的 SPIIF 位置 1。如果 SPI 中断被允许，当 SPIIF 位置 1，会产生一个中断请求。

当主设备通过 MOSI 传送数据给从设备时，相应的从设备同时也通过 MISO 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。因此，SPIIF 标志位置 1 既表示数据发送完成，又表示接收完成。用户可以从 SPDAT 寄存器中读出接收的数据，并清除 SPIIF。

13.3.2 从机模式

当 SPCON 寄存器的 MSTR 位清 0 时，SPI 模块工作在从机模式。在数据传送之前，从设备的 nSS 引脚必须拉低，且必须保持低电平直到一个字节数据传送完毕。当从机设备的 nSS 管脚不为低时，主机设备不能与从机交换数据。在数据传输开始前和数据传输完成前，nSS 管脚都需要保持低电平状态。如果 nSS 变为高电平，SPI 模块将被迫进入闲置状态。如果 nSS 管脚在传输的过程被置高，那么传输将被取消，接收移位缓存区里剩下的位数将变高，同时也将进入闲置

状态。

在从机模式下，按照主机的 SCLK 的时钟信号，数据通过 MOSI 管脚由主机向从机传输，通过 MISO 管脚由从机向主机传输。一个位计数器记录 SCLK 的边沿数，当数据移位寄存器移入 8 位数据（一字节）；同时，在移位寄存器接收到 8 位数据（一字节），数据将移到接收数据缓存器，同时 SPIIF 置 1。若 SPI 中断被允许，当 SPIIF 置 1 时，会产生一个中断请求。此时接收移位寄存器将保持且 SPIIF 位置 1，这样 SPI 从设备将不会接收到任何数据，直到 SPIIF 位清 0。SPI 从设备必须在主设备开始一次新的数据传输前，将要传的数据写入发送移位寄存器。如果发送开始前未写入数据，从设备将传送“0x00”给主设备。如果写 SPDAT 操作发生在传输过程中，那么 SPI 从设备的 WCOL 标志位置 1，表示写 SPDAT 冲突。但是移位寄存器的数据不受影响，本次传输也不会中断。

13.4 数据传输格式

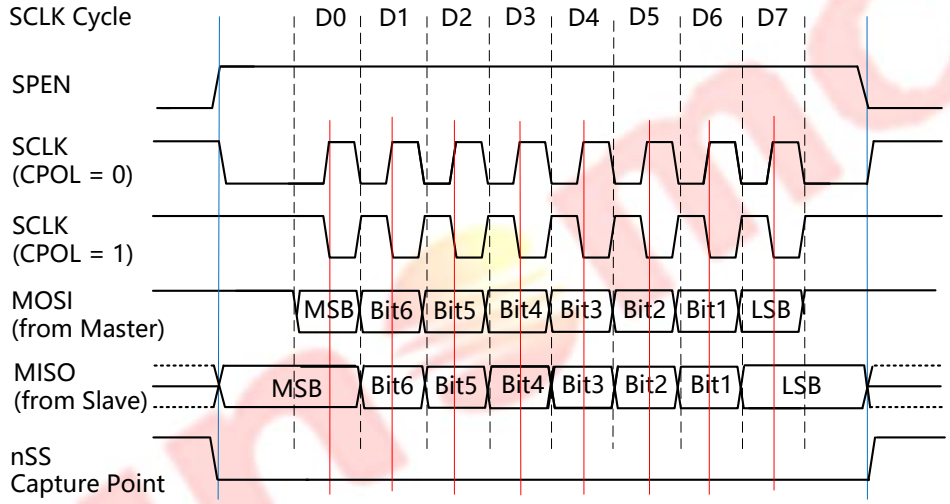
通过软件设置 SPCON 寄存器的 CPOL 位和 CPHA 位，用户可以选择 SPI 时钟 SCLK 的极性和相位的组合。

CPOL 位定义时钟的极性，即空闲时的电平状态。

CPHA 位定义时钟的相位，即定义了允许数据采样移位的时钟边沿。在 SPI 通信中主从通讯的两个设备，时钟极性相位的设置应一致。

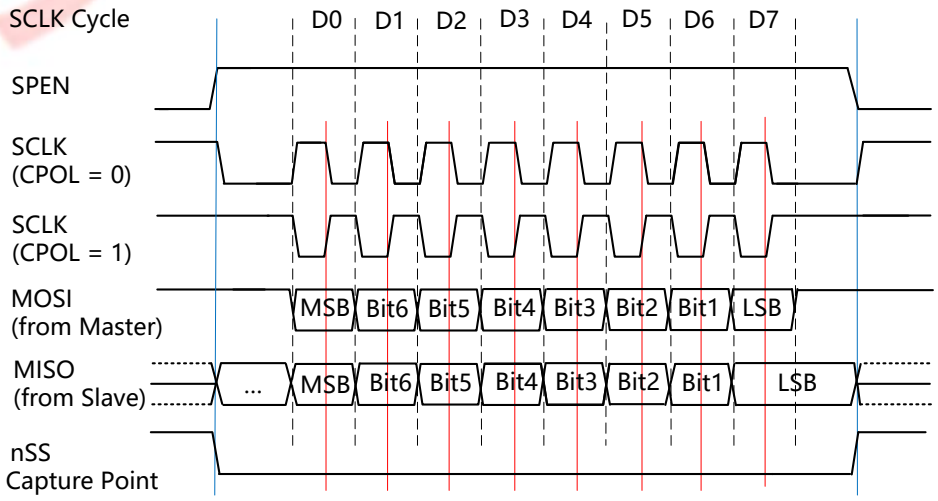
当 CPHA=0，SCLK 的第一个捕获数据，从设备必须在 SLCK 的第一个沿之前将数据准备好。

CPHA = 0 时的数据传输示意图



当 CPHA=1，主设备在 SCLK 的第一个沿将数据输出到 MOSI 上，从设备把 SCLK 的第一沿作为开始发送信号，SCLK 的第二个沿开始捕获数据，因此用户必须在第一个 SLCK 的两个沿内完成写 SPDAT 的操作。这种数据传输格式是一个主设备一个从设备之间通讯的首选。

CPHA = 1 时的数据传输示意图



13.5 出错检测

出错情形一：模式故障（MODF 位）

SPI 主模式下的模式故障出错表明 nSS 引脚上的电平状态与实际的设备模式不一致。SPSTA 寄存器中 MODF 位置 1 后，表明系统控制存在多主设备冲突的问题。这种情况下，由于 SPCON 寄存器的 nSS 引脚禁止位（SSDIS）清 0，nSS 引脚信号为低时，MODF 标志位置 1。然而，对于只有一个主设备的系统来说，主设备的 nSS 引脚被拉低，那是绝对不允许的。这种情况下，为防止 MODF 置 1，可使 SPCON 寄存器中的 SSDIS 位置 1，让 nSS 引脚作为普通 I/O 口或是其它功能引脚。

此时，在重启 SPI 通信前，用户必须将 MODF 位以软件方式清 0，将 SPCON 寄存器中的 MSTR 位和 SPSTA 寄存器的 SPIEN 位置 1，重新启动主模式。

注：当 SSDIS=0 且 MODF=1 时将产生 SPI 错误中断。

出错情形二：写冲突（WCOL 位）

在发送数据序列期间写入 SPDAT 寄存器会引起写的冲突，SPSTA 寄存器中的 WCOL 标志位置 1。WCOL 位置 1 不会引起中断，发送也不会中止。该标志位仅用于通知用户，SPI 数据写入的时机不对，WCOL 位需由软件清 0。

出错情形三：超限情况（SPIOV 位）

当作为主设备或从设备时，软件尚未清除 SPIIF 位，此时主或从设备又试图发送/接受几个数据字节时，发生超限情况。在这种情况下，接收移位寄存器保持原有数据，SPIIF 置 1，同样 SPI 设备直到 SPIIF 被清除后才会再接收新的数据。在 SPIIF 位被清除之前继续调用中断，SPIOV 位置 1 不会引起中断，发送也不会中止。该标志位仅用于通知用户，仍有接收的 SPI 数据未读取，SPIOV 位需由软件清 0。

13.6 SPI 相关寄存器

SPI 寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)
SFR0 /AF	**SPDAT**	SPI 数据寄存器	SPDAT 7	SPDAT 6	SPDAT 5	SPDAT 4	SPDAT 3	SPDAT 2	SPDAT 1	SPDA T0	xxxx xxxx
SFR0 /AE	**SPSTA**	SPI 状态寄存器	SPIEN	SPIIF	MODF	WCOL	SPIOV	-	-	-	0000 0---
SFR0 /AD	**SPCON**	SPI 控制寄存器	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0	0000 0000

SPI 控制寄存器（0xAD，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPCON	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] **DIR** – 传输方向选择位

0：发送时，由 MSB 开始依次发送；

1：发送时，由 LSB 开始依次发送。

BIT[6] **MSTR** – SPI 主从选择位

0：片上 SPI 做从设备使用；

1：片上 SPI 做主设备使用。

注：如果要设置 MSTR=1，务必先让 SSDIS=1；或者在 SSDIS=0 的状态，但请提前上拉外部 nSS 引脚，否则 SPI 保护功能生效，并不能正常进入主模式，该位无法置 1，仍保持 0（从模式状态）。

BIT[5] **CPHA** – SPI 时钟相位控制位

0：SPI 周期的第一个沿采集数据；

1：SPI 周期的第二个沿采集数据。

BIT[4] **CPOL** – SPI 时钟极性控制位

0：在 Idle 状态下 SCK 处于低电平；

1：在 Idle 状态下 SCK 处于高电平。

BIT[3] **SSDIS** – SPI 片选信号 nSS 控制位

0：在主/从模式下，打开 nSS 引脚；

1：在主/从模式下，关闭 nSS 引脚，nSS 脚位可另作他用。

注：如果 SSDIS 置 1，不产生 MODF 中断请求；在从模式下，若 CPHA 为 0，该位无效。

BIT[2:0] **SPR[2:0]** – SPI 时钟速率选择位

SPR[2:0]	SPI 工作时钟 F_{SPI}
000	Fsys/2
001	Fsys/4
010	Fsys/8
011	Fsys/16
100	Fsys/32
101	Fsys/64
110	Fsys/128
111	Fsys/256

注：作从模式时，SPI 的时钟由主机决定，但不应超过 8Mbit/s；作主模式时，SPI 的时钟由上表决定，最大设置也不应超过 8Mbit/s。

SPI 状态寄存器（0xAE，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPSTA	SPIEN	SPIIF	MODF	WCOL	SPIOV	-	-	-
R/W	R/W	R/W	R/W	R/W	R/W	-	-	-
复位值	0	0	0	0	0	-	-	-

BIT[7] **SPIEN** – SPI 模块使能位

0：关闭 SPI 模块，端口用于通用 IO 或其他复用模块；

1：打开 SPI 模块，MOSI、MISO、SCLK 端口有效，开始数据的收发。

***注**：nSS 端口的使能与否，直接由 SSDIS 位来控制。*

BIT[6] **SPIIF** – SPI 数据传送标志位

0：由软件清 0；

1：表明数据传输已完成，由硬件置 1。

BIT[5] **MODF** – 模式故障标志位

0：由软件清 0；

1：表明 nSS 引脚电平与 SPI 模式不匹配，由硬件置 1。

BIT[4] **WCOL** – 写入冲突标志位

0：表明没有写入冲突发生，由软件清 0；

1：表明已检测到写入冲突发生，由硬件置 1。

BIT[3] **SPIOV** – 接收超限标志位

0：表明以处理接收超限，由软件清 0；

1：表明已检测到接收超限，由硬件置 1。

SPI 数据寄存器（0xAF，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPDAT	SPDAT7	SPDAT6	SPDAT5	SPDAT4	SPDAT3	SPDAT2	SPDAT1	SPDAT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	X	X	X	X	X	X	X	X

BIT[7:0] **SPDAT[7:0]** – SPI 数据缓存寄存器

注 1：写入 SPDAT 的数据被放置到发送移位寄存器中，而读取 SPDAT 时将获得接收移位寄存器中的数据，本质上对应了两个不同的寄存器，因此用户无法直接读取到自己写入的值。

注 2：SPDAT 的复位值为随机值，因此作主模式发送数据前必须设定其初值；当关闭 SPI 功能后，再读取 SPDAT 的内容无效。

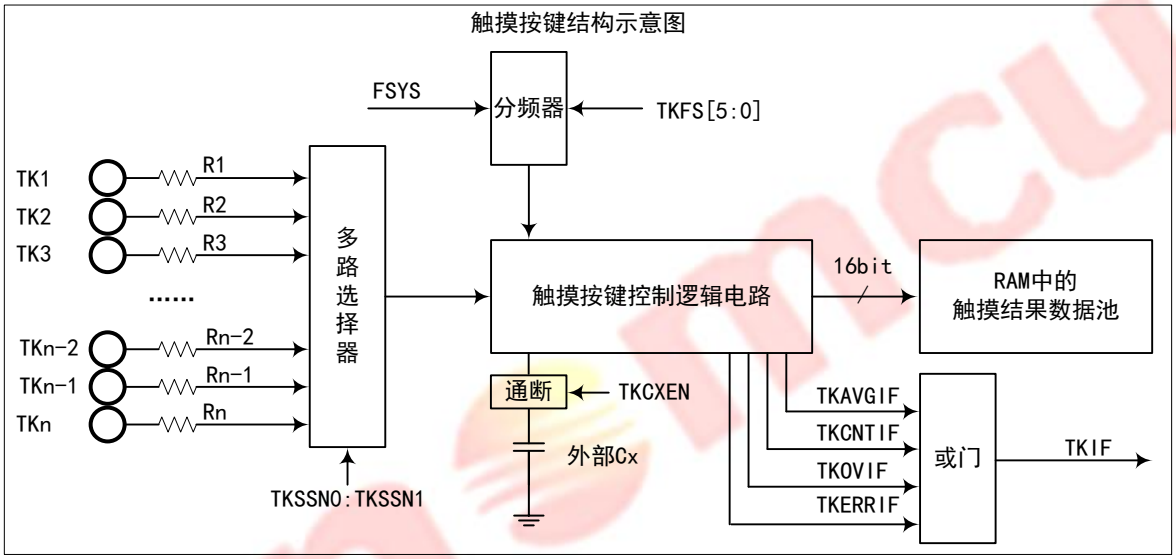
14 触摸按键

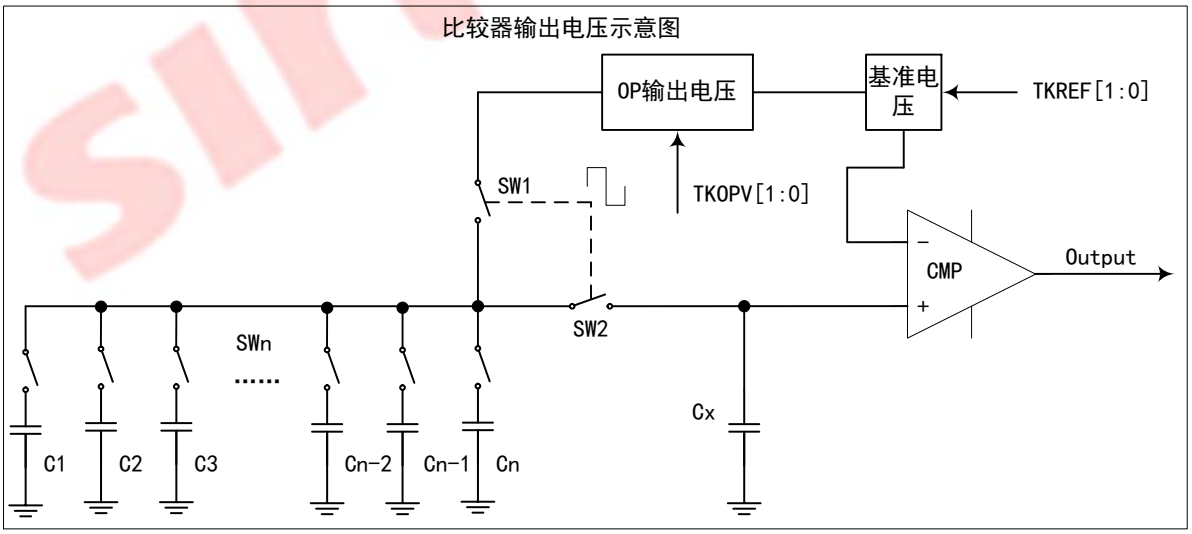
本芯片内建硬件触摸功能模块，最大支持 16 通道的触摸按键输入。支持 4MHz、2MHz、1MHz、500KHz 四种工作频率，支持可选基准电压，支持 1 路高精度内部参考电压，支持 1 路独立触摸按键扫描中断，借助外部 Cx 电容实现“电荷迁移”，用户可以根据不同应用场合外接不同的 Cx 电容。

本模块实现的原理是一种电容式感应触摸检测，即由于存在“电荷的迁移”，简单地将就是“触摸按键按下”与“触摸按键未按下”两者的电容电荷是不一样的。本触摸按键单元可以检测到前后两者之间细微的差别，以扫描方式实现对“触摸按键是否动作”的检测。

14.1 触摸按键内部结构图

触摸按键（TK）单元内部结构示意图和内部比较器示意图。





本模块扫描式工作的实现原理，大致如下：

1. 使能 TK 模块，对 TK 模块的工作频率、采样次数、扫描通道使能、放电时间、比较器电压等参数的初始化设定。
2. 设置 TK 基准电压，若选择的基准电压为内部基准电源电压，需要开启后，软件延时 50us，等待其稳定后再使用。
3. 设置一个 28 位的放大系数，设置 TKSSN0 和 TKSSN1 安排好一组扫描顺序，将 TKSAT 位置 1，启动对一个有

效 TK 通道的扫描。

4. 扫描中，TK 模块先让 SWn（这里 n=0～15）和 SW1 闭合，SW2 断开，基准电源电压对 Cn（这里 n=0～15）进行充电。同时充电计时器启动，开始统计电容 Cx 的充电时间。

5. 再让 SWn（这里 n=0～15）和 SW2 闭合，SW1 断开，此时 Cn（这里 n=0～15）对 Cx 进行放电。

6. 反复以上步骤 4 和步骤 5 两个动作，直到 Cx 上的电压大于比较器基准电压，则比较器的 Output 输出高电平，滤波器开始采样，同时充电计时器停止计数。为消除干扰信号的影响，提高系统稳定性，需要对 Output 做滤波处理。

7. TK 模块的放电计时器输出高电平，并让 SW1 和 SW2 断开，对 Cx 进行放电，开始统计电容 Cx 放电时间。

8. 当滤波器输出为低电平且电容 Cx 放电结束时，若设定为多次采样且多次采样未结束时，跳转到步骤 4 开始执行下一次采样；若采样结束，则跳转到下一个步骤 9，开始计算采样结果。

9. 把一次采样或多次采样（取其平均值）的采样结果，通过放大系数运算后得到一个扫描结果。若没有发生启动错误也没有发生扫描结果溢出，则 TK 模块自动把 16 位扫描结果加载到输出结果数据池中，一个 TK 通道扫描完成。若 TKIE=1 有异常产生（TKCNTIF、TKOVIF、TKERRIF 其中有一个置 1），则结束本次扫描并产生 TK 中断，一组 TK 通道扫描也将因此停止。

10. 根据 TKSSN0 和 TKSSN1 的顺序，切换到下一个有效的 TK 通道，跳转到步骤 4 开始扫描，依次类推直到一组 TK 通道全部扫描结束，若 TKIE=1 且无异常产生 TKAVGIF 置 1 并产生 TK 中断，一次按键通道的扫描完成。

11. TK 中断处理

当 TKIF 置 1（TKIE=1 时，进入 TK 中断），说明 TKAVGIF、TKCNTIF、TKOVIF、TKERRIF 四个标志位中有一个置 1 了。

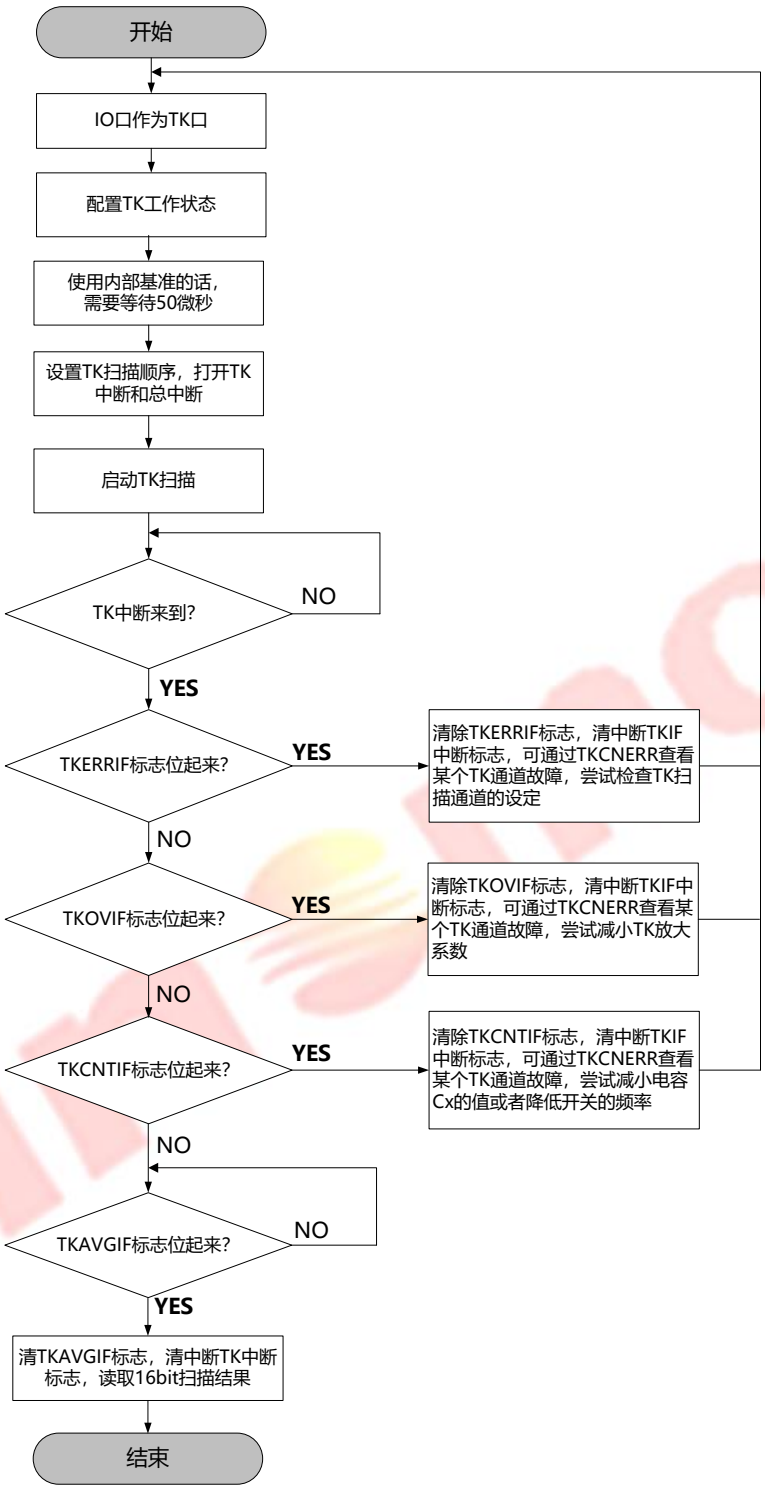
当 TKAVGIF 置 1，说明 TK 扫描正常结束，读出 RAM 数据池中的数据，需要手动清除 TKAVGIF 标志位和 TKIF 标志位，退出中断。

当 TKCNTIF 置 1，说明按键扫描计数溢出错误，可以减小电容 Cx 的值或者降低开关的频率，手动清除 TKCNTIF 标志位，退出中断，重新启动 TK 扫描。

当 TKOVIF 置 1，说明数据寄存器运算溢出错误，重新设置（减小）放大系数寄存器的值，手动清除 TKOVIF 标志位，退出中断，重新启动 TK 扫描。

当 TKERRIF 置 1，说明 TK 按键控制器启动错误，可以确认 TK 扫描的通道是否正确，手动清除 TKERRIF 标志位，退出中断，重新启动 TK 扫描。

14.2 触摸按键控制流程



14.3 触摸按键模块低功耗模式

使能控制位 TKLPMD，触摸按键模块进入低功耗模式，LIRC 时钟分频（2048/4096/8192/16384 可选）后作为 TK 模块的启动触发信号，触发信号硬件启动 HIRC 时钟并使能 TKEN，等待 2 个 LIRC clk（HIRC 和 TK 参考建立 warmup），启动 TK 通道扫描，扫描结束数据结果与 TK 比较阈值比较，跟据预设的比较方向控制 TKCMPDIR，满足阈值条件，退出低功耗模式置比较标志 TKCMPIF，并记录当前扫描通道号 TKCNER[3:0]，若开启中断使能，产生中断；若不满足阈值

条件，继续扫描下个按键通道直至扫描结束，关闭 TKEN 和并恢复 HIRC 状态。

此中断可唤醒 sleep 和 stop 模式。

低功耗模式 TKEN 关闭后，所有的 TK 模块寄存器不能复位，保证下次触发信号启动 TKEN 后，用户配置有效。

TK 低功耗操作流程：

(1) 打开 TKEN，配置 TK 模块寄存器；

(2) 使能控制位 TKLPMD；

(3) 关闭 TKEN

(4) 进入低功耗模式

14.4 触摸按键相关寄存器

触摸按键（TK）寄存器汇总表

地址(SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0/F7	**TKCMPL**	触摸比较阈值低位寄存器	CMP15	CMP14	CMP13	CMP12	CMP11	CMP10	CMP9	CMP8	00000000
SFR0/F6	**TKCMPH**	触摸比较阈值高位寄存器	CMP7	CMP6	CMP5	CMP4	CMP3	CMP2	CMP1	CMP0	00000000
SFR0/F5	**TKSSN1**	触摸扫描顺序控制寄存器 1	TK15	TK14	TK13	TK12	TK11	TK10	TK9	TK8	00000000
SFR0/F4	**TKSSN0**	触摸扫描顺序控制寄存器 0	TK7	TK6	TK5	TK4	TK3	TK2	TK1	TK0	00000000
SFR0/F3	**TKCHEN1**	触摸端口控制寄存器 1	TKCH15	TKCH14	TKCH13	TKCH12	TKCH11	TKCH10	TKCH9	TKCH8	00000000
SFR0/F2	**TKCHEN0**	触摸端口控制寄存器 0	TKCH7	TKCH6	TKCH5	TKCH4	TKCH3	TKCH2	TKCH1	TKCH0	00000000
SFR0/E7	**TKCHERR**	触摸扫描出错提示寄存器	-	-	-	-	TKCNER3	TKCNER2	TKCNER1	TKCNER0	----0000
SFR0/E6	**TKANA**	触摸模拟参数选择寄存器	CMPSAMMD1	CMPSAMMD10	TKPULLCUR2	TKPULLCUR1	TKPULLCUR0	TKRESVAL2	TKRESVAL1	TKRESVAL0	00000000
SFR0/E5	**TKVREF**	触摸模块电压选择寄存器	TKREF1	TKREF0	TKCCT1	TKCCT0	TKOPV1	TKOPV0	TKDCT1	TKDCT0	00000000
SFR0/E4	**TKCON3**	触摸选择控制寄存器 3	TKAVGIF	TKCNTIF	TKOVIF	TKERRIF	-	-	TKCMPIF	TKIF	0000--00
SFR0/E3	**TKCON2**	触摸选择控制寄存器 2	TKRDNEN	TKCMPDIR	TKRDSL1	TKRDSL0	TKCXEN	TKCOMEN	TKCOMVS	TKCOMMD	00000000
SFR0/E2	**TKCON1**	触摸选择控制寄存器 1	TKTRGDIV1	TKTRGDIV0	TKFS5	TKFS4	TKFS3	TKFS2	TKFS1	TKFS0	00000000
SFR0/E1	**TKCON0**	触摸选择控制寄存器 0	TKEN	TKLPMD	TKSAT	TKPAREN	TKMOD	TKOPSEL	TKSAMP1	TKSAMP0	00000000
SFR0/C7	**TKAMP3**	触摸放大系数控制寄存器 3	-	-	-	-	TKAMP27	TKAMP26	TKAMP25	TKAMP24	----0000
SFR0/C6	**TKAMP2**	触摸放大系数控制寄存器 2	TKAMP23	TKAMP22	TKAMP21	TKAMP20	TKAMP19	TKAMP18	TKAMP17	TKAMP16	00000000
SFR0/C5	**TKAMP1**	触摸放大系数控制寄存器 1	TKAMP15	TKAMP14	TKAMP13	TKAMP12	TKAMP11	TKAMP10	TKAMP9	TKAMP8	00000000

SFR0 /C4	**TKAMP 0**	触摸放大系数控制寄存器 0	TKAMP 7	TKAMP 6	TKAMP 5	TKAMP 4	TKAMP 3	TKAMP 2	TKAMP 1	TKAMP 0	0000 0000

触摸控制寄存器 0（0xE1，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON0	TKEN	TKLPMD	TKSAT	TKPAREN	TKMOD	TKOPSEL	TKSAMP1	TKSAMP0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] **TKEN** – TK 模块使能位

0：关闭整个 TK 模块；

1：打开整个 TK 模块。

BIT[6] **TKLPMD** – TK 模块低功耗模式使能位

0：关闭低功耗模式；

1：使能低功耗模式。

BIT[5] **TKSAT** – 启动 TK 模块触摸扫描位

0：未启动按键扫描或按键扫描结束；

1：启动按键扫描或正在执行按键扫描中。

BIT[4] **TKPAREN** – TK 所有被使用的通道内部并联控制位

0：所有被使用的 TK 通道各自独立使用；

1：将所有被使用的 TK 通道内部并联。

注：并联仅针对被选中的通道,未选中通道不做并联处理。

BIT[3] **TKMOD** – TK 模块扫描模式控制位

0：选择充电次数来作为扫描参数；

1：选择充电时间来作为扫描参数。

BIT[2] **TKOPSEL** – TK 模块输出电压控制位

0：选择 OP 作为输出电压；

1：选择 VDD 作为输出电压。

注：该位为 0 时，OP 的输出电压值由 TKOPV[1:0]来选择；该位为 1 时，由 VDD 直接输出电压。

BIT[1:0] **TKSAMP [1:0]** – TK 模块采样次数选择位

TKSAMP1	TKSAMP0	采样次数选择
0	0	按键采样 1 次输出数据，则 D15-D0 的结果为 1 次采样的平均值
0	1	按键采样 3 次输出数据，则 D15-D0 的结果为 1 次采样的平均值(去除最大值及最小值)
1	0	按键采样 6 次输出数据，则 D15-D0 的结果为 4 次采样的平均值(去除最大值及最小值)
1	1	按键采样 10 次输出数据，则 D15-D0 的结果为 8 次采样的平均值(去除最大值及最小值)

触摸控制寄存器 1（0xE2，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON1	TKTRGDIV1	TKTRGDIV0	TKFS5	TKFS4	TKFS3	TKFS2	TKFS1	TKFS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:6] **TKTRGDIV [1:0]** – TK 低功耗模式触发频率选择位

00：LIRC/2048

01：LIRC/4096

10：LIRC/8192

11：LIRC/16384

BIT[5:0] **TKFS[5:0]** – TK 工作频率选择位

TKFS5	TKFS4	TKFS3	TKFS2	TKFS1	TKFS0	工作频率 F_{TK}
0	0	0	0	0	0	F_{TK} = Fsys
0	0	0	0	0	1	F_{TK} = Fsys/2
000010 B（2）~111101 B（62）						F_{TK} = Fsys/ (n + 1)
1	1	1	1	1	0	F_{TK} = Fsys/63
1	1	1	1	1	1	F_{TK} = Fsys/64

注：该频率设置越高，则 TK 模块的扫描速度越快，工作频率推荐为 4MHz；其中 n 代表 TKFS 中设定的实际值。

触摸控制寄存器 2（0xE3，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON2	TKRDNEN	TKCMPDIR	TKRDSL1	TKRDSL0	TKCXEN	TKCOMEN	TKCOMVS	TKCOMMD
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] **TKRDEN** – TK 模块随机频率使能位

0：禁止触摸按键随机频率功能；

1：允许触摸按键随机频率功能。

BIT[6] **TKCMPDIR** – TK 模块比较阈值方向控制

0：大于预设阈值产生标志；

1：小于等于预设阈值产生标志。

注：此控制位仅在低功耗模式使能后有效

BIT[5:4] **TKRDSL[1:0]** – TK 模块随机频率使能位

TKRDSL[1:0]	TK 随机抖动设置位（在 TKFS[5:0]的基础上）
00	随机抖动±1
01	随机抖动±1，±2
10	随机抖动±1，±2，±3
11	随机抖动±1，±2，±3，±4

注 1：该位与 TKRDEN 位联动，当 TKRDEN 为 0 时，TKRDSL[1:0]位将无效；当 TKRDEN 为 1 时，TK 随机抖动的选择由 TKRDSL [1:0]位决定。

注 2：该位与 TKFS[5:0]位联动，当 TKFS[5:0]设定值≤0x01 时，不能进行抖动时钟；当 TKFS[5:0]设定值=0x02 时，TKRDSL [1:0]位只能选择 00 档；当 TKFS[5:0]设定值=0x03 时，TKRDSL [1:0]位有 00、01 两档可以选择；当 TKFS[5:0]设定值=0x04 时，TKRDSL [1:0]位有 00、01、10 三档可以选择；当 TKFS[5:0]设定值≥0x05 时，TKRDSL [1:0]位，四档全部可以选择。

BIT[3] **TKCXEN** – TK 外部 Cx 电容使能位

0：保留位

1：选择 Cx 电容引脚，对应的 TKCx 端口作为触摸模块的指定 Cx 电容接口，GPIO 和其他复用功能无效，用户必须接外部 Cx 电容。

注 1： 此位必须固定为 1。

注 2：使能（TKCXEN=1），用户使用到 TK 模块时，必须在外部接一个电容值，推荐使用在 1nF～47nF 之间的 10%精度的涤纶电容 Cx，Cx 的电容值越小灵敏度越高，电容越大灵敏度越低，是具体应用而定。

BIT[2] **TKCOMEN** – TK 模块波形补偿使能位

0：禁止触摸按键波形补偿；

1：允许触摸按键波形补偿。

BIT[1] **TKCOMVS** – TK 模块波形补偿电压选择位

0：保留位；

1：触摸按键补偿波形电平由 VDD 提供。

注：该位与 TKCOMEN 位联动，当 TKCOMEN 为 0 时，TKCOMV 位将无效；当 TKCOMEN 为 1 时，TKCOMV 位必须配置为 1。

BIT[0] **TKCOMMD** – TK 模块波形补偿模式选择位

0：触摸按键补偿波形为弱输出模式；

1：触摸按键补偿波形为强输出模式。

***注：**该位与 TKCOMEN 位联动，当 TKCOMEN 为 0 时，TKCOMMD 位将无效；当 TKCOMEN 为 1 时，波形补偿模式的选择由 TKCOMMD 位决定。*

触摸状态标志寄存器（0xE4，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON3	TKAVGIF	TKCNTIF	TKOVIF	TKERRIF	-	-	TKCMPIF	TKIF
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
复位值	0	0	0	0	-	-	0	0

BIT[7] **TKAVGIF** – TK 扫描正常完成标志位

0：TK 无采样或者一次 TK 扫描未完成，由软件清 0；

1：一次 TK 扫描已正常结束，硬件置 1。

BIT[6] **TKCNTIF** – TK 扫描计数溢出标志位

0：TK 扫描计数正常，未溢出，由软件清 0；

1：TK 扫描计数溢出，硬件置 1。

***注：**若出现该标志位，首先应该增大“TK 放电时间”；若放电时间调至最大，仍出现该标志位，应该考虑减小外部 Cx 的电容值。*

BIT[5] **TKOVIF** –TK 扫描结果溢出标志位

0：运算结果正常，无溢出，由软件清 0；

1：运算结果产生高位溢出，TK 运算结果有溢出，硬件置 1。

***注：**若出现该标志位，应该减小“TK 放大系数”，直到控制输出结果在一个合理的范围之内。*

BIT[4] **TKERRIF** – TK 扫描异常标志位

0：TK 扫描正常，由软件清 0；

1：TK 扫描启动异常，硬件置 1。

***注：**若出现该标志位，内部比较输出一直为高，应该确认 TK 扫描的通道是否正确。*

BIT[1] **TKCMPIF** – TK 比较器标志位

0：TK 扫描结果数据未满足比较阈值条件；

1：TK 扫描结果数据满足比较阈值条件，硬件置 1。

***注：**此位仅低功耗模式开启有效。*

BIT[0] **TKIF** – TK 中断标志位

0：TK 无中断，由软件清 0；

1：TK 有中断，即 TKAVGIF、TKCNTIF、TKOVIF、TKERRIF、TKCMPIF 五个标志位中其中有一个标志位置位，硬件置 1。

***注 1：**TKAVGIF、TKCNTIF、TKOVIF、TKERRIF 四个标志位之间，虽然没有优先级之分，但同一时刻只会产生一*

个。当产生任意一个标志位后，TK 模块处于停止状态，直到重新开始扫描之前，不会产生新的标志位。

***注 2：**当 TKIE 使能（TKIE=1）时，且 TKIF 置位，则系统会产生相应的 TK 中断。*

触摸放大系数控制寄存器 0-3（0xC4、0xC5、0xC6、0xC7，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKAMP3	-	-	-	-	TKAMP27	TKAMP26	TKAMP25	TKAMP24
TKAMP2	TKAMP23	TKAMP22	TKAMP21	TKAMP20	TKAMP19	TKAMP18	TKAMP17	TKAMP16
TKAMP1	TKAMP15	TKAMP14	TKAMP13	TKAMP12	TKAMP11	TKAMP10	TKAMP9	TKAMP8
TKAMP0	TKAMP7	TKAMP6	TKAMP5	TKAMP4	TKAMP3	TKAMP2	TKAMP1	TKAMP0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[27:0] **TKAMP[27:0]** – 28 位 TK 放大系数

其取值范围从 0x00000000 到 0x0FFFFFFF。

***注：**TK 放大系数的作用是将扫描后的结果放大，用户因根据不同应用场合进行调整。*

触摸端口控制寄存器 0-1（0xF2、0xF3，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCHEN0	TKCH7	TKCH6	TKCH5	TKCH4	TKCH3	TKCH2	TKCH1	TKCH0
TKCHEN1	TKCH15	TKCH14	TKCH13	TKCH12	TKCH11	TKCH10	TKCH9	TKCH8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **TKCHENn（n = 0-15）** – TK 端口使能位

0：禁止对应的 TKCHn（n = 0-15）的 TKn（n = 0-15）通道，该端口用于通用 IO 或其他复用功能；

1：允许对应的 TKCHn（n = 0-15）的 TKn（n = 0-15）通道，该端口仅用于 TK 功能。

***注：**外接 Cx 电容的端口的使能，通过 TKCXEN 位控制。*

触摸扫描出错提示寄存器（0xE7，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCHERR	-	-	-	-	TKCNER3	TKCNER2	TKCNER1	TKCNER0
R/W	-	-	-	-	R	R	R	R
复位值	-	-	-	-	0	0	0	0

BIT[3:0] **TKCHERR[3:0]** – TK 扫描出错提示位

TKCHER3	TKCHER2	TKCHER1	TKCHER0	扫描出错提示
0	0	0	0	TK 通道 0 扫描出错
0	0	0	1	TK 通道 1 扫描出错
0	0	1	0	TK 通道 2 扫描出错
0	0	1	1	TK 通道 3 扫描出错
0	1	0	0	TK 通道 4 扫描出错
0	1	0	1	TK 通道 5 扫描出错
0	1	1	0	TK 通道 6 扫描出错
0	1	1	1	TK 通道 7 扫描出错
1	0	0	0	TK 通道 8 扫描出错
1	0	0	1	TK 通道 9 扫描出错
1	0	1	0	TK 通道 10 扫描出错
1	0	1	1	TK 通道 11 扫描出错

1	1	0	0	TK 通道 12 扫描出错
1	1	0	1	TK 通道 13 扫描出错
1	1	1	0	TK 通道 14 扫描出错
1	1	1	1	TK 通道 15 扫描出错

注：当按键有错误标志位置起时，除 TKAVGIF 标志位（不会停止 TK 扫描的继续运行）外，其余异常标志位都会停止 TK 扫描的继续运行，并将对应一个出错通道的位信息保存于该寄存器中。该寄存器为只读寄存器，启动 TK 扫描后自动清 0 该标志位。

触摸模控制寄存器 3（0xE6，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKANA	CMPSAMMD1	CMPSAMMD0	TKPULLCUR2	TKPULLCUR1	TKPULLCUR0	-	TKRESVAL1	TKRESVAL0
R/W	R/W	R/W	R/W	R/W	R/W	-	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:6] **CMPSAMMD [1:0]** – 比较器工作模式选择

00：模式 0，TK 扫描流程中，比较器和储能大电容 Cx 之间开关常闭，输出实时采样；

01：模式 1，TK 扫描流程中，比较器和储能大电容 Cx 之间开关常闭， pad 电容（TK）向大电容 Cx 搬移电荷阶段比较器输出不采样，其余阶段比较输出实时采样；

其他： 保留位，禁止使用。

BIT[5:3] **TKPULLCUR [2:0]**

此位必须配置为 111B。

BIT[1:0] **TKRESVAL [1:0]**

此位必须配置 00B。

触摸控制寄存器 4（0xE5，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKVREF	TKREF1	TKREF0	TKCCT1	TKCCT0	TKOPV1	TKOPV0	TKDCT1	TKDCT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:6] **TKREF[1:0]** – TK 的内部基准电压选择位

TKREF[1:0]	TK 内部基准电压
00	Vtkref = 2.5V
01	Vtkref = 2V
10	Vtkref = 1.5V
11	Vtkref = 1V

BIT[5:4] **TKCCT[1:0]** – TK 去抖动时间选择位

TKTSP[1:0]	TK 去抖动时间
00	约为 4/Fsys
01	约为 8/Fsys
10	约为 16/Fsys
11	约为 32/Fsys

BIT[3:2] **TKOPV[1:0]** – TK 的 OP 输出电压选择位

TKOPV[1:0]	TK（OP）输出电压
00	Vtkop = 4V
01	Vtkop = 3V

10	Vtkop = 2.5V
11	Vtkop = 2V

BIT[1:0] **TKDCT[1:0]** – TK 放电时间选择位

TKDCT[1:0]	Cx 放电时间
00	约为 128/Fsys
01	约为 256/Fsys
10	约为 384/Fsys
11	约为 512/Fsys

触摸扫描序列控制寄存器 0-1（0xF4、0xF5，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKSSN0	TK7	TK6	TK5	TK4	TK3	TK2	TK1	TK0
TKSSN1	TK15	TK14	TK13	TK12	TK11	TK10	TK9	TK8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[15:0] **TKn（n = 0-15）** – TK 扫描序列控制

这里的 TKn（n = 0-15）分别对应了 TK（0 至 15）的 16 个通道 ，当 TKn（n = 0-15）设为 1 时，启动 TK 扫描后硬件自动扫描该通道，否则不扫描。

注: *TK 硬件的扫描顺序是从 TKSSN0 的 TK0 通道开始扫描到 TK7,之后切换至 TKSSN1,从 TK8 开始扫描到 TK15,至此完成一轮 TK 的扫描工作。该寄存器的值不会自动清 0。故用户启动 TK 扫描前，首先要确认扫描的通道号，并提前写入该寄存器。*

触摸按键扫描结果寄存器（0x4E0 - 0x4FF，XRAM）

XRAM 地址	TK 扫描结果低/高位寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
4FFH	**TK0L**	D7	D6	D5	D4	D3	D2	D1	D0
4FEH	**TK0H**	D15	D14	D13	D12	D11	D10	D9	D8
4FDH	**TK1L**	D7	D6	D5	D4	D3	D2	D1	D0
4FCH	**TK1H**	D15	D14	D13	D12	D11	D10	D9	D8
4FBH	**TK2L**	D7	D6	D5	D4	D3	D2	D1	D0
4FAH	**TK2H**	D15	D14	D13	D12	D11	D10	D9	D8
4F9H	**TK3L**	D7	D6	D5	D4	D3	D2	D1	D0
4F8H	**TK3H**	D15	D14	D13	D12	D11	D10	D9	D8
4F7H	**TK4L**	D7	D6	D5	D4	D3	D2	D1	D0
4F6H	**TK4H**	D15	D14	D13	D12	D11	D10	D9	D8
4F5H	**TK5L**	D7	D6	D5	D4	D3	D2	D1	D0
4F4H	**TK5H**	D15	D14	D13	D12	D11	D10	D9	D8
4F3H	**TK6L**	D7	D6	D5	D4	D3	D2	D1	D0
4F2H	**TK6H**	D15	D14	D13	D12	D11	D10	D9	D8
4F1H	**TK7L**	D7	D6	D5	D4	D3	D2	D1	D0
4F0H	**TK7H**	D15	D14	D13	D12	D11	D10	D9	D8
4EFH	**TK8L**	D7	D6	D5	D4	D3	D2	D1	D0
4EEH	**TK8H**	D15	D14	D13	D12	D11	D10	D9	D8
4EDH	**TK9L**	D7	D6	D5	D4	D3	D2	D1	D0
4ECH	**TK9H**	D15	D14	D13	D12	D11	D10	D9	D8

4EBH	**TK10L**	D7	D6	D5	D4	D3	D2	D1	D0
4EAH	**TK10H**	D15	D14	D13	D12	D11	D10	D9	D8
4E9H	**TK11L**	D7	D6	D5	D4	D3	D2	D1	D0
4E8H	**TK11H**	D15	D14	D13	D12	D11	D10	D9	D8
4E7H	**TK12L**	D7	D6	D5	D4	D3	D2	D1	D0
4E6H	**TK12H**	D15	D14	D13	D12	D11	D10	D9	D8
4E5H	**TK13L**	D7	D6	D5	D4	D3	D2	D1	D0
4E4H	**TK13H**	D15	D14	D13	D12	D11	D10	D9	D8
4E3H	**TK14L**	D7	D6	D5	D4	D3	D2	D1	D0
4E2H	**TK14H**	D15	D14	D13	D12	D11	D10	D9	D8
4E1H	**TK15L**	D7	D6	D5	D4	D3	D2	D1	D0
4E0H	**TK15H**	D15	D14	D13	D12	D11	D10	D9	D8
-	**R/W**	R	R	R	R	R	R	R	R
-	**复位值**	X	X	X	X	X	X	X	X

BIT[15:0] **TKnL/H（n = 0-15）[15:0]** – 16 个 TK 触摸按键的 16 位扫描结果

注 1：TK 扫描结果保存在 RAM 中，且为只读，掉电后数据会丢失。当 TK 一次扫描完成后，建议及时取出需要的结果，以免当前的结果被后一次扫描的结果覆盖。

*注 2：**芯片复位后，TK 结果寄存器地址中保存的值将为随机值。***

15 中断

芯片的中断包括外部中断 INT0（有 5 个端口共用该中断源）、INT1（有 5 个端口共用该中断源）、定时器中断（T0、T1、T2、T3）、ADC 转换中断、PWM0 中断、PWM1 中断、UART0 中断、UART1 中断、SPI 中断、PWM1 刹车保护中断（PWM1FB）、CRC 中断共 14 个中断源，分为 2 个中断优先级。

每个中断均具有独立的中断标志位，中断使能位、中断向量和优先级设置位，还可以通过 1 个全局中断使能位将所有中断同时禁止。

15.1 中断源列表

芯片的中断源及其中断向量、相关控制位等列表如下：

中断源	触发条件	中断标志	中断使能位	优先级设置位	中断向量	中断号(C51)	查询优先级	中断标志清除方式	停止模式唤醒	休眠模式唤醒
Reset	-	-	-	-	0000H	-	0(最高)	-	Yes	Yes
INT0	上升沿/下降沿/双沿	EXT0IF4/ EXT0IF3/ EXT0IF2/ EXT0IF1/ EXT0IF0	EXT0IE	EXT0IP	0003H	0	1	软件清除	Yes	Yes
T0	T0 溢出	TF0	T0IE	T0IP	000BH	1	2	硬件自动	Yes	No
INT1	上升沿/下降沿/双沿	EXT1IF4/ EXT1IF3/ EXT1IF2/ EXT1IF1/ EXT1IF0	EXT1IE	EXT1IP	0013H	2	3	软件清除	Yes	Yes
T1	T1 溢出	TF1	T1IE	T1IP	001BH	3	4	硬件自动	Yes	No
UART0	UART0 收/发完成	RI0/TI0	UART0IE	UART0IP	0023H	4	5	软件清除	Yes	No
SPI	SPI 收发完成/模式故障	SPIIF/ SSDIS=0 且 MODF=1	SPIIE	SPIIP	002BH	5	6	软件清除	Yes	No
ADC	ADC 转换完成	ADCIF	ADCIE	ADCIP	0033H	6	7	软件清除	Yes	No
T2	T2 溢出、比较/T2 重载、捕获	TF2/ EXTF2	T2IE	T2IP	003BH	7	8	软件清除	Yes	No
T3	T3 溢出	TF3	T3IE	T3IP	0043H	8	9	硬件自动	Yes	Yes
PWM1	PWM1 溢出	PWM1IF	PWM1IE	PWM1IP	004BH	9	10	硬件自动	Yes	No
UART1	UART1 收/发完成	RI1/TI1	UART1IE	UART1IP	0053H	10	11	软件清除	Yes	No
PWM0	PWM0 溢出	PWM0IF	PWM0IE	PWM0IP	005BH	11	12	硬件自动	Yes	No
PWM1FB	PWM1FB 上升沿/下降沿	PWM1FBIF	PWM1FBIE	PWM1FBIP	0063H	12	13	软件清除	Yes	No
TK	TK 扫描完成/扫描异常	TKIF (TKAVGIF /TKCNTIF/ TKOVIF/ TKERRIF)	TKIE	TKIP	006BH	13	14	软件清除	Yes	Yes
CRC	CRC 校验完成	CRCIF	-	CRCIP	0073H	14	15(最低)	软件清除	No	No

15.2 中断优先级

芯片的中断具有 2 个中断优先级，中断源的中断请求可设定为高优先级或低优先级，通过优先级的设定可实现中断服务程序的 2 级嵌套。

低优先级的中断服务程序能被高优先级请求所中断，反之则不行；中断服务程序不能被同一级的中断请求所中断。

当多个同一优先级的中断同时产生中断请求时，根据预定的查询优先级顺序响应。

15.3 中断处理流程

当一个中断产生并且被芯片响应，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 值出栈，并返回执行中断前的程序继续执行。

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，再执行一条指令，之后转去执行新的中断请求。

当用户开启优先级中断，高优先级中断可以打断低优先级的中断，本芯片可以实现 2 级中断嵌套。

15.4 外部中断

本芯片有 2 个外部中断源 INT0 和 INT1，每个中断源分别有 5 路外部输入，对应端口 INTxy（xy = 0-1；y = 0-4）。外部中断为边沿触发，配置控制位 ExnIS[1:0]（n=0-1)选择不同的触发沿（不触发/上升沿/下降沿/双沿）。用户可以通过 EXTEN/ EXTEN1 寄存器灵活地在 5 个口中选择一个或多个端口作为输入，端口 INT0y（y=0-4）复用同一个中断源 INT0，端口 INT1y（y=0-4）复用同一个中断源 INT1。

若设置触发沿为下降沿，一个周期内引脚输入连续采样为高电平，下个周期采样为低电平，则对应中断标志置起，发出中断请求；若多个中断请求同时发生，同一个中断源只响应一个中断请求，用户可以在中断处理程序中进行标志判断区分；中断标志需要软件清零，未清零标志，硬件将会在中断退出后再次发出中断请求。

同一个中断源的 5 路输入，进入中断处理程序，不影响新的中断标志建立，只是不会响应新的中断请求。

注：外部中断输入通道内建最小 100ns 模拟滤波，高温低压条件下滤波偏大（不超过 500ns）。外部中断沿信号的高低电平必须大于（1 个系统时钟周期+模拟滤波时间），才能保证有效的触发。

外部中断使能寄存器（0xAC，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTEN	-	EXT0EN4	EXT0EN3	EXT0EN2	EXT0EN1	EXT0EN0	-	EXT0A
R/W	-	R/W	R/W	R/W	R/W	R/W	-	R/W
复位值	-	0	0	0	0	0	-	0

外部中断使能寄存器 1（0xAB，SFR1）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTEN1	-	EXT1EN4	EXT1EN3	EXT1EN2	EXT1EN1	EXT1EN0	-	EXT1A
R/W	-	R/W	R/W	R/W	R/W	R/W	-	R/W
复位值	-	0	0	0	0	0	-	0

BIT[6:2] **EXTnENm（n = 0-1; m = 0-4）** – 外部中断 INTxy（x = 0-1；y = 0-4）端口使能位

0：禁止对应某一路 INTxy（x = 0-1；y = 0-4）端口的复用功能，端口用于通用 IO 或其他复用模块；

1：使能对应某一路 INTxy（x = 0-1；y = 0-4）端口的复用功能，端口作为外部中断输入用。

注 1：要使能对应的 INTxy（x = 0-1；y = 0-4）端口，还必须先使能 EXTnA（n = 0-1）位。中断源 INTx（x = 0-1）一次只允许一个外部中断信号（五选一）通过端口进入芯片中断系统，输入信号有效后将产生对应的 EXTnIFm

（n = 0-1; m = 0-4）标志位置 1。

注 2：端口的外部中断功能使能后，其上拉电阻自动无效。

注 3：当使用 T0/T1 测量电平宽度时，T0/T1 不能用作他用，且只通过 INT04/ INT14，即这两个外设联动工作，资源同时被使用。此时 INT04 使用 T0，计算的方式可对应 T0 的方式 0-3，四种方式；而 INT14 使用 T1，但计算的方式可对应 T1 的方式 0-2，仅三种方式。

BIT[0] **EXTnA （n = 0-1）** – 外部中断 INTxy（xy = 0,1；y = 0-4）总使能位

0：禁止所有物理 INTxy（xy = 0-1；y = 0-4）端口上外部中断功能，即关闭 INTx 模块；

1：使能所有物理 INTx（xy = 0-1；y = 0-4）端口上外部中断功能，即打开 INTx 模块。

外部中断控制寄存器（0xAC，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTIS	-	EXT0IF4	EXT0IF3	EXT0IF2	EXT0IF1	EXT0IF0	EXT0IS1	EXT0IS0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	0	0	0	0	0	0	0

外部中断控制寄存器 1（0xAB，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXTIS1	-	EXT1IF4	EXT1IF3	EXT1IF2	EXT1IF1	EXT1IF0	EXT1IS1	EXT1IS0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	0	0	0	0	0	0	0

BIT[6:2] **EXTnIFm（n = 0-1; m = 0-4）** – 外部中断 INTxy（x = 0,1；y = 0-4）标志位

0：在对应物理 INTxy（x = 0,1；y = 0-4）端口上没有外部中断产生，由软件清 0；

1：在对应物理 INTxy（x = 0,1；y = 0-4）端口上有外部中断产生，由硬件置 1，若同时 EXnIE（n = 0,1）位置位，同时将产生 INTx（x = 0,1）中断，则芯片会响应该中断请求。

注 1：对 EXT0Ifm（m = 0～4），对应于 INT0 中断源；EXT1Ifm（m = 0～4），对应于 INT1 中断源；每个物理端口标志独立建立，但每个中断源的 5 个子中断口并不会嵌套，同一个中断源不会同时响应多个中断请求。

注 2：这些标志位置 1 后需要软件清 0。

BIT[1:0] **EXTnIS[1:0] （n = 0-1）** – 外部中断 INTxy（x = 0,1；y = 0-4）触发方式选择位

EXnIS[1:0]（n = 0-1）	INTxy（x = 0,1；y = 0-4）的端口触发方式
00	不触发
01	下降沿触发
10	上升沿触发
11	双沿触发

注：同一个中断源 INTx（x = 0-1）而言，对 INTxy（x = 0,1；y = 0-4）这 5 个端口将使用共同的触发方式，不可单独设置。

EXnISm[1:0]	在使用 T0/T1 脉冲宽度测量方式下，INT04/INT14 的端口触发方式
XX	始终有效（用于测量高电平的脉冲宽度）

注：在使用 T0/T1 脉冲宽度测量方式下，仅 INT04 和 INT14 两个端口可用，且 ExnISm[1:0]控制位不参与控制，此时可用于测量高电平的脉冲宽度；在 INT04 对应于 T0 的资源，INT14 则对应于 T1 的资源。

15.5 中断相关寄存器

中断寄存器汇总表

地址 (SFR n/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值 (B)

SFR0 /BB	**IPH1**	中断优先级控制 1 高位寄存器	CRCIPH	TKIPH	PWM1FBIPH	PWM0IPH	UART1IPH	PWM1IPH	T3IPH	T2IPH	00000000
SFR0 /BA	**IP1**	中断优先级控制 1 低位寄存器	CRCIPL	TKIPL	PWM1FBIPL	PWM0IPL	UART1IPL	PWM1IPL	T3IPL	T2IPL	00000000
SFR0 /B9	**IPH**	中断优先级控制高位寄存器	-	ADCIPH	SPIIPH	UART0IPH	T1IPH	EXT1IPH	T0IPH	EXT0IPH	-0000000
SFR0 /B8	**IP**	中断优先级控制低位寄存器	-	ADCIPL	SPIIPL	UART0IPL	T1IPL	EXT1IPL	T0IPL	EXT0IPL	-0000000
SFR0 /AC	**EXTIS**	外部中断控制寄存器	-	EXT0IF4	EXT0IF3	EXT0IF2	EXT0IF1	EXT0IF0	EXT0IS1	EXT0IS0	-00000--
SFR1 /AC	**EXTEN**	外部中断使能寄存器	-	EXT0EN4	EXT0EN3	EXT0EN2	EXT0EN1	EXT0EN0	-	EXT0A	-00000-0
SFR0 /AB	**EXTIS1**	外部中断控制寄存器 1	-	EXT1IF4	EXT1IF3	EXT1IF2	EXT1IF1	EXT1IF0	EXT1IS1	EXT1IS0	-00000--
SFR1 /AB	**EXTEN1**	外部中断使能寄存器 1	-	EXT1EN4	EXT1EN3	EXT1EN2	EXT1EN1	EXT1EN0	-	EXT1A	-00000-0
SFR0 /A9	**IE1**	中断使能寄存器 1	CRCIE	TKIE	PWM1FBIE	PWM0IE	UART1IE	PWM1IE	T3IE	T2IE	00000000
SFR0 /A8	**IE**	中断使能寄存器	EA	ADCIE	SPIIE	UART0IE	T1IE	EXT1IE	T0IE	EXT0IE	00000000

中断使能寄存器（0xA8，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IE	EA	ADCIE	SPIIE	UART0IE	T1IE	EXT1IE	T0IE	EXT0IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] **EA** – 总中断使能位

0：关闭所有中断；

1：打开中断，但具体中断使能控制由相应的控制位决定。

BIT[6] **ADCIE** – ADC 中断使能位

0：关闭 ADC 中断；

1：打开 ADC 中断。

BIT[5] **SPIIE** – SPI 中断允许控制位

0：禁止 SPI 中断；

1：允许 SPI 中断，当标志位 SPIIF=1 直接产生正常中断或者 SSDIS=0 且 MODF=1 时将产生 SPI 错误中断。

BIT[4] **UART0IE** – UART0 中断使能位

0：关闭 UART0 中断；

1：打开 UART0 中断。

BIT[3] **T1IE** – T1 中断使能位

0：关闭 T1 中断；

1：打开 T1 中断。

BIT[2] **EXT1IE** – INT1 中断使能位

0：关闭 INT1 中断；

1：打开 INT1 中断。

BIT[1] **T0IE** – T0 中断使能位

0：关闭 T0 中断；

1：打开 T0 中断。

BIT[0] **EXT0IE** – INT0 中断使能位

0：关闭 INT0 中断；

1：打开 INT0 中断。

中断使能寄存器 1（0xA9，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IE1	CRCIE	TKIE	PWM1FBIE	PWM0IE	UART1IE	PWM1IE	T3IE	T2IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7] **CRCIE** – CRC 中断使能位

0：禁止 CRC 中断；

1：允许 CRC 中断。

BIT[6] **TKIE** – TK 中断使能位

0：禁止 TK 中断；

1：允许 TK 中断。

BIT[5] **PWM1FBIE** – PWM1 阵列刹车保护中断使能位

0：禁止 PWM1 刹车保护中断；

1：允许 PWM1 刹车保护中断。

BIT[4] **PWM0IE** – PWM0 中断使能位

0：禁止 PWM0 中断；

1：打开 PWM0 中断。

BIT[3] **UART1IE** – UART1 中断使能位

0：关闭 UART1 中断；

1：打开 UART1 中断。

BIT[2] **PWM1IE** – PWM1 总中断使能位

0：禁止 PWM1 总中断；

1：打开 PWM1 总中断。

BIT[1] **T3IE** – T3 中断使能位

0：关闭 T3 中断；

1：打开 T3 中断。

BIT[0] **T2IE** – T2 中断使能位

0：关闭 T2 中断；

1：打开 T2 中断。

注：该中断源可同时受 TF2 和 EXTF2 两个中断标志位影响，但原则上 T2 的这两个功能不可同时使用。

中断优先级控制低位寄存器（0xB8，SFR0）和中断优先级控制高位寄存器（0xB9，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP	-	ADCIPL	SPIIPL	UART0IPL	T1IPL	EXT1IPL	T0IPL	EXT0IPL
IPH	-	ADCIPH	SPIIPH	UART0IPH	T1IPH	EXT1IPH	T0IPH	EXT0IPH
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	-	0	0	0	0	0	0	0

中断优先级控制 1 低位寄存器（0xBA，SFR0）和中断优先级控制 1 高位寄存器（0xBB，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP1	CRCIPL	TKIPL	PWM1FBIPL	PWM0IPL	UART1IPL	PWM1IPL	T3IPL	T2IPL
IPH1	CRCIPH	TKIPH	PWM1FBIPH	PWM0IPH	UART1IPH	PWM1IPH	T3IPH	T2IPH
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **IPH(1) /IP(1)** – 中断优先级控制位

IPH/IPH1	IP/IP1	中断优先级等级
0	0	等级 0（低）
0	1	等级 1（高）
1	0	禁止
1	1	

注：“高优先级中断”可以打断正在执行的“低优先级中断”，实现 1 层中断嵌套；在同一优先级情况下则按其“中断源列表”上的顺序排列。特别注意，一个中断产生后，若用户不进行处理，相同优先级的任何新中断将不会再产生。

16 其他辅助功能

16.1 双 DPTR

传统 8051 架构仅有一组 DPTR（数据指针寄存器）。DPTR 本质是一个 16 位的特殊功能寄存器，其高位字节寄存器用 DPH 表示，低位字节寄存器用 DPL 表示。

本芯片重新设计了架构，升级了内核，增加了另一组数据指针 DPTR1，其高位字节寄存器用 DPH1 表示，低位字节寄存器用 DPL1 表示。这样的做法，既可以兼容了老的 51 代码，软件还能利用提供的两组数据指针 DPTR 及 DPTR1，进一步加快程序的执行速度同时缩减代码量。

数据指针低字节寄存器（0x82，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

数据指针高字节寄存器（0x83，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPH	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

数据指针 1 低字节寄存器（0x84，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPL1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

数据指针 1 高字节寄存器（0x85，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPH1	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

数据指针选择寄存器（0xD8，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPS	-	SFRSL0	-	-	DIV16	MUL16	-	SEL
R/W	-	R/W	-	-	R/W	R/W	-	R/W
复位值	-	0	-	-	0	0	-	0

BIT[0] **SEL** – DPTR/DPTR1 选择控制位

0：指令“MOVC/MOVX @DPTR”中的操作对象为 DPTR；

1：指令“MOVC/MOVX @DPTR”中的操作对象为 DPTR1。

CORE 切换 DPTR，汇编代码应用举例：

```
……//假设当前使用的是 DPTR
    INC DPS;          //切换至 DPTR1
……
    INC DPS;          //再切换回 DPTR
```

16.2 软件 LCD 驱动

本芯片的部分 IO 口，可以输出 1/2 VDD 电压，作为软件 LCD 的 COM 口使用。

用户可以根据实际情况配置成需要的 COM 数，最大可配置成 24 个 COM 口。

GPIO 与 COM 对应表

GPIO	COMx	GPIO	COMx	GPIO	COMx
P1.0	COM0	P0.1	COM8	P3.1	COM16
P1.1	COM1	P0.0	COM9	P3.0	COM17
P1.2	COM2	P2.7	COM10	P0.2	COM18
P1.3	COM3	P2.6	COM11	P0.3	COM19
P1.4	COM4	P2.3	COM12	P0.4	COM20
P1.5	COM5	P2.2	COM13	P0.5	COM21
P1.6	COM6	P2.1	COM14	P0.6	COM22
P1.7	COM7	P2.0	COM15	P0.7	COM23

注:LCD 复用优先级最低,仅高于 GOIO

LCD 控制寄存器 0（0xD9，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LCDCR0	LCDEN	-	-	-	-	-	-	-
R/W	R/W	-	-	-	-	-	-	-
复位值	0	-	-	-	-	-	-	-

BIT[7] **LCDEN** – LCD 模块使能位

0：关闭 LCD 模块；

1：打开 LCD 模块。

LCDEN	LCD 电压输出口分压电阻选择
0	关闭内部分压电阻
1	内部分压电阻=20K

LCD 软件控制寄存器（0xDA，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LCDSW	COM7EN	COM6EN	COM5EN	COM4EN	COM3EN	COM2EN	COM1EN	COM0EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **COMnEN（n=0~7）** – 使能 COMn 口（n=0~7）的 1/2VDD 功能

0：禁止软件 LCD 输出功能，端口用于通用 IO 或其他复用模块；

1：打开软件 LCD 输出功能，端口输出 1/2 VDD 电压。

注：打开软件 LCD 功能后，对应 COMn 口 (n=0~7) 的电平将变化成 1/2 VDD，且不会反映在端口数据寄存器上。

LCD 软件控制寄存器（0xDB，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LCDSW1	COM15EN	COM14EN	COM13EN	COM12EN	COM11EN	COM10EN	COM9EN	COM8EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **COMnEN（n=8~15）** – 使能 COMn 口（n=8~15）的 1/2VDD 功能

0：禁止软件 LCD 输出功能，端口用于通用 IO 或其他复用模块；

1：打开软件 LCD 输出功能，端口输出 1/2 VDD 电压。

注：打开软件 LCD 功能后，对应 COMn 口（n=8~15）的电平将变化成 1/2 VDD，且不会反映在端口数据寄存器上。

LCD 软件控制寄存器（0xDC，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LCDSW2	COM23EN	COM22EN	COM21EN	COM20EN	COM19EN	COM18EN	COM17EN	COM16EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **COMnEN（n=16~23）** – 使能 COMn 口（n=16~23）的 1/2VDD 功能

0：禁止软件 LCD 输出功能，端口用于通用 IO 或其他复用模块；

1：打开软件 LCD 输出功能，端口输出 1/2 VDD 电压。

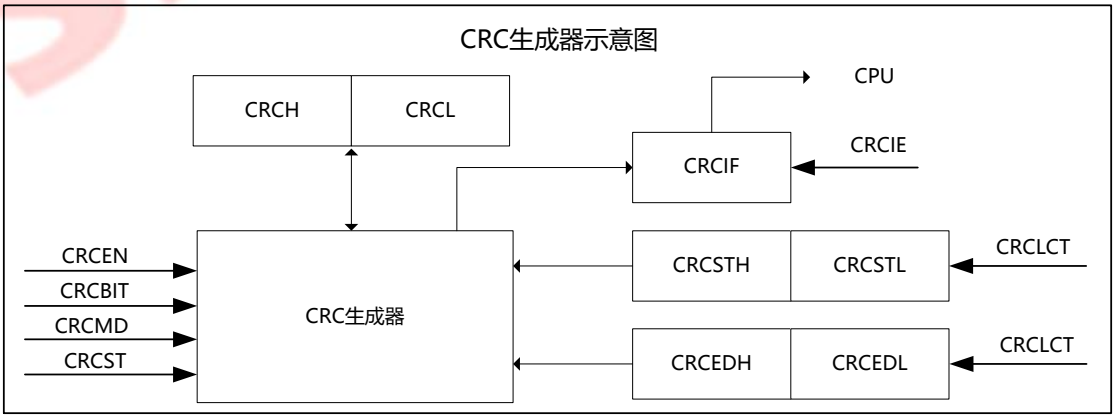
注：打开软件 LCD 功能后，对应 COMn 口（n=16~23）的电平将变化成 1/2 VDD，且不会反映在端口数据寄存器上。

16.3 循环冗余校验（CRC）

循环冗余校验(Cyclic Redundancy Check, CRC)是一种根据输入数据产生简短固定位数校验码的一种散列函数，主要用来提高系统的可靠性。

16.3.1 循环冗余校验（CRC）特性

集成硬件 16 位 CRC 单元，符合 CRC-16 标准，生成多项式为 $X_{16} + X_{15} + X_2 + 1$，即 0x8005，初值可设为任意值，默认为 0xFFFF。



16.3.2 循环冗余校验（CRC）相关寄存器

CRC 控制寄存器（0xD1，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCCR	CRCEN	-	CRCIF	CRCST	CRCMD	CRCLCT	-	CRCENC
R/W	R/W	-	R/W	R/W	R/W	R/W	-	R/W
复位值	0	-	0	0	0	0	-	0

BIT[7] **CRCEN** – CRC 模块使能控制位

0：CRC 模块关闭；

1：CRC 模块打开。

BIT[5] **CRCIF** – CRC 中断标志位

0：未启动或启动 CRC 校验后未完成，由软件清 0；

1：本次 CRC 校验计算完成，由硬件置 1，若同时 CRCIE 位置位，将产生 CRC 中断，则芯片会响应该中断请求。

BIT[4] **CRCST** – CRC 校验计算启动位

0：本次 CRC 校验完成，由硬件清 0；

1：启动一次 CRC 校验或者处于 CRC 校验计算当中，由软件置 1。

注：写的话，0 可以写入，但无实际意义；读的话，该位根据 CRC 模块的实际状态返回 1/0。

BIT[3] **CRCMD** – CRC 校验模式选择位

0：对整个 FLASH ROM 区域或者整个 EEPROM 区域做 CRC 校验；

1：保留位，禁止使用。

注：此位必须设置为 0。

BIT[2] **CRCLCT** – CRC 校验区域选择位

0：对 FLASH ROM 区域做 CRC 校验；

1：对 EEPROM 区域做 CRC 校验。

注:这里 FLASH ROM 有效区域指 0x0000~0x3FFF 的地址范围;EEPROM 有效区域指 0x000~0x3FF 的地址范围。

BIT[0] **CRCENC** – CRC 加密相关控制位

0：应用于 FLASH 无加密的状态，对指定起始地址和结束地址的区域做 CRC 校验；

1：应用于 FLASH 有加密的状态，对起始地址所在扇区进行 CRC 校验，结束地址无效。

注:对于 FLASH 不加密的区域(CRCENC=0)做 CRC 校验时,用户可以指定任意的起始地址和结束地址;对于 FLASH 加密的区域（CRCENC=1）做 CRC 校验时，用户只要指定一个起始地址即可，无需指定结束地址，芯片会自动对当前地址所在的扇区做 CRC 校验，并给出 CRC 计算结果，此时结束地址并不参与本次 CRC 的运算。

CRC 结果低字节寄存器（0xD2，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCL	CRCL7	CRCL6	CRCL5	CRCL4	CRCL3	CRCL2	CRCL1	CRCL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

BIT[7:0] **CRCL[7:0]** – CRC 结果低字节

CRC 结果高字节寄存器（0xD3，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCH	CRCH7	CRCH6	CRCH5	CRCH4	CRCH3	CRCH2	CRCH1	CRCH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	1	1	1	1	1	1	1	1

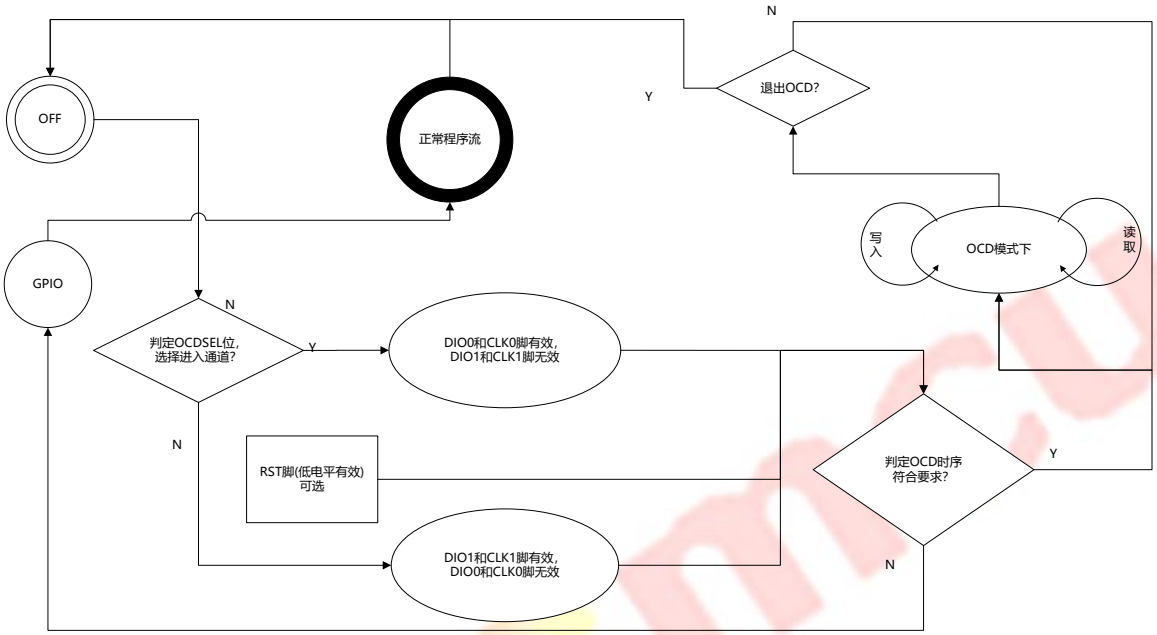
BIT[7:0] **CRCH[7:0]** – CRC 结果高字节

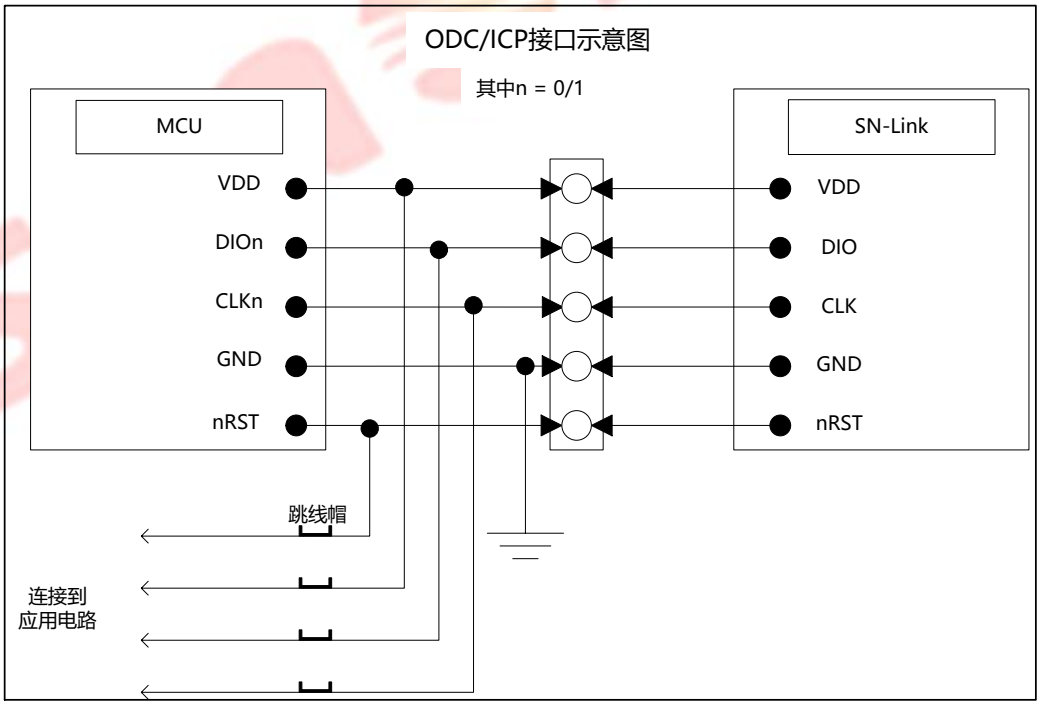
注：CRC[15:0]的内容可以在 CRC 校验启动前，由用户写入任意值，作为 CRC 初值，默认为 0xFFFF；其内容，在 CRC 校验计算过程中，不允用户修改，否则 CRC 结果将不可预知。

17 片上调试OCD和在电路编程ICP接口

本芯片内嵌片上调试（OCD）功能，这为用户提供了低成本的调试方法，并且本芯片充分考虑了用户的需求，提供了精简的 2 线调试接口电路，以及自动二选一的两组调试实物接口（在用户配置字中设定，默认为第一组调试接口）。当一组连接后，另一组自动失效。在引脚复用的场合可以真正做到全仿真。

ODC状态机示意图





片上调试 OCD 和在电路编程 ICP 接口整理

名称	对应 PIN	名称	对应 PIN	要求
电源正	VDD	电源正	VDD	必须

DIO0	P3.1	DIO1	P1.7	（二选一）
CLK0	P3.0	CLK1	P1.5	必须
nRST	P0.2	nRST	P0.2	可选
电源负	GND	电源负	GND	必须

注： DIOn 和 CLKn（n=0、1）的选择与“用户”实际电路预留调试的接口有关，正常情况下选择其中一对使用即可。

18 EEPROM的IAP操作

片上 EEPROM 数据存储器，支持用户程序在带电工作中实时地读出或写入数据。当然，用户也可以通过相关的工具，由外部直接操作 EEPROM，其原理和操作 FLASH 类似。在读取数据时，利用 MOVC 指令间接寻址，有一个 EERD 标志位进行区分读取的是 FLASH 区，还是 EEPROM 区；而写入数据时，则通过操作地址寄存器 IAPAD、数据寄存器 IAPDB 进行。因为本芯片使用的是独立 EERPOM 编址，用户可以在正常程序的执行中，同时擦/写 EEPROM，以提高芯片的执行效率。

EEPROM 每 16 字节为 1 页，合计 1024 字节的 EEPROM 共分 64 页，用户可对 EEPROM 进行全空间擦除、页擦除或页内 1~16 字节的编程操作。

EEPROM 编程之前需将对应地址内的原数据擦除，才能写入新数据。即使写入相同内容也需要先擦除再编程，否则不能保证数据正确。

数据寄存器 IAPDB 为数据锁存器，对 IAPDB 的“写”操作实际将数据写入数据锁存区的对应地址中。

EEPROM 在无操作时，将自动进入待机状态，以降低芯片功耗；任意操作都将使其自动恢复其至正常的工作状态。

18.1 EEPROM 相关寄存器

EEPROM 操作寄存器汇总表

地址(SFRn/H)	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值(B)
SFR0/EF	**IAPKEY**	IAP 写保护寄存器	IAPKEY7	IAPKEY6	IAPKEY5	IAPKEY4	IAPKEY3	IAPKEY2	IAPKEY1	IAPKEY0	00000000
SFR0/EE	**IAPMD**	IAP 模式寄存器	IAPEN	-	-	-	ECHIPERS	EPAGEERS	EPAGEWR	-	0---000-
SFR0/EC	**IAPDB**	IAP 数据寄存器	IAPDB7	IAPDB6	IAPDB5	IAPDB4	IAPDB3	IAPDB2	IAPDB1	IAPDB0	00000000
SFR0/EB	**IAPADH**	IAP 高地址寄存器	IAPAD15	IAPAD14	IAPAD13	IAPAD12	IAPAD11	IAPAD10	IAPAD9	IAPAD8	00000000
SFR0/EA	**IAPADL**	IAP 低地址寄存器	IAPAD7	IAPAD6	IAPAD5	IAPAD4	IAPAD3	IAPAD2	IAPAD1	IAPAD0	00000000
SFR0/E9	**IAPCR**	IAP 控制寄存器	IAPSTR	-	-	-	IAPTRIG2	IAPTRIG1	IAPTRIG0	EERD	0---0000

IAP 模式寄存器（0xEE，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPMD	IAPEN	-	-	-	ECHIPERS	EPAGEERS	EPAGEWR	-
R/W	R/W	-	-	-	R/W	R/W	R/W	-
复位值	0	-	-	-	0	0	0	-

BIT[7] **IAPEN** – IAP 使能位

0：关闭 IAP 功能（写一遍有效）；

1：打开 IAP 功能（连续写两遍有效）。

注：只有对该位连续写 2 条相同的指令，才能使系统进行打开 IAP 使能位，否则下个时钟周期到来后，寄存器位中先前写入的值将被硬件自动清掉，保持为 0，这是一种简单的保护机制；关闭时写一次 0 即可。

BIT[3] **ECHIPERS** – EEPROM 全空间擦除模式位

0：无（写一遍有效）；

1：EEPROM 全空间擦除模式（连续写两遍有效）。

BIT[2] **EPAGEERS** – EEPROM 页擦除模式位

0：无（写一遍有效）；

1：EEPROM 页擦除模式（连续写两遍有效）。

BIT[1] **EPAGEWR** – EEPROM 页编程（写入）模式位

0：无（写一遍有效）；

1：EEPROM 页编程模式（连续写两遍有效）。

注：特别注意，IAPMD 寄存器中的每一位置 1 的话，都需要连续写 2 次操作。另外，以上 3 种 EEPROM 的操作模式，在同一时刻，本芯片只允许进入一种模式执行，否则芯片会出现不可预知的情况。

IAP 写保护寄存器（0xEF，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPKEY	IAPKEY7	IAPKEY6	IAPKEY5	IAPKEY4	IAPKEY3	IAPKEY2	IAPKEY1	IAPKEY0
R/W	W	W	W	W	W	W	W	W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **IAPKEY[7:0]** – IAP 保护锁

IAP 保护用，当用户在启动 IAP 操作的时候，必须先向其写入 0x5A，紧接着写 0xA5，解除 IAP 保护，对 IAPCR 的操作才有效；否则本次操作视为无效。

注：IAPKEY 为只写寄存器，不可读取其值，其保护的保护对象为 IAPCR 寄存器中的 IAPSTR 位和 IAPTRIG[2:0]位。

IAP 数据寄存器（0xEC，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPDB	IAPDB7	IAPDB6	IAPDB5	IAPDB4	IAPDB3	IAPDB2	IAPDB1	IAPDB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[7:0] **IAPDB[7:0]** – IAP 写入数据。

IAP 高地址寄存器（0xEB，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPADH	IAPAD15	IAPAD14	IAPAD13	IAPAD12	IAPAD11	IAPAD10	IAPAD9	IAPAD8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[5:0] **IAPAD[15:8]** – IAP 写入地址的高 8 位。

IAP 低地址寄存器（0xEA，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPADL	IAPAD7	IAPAD6	IAPAD5	IAPAD4	IAPAD3	IAPAD2	IAPAD1	IAPAD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位值	0	0	0	0	0	0	0	0

BIT[3:0] **IAPAD[7:0]** – IAP 写入地址的低 8 位。

注：因为本芯片的 EEPROM 的容量为 1024 bytes，所以可操作的范围应该在 0～0x3FF 之间，对 IAPAD 寄存器[15:10]位的操作是禁止的。

IAP 控制寄存器（0xE9，SFR0）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPCR	IAPSTR	-	-	-	IAPTRIG2	IAPTRIG1	IAPTRIG0	EERD

R/W	R/W	-	-	-	R/W	R/W	R/W	R/W
复位值	0	-	-	-	0	0	0	0

BIT[7] **IAPSTR** – IAP 操作启动控制位
0：由硬件清 0，表示当前不在 IAP 流程中，或者 IAP 流程已结束；
1：写 1 后启动一次 IAP 操作流程。

BIT[3] **IAPTRIG2** – IAP 操作执行控制位 2
0：由硬件清 0，无 IAP 操作，或者当前 IAP 操作已结束；
1：写 1 后，硬件开始执行一次 IAP 操作。

BIT[2] **IAPTRIG1** – IAP 操作执行控制位 1
0：由硬件清 0，无 IAP 操作，或者当前 IAP 操作已结束；
1：写 1 后，硬件开始执行一次 IAP 操作。

BIT[1] **IAPTRIG0** – IAP 操作执行控制位 0
0：由硬件清 0，无 IAP 操作，或者当前 IAP 操作已结束；
1：写 1 后，硬件开始执行一次 IAP 操作。

注：特别注意，为了防止芯片误动作，以上四个位的设置需要配合 IAPKEY 寄存器的设置联动操作，否则设置无效，详细可以参考“IAP 操作标准流程”。

BIT[0] **EERD** – 读取 EEPROM 区操作选择位
0：由软件清 0，用 MOVC 指令读取 FLASH 中的内容；
1：由软件置 1，用 MOVC 指令读取 EEPROM 中的内容。

注 1：因 FLASH 和 EEPROM 地址重叠，芯片通过该位识别 MOVC 指令访问的是 EEPROM 区还是 FLASH 区。

注 2：当用 MOVC 指令读取“用户配置字”中设置了代码区加密功能时（ENCR 位=1），FLASH 区读取到的内容将为零。

注 3:EERD 位仅与 EEPROM 的读操作有关,与擦/写操作无关,即对 EEPROM(ECHIPERS / EPAGEERS / EOAGEWR =1）的 IAP 操作，无视该位的值。

18.2 EEPROM 的 IAP 操作标准流程

1．EEPROM 内容读取的标准操作流程，如下：

(1) 首先，关闭中断；若使用看门狗的话，必须将看门狗溢出时间调整到足够大；
(2) 将 EERD 置 1；
(3) 用 MOVC 指令读取对应的 EEPROM 地址，若需要可以重复操作多次；
(4) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。

2．FLASH 内容读取的标准操作流程，如下：

(1) 首先，关闭中断；若使用看门狗的话，必须将看门狗溢出时间调整到足够大；
(2) 将 EERD 置 0（默认情况下）；
(3) 用 MOVC 指令读取对应的 FLASH 地址，若需要可以重复操作多次；
(4) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。

3．EEPROM 全空间擦除的标准操作流程，如下：

(1) 首先，关闭中断；若使用看门狗的话，必须将看门狗溢出时间调整到足够大；
(2) 将 IAPEN 位重复执行 2 次置 1 操作，将 ECHIPERS 重复执行 2 次置 1 操作；
(3) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPSRT 位置 1；
(4) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG0 位置 1；
(5) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG1 位置 1；
(6) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG2 位置 1；
(7) 若一切正常，在 EEPROM 全擦除的典型时间后，原本置 1 的 IAPSTR 位将硬件清 0，表示本次 EEPROM 全

擦除成功，并退出本次 IAP 流程，用户可以通过查询该标志位来确认本次操作的完成情况；

(8) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。

4．EEPROM 页擦除（1~16 个字节可选择）的标准操作流程，如下：

(1) 首先，关闭中断；若使用看门狗的话，必须将看门狗溢出时间调整到足够大；

(2) 将 IAPEN 位重复执行 2 次置 1 操作，将 EPAGEERS 重复执行 2 次置 1 操作；

(3) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPSRT 位置 1；

(4) 向 IAPADH 中写入预备页擦除的 EEPROM 字节的地址高 8 位；

(5) 向 IAPADL 中写入预备页擦除的 EEPROM 字节的地址低 8 位；

(6) 向 IAPDB 中写入 0 值；

(7) 重复（4）至（6）三个步骤，直至选择完毕所有需要擦除的字节；

(8) 将 IAPADL 的低 4 位清 0；

(9) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG0 位置 1；

(10) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG1 位置 1；

(11) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG2 位置 1；

(12) 若一切正常，在 EEPROM 页擦除的典型时间后，原本置 1 的 IAPSTR 位将硬件清 0，表示本次 EEPROM 页擦除成功，并退出本次 IAP 流程，用户可以通过查询该标志位来确认本次操作的完成情况；

(13) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。

5．EEPROM 页写入（1~16 个字节可选择）的标准操作流程，如下：

(1) 首先，关闭中断，将看门狗溢出时间调整到足够大；

(2) 将 IAPEN 位重复执行 2 次置 1 操作，将 EPAGEWR 重复执行 2 次置 1 操作；

(3) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPSRT 位置 1；

(4) 向 IAPADH 中写入预备页写入的 EEPROM 字节的地址高 8 位；

(5) 向 IAPADL 中写入预备页写入的 EEPROM 字节的地址低 8 位；

(6) 向 IAPDB 中写入预备页写入的新数值；

(7) 重复（4）至（6）三个步骤，直至选择完毕所有需要写入的字节；

(8) 将 IAPADL 的低 4 位清 0；

(9) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG0 位置 1；

(10) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG1 位置 1；

(11) 对 IAPKEY[7:0]写入 0x5A，再写入 0xA5，解锁后对 IAPTRIG2 位置 1；

(12) 若一切正常，在 EEPROM 页擦除的典型时间后，原本置 1 的 IAPSTR 位将硬件清 0，表示本次 EEPROM 页写入成功，并退出本次 IAP 流程，用户可以通过查询该标志位来确认本次操作的完成情况；

(13) 完成后，退出本次 IAP 操作，打开中断；若使用看门狗的话，复原看门狗溢出时间。

注 1： EEPROM 的多字节操作时，1～16 个数据必须在同一页，即除地址的最低 4 位以外的其它地址位必须相同，否则会发生错误；

注 2：对 EEPROM 进行 IAP 的同时，芯片可继续执行；

注 3：芯片复位后，需等待至少 20uS 后才能进行 EEPROM 以及 IAP 的正常操作；

注 4：若 FLASH 地址处在加密区域中，“用户”是无法读取 FLASH 内容的，即使强行读出则内容皆为 0。

注 5：第一次使用 EEPROM 前，建议“用户”进行一次全擦除操作。

19 附录

19.1 指令集总览表

助记符	说明	周期(T)	长度(B)
ADD A, Rn	寄存器 Rn 和累加器 ACC 相加，结果存到 ACC	1	1
ADD A, direct	直接寻址字节和累加器 ACC 相加，结果存到 ACC	2	2
ADD A, @Ri	内部 RAM Ri 和累加器 ACC 相加，结果存到 ACC	2	1
ADD A, #data	立即数和累加器 ACC 相加，结果存到 ACC	1	1
ADDC A, Rn	寄存器 Rn 和累加器 ACC 相加（带 C 标志），结果存到 ACC	1	1
ADDC A, direct	直接寻址字节和累加器 ACC 相加（带 C 标志），结果存到 ACC	2	2
ADDC A, @Ri	内部 RAM Ri 和累加器 ACC 相加（带 C 标志），结果存到 ACC	2	1
ADDC A, #data	立即数和累加器 ACC 相加（带 C 标志），结果存到 ACC	1	1
SUBB A,Rn	累加器 ACC 减寄存器 Rn（带 C 标志），结果存到 ACC	1	1
SUBB A, direct	累加器 ACC 减直接寻址字节（带 C 标志），结果存到 ACC	2	2
SUBB A, @Ri	累加器 ACC 减内部 RAM Ri（带 C 标志），结果存到 ACC	2	1
SUBB A, #data	累加器 ACC 减立即数（带 C 标志），结果存到 ACC	1	1
INC A	累加器 ACC 加 1，结果存到 ACC	1	1
INC Rn	寄存器 Rn 加 1，结果存到 Rn	2	1
INC direct	直接寻址字节加 1，结果存到直接选址字节	3	2
INC @Ri	内部 RAM Ri 加 1，结果存到 Ri	3	1
INC DPTR	数据指针 DPTR/DPRT1 加 1，结果存到 DPTR/DPRT1	4	1
DEC A	累加器 ACC 自减 1，结果存到 ACC	1	1
DEC Rn	寄存器 Rn 自减 1，结果存到 Rn	2	1
DEC direct	直接寻址字节减 1，结果存到直接选址字节	3	2
DEC @Ri	内部 RAM Ri 减 1，结果存到 Ri	3	1
MUL AB 8 X 8	累加器 ACC 乘寄存器 B	11	1
16 X 8		20	
DIV AB 8 / 8	累加器 ACC 除寄存器 B	11	1
16 / 8		20	
DA A	BCD 码调整	1	1
ANL A, Rn	累加器 ACC 与寄存器 Rn，结果存到 ACC	1	1
ANL A, direct	累加器 ACC 与直接寻址字节，结果存到 ACC	2	2
ANL A, @Ri	累加器 ACC 与内部 RAM Ri，结果存到 ACC	2	1
ANL A, #data	累加器 ACC 与立即数，结果存到 ACC	2	2
ANL direct, A	直接寻址字节与累加器 ACC，结果存到直接寻址字节	3	2
ANL direct, #data	直接寻址字节与立即数，结果存到直接寻址字节	3	3
ORL A, Rn	累加器 ACC 或寄存器 Rn，结果存到 ACC	1	1
ORL A, direct	累加器 ACC 直接寻址字节，结果存到 ACC	2	2
ORL A, @Ri	累加器 ACC 或内部 RAM Ri，结果存到 ACC	2	1
ORL A, #data	累加器 ACC 或立即数，结果存到 ACC	2	2
ORL direct, A	直接寻址字节或累加器 ACC，结果存到直接寻址字节	3	2
ORL direct, #data	直接寻址字节或立即数，结果存到直接寻址字节	3	3

XOL A, Rn	累加器 ACC 异或寄存器 Rn，结果存到 ACC	1	1
XOL A, direct	累加器 ACC 直接寻址字节，结果存到 ACC	2	2
XOL A, @Ri	累加器 ACC 异或内部 RAM Ri，结果存到 ACC	2	1
XOL A, #data	累加器 ACC 异或立即数，结果存到 ACC	2	2
XOL direct, A	直接寻址字节异或累加器 ACC，结果存到直接寻址字节	3	2
XOL direct, #data	直接寻址字节异或立即数，结果存到直接寻址字节	3	3
CLR A	对 ACC 清零	1	1
CPL A	对 ACC 取反	1	1
RL A	累加器 ACC 循环左移	1	1
RLC A	累加器 ACC 循环左移（带 C 标志）	1	1
RR A	累加器 ACC 循环右移	1	1
RRC A	累加器 ACC 循环右移（带 C 标志）	1	1
SWAP A	交换累加器 ACC 的高低半字节，结果存到 ACC	4	1
MOV A, Rn	将寄存器 Rn 送到累加器 ACC	1	1
MOV A, direct	将直接寻址字节送到累加器 ACC	2	2
MOV A, @Ri	将内部 RAM Ri 送到累加器 ACC	2	1
MOV A, #data	将立即数送到累加器 ACC	2	2
MOV Rn, A	将累加器 ACC 送到寄存器 Rn	2	1
MOV Rn, direct	将直接寻址直接送到寄存器 Rn	3	2
MOV Rn, #data	将立即数送到寄存器 Rn	2	2
MOV Rn, A	将累加器 ACC 送到寄存器 Rn	2	1
MOV direct, A	将累加器 ACC 送到直接寻址字节	3	2
MOV direct, Rn	将寄存器 Rn 送到直接寻址字节	2	2
MOV direct1, direct2	将直接寻址字节 2 送到直接寻址字节 1	3	3
MOV direct, @Ri	将内部 RAM Ri 送到直接寻址字节	3	2
MOV direct, #data	将立即数送到直接寻址字节	3	3
MOV @Ri, A	将累加器 ACC 送到内部 RAM Ri	2	1
MOV @Ri, direct	将直接寻址字节送到内部 RAM Ri	3	2
MOV @Ri, #data	将立即数送到内部 RAM Ri	2	2
MOV DPTR, #data16	将 16 位立即数送到数据指针	3	3
MOVC A, @A+DPTR	将代码程序送到累加器 ACC（相对数据指针）	7	1
MOVC A, @A+PC	将代码程序送到累加器 ACC（相对程序计数器）	8	1
MOVX A, @Ri	通过内部 RAM Ri 将 8 位外存送到累加器 ACC	5	1
MOVX A, @DPTR	通过数据指针 DPTR 将 16 位外存送到累加器 ACC	6	2
MOVX @Ri, A	通过内部 RAM Ri 将累加器 ACC 送到 8 位外存	4	1

MOVX @DPTR, A	通过数据指针 DPTR 将累加器 ACC 送到 16 位外存	5	1
PUSH direct	直接寻址字节压入栈顶	5	2
POP direct	栈顶弹至直接寻址字节	4	2
XCH A, Rn	累加器 ACC 与寄存器 Rn 交换	3	1
XCH A, direct	累加器 ACC 与直接寻址字节交换	4	2
XCH A, @Ri	累加器 ACC 与内部 RAM Ri 交换	4	1
XCHD A, @Ri	累加器 ACC 低 4 位与内部 RAM Ri 低 4 位交换	4	1
ACALL addr11	2KB 内绝对调用	7	2
LCALL addr16	64KB 内长调用	7	3
RET	子程序返回	8	1
RETI	中断返回	8	1
AJMP addr11	2KB 内绝对转移	4	2
LJMP addr16	64KB 内长转移	5	3
SJMP rel	相对短转移	4	2
JMP @A+DPTR	相对长转移	6	1
JZ rel (不发生转移) (发生转移)	累加器为零转移	3 5	2
JNZ rel (不发生转移) (发生转移)	累加器为非零转移	3 5	2
JC rel (不发生转移) (发生转移)	C 置位转移	2 4	2
JNC rel (不发生转移) (发生转移)	C 清零转移	2 4	2
JB bit, rel (不发生转移) (发生转移)	直接寻址位置位转移	4 6	3
JNB bit, rel (不发生转移) (发生转移)	直接寻址位清零转移	4 6	3
JBC bit, rel (不发生转移) (发生转移)	直接寻址位置位转移并清该位	4 6	3
CJNE A, direct, rel (不发生转移) (发生转移)	累加器与直接寻址字节不等转移	4 6	3
CJNE A, #data, rel (不发生转移) (发生转移)	累加器与立即数不等转移	4 6	3
CJNE Rn, #data, rel (不发生转移) (发生转移)	寄存器与立即数不等转移	4 6	3
CJNE @Ri, #data, rel (不发生转移) (发生转移)	内部 RAM 与立即数不等转移	4 6	3
DJNZ Rn, rel (不发生转移) (发生转移)	寄存器减 1 不为零转移	3 5	2
DJNZ direct, rel (不发生转移) (发生转移)	直接寻址字节减 1 不为零转移	4 6	3

NOP	空操作	1	1
CLR C	C 清零	1	1
CLR bit	直接寻址位清零	3	2
SETB C	C 置位	1	1
SETB bit	直接寻址位置位	3	2
CPL C	C 取反	1	1
CPL bit	直接寻址位取反	3	2
ANL C, bit	C 逻辑与直接寻址位	2	2
ANL C, /bit	C 逻辑与直接寻址位的反	2	2
ORL C, bit	C 逻辑或直接寻址位	2	2
ORL C, /bit	C 逻辑或直接寻址位的反	2	2
MOV C, bit	直接寻址位送 C	2	2
MOV bit, C	C 送直接寻址位	3	2

注：对于条件跳转类指令，若跳转条件成立，则指令需 2 个周期，否则只需 1 个周期。

19.2 特殊功能寄存器总览表

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
FF	SFR0	**OSADJCR**	零点偏移修正寄存器	OSADJEN	OSADJT6	OSADJT5	OSADJT4	OSADJT3	OSADJT2	OSADJT1	OSADJT0	0xxx xxxx
	SFR1	**PWM145F**	PWM1的45组死区时间设定寄存器	PWM145F7	PWM145F6	PWM145F5	PWM145F4	PWM145F3	PWM145F2	PWM145F1	PWM145F0	1111 1111
FE	SFR0	**ADRH**	(ADRSEL = 0) ADC 转换结果高位寄存器	ADR11	ADR10	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4	xxxx xxxx
			(ADRSEL = 1) ADC 转换结果高位寄存器	-	-	-	-	ADR11	ADR10	ADR9	ADR8	---- xxxx
FD	SFR0	**ADRL**	(ADRSEL = 0) ADC 转换结果低位寄存器	-	-	-	-	ADR3	ADR2	ADR1	ADR0	---- xxxx
			(ADRSEL = 1) ADC 转换结果低位寄存器	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	xxxx xxxx
FC	SFR0	**ADRIOS1**	ADC 端口配置寄存器 1	-	-	EX_VREF_EN	-	AN11EN	AN10EN	AN9EN	AN8EN	--0- 0000
FB	SFR0	**ADRIOS0**	ADC 端口配置寄存器 0	AN7EN	AN6EN	AN5EN	AN4EN	AN3EN	AN2EN	AN1EN	AN0EN	0000 0000
FA	SFR0	**ADCCR1**	ADC 控制寄存器 1	ADRSEL	ADCKS2	ADCKS1	ADCKS0	ADSPS3	ADSPS2	ADSPS1	ADSPS0	0000 1111
F9	SFR0	**ADCCR0**	ADC 控制寄存器 0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	VRS1	VRS0	ADEOC	ADEN	1111 1110
	SFR1	**PWRCR1**	高级能耗控制寄存器 1	TK_CLKEN	-	CRC_CLKEN	-	-	T3_CLKEN	T2_CLKEN	T01_CLKEN	1111 1111
F8	SFR0	**PWRCR**	高级能耗控制寄存器	-	-	PWM1_CLKEN	PWM0_CLKEN	SPI_CLKEN	ADC_CLKEN	-	OCD_CLKEN	--11 1111

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
F7	SFR0	**TKCMPL**	触摸比较阈值低位寄存器	CMP15	CMP14	CMP13	CMP12	CMP11	CMP10	CMP9	CMP8	0000 0000
	SFR1	**PWM13DL**	PWM1 通道 3 占空比设定低位寄存器	PWM13D7	PWM13D6	PWM13D5	PWM13D4	PWM13D3	PWM13D2	PWM13D1	PWM13D0	0000 0000
F6	SFR0	**TKCMPH**	触摸比较阈值高位寄存器	CMP7	CMP6	CMP5	CMP4	CMP3	CMP2	CMP1	CMP0	0000 0000
	SFR1	**PWM13DH**	PWM1 通道 3 占空比设定高位寄存器	PWM13D15	PWM13D14	PWM13D13	PWM13D12	PWM13D13	PWM13D13	PWM13D9	PWM13D8	0000 0000
F5	SFR0	**TKSSN1**	触摸扫描顺序控制寄存器 1	TK15	TK14	TK13	TK12	TK11	TK10	TK9	TK8	0000 0000
	SFR1	**PWM123F**	PWM1的 23 组死区时间设定寄存器	PWM123F7	PWM123F6	PWM123F5	PWM123F4	PWM123F3	PWM123F2	PWM123F1	PWM123F0	1111 1111
F4	SFR0	**TKSSN0**	触摸扫描顺序控制寄存器 0	TK7	TK6	TK5	TK4	TK3	TK2	TK1	TK0	0000 0000
	SFR1	**PWM14DL**	PWM1 通道 4 占空比设定低位寄存器	PWM14D7	PWM14D6	PWM14D5	PWM14D4	PWM14D3	PWM14D2	PWM14D1	PWM14D0	0000 0000
F3	SFR0	**TKCHEN1**	触摸端口控制寄存器 1	TKCH15	TKCH14	TKCH13	TKCH12	TKCH11	TKCH10	TKCH9	TKCH8	0000 0000
	SFR1	**PWM14DH**	PWM1 通道 4 占空比设定高位寄存器	PWM14D15	PWM14D14	PWM14D13	PWM14D12	PWM14D11	PWM14D14	PWM14D9	PWM14D8	0000 0000
F2	SFR0	**TKCHEN0**	触摸端口控制寄存器 0	TKCH7	TKCH6	TKCH5	TKCH4	TKCH3	TKCH2	TKCH1	TKCH0	0000 0000
	SFR1	**PWM15DL**	PWM1 通道 5 占空比设定低位寄存器	PWM15D7	PWM15D6	PWM15D5	PWM15D4	PWM15D3	PWM15D2	PWM15D1	PWM15D0	0000 0000
F1	SFR0	**AUXC**	乘除法增强寄存	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
			器									
	SFR1	**PWM15DH**	PWM1 通道 5 占空比设定高位寄存器	PWM15D15	PWM15D14	PWM15D13	PWM15D12	PWM15D15	PWM15D15	PWM15D9	PWM15D8	0000 0000
F0	SFR0	**B**	B 寄存器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
EF	SFR0	**IAPKEY**	IAP 写保护寄存器	IAPKEY7	IAPKEY6	IAPKEY5	IAPKEY4	IAPKEY3	IAPKEY2	IAPKEY1	IAPKEY0	0000 0000
	SFR1	**PWM10DL**	PWM1 通道 0 占空比设定低位寄存器	PWM10D7	PWM10D6	PWM10D5	PWM10D4	PWM10D3	PWM10D2	PWM10D1	PWM10D0	0000 0000
EE	SFR0	**IAPMD**	IAP 模式寄存器	IAPEN	-	-	-	ECHIPERS	EPAGEERS	EPAGEWR	-	0--- 000-
	SFR1	**PWM10DH**	PWM1 通道 0 占空比设定高位寄存器	PWM10D15	PWM10D14	PWM10D13	PWM10D12	PWM10D11	PWM10D10	PWM10D9	PWM10D8	0000 0000
ED	SFR1	**PWM11DL**	PWM1 通道 1 占空比设定低位寄存器	PWM11D7	PWM11D6	PWM11D5	PWM11D4	PWM11D3	PWM11D2	PWM11D1	PWM11D0	0000 0000
EC	SFR0	**IAPDB**	IAP 数据寄存器	IAPDB7	IAPDB6	IAPDB5	IAPDB4	IAPDB3	IAPDB2	IAPDB1	IAPDB0	0000 0000
	SFR1	**PWM11DH**	PWM1 通道 1 占空比设定高位寄存器	PWM11D15	PWM11D14	PWM11D13	PWM11D12	PWM11D11	PWM11D11	PWM11D9	PWM11D8	0000 0000
EB	SFR0	**IAPADH**	IAP 高地址寄存器	IAPAD15	IAPAD14	IAPAD13	IAPAD12	IAPAD11	IAPAD10	IAPAD9	IAPAD8	0000 0000
	SFR1	**PWM101F**	PWM1的01组死区时间设定寄存器	PWM101F7	PWM101F6	PWM101F5	PWM101F4	PWM101F3	PWM101F2	PWM101F1	PWM101F0	1111 1111
EA	SFR0	**IAPADL**	IAP 低地址寄存器	IAPAD7	IAPAD6	IAPAD5	IAPAD4	IAPAD3	IAPAD2	IAPAD1	IAPAD0	0000 0000
	SFR1	**PWM12DL**	PWM1 通道 2 占空比设定低位寄存器	PWM12D7	PWM12D6	PWM12D5	PWM12D4	PWM12D3	PWM12D2	PWM12D1	PWM12D0	0000 0000
E9	SFR0	**IAPCR**	IAP 控制寄存器	IAPSTR	-	-	-	IAPTRIG2	IAPTRIG1	IAPTRIG0	EERD	0--- 0000
	SFR1	**PWM12DH**	PWM1 通道 2 占	PWM12D15	PWM12D14	PWM12D13	PWM12D12	PWM12D11	PWM12D12	PWM12D9	PWM12D8	0000 0000

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
			空比设定高位寄存器									
E8	SFR0	**RSTFLAG**	复位标志寄存器	PORF	LVRF	RSTF	WDTRF	SWRF	-	-	SWRST	0000 0--0
E7	SFR0	**TKCHERR**	触摸扫描出错提示寄存器	-	-	-	-	TKCNER3	TKCNER2	TKCNER1	TKCNER0	00-- --00
	SFR1	**PWM1IER**	PWM1 中断控制寄存器	PWM1IF	PWM1FBIF	-	-	-	-	PWM1FBIS	PWM1FBEN	00-- --00
E6	SFR0	**TKANA**	触摸模拟参数选择寄存器	CMPSAMMD1	CMPSAMMD0	TKPULLCUR2	TKPULLCUR1	TKPULLCUR0	TKRESVAL2	TKRESVAL1	TKRESVAL0	0000 0000
	SFR1	**PWM1CR0**	PWM1 控制寄存器 0	PWM1EN	-	PWM15OE	PWM14OE	PWM13OE	PWM12OE	PWM11OE	PWM10OE	0-00 0000
E5	SFR0	**TKVREF**	触摸模块电压选择寄存器	TKREF1	TKREF0	TKCCT1	TKCCT0	TKOPV1	TKOPV0	TKDCT1	TKDCT0	0000 0000
	SFR1	**PWM1CR1**	PWM1 控制寄存器 1	PWM1CS	PWM145MD	PWM123MD	PWM101MD	-	PWM1CLK2	PWM1CLK1	PWM1CLK0	0000 -000
E4	SFR0	**TKCON3**	触摸选择控制寄存器 3	TKAVGIF	TKCNTIF	TKOVIF	TKERRIF	-	-	TKCMPIF	TKIF	0000 --00
	SFR1	**PWM1CR2**	PWM1 控制寄存器 2	PWM1RL	PWM145FEN	PWM123FEN	PWM101FEN	-	PWM145S	PWM123S	PWM101S	0000 -000
E3	SFR0	**TKCON2**	触摸选择控制寄存器 2	TKRDNEN	TKCMPDIR	TKRDSL1	TKRDSL0	TKCXEN	TKCOMEN	TKCOMVS	TKCOMMD	0000 0000
	SFR1	**PWM1CR3**	PWM1 控制寄存器 3	PWM1CB	-	PWM15B	PWM14B	PWM13B	PWM12B	PWM11B	PWM10B	0-00 0000
E2	SFR0	**TKCON1**	触摸选择控制寄存器 1	TKTRGDIV1	TKTRGDIV0	TKFS5	TKFS4	TKFS3	TKFS2	TKFS1	TKFS0	0000 0000
	SFR1	**PWM1PL**	PWM1 周期设定低位寄存器	PWM101P7	PWM101P6	PWM101P5	PWM101P4	PWM101P3	PWM101P2	PWM101P1	PWM101P0	0000 0000
E1	SFR0	**TKCON0**	触摸选择控制寄存器 0	TKEN	TKLPMD	TKSAT	TKPAREN	TKMOD	TKOPSEL	TKSAMP1	TKSAMP0	0000 0000
	SFR1	**PWM1PH**	PWM1 周期设定	PWM101P1	PWM101P1	PWM101P1	PWM101P1	PWM101P1	PWM101P1	PWM101P9	PWM101P8	0000 0000

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
			高位寄存器	5	4	3	2	1	0			
E0	SFR0	**A**	累加器	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
DF	SFR0	**PWM0D**	PWM0 占空比设置寄存器	PWM0D7	PWM0D6	PWM0D5	PWM0D4	PWM0D3	PWM0D2	PWM0D1	PWM0D0	0000 0000
DE	SFR0	**PWM0P**	PWM0 周期寄设置存器	PWM0P7	PWM0P6	PWM0P5	PWM0P4	PWM0P3	PWM0P2	PWM0P1	PWM0P0	0000 0000
DD	SFR0	**PWM0CR**	PWM0 控制寄存器	PWM0EN	-	PWM0IF	PWM0OE	PWM0S	PWM0C2	PWM0C1	PWM0C0	0-00 0000
DC	SFR0	**LCDSW2**	LCD 软件控制寄存器 2	COM23EN	COM22EN	COM21EN	COM20EN	COM19EN	COM18EN	COM17EN	COM16EN	0000 0000
DB	SFR0	**LCDSW1**	LCD 软件控制寄存器 1	COM15EN	COM14EN	COM13EN	COM12EN	COM11EN	COM10EN	COM9EN	COM8EN	0000 0000
DA	SFR0	**LCDSW**	LCD 软件控制寄存器	COM7EN	COM6EN	COM5EN	COM4EN	COM3EN	COM2EN	COM1EN	COM0EN	0000 0000
D9	SFR0	**LCDCR0**	LCD 控制寄存器 0	LCDEN	-	-	-	-	-	-	-	0--- ----
D8	SFR0	**DPS**	数据指针选择寄存器	-	SFRSL0	-	-	DIV16	MUL16	-	SEL	00-0 0--0
D3	SFR0	**CRCH**	CRC 结果高字节寄存器	CRCH7	CRCH6	CRCH5	CRCH4	CRCH3	CRCH2	CRCH1	CRCH0	0000 0000
D2	SFR0	**CRCL**	CRC 结果低字节寄存器	CRCL7	CRCL6	CRCL5	CRCL4	CRCL3	CRCL2	CRCL1	CRCL0	1111 1111
D1	SFR0	**CRCCR**	CRC 控制寄存器	CRCEN	-	CRCIF	CRCST	CRCMD	CRCLCT	-	CRCENC	0-00 00-0
D0	SFR0	**PSW**	程序状态寄存器	CY	AC	F0	RS1	RS0	OV	F1	P	0000 0000
CF	SFR0	**WDTCR**	看门狗定时器控制寄存器	-	-	-	-	-	WDTOT2	WDTOT1	WDTOT0	---- -111
CE	SFR0	**T2CON1**	T2 控制寄存器 1	CM2ON	CM2	BR2EN	RCLKU1	TCLKU1	EX2SL	-	T2EN	0000 0010
CD	SFR0	**TH2**	T2 计数高位寄存器	TH27	TH26	TH25	TH24	TH23	TH22	TH21	TH20	0000 0000
CC	SFR0	**TL2**	T2 计数低位寄存器	TL27	TL26	TL25	TL24	TL23	TL22	TL21	TL20	0000 0000
CB	SFR0	**RCAPH2**	T2 重载/捕获高位	RCAPH27	RCAPH26	RCAPH25	RCAPH24	RCAPH23	RCAPH22	RCAPH21	RCAPH20	0000 0000

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
			寄存器									
CA	SFR0	**RCAPL2**	T2 重载/捕获低位寄存器	RCAPL27	RCAPL26	RCAPL25	RCAPL24	RCAPL23	RCAPL22	RCAPL21	RCAPL20	0000 0000
C9	SFR0	**T2MOD**	工作模式寄存器	LD2EN	T2CKS2	T2CKS1	T2CKS0	T2OE	T2SL1	T2SL0	-	0000 0000
C8	SFR0	**T2CON**	T2 控制寄存器	TF2	EXTF2	RCLKU0	TCLKU0	EX2EN	TR2	CT2	CPRL2	0000 0000
C7	SFR0	**TKAMP3**	触摸放大系数控制寄存器 3	-	-	-	-	TKAMP27	TKAMP26	TKAMP25	TKAMP24	---- 0000
C6	SFR0	**TKAMP2**	触摸放大系数控制寄存器 2	TKAMP23	TKAMP22	TKAMP21	TKAMP20	TKAMP19	TKAMP18	TKAMP17	TKAMP16	0000 0000
C5	SFR0	**TKAMP1**	触摸放大系数控制寄存器 1	TKAMP15	TKAMP14	TKAMP13	TKAMP12	TKAMP11	TKAMP10	TKAMP9	TKAMP8	0000 0000
C4	SFR0	**TKAMP0**	触摸放大系数控制寄存器 0	TKAMP7	TKAMP6	TKAMP5	TKAMP4	TKAMP3	TKAMP2	TKAMP1	TKAMP0	0000 0000
C3	SFR0	**P0PU**	P0 端口上拉电阻控制寄存器	P07PU	P06PU	P05PU	P04PU	P03PU	P02PU	P01PU	P00PU	0000 0000
C2	SFR0	**P0MOD1**	P0 端口模式寄存器 1	P03M1	P03M0	P02M1	P02M0	P01M1	P01M0	P00M1	P00M0	1010 1010
C1	SFR0	**P0MOD0**	P0 端口模式寄存器 0	P07PU	P06PU	P05PU	P04PU	P03PU	P02PU	P01PU	P00PU	0000 0000
C0	SFR0	**OSCM**	振荡模式寄存器	XTSPD	-	STBH	STBXT	-	CLKS	LFEN	HFEN	1-00 -x1x
BB	SFR0	**IPH1**	中断优先级控制 1 高位寄存器	CRCIPH	-	PWM1FBIPH	PWM0IPH	UART0IPH	PWM1IPH	T3IPH	T2IPH	0-00 0000
BA	SFR0	**IP1**	中断优先级控制 1 低位寄存器	CRCIPL	-	PWM1FBIPL	PWM0IPL	UART0IPL	PWM1IPL	T3IPL	T2IPL	0-00 0000
B9	SFR0	**IPH**	中断优先级控制高位寄存器	-	ADCIPH	SPIIPH	UART0IPH	T1IPH	EXT1IPH	T0IPH	EXT0IPH	-000 0000
B8	SFR0	**IP**	中断优先级控制低位寄存器	-	ADCIPL	SPIIPL	UART0IPL	T1IPL	EXT1IPL	T0IPL	EXT0IPL	-000 0000
B3	SFR0	**P3PU**	P3 端口上拉电阻控制寄存器	-	-	-	-	-	-	P31PU	P30PU	---- --00

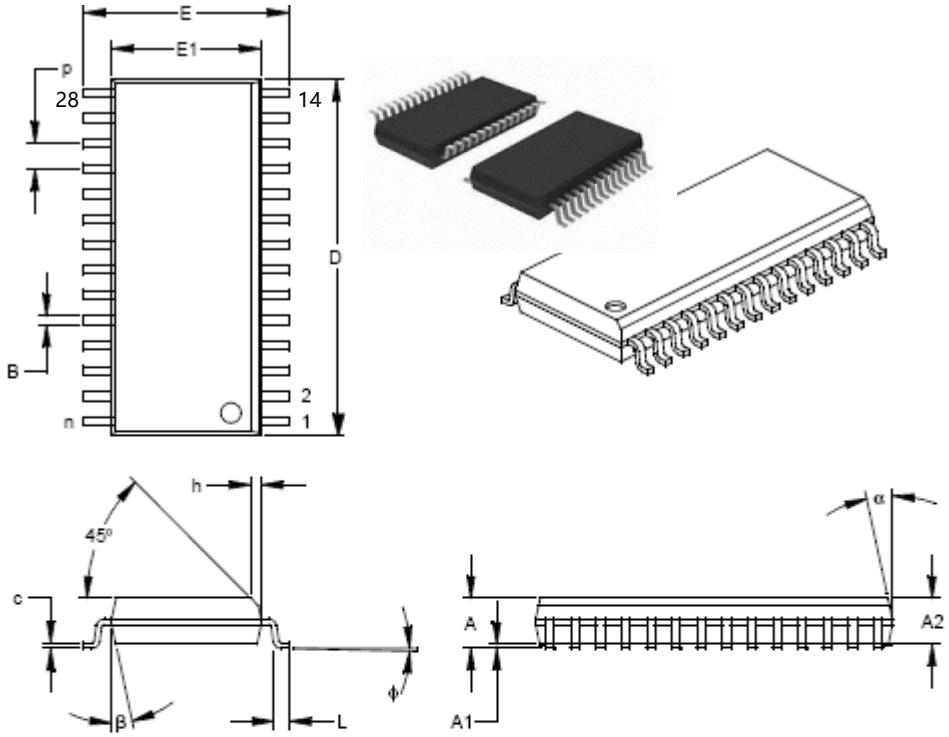
地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
B1	SFR0	**P3MOD0**	P3 端口模式寄存器 0	-	-	-	-	P31M1	P31M0	P30M1	P30M0	---- 1010
B0	SFR0	**P3**	P3 端口数据寄存器	-	-	-	-	-	-	P31D	P30D	---- --xx
AF	SFR0	**SPDAT**	SPI 数据寄存器	SPDAT7	SPDAT6	SPDAT5	SPDAT4	SPDAT3	SPDAT2	SPDAT1	SPDAT0	xxxx xxxx
AE	SFR0	**SPSTA**	SPI 状态寄存器	SPIEN	SPIIF	MODF	WCOL	SPIOV	-	-	-	0000 0---
AD	SFR0	**SPCON**	SPI 控制寄存器	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0	0000 0000
AC	SFR0	**EXTIS**	外部中断控制寄存器	-	EXT0IF4	EXT0IF3	EXT0IF2	EXT0IF1	EXT0IF0	EXT0IS1	EXT0IS0	-000 0000
	SFR1	**EXTEN**	外部中断使能寄存器	-	EXT0EN4	EXT0EN3	EXT0EN2	EXT0EN1	EXT0EN0	-	EXT0A	-000 00-0
AB	SFR0	**EXTIS1**	外部中断控制寄存器 1	-	EXT1IF4	EXT1IF3	EXT1IF2	EXT1IF1	EXT1IF0	EXT1IS1	EXT1IS0	-000 0000
	SFR1	**EXTEN1**	外部中断使能寄存器 1	-	EXT1EN4	EXT1EN3	EXT1EN2	EXT1EN1	EXT1EN0	-	EXT1A	-000 00-0
A9	SFR0	**IE1**	中断使能寄存器 1	CRCIE	-	PWM1FBIE	PWM0IE	UART1IE	PWM1IE	T3IE	T2IE	0-00 0000
A8	SFR0	**IE**	中断使能寄存器	EA	ADCIE	SPIIE	UART0IE	T1IE	EXT1IE	T0IE	EXT0IE	0000 0000
A3	SFR0	**P2PU**	P2 端口上拉电阻控制寄存器	P27PU	P26PU	P25PU	P24PU	P23PU	P22PU	P21PU	P20PU	0000 0000
A2	SFR0	**P2MOD1**	P2 端口模式寄存器 1	P27M1	P27M0	P26M1	P26M0	P25M1	P25M0	P24M1	P24M0	1010 1010
A1	SFR0	**P2MOD0**	P2 端口模式寄存器 0	P23M1	P23M0	P22M1	P22M0	P21M1	P21M0	P20M1	P20M0	1010 1010
A0	SFR0	**P2**	P2 端口数据寄存器	P27D	P26D	P25D	P24D	P23D	P22D	P21D	P20D	xxxx xxxx
9F	SFR0	**PXODM**	端口输出驱动控制寄存器	-	-	-	-	P1ODM1	P1ODM0	P0ODM1	P0ODM0	---- 0000
9D	SFR0	**SBUF_1**	串口1缓存寄存器	SBUF7_1	SBUF6_1	SBUF5_1	SBUF4_1	SBUF3_1	SBUF2_1	SBUF1_1	SBUF0_1	xxxx xxxx
9C	SFR0	**SCON_1**	串口1控制寄存器	SM0_1	SM1_1	SM2_1	REN_1	TB8_1	RB8_1	TI_1	RI_1	0000 0000
9B	SFR0	**SADEN**	串口0地址掩码寄	SADEN7	SADEN6	SADEN5	SADEN4	SADEN3	SADEN2	SADEN1	SADEN0	0000 0000

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
			存器									
9A	SFR0	**SADDR**	串口0从机地址寄存器	SADDR7	SADDR6	SADDR5	SADDR4	SADDR3	SADDR2	SADDR1	SADDR0	0000 0000
99	SFR0	**SBUF**	串口0缓存寄存器	SBUF7	SBUF6	SBUF5	SBUF4	SBUF3	SBUF2	SBUF1	SBUF0	xxxx xxxx
98	SFR0	**SCON**	串口0控制寄存器	SM0/FE	SM1/RXOV	SM2/TXCOL	REN	TB8	RB8	TI	RI	0000 0000
97	SFR0	**TH3**	T3 计数高位寄存器	TH37	TH36	TH35	TH34	TH33	TH32	TH31	TH30	0000 0000
96	SFR0	**TL3**	T3 计数低位寄存器	TL37	TL36	TL35	TL34	TL33	TL32	TL31	TL30	0000 0000
93	SFR0	**P1PU**	P1 端口上拉电阻控制寄存器	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU	0000 0000
92	SFR0	**P1MOD1**	P1 端口模式寄存器 1	P17M1	P17M0	P16M1	P16M0	P15M1	P15M0	P14M1	P14M0	1010 1010
91	SFR0	**P1MOD0**	P1 端口模式寄存器 0	P13M1	P13M0	P12M1	P12M0	P11M1	P11M0	P10M1	P10M0	1010 1010
90	SFR0	**P1**	P1 端口数据寄存器	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D	xxxx xxxx
8F	SFR0	**T3CR**	T3 控制寄存器	TR3	CT3	TF3	-	T3EN	T3CKS2	T3CKS1	T3CKS0	000- 0000
8E	SFR0	**TCON1**	定时器控制寄存器 1	T1FS	T1OE	T1CKS1	T1CKS0	T0FS	T0OE	T0CKS1	T0CKS0	0000 0000
8D	SFR0	**TH1**	定时器计数高位寄存器 1	TH17	TH16	TH15	TH14	TH13	TH12	TH11	TH10	0000 0000
8C	SFR0	**TH0**	定时器计数高位寄存器 0	TH07	TH06	TH05	TH04	TH03	TH02	TH01	TH00	0000 0000
8B	SFR0	**TL1**	定时器计数低位寄存器 1	TL17	TL16	TL15	TL14	TL13	TL12	TL11	TL10	0000 0000
8A	SFR0	**TL0**	定时器计数低位寄存器 0	TL07	TL06	TL05	TL04	TL03	TL02	TL01	TL00	0000 0000
89	SFR0	**TMOD**	定时器模式寄存	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0	0000 0000

地址(H)	SFR号	寄存器名	定义	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值（B）
			器									
88	SFR0	**TCON**	定时器控制寄存器	TF1	TR1	TF0	TR0	-	-	-	ADCIF	0000 ---0
87	SFR0	**PCON**	电源控制寄存器	UART1EN	-	-	UART0EN	SMOD0	SSTAT0	SLEEP	STOP	0--0 0000
86	SFR0	**SLPCR**	模式保护控制寄存器	SLPCR7	SLPCR6	SLPCR5	SLPCR4	SLPCR3	SLPCR2	SLPCR1	SLPCR0	0000 0000
85	SFR0	**DPH1**	数据指针1高字节	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
84	SFR0	**DPL1**	数据指针1低字节	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
83	SFR0	**DPH**	数据指针高字节	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
82	SFR0	**DPL**	数据指针低字节	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0000
81	SFR0	**SP**	堆栈指针	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	0000 0111
80	SFR0	**P0**	P0 端口数据寄存器	P07D	P06D	P05D	P04D	P03D	P02D	P01D	P00D	xxxx xxxx

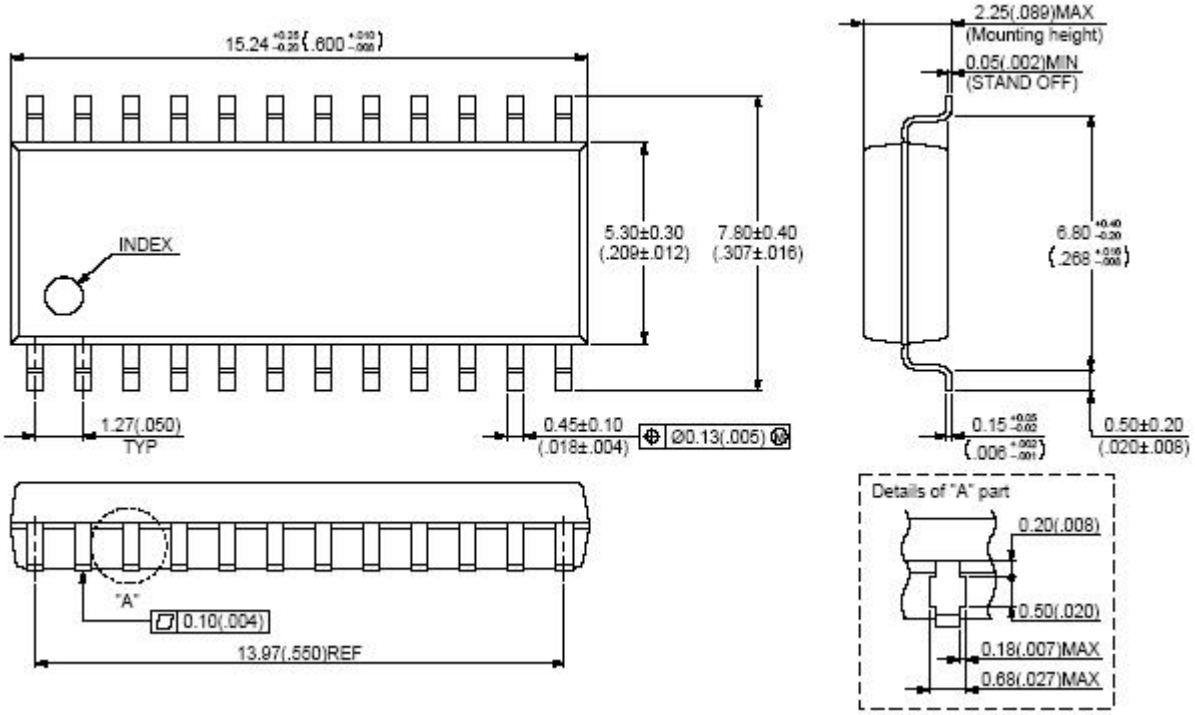
20 封装信息

20.1 SOP28（1.27mm pitch）

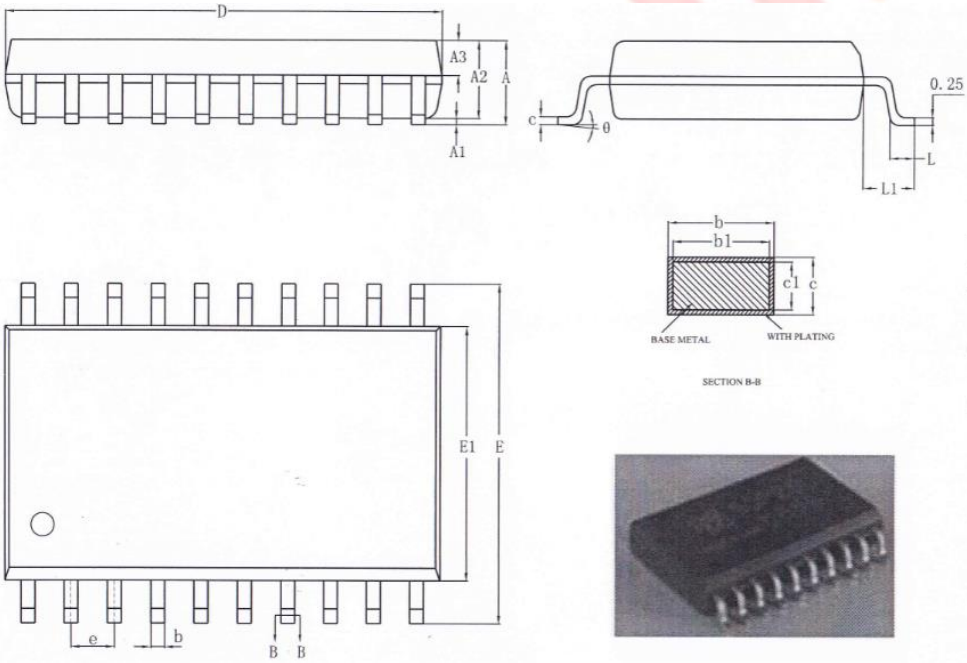


Units		INCHES*			MILLIMETERS		
Dimension Limits		MIN	NOM	MAX	MIN	NOM	MAX
Number of Pins	n		28			28	
Pitch	p		.050			1.27	
Overall Height	A	.093	.099	.104	2.36	2.50	2.64
Molded Package Thickness	A2	.088	.091	.094	2.24	2.31	2.39
Standoff §	A1	.004	.008	.012	0.10	0.20	0.30
Overall Width	E	.394	.407	.420	10.01	10.34	10.67
Molded Package Width	E1	.288	.295	.299	7.32	7.49	7.59
Overall Length	D	.695	.704	.712	17.65	17.87	18.08
Chamfer Distance	h	.010	.020	.029	0.25	0.50	0.74
Foot Length	L	.016	.033	.050	0.41	0.84	1.27
Foot Angle Top	ϕ	0	4	8	0	4	8
Lead Thickness	c	.009	.011	.013	0.23	0.28	0.33
Lead Width	B	.014	.017	.020	0.36	0.42	0.51
Mold Draft Angle Top	α	0	12	15	0	12	15
Mold Draft Angle Bottom	β	0	12	15	0	12	15

20.2 SOP24（1.27mm pitch）

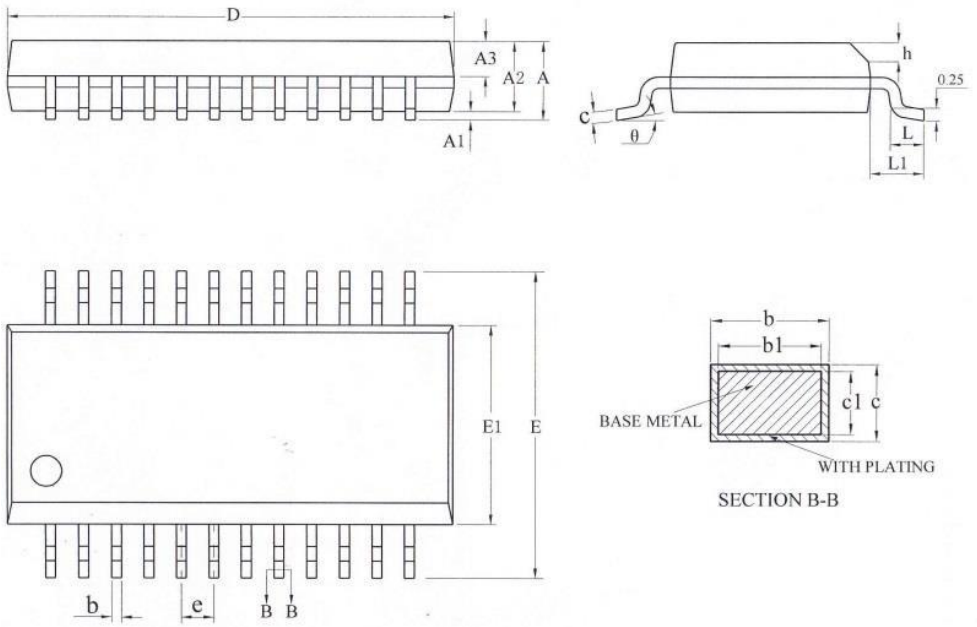


20.3 SOP20（1.27mm pitch）



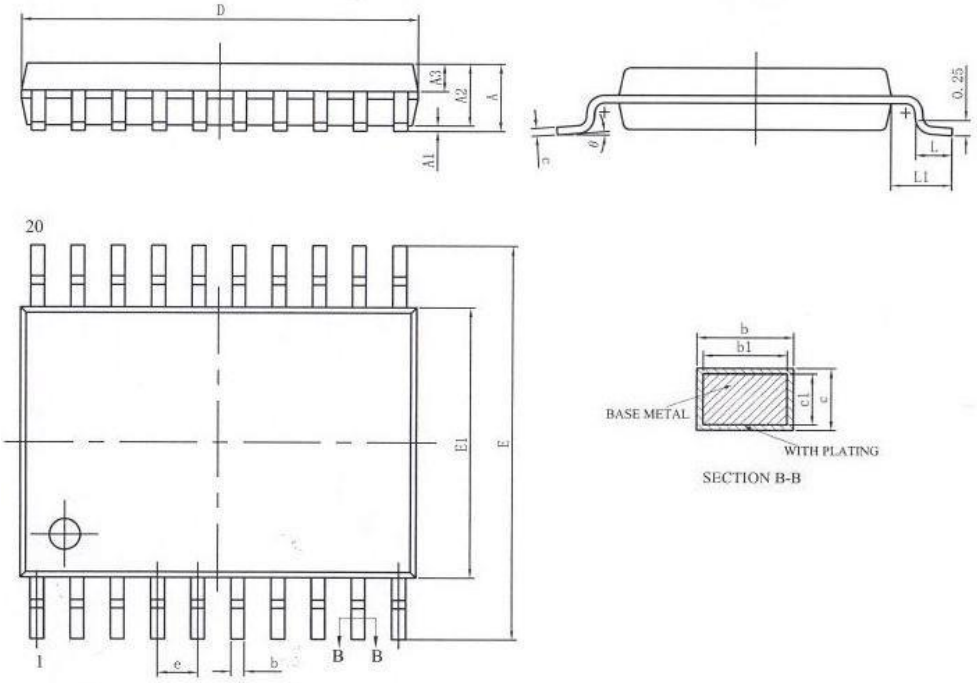
	SYMBOL	MILLIMETER		
		MIN	NOM	MAX
	A	—	—	2.65
	A1	0.10	—	0.30
	A2	2.25	2.30	2.35
	A3	0.97	1.02	1.07
⚠	b	0.35	—	0.43
⚠	b1	0.34	0.37	0.40
⚠	c	0.25	—	0.29
	c1	0.24	0.25	0.26
⚠	D	12.70	12.80	12.90
	E	10.10	10.30	10.50
⚠	E1	7.40	7.50	7.60
	e	1.27BSC		
	L	0.70	—	1.00
	L1	1.40REF		
	θ	0	—	8°

20.4 SSOP24（0.635mm pitch）



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.10	0.15	0.25
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.23	—	0.31
b1	0.22	0.25	0.28
c	0.20	—	0.24
c1	0.19	0.20	0.21
D	8.55	8.65	8.75
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	0.635BSC		
h	0.30	—	0.50
L	0.50	—	0.80
L1	1.05REF		
θ	0	—	8°

20.5 TSSOP20（0.65mm pitch）



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.20
A1	0.05	—	0.15
A2	0.80	1.00	1.05
A3	0.39	0.44	0.49
b	0.20	—	0.29
b1	0.19	0.22	0.25
c	0.13	—	0.18
c1	0.12	0.13	0.14
D	6.40	6.50	6.60
E1	4.30	4.40	4.50
E	6.20	6.40	6.60
e	0.65BSC		
L	0.45	0.60	0.75
L1	1.00BSC		
θ	0	—	8°

21 版本修订记录

版本号	修订日期	修订内容
V1.0	2020-12-02	初始版本
V1.1	2021-03-17	新增 SSOP24 封装 A0YK；
V1.2	2021-06-15	调整 T2 规格描述及示意图，修订 T0 中波特率计算公式的笔误，增加 T3 中关于停止/休眠模式下唤醒的说明； 增加休眠模式中唤醒后时钟预热的相关说明；增加 ADC 关闭后端口模式的相关注释；增加端口外部中断功能使能后其上拉电阻无效的注释； 增加 UART 中端口与 IO 功能转换的说明，统一 UART 端口命名；
V1.3	2021-08-12	修正 ADC 参考电压选择位 VRS 中选项 VDD 与 2.0V 定义互反的错误；
V1.4	2021-09-13	新增 TSSOP20 封装 A0Y；
V1.5	2021-10-08	修正 T2 溢出标志 TF2 定义中“硬件清 0”的错误说明； 去除触摸按键内部 Cx 电容的描述；
V1.6	2021-10-28	补全 LCD 软件控制寄存器 LCDSW1,LCDSW2 定义；

22 免责声明

本资料内容为广东晟矽微电子有限公司（以下简称“我公司”）版权所有。

我公司将力求本资料的内容做到准确无误，但同时保留在不通知用户的情况下，对本资料内容的修改权。如您需要获得最新的资料，请及时联系我公司。

我公司将尽最大努力为您提供高品质、高稳定性的产品。尽管如此，由于一般半导体器件的电气敏感性及易受到外部物理损伤等固有特性存在，所以难免造成半导体器件出现故障或失效的可能。当您使我公司产品时，有责任按照本资料以及相连资料中提到的“规则”来设计一个稳定及安全可靠的系统环境。另外，在遇到超规格（本资料中未描述到内容）的使用，请您提前咨询我公司，以免因我公司产品在一些特殊设备中或者特殊环境下的使用，导致财产损失、人员伤亡等严重后果的发生。因超规格的使用、未经咨询授权我公司产品的使用，我公司对此不承担任何法律责任。