



十速

TM52eF1375A

规格书

Rev 0.90

(使用前请阅读第二页的注意事项)

tenx reserves the right to change or discontinue the manual and online documentation to this product herein to improve reliability, function or design without further notice. **tenx** does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. **tenx** products are not designed, intended, or authorized for use in life support appliances, devices, or systems. If Buyer purchases or uses **tenx** products for any such unintended or unauthorized application, Buyer shall indemnify and hold **tenx** and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that **tenx** was negligent regarding the design or manufacture of the part.

使用注意事项

1. 在进入停止/暂停模式（PDOWN）之前，必须设置 ENVPULL（ENVPULL = 1）。
2. 在进入停止/暂停模式（PDOWN）之前，必须将系统时钟设置为慢时钟模式（SELFCK = 0）。
3. IAP 写入前，用户应先关闭LVR，等IAP 写入完成后再开启LVR。

修改纪录

版次	生效日	修订内容概要
V0.90	Dec, 2022	新颁。

目录

使用注意事项.....	2
修改纪录.....	3
TM52 系列 eF1xxx 家族.....	7
概述.....	9
系统框图.....	9
基本功能.....	10
IC 引脚图	14
引脚描述.....	15
引脚汇总.....	16
功能描述.....	17
1. CPU 核心.....	17
1.1 累加器 (ACC)	17
1.2 B 寄存器 (B)	17
1.3 堆栈指针 (SP)	18
1.4 数据指针 (DPTRs)	18
1.5 程序状态字 (PSW)	19
2. 存储器.....	20
2.1 程序存储器.....	20
2.1.1 程序存储器的功能分区.....	20
2.1.2 FLASH ICP 模式.....	21
2.1.3 Flash IAP 模式（类似于 EEPROM）	21
2.1.4 IAP 模式访问例程	22
2.2 数据存储器.....	25
2.2.1 IRAM	26
2.2.2 XRAM.....	26
2.2.3 SFRs	26
3. 低电压复位和低电压检测.....	28
4. 复位.....	30
4.1 上电复位	30
4.2 外部引脚复位	30
4.3 软件复位	30
4.4 看门狗定时器复位	30
4.5 低电压复位	30

5. 时钟电路和工作模式.....	32
5.1 时钟电路.....	32
5.2 操作模式.....	34
6. 中断和唤醒.....	36
6.1 中断使能和优先级控制.....	36
6.2 关于中断子程序的建议.....	36
6.3 引脚中断和 LVD 中断.....	41
6.4 空闲模式唤醒和中断.....	45
6.5 暂停/停止模式唤醒和中断.....	45
7. I/O 端口.....	47
7.1 端口 1/端口 2 /端口 3.....	47
7.2 端口 0.....	55
8. 定时器.....	58
8.1 Timer0/1.....	58
8.2 Timer2.....	61
8.3 Timer3.....	63
8.4 T0O 和 T2O 输出控制.....	63
9. UART.....	64
10. PWMs.....	66
11. ADC.....	71
11.1 ADC 通道.....	72
11.2 ADC 转换时间.....	72
12. 触摸按键 (FTK).....	75
13. 软件控制的 LCD 驱动器.....	85
14. LED 控制器/驱动器.....	86
14.1 LED 正反扫模式 (BiD).....	87
14.2 LED 点矩阵模式 (DMX).....	90
15. 串行外围设备接口 (SPI).....	95
16. 循环冗余校验码 (CRC).....	100
17. 乘法器和除法器.....	101
18. 主 I ² C 接口.....	103
19. 从机 I ² C 接口.....	107
20. 在线仿真器 (ICE) 模式.....	110

SFR & CFGW 映像	111
SFR & CFGW 说明	113
指令集.....	129
电气特性.....	132
1. 最大绝对额定值.....	132
2. DC 特性	133
3. 时钟时序.....	135
4. 复位时序特性.....	135
5. ADC 电气特性	135
6. 特性曲线图.....	136
封装说明.....	139

TM52 系列 eF1xxx 家族
共同特性

CPU	闪存程序 存储器	RAM 字节	双 时钟	工作模式	Timer0 Timer1 Timer2	UART	实时 Timer3	LVD	LVR
快速8051 (2T)	16K~64K 有 IAP, ISP, ICP	1280 ~ 4352	SXT SRC FXT FRC	快钟 慢钟 空闲 停止 暂停	8051标准		15-bit	16 阶	8~16 阶

家族成员特性

P/N	程序 存储器	数据 存储器	RAM 字节	IO 引脚	PWM	SAR ADC	触摸 按键	LCD	LED	串口
TM52-eF1716	闪存 16K 字节	EEPROM 128字节	1280	30	16位x3 8位x3	12位 16通道	20通道	8com	BiD 4Cx6S	SPI UARTx2 I ² C
TM52-eF1732	闪存 32K 字节	EEPROM 128字节	1280	30	16位x3 8位x3	12位 16通道	20通道	8com	BiD 4Cx6S	SPI UARTx2 I ² C
TM52-eF1374	闪存 20K 字节	EEPROM 128字节	1280	26	16位x3	12位 16通道	20通道	8com	BiD 4Cx6S DMX 8x8	SPI UARTx2 I ² C
TM52-eF1375	闪存 32K 字节	EEPROM 128字节	1280	26	16位x3	12位 16通道	20通道	8com	BiD 4Cx6S DMX 8x8	SPI UARTx2 I ² C
TM52-eF1375A	闪存 32K 字节	EEPROM 128字节	1280	26	16位x3	12位 16通道	20通道	8com	BiD 4Cx6S DMX 8x8	SPI UARTx2 I ² C
TM52-eF1385	闪存 32K 字节	EEPROM 128字节	4352	42	16位x9	12位 45通道	21通道 x 2	4Cx20S ~ 8Cx16S	MX 8x8 DMX 7x8	SPI UARTx3 I ² C
TM52-eF1386	闪存 64K 字节	EEPROM 128字节	4352	42	16位x9	12位 45通道	21通道 x 2	4Cx20S ~ 8Cx16S	MX 8x8 DMX 7x8	SPI UARTx3 I ² C

P/N	工作电压	工作电流					最大系统时钟 (Hz)			
		快钟 FRC	慢钟 SRC	空闲 SRC	停止	暂停	SXT	SRC	FXT	FRC
TM52-eF1716 TM52-eF1732	2.5~5.5V	3.5mA	0.18mA	0.15mA	7uA@5V 1.4uA@3V	11uA@5V 4uA@3V	32K	80K	16M	14.7456 M
TM52-eF1374 TM52-eF1375	2.2~5.5V	3.4mA	0.2mA	0.17mA	12uA@5V 5uA@3V	15uA@5V 7uA@3V	32K	80K	18M	18.432M
TM52-eF1375A	2.5~5.5V	3.2mA	0.19mA	0.15mA	7.7uA@5V 1.5uA@3V	11uA@5V 4uA@3V	32K	80K	18M	18.432M
TM52-eF1386	2.3~5.5V	3.5mA	0.2mA	0.18 mA	11uA@5V 4uA@3V	14uA@5V 6uA@3V	32K	80K	18M	18.432M

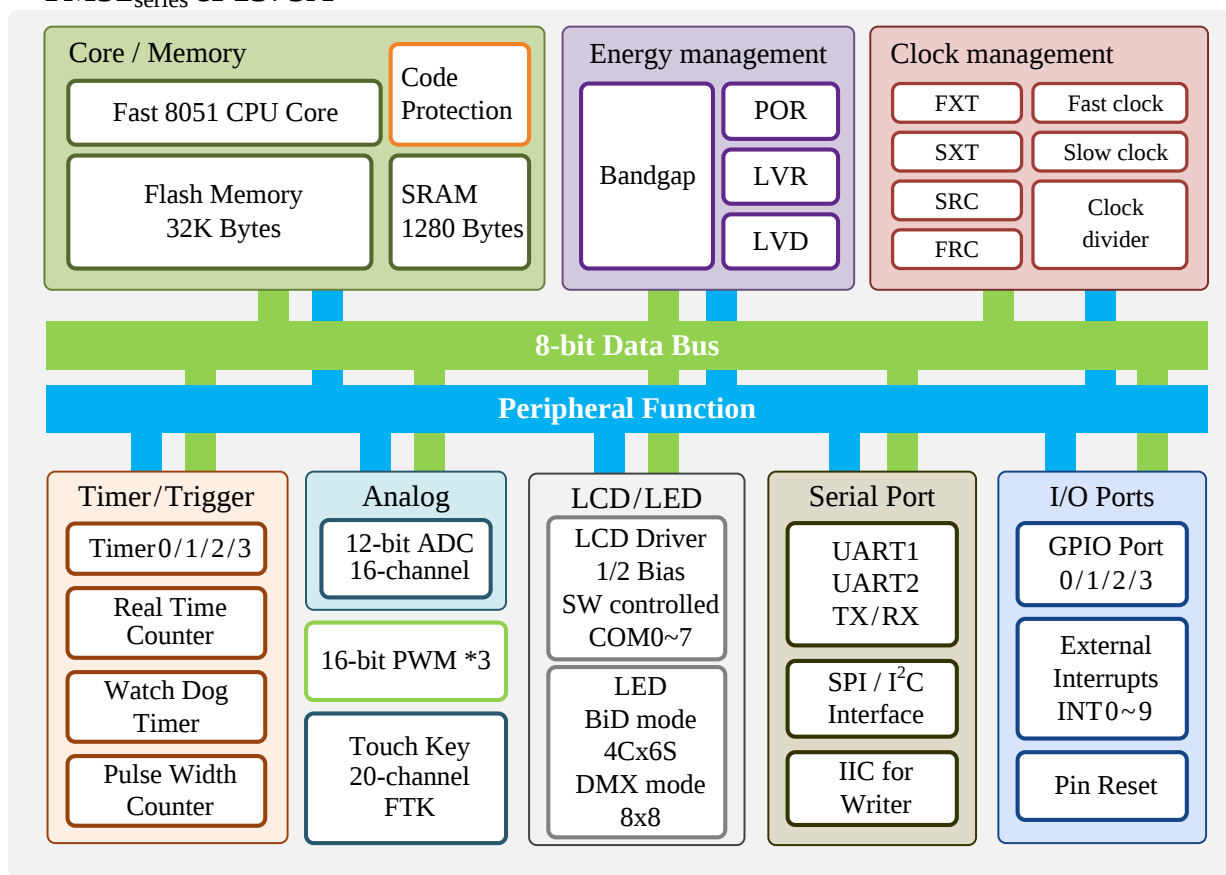
概述

TM52 系列 eF1375A 是一个新的,快速的 8051 架构,与业界标准 8051 指令集完全兼容的 8 位单片机,并保持了 8051 外围的功能模块。通常情况下,TM52 执行指令,比传统的 8051 架构快六倍。

TM52-eF1375A 通过集成多种功能在芯片上,提供更高的性能,更低的成本,能快速进入市场,包括 32K 字节的 Flash 程序存储器,1280 字节 SRAM,低电压复位 (LVR),低电压检测 (LVD),双时钟省电工作模式,8051 标准 UART 和定时器 Timer0/Timer1/Timer2,实时计时器 Timer3,LCD/LED 驱动器,3 组 16 位脉冲宽度调制器,16 通道的 12 位模数转换器 (ADC),20 通道触摸按键, I²C/SPI 串口和看门狗定时器 (WDT)。它的高可靠性和低功耗的特性,可广泛适用于消费电子及家用电器产品。

系统框图

TM52_{series} eF1375A



基本功能

1. 标准 8051 指令集,快速的机器周期
 - 指令执行比传统 8051 快六倍
2. FLASH 程序存储器
 - 32K 字节 FLASH 程序存储器
 - 支持 IAP(类 EEPROM)
 - 程序码保护功能
 - 最低擦除 10K 次
 - 最低 10 年数据保留
3. 总计 1280 字节 SRAM(IRAM+XRAM)
 - 256 字节 IRAM 在 8051 内部数据存储器区
 - 1024 字节 XRAM 在 8051 外部数据存储器区(由 MOVX 指令存取)
4. 4 种系统时钟类型选择
 - 快时钟使用外部 1~18 MHz 晶体(FXT)
 - 快时钟使用内部 RC(FRC, 18.432 MHz)
 - 慢时钟使用外部 32768 Hz 晶体(SXT)
 - 慢时钟使用内部 RC(SRC, 80 KHz)
 - 系统时钟可以通过 1/2/4/16 选项除频
5. 8051 标准定时器 – Timer0/1/2
 - 16 位 Timer0, 支持 T0O 时钟输出供蜂鸣器应用
 - 16 位 Timer1
 - 16 位 Timer2, 支持 T2O 时钟输出供蜂鸣器应用
6. 15 位 Timer3
 - 时钟源为慢时钟
 - 中断期可选时钟除以 32768/16384/8192/128 选项
7. UARTs
 - UART1, 8051 标准 UART
 - UART2, 仅支持模式 1 和模式 3
8. 三个独立的 16 位的 PWM 有周期调整
 - 具有 PWM0/PWM1/PWM2 中断

9. SPI 串口

- 主机模式或从机模式可选
- 可编程的传输比特率
- 串行时钟相位和极性选项
- 优先选择 MSB 优先或 LSB 优先

10. I2C 接口（主/从）

11. 20 通道触摸键（FTK）

- 内部参考电容支持
- 支持四种扫描模式

12. 12 位 ADC, 具有 13 个通道的外部引脚输入和 3 通道内部参考电压

- 内部基准电压源 (V_{BG}): 1.22V @ $V_{CC}=5V\sim3V$, 25°C
- 内部基准电压源: $1/4V_{CC}$
- 2 种 ADC 内部基准电压源: $2.5V/V_{CC}$

13. LCD 控制器/驱动器

- 1/8 占空比
- 软件控制的 COM0~7
- 1/2 LCD 偏压

14. LED 控制器/驱动器

- 支持 COM 死区防闪烁
- 支持暂停功能
- 支持亮度均匀/增强模式

【正反扫模式 (BiD)】

- 4C x 6S, 10 引脚, 最多 48 点
- 三组 8 段亮度可调

【点矩阵模式 (DMX)】

- 8C x 8S, 9 引脚, 最高 64 点
- 8 段亮度可调

15. 引脚中断能将暂停/停止模式下的 CPU 唤醒

- INT0~INT9 中断与唤醒
- 端口 1 每个引脚可以定义为唤醒和中断引脚(通过引脚电平变化)

注: 如果 INTn 引脚为低电平并且使能了唤醒功能, 则芯片无法进入暂停/停止模式。(INTn = 0 和 EXn = 1, n = 0~9)

16. 14 来源, 4 中断优先级

- Timer0/Timer1/Timer2/Timer3 中断
- INT0/INT1 下降沿/低电平中断
- INT2~INT9 下降沿中断
- 端口 1 引脚电平变化中断
- UART1/UART2 TX/RX 中断
- ADC/触摸键中断
- SPI 中断
- I²C 中断
- PWM0/PWM1/PWM2 中断

17. 最大 26 可编程 I/O 引脚

- CMOS 推挽输出
- 伪开漏或开漏输出
- 施密特触发输入
- 引脚上拉可以使能/禁止
- 所有引脚均具有高灌电流 (80mA @ $V_{CC} = 5V$, $V_{OL} = 0.1V_{CC}$)

18. 独立的 RC 振荡看门狗定时器

- 400ms/200ms/100ms/50ms 可选择的看门狗超时选项

19. 5 种复位

- 上电复位
- 可选的外部引脚复位
- 可选的看门狗复位
- 软件命令复位
- 可选的低电压复位

20. 16 级低电压复位

- 4.3V/4.2V/4.0V/3.9V/3.8V/3.7V/3.6V/3.4V/
3.3V/3.2V/3.1V/3.0V/2.8V/2.7V/2.6V/2.5V

21. 8 级低电压检测

- 4.2V/3.9V/3.7V/3.4V/3.2V/3.0V/2.7V/2.5V

22. 5 种电源工作模式

- 快钟模式/慢钟模式/空闲模式/暂停模式/停止模式

23. 集成的 16 位循环冗余校验功能

24. 乘法与除法

- 8 位乘法器和除法器（标准 8051）
- 16 位乘法器和除法器
- 32 位÷16 位除法器

25. 在板仿真/ICE 接口

- 使用 P3.0/P3.1 引脚或 P0.0/P0.1 引脚
- 与 ICP 编程引脚共享

26. 工作电压

- $V_{CC} = 2.5V \sim 5.5V$ @ $F_{SYSCLK} = 18.432MHz$
- $I_{CC} = 7.7\mu A$ @停止模式, $V_{CC} = 5V$
- $I_{CC} = 1.5\mu A$ @停止模式, $V_{CC} = 3V$
- $I_{CC} = 11\mu A$ @暂停模式, $V_{CC} = 5V$
- $I_{CC} = 4.0\mu A$ @暂停模式, $V_{CC} = 3V$
- $I_{CC} = 150\mu A$ @空闲模式, $V_{CC} = 5V$

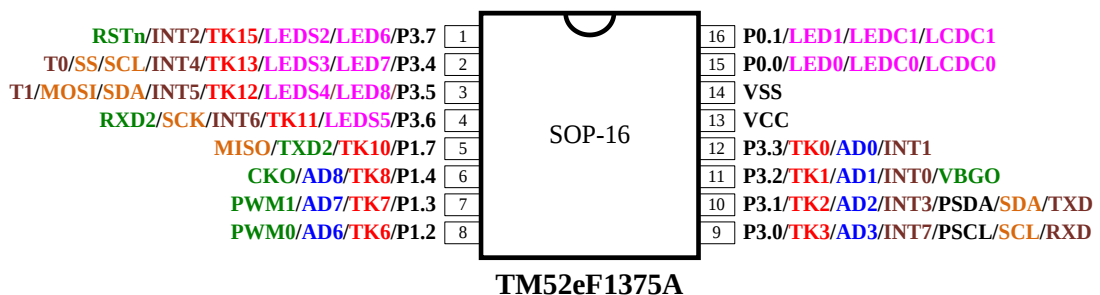
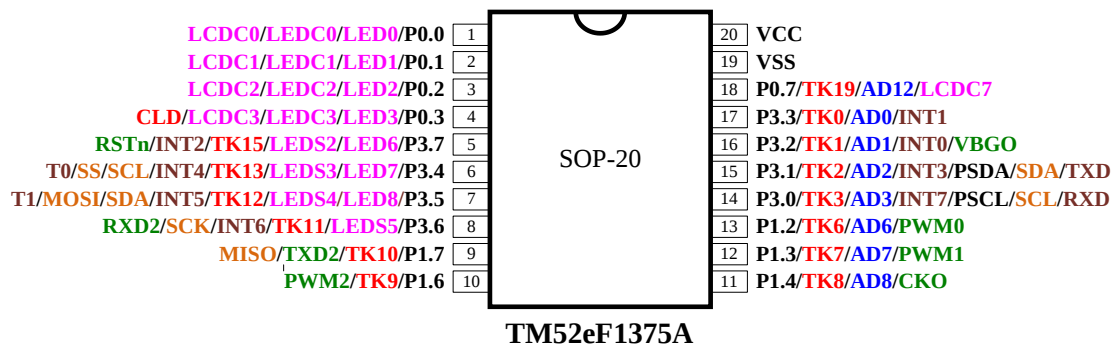
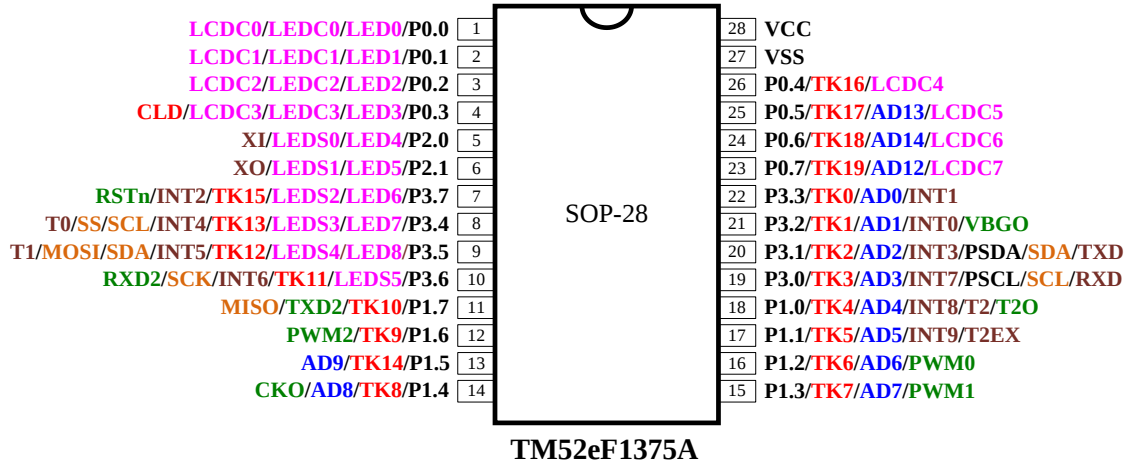
27. 工作温度范围

- $-40^{\circ}C \sim +105^{\circ}C$

28. 封装类型

- 28-pin SOP (300 mil)
- 20-pin SOP (300 mil)
- 16-pin SOP (150 mil)

IC 引脚图



引脚描述

Name	In/Out	Pin Description
P0.0~P0.7	I/O	施密特触发器输入或 CMOS 推挽输出的位可编程 I/O 端口。上拉电阻可由软件分配。
P1.0~P1.7	I/O	施密特触发器输入，CMOS 推挽输出或开漏输出的位可编程 I/O 端口。上拉电阻可由软件分配。这些管脚的电平变化可以中断 CPU 使其从空闲/暂停/停止模式唤醒。
P2.0~P2.1	I/O	施密特触发器输入，CMOS 推挽输出或开漏输出输出的位可编程 I/O 端口。上拉电阻可由软件分配。
P3.0~P3.2	I/O	施密特触发器输入，CMOS 推挽输出或“伪开漏输出”输出的位可编程 I/O 端口。上拉电阻可由软件分配。
P3.3~P3.7	I/O	施密特触发器输入，CMOS 推挽输出或开漏输出的位可编程 I/O 端口。上拉电阻可由软件分配。
INT0, INT1	I	外部低电平或下降沿中断输入，空闲/暂停/停止模式唤醒输入。
INT2~9	I	外部下降沿中断输入，空闲/暂停/停止模式唤醒输入。
RXD	I/O	UART1 模式 0 发送和接收数据，模式 1/2/3 接收数据
RXD2	I/O	UART2 模式 1/3 接收数据
TXD	I/O	UART1 模式 0 发送时钟，模式 1/2/3 发送数据。在单线 UART 模式下，此引脚发送和接收串行数据。
TXD2	I/O	UART2 模式 1/3 发送数据
T0, T1, T2	I	Timer0, Timer1, Timer2 事件计数引脚输入。
T2EX	I	Timer2 外部触发输入
T0O	O	Timer0 溢出除以 64 输出
T2O	O	Timer2 溢出除以 2 输出
CKO	O	系统时钟除以 2 输出
VBGO	O	带隙电压输出
PWM0~PWM2	O	16 位 PWM 输出
AD0~AD9 AD12~AD14	I	ADC 输入
TK00~TK19	I	触摸键输入
CLD	I	触控按键电荷收集电容器连接引脚
LCDC0~LCDC7	O	LCD 1/2 偏压输出
LEDC0~LEDC3	O	LED 正反扫模式 COM 输出
LEDS0~LEDS5	O	LED 正反扫模式 SEG 输出
LED0~LED8	O	LED 点矩阵模式输出
MISO	I/O	SPI 数据输入用于主模式，数据输出用于从模式
MOSI	I/O	SPI 数据输出用于主机模式，数据输入用于从机模式
SS	I	SPI 低电平有效从机选择输入，用于从机模式
SCK	I/O	用于主机的 SPI 时钟输出或用于从机模式的时钟输入
SCL	I/O	I ² C SCL
SDA	I/O	I ² C SDA
PSCL	I/O	程序用 I ² C SCL
PSDA	I/O	程序用 I ² C SDA
RSTn	I	外部低电平有效复位输入，上拉电阻固定使能。
XI, XO	—	晶振/谐振器振荡器连接，用于系统时钟（FXT 或 SXT）
VCC, VSS	P	电源输入引脚和接地

引脚汇总

Pin #	引脚名称	类型	初始状态	输入			输出			交替功能										其它
SOP-28				上拉电阻	唤醒	外部中断	推挽	伪开漏	开漏	LCD	LED 正反转	LED 点阵	ADC	Touch Key	UART	PWM	Timer	SPI	I ² C	
1	P0.0	I/O	Hi-Z	⊙			•			•	•	•								
2	P0.1	I/O	Hi-Z	⊙			•			•	•	•								
3	P0.2	I/O	Hi-Z	⊙			•			•	•	•								
4	P0.3	I/O	Hi-Z	⊙			•			•	•	•								CLD
5	P2.0	I/O	Hi-Z	⊙			•		•	•	•	•								Crystal
6	P2.1	I/O	Hi-Z	⊙			•		•	•	•	•								Crystal
7	P3.7	I/O	PU	⊙	•	•	•		•	•	•	•		•						Reset
8	P3.4	I/O	Hi-Z	⊙	•	•	•		•	•	•	•		•			•	•	•	
9	P3.5	I/O	Hi-Z	⊙	•	•	•		•	•	•	•		•			•	•	•	
10	P3.6	I/O	Hi-Z	⊙	•	•	•		•	•	•	•		•	•			•		
11	P1.7	I/O	Hi-Z	⊙	•		•		•	•	•	•		•	•			•		
12	P1.6	I/O	Hi-Z	⊙	•		•		•	•	•	•		•		•				
13	P1.5	I/O	Hi-Z	⊙	•		•		•	•	•	•		•	•					
14	P1.4	I/O	Hi-Z	⊙	•		•		•	•	•	•		•	•					CKO
15	P1.3	I/O	Hi-Z	⊙	•		•		•	•	•	•		•	•	•				
16	P1.2	I/O	Hi-Z	⊙	•		•		•	•	•	•		•	•	•				
17	P1.1	I/O	Hi-Z	⊙	•	•	•		•	•	•	•		•	•		•			
18	P1.0	I/O	Hi-Z	⊙	•	•	•		•	•	•	•		•	•		•			T2O
19	P3.0	I/O	Hi-Z	⊙	•	•	•	•		•	•	•		•	•	•			•	
20	P3.1	I/O	Hi-Z	⊙	•	•	•	•		•	•	•		•	•	•			•	
21	P3.2	I/O	Hi-Z	⊙	•	•	•	•		•	•	•		•	•	•				VBGO
22	P3.3	I/O	Hi-Z	⊙	•	•	•		•	•	•	•		•	•	•				
23	P0.7	I/O	Hi-Z	⊙			•			•	•	•		•	•					
24	P0.6	I/O	Hi-Z	⊙			•			•	•	•		•	•					
25	P0.5	I/O	Hi-Z	⊙			•			•	•	•		•	•					
26	P0.4	I/O	Hi-Z	⊙			•			•	•	•		•	•					
27	VSS	P																		
28	VCC	P																		

PS:

- 端口 1, P2.0, P2.1, 端口 3 这些引脚通过工作模式控制上拉电阻
- ⊙ 端口 0, 在 PxOE.n = 0 和 Px.n = 1 时控制上拉电阻

功能描述

1. CPU 核心

采用 8051 的架构, C 语言作为开发平台。TM52 装置拥有一个快速 8051 内核的高度集成微控制器, 可以使开发人员实现比传统 8051 芯片更高的性能。TM52 系列微控制器提供标准 8051 指令集兼容的完整的二进制代码, 以确保一个简单的移植路径, 以加快系统产品的开发速度。CPU 核心包括了 ALU, 程序状态字 (PSW), 累加器 (ACC), B 寄存器, 堆栈指针 (SP), 数据指针, 编程计数器, 指令译码器, 以及核心的特殊功能寄存器 (SFR)。

1.1 累加器 (ACC)

该寄存器提供了一个运算数供给大多数的 ALU 操作。累加器通常被称为 A 或 ACC 和有时被称为寄存器 A。在本文档中, 累加器被表示为“A”或“ACC”, 包括指令表。累加器, 正如其名称所示, 被用作通用寄存器累积了大量的指令的中间结果。累加器是完成算术运算和逻辑运算的最重要、最频繁的寄存器。它保存大多数算术和逻辑运算的中间结果, 以协助数据运送。

SFR E0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ACC	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

E0h.7~0 ACC: 累加器

1.2 B 寄存器 (B)

“B”寄存器和 ACC 是非常相似的, 可容纳 1 个字节的值。该寄存器提供了乘法或除法指令的第二个运算数。否则, 它可被用作一个暂存寄存器。B 寄存器只有用于两个 8051 的指令, MUL 和 DIV。当 A 乘或除以另一个数, 结果数存储在 B。对于 MUL 和 DIV 指令, 有必要将这两个运算数放在 A 和 B。

ex: DIV AB

当执行该指令, A 里面的数会除以 B 的数, 得到的答复是存储在 A。

SFR F0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
B	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F0h.7~0 B: B 寄存器

1.3 堆栈指针 (SP)

SP 寄存器包含堆栈指针。执行 LCALL,ACALL 和 PUSH 指令时,堆栈指针先加 1,再将程序计数器加载到堆栈中。执行 RET,RETI 和 POP 指令时,堆栈数据退回程序计数器后,堆栈指针再减 1。

SFR 81h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SP	SP							
R/W	R/W							
Reset	0	0	0	0	0	1	1	1

81h.7~0 **SP**:堆栈指针

1.4 数据指针 (DPTRs)

TM52 装置有两个数据指针,它们共享相同的 SFR 地址。每个 DPTR 的大小是 16 位,有两个数据指针寄存器:高字节 (DPH) 和低字节 (DPL)。该 DPTR 用于 16 位地址的外部存储器存取,偏移字节代码读取和偏移程序跳转。设置 DPSEL 控制位允许程序代码在两个物理数据指针之间进行切换。

SFR 82h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPL	DPL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

82h.7~0 **DPL**:数据指针低字节

SFR 83h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DPH	DPH							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

83h.7~0 **DPH**:数据指针高字节

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.0 **DPSEL**:活动数据指针选择

1.5 程序状态字 (PSW)

该寄存器包含 CPU 和 ALU 操作导致的状态信息。会影响 PSW 的指令如下所示。

指令	标志		
	C	OV	AC
ADD	X	X	X
ADDC	X	X	X
SUBB	X	X	X
MUL	0	X	
DIV	0	X	
DA	X		
RRC	X		
RLC	X		
SETB C	1		

指令	标志		
	C	OV	AC
CLR C	0		
CPL C	X		
ANL C,bit	X		
ANL C,/bit	X		
ORL C,bit	X		
ORL C,/bit	X		
MOV C,bit	X		
CJNE	X		

“0”表示标志被清零,“1”表示标志被设置和“X”表示标志的状态取决于操作的结果。

SFR D0h	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
Reset	0	0	0	0	0	0	0	0

D0h.7 **CY**:ALU 进位标志

D0h.6 **AC**:ALU 辅助进位标志

D0h.5 **F0**:通用的使用者定义标志

D0h.4~3 **RS1,RS0**: (RS1,RS0) 的内容所启动之工作寄存器存储区为:

00:存储区 0 (00h~07h)

01:存储区 1 (08h~0Fh)

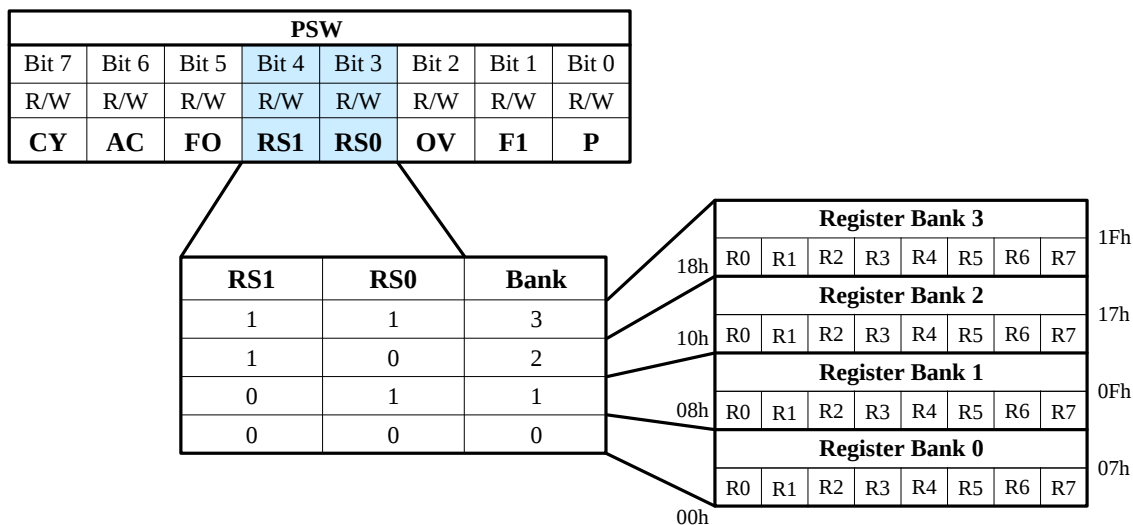
10:存储区 2 (10h~17h)

11:存储区 3 (18h~1Fh)

D0h.2 **OV**:ALU 溢出标志

D0h.1 **F1**:通用的使用者定义标志

D0h.0 **P**:奇偶标志。由硬件于每个指令周期设置/清零来表示在累加器“1”位之奇/偶数。



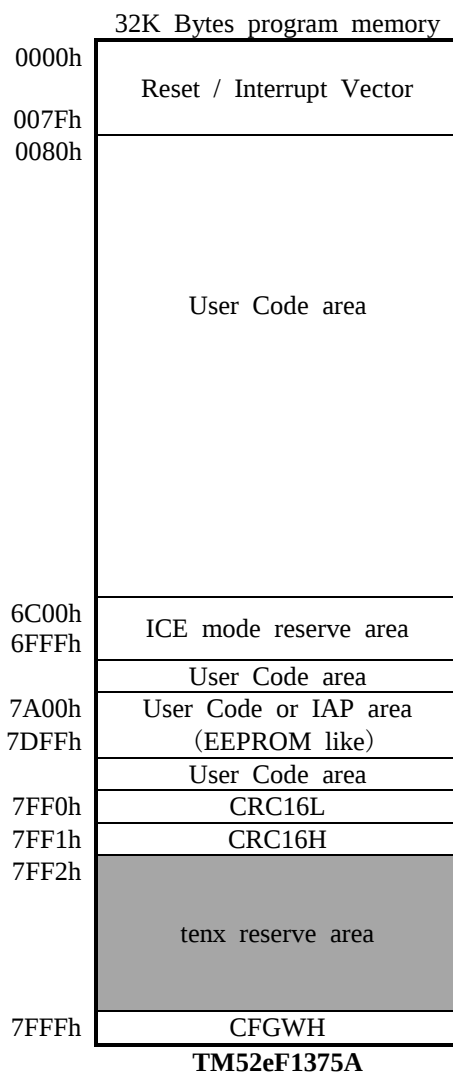
2. 存储器

2.1 程序存储器

该芯片具有一个用于 TM52eF1375A 的 32K 字节闪存程序存储器，它们可以支持应用程序编程(IAP)功能模式。Flash 的写入寿命至少为 100K 个周期。程序存储器地址的连续空间(0000h~7FFFh)被划分为多个扇区用于设备操作。

2.1.1 程序存储器的功能分区

程序存储器的最后一个字节被定义为芯片配置字 (CFGW)，它在上电复位 (POR) 时被加载到设备控制寄存器中。0000h~007Fh 被复位/中断向量占用，作为标准的 8051 定义。对于 TM52eF1375A，地址空间 7A00h~7DFFh 被定义为 IAP 区域。在在线仿真 (ICE) 模式下，用户还需要为 ICE 系统通信保留地址空间 6C00h~6FFFh。CRC16H / L 是校验和的保留区域。Tenx 可以提供 CRC 验证子例程。用户可以通过 CRC 验证子例程计算校验和，以与 CRC16H / L 进行比较，并检查 ROM 代码的有效性。



2.1.2 FLASH ICP 模式

FLASH 存储器可以通过 tenx 专有的烧录器 (TWR98/TWR99), 这需要至少四根线连接到该芯片以进行编程。如果用户想在目标电路板上的 FLASH 进行编程 (在电路编程, ICP), 这些引脚必须保留足够的自由来连接到烧录器, 最好不要连接电路; 如果要连接电路的话, 请参考相关 AP 资料。连接更多的烧录引脚可以提高烧录器的烧录效率和速度。

连接线数目	连接管脚
4-Wire	VCC, VSS, P3.0, P3.1

2.1.3 Flash IAP 模式 (类似于 EEPROM)

eF1375A 具有“应用程序内” (IAP) 功能, 该功能允许软件在 CPU 运行时从闪存读取数据/向闪存写入数据。有两页 (7A00h~7BFFh 和 7C00h~7DFFh) 可以进行 IAP 写入和擦除。使用 IAP 进行写入时, 需要先擦除然后再写入字节。删除后, 每个地址只能写入一次

IAP 擦除操作将一次从 7A00h~7BFFh 或 7C00h~7DFFh 擦除 512 个字节。在地址 7B2Dh 中写入任何值时, 可以擦除 512 个字节的 7A00h~7BFFh。同样, 在地址 7D69h 中写入任何值时, 可以擦除 512 个字节的 7C00h~7DFFh。在进行 IAP 写入或擦除之前, 应将两个 SFR (IAPWE 和 SWCMD) 设置如下表。在 IAP 写入或擦除之后, 应立即清除 IAPWE 和 SWCMD。

SFR 设定	IAP 写入	IAP 页擦除 (擦除 512 字节)	IAP 禁用
地址 7A00h~7BFFh	SWCMD = 65h IAPWE = 4Ah	SWCMD = 65h IAPWE = BAh	SWCMD = 0h IAPWE = 0h
地址 7C00h~7DFFh	SWCMD = 65h IAPWE = 4Ch	SWCMD = 65h IAPWE = BCh	SWCMD = 0h IAPWE = 0h

通过“MOVX @DPTR, A”指令, IAP 可以简单地写入和擦除, 并且通过“MOVC”指令可以轻松完成 IAP 读取。

地址	字节写入	页擦除
0000h ~ 79FFh	N	N
7A00h ~ 7BFFh	Y 字节写入	Y 页擦除
7C00h ~ 7DFFh	Y 字节写入	Y 页擦除
7E00h ~ 7FFFh	N	N

2.1.4 IAP 模式访问例程

通过“MOVX @DPTR, A”指令可以简单地完成闪存 IAP 写入，而 DPTR 包含从 7A00h 到 7DFEh 的目标闪存地址，而 ACC 包含正在写入的数据。仅当 IAPWE 和 SWCMD 设置为适当的值时，eF1375A 才接受 IAP 写入命令。Flash IAP 写入一个字节大约需要 20 μ s，擦除一页大约需要 2ms。当 IAP 写入或擦除 CPU 处于等待状态时，但是所有外围模块（定时器，LED 等）在写入/擦除时间内继续运行。IAP 写入后，软件必须处理未决的中断。eF1375A 具有内置的 IAP 超时功能，可避免写入失败状态。闪存 IAP 写入需要更高的 VCC 电压，VCC > 2.5V。

在写入 IAP 之前，用户应先关闭 LVR。

如何擦除页 7A00h~7BFFh

- (1) 将 DPTR 设置为 7B2Dh
- (2) 将 SWCMD 设置为 65h
- (3) 将 IAPWE 设置为 BAh
- (4) MOVX @DPTR, A（将任何数据写入 7B2Dh 以擦除 7A00h~7BFFh）

```
; IAP example code
; need 2.5V < VCC < 5.5V
SETB     LVRPD                ; Disable LVR
MOV      DPTR, #7B2Dh         ; DPTR=7B2Dh=target IAP address
MOV      SWCMD, #65h          ; IAP write enable
MOV      IAPWE, #BAh          ; IAP 7A00h~7BFFh erase enable
MOVX     @DPTR, A              ; write any data to 7B2Dh to erase 7A00h~7BFFh
                                   ; 7A00h~7BFFh convert to '1' after IAP erase
                                   ; 2ms H/W writing time, CPU wait
MOV      IAPWE, #00h          ; IAP write disable, immediately after IAP write
CLR      LVRPD                ; Enable LVR
```

如何擦除页 7C00h~7DFFh

- (1) 将 DPTR 设置为 7D69h
- (2) 将 SWCMD 设置为 65h
- (3) 将 IAPWE 设置为 BCh
- (4) MOVX @DPTR, A（将任何数据写入 7D69h 以擦除 7C00h~7DFFh）

```
; IAP example code
; need 2.5V < VCC < 5.5V
SETB     LVRPD                ; Disable LVR
MOV      DPTR, #7D69h         ; DPTR=7D69h=target IAP address
MOV      SWCMD, #65h          ; IAP write enable
MOV      IAPWE, #0BCh         ; IAP 7C00h~7DFFh erase enable
MOVX     @DPTR, A              ; write any data to 7D69h to erase 7C00h~7DFFh
                                   ; 7C00h~7DFFh convert to '1' after IAP erase
                                   ; 2ms H/W writing time, CPU wait
MOV      IAPWE, #00h          ; IAP write disable, immediately after IAP write
CLR      LVRPD                ; Enable LVR
```

如何写入 7A00h 至 7BFFh 范围内的字节

- (1) 将 DPTR 设置为 7A00h
- (2) 将 SWCMD 设置为 65h
- (3) 将 IAPWE 设置为 4Ah
- (4) MOVX @DPTR, A (将数据写入 7A00h)

```
; IAP example code
; need 2.5V < VCC < 5.5V
SETB    LVRPD                ; Disable LVR
MOV      DPTR, #7A00h         ; DPTR=7A00h=target IAP address
MOV      A, #5Ah              ; A=5Ah=target IAP write data
MOV      SWCMD, #65h          ; IAP write enable
MOV      IAPWE, #4Ah          ; IAP write range 7A00h~7BFFh enable

MOVX     @DPTR, A             ; Flash[7A00h] =5Ah, after IAP write
                                   ; 20us H/W writing time, CPU wait

MOV      IAPWE, #00h          ; IAP write disable, immediately after IAP write
CLR      A                    ; A=0
MOVC     A, @A+DPTR           ; A=5Ah
CLR      LVRPD                ; Enable LVR
```

如何写入 7C00h 至 7DFFh 范围内的字节

- (1) 将 DPTR 设置为 7C00h
- (2) 将 SWCMD 设置为 65h
- (3) 将 IAPWE 设置为 4Ch
- (4) MOVX @DPTR, A (将数据写入 7C00h)

```
; IAP example code
; need 2.5V < VCC < 5.5V
SETB    LVRPD                ; Disable LVR
MOV      DPTR, #7C00h         ; DPTR=7C00h=target IAP address
MOV      A, #5Ah              ; A=5Ah=target IAP write data
MOV      SWCMD, #65h          ; IAP write enable
MOV      IAPWE, #4Ch          ; IAP write range 7C00h~7DFFh enable

MOVX     @DPTR, A             ; Flash[7C00h] =5Ah, after IAP write
                                   ; 20us H/W writing time, CPU wait

MOV      IAPWE, #00h          ; IAP write disable, immediately after IAP write
CLR      A                    ; A=0
MOVC     A, @A+DPTR           ; A=5Ah
CLR      LVRPD                ; Enable LVR
```

Flash 7FFFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CFGWH	PROTN	XRSTEN	LVRE			–	MVCLOCKN	FRCPCSC

7FFFh.1 **MVCLOCKN**: 如果为 0, 则 MOVN 和 MOVX 无法访问从 0000h 到 01FFh 的地址。

SFR 97h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SWCMD	IAPEN/SWRST							
	–						WDT0	IAPEN
R/W	W						R	R
Reset	–						0	0

97h.7~0 **IAPEN (W)**:

写入 65h 以启用 IAP 写入/擦除;

写入其他值以禁用 IAP 写入/擦除。建议在访问 IAP 之后立即将其清除。

97h.0 **IAPEN (R)**: 该标志指示 IAP 可以访问或不能访问闪存扇区。

SFR C9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IAPWE	IAPWE							
	IAPWE	IAPTO	–					
R/W	R	R	W					
Reset	0	0	–					

C9h.7~0 **IAPWE (W)**:

写入 4Ah 以启用 IAP, 将一个字节写入 ROM [7A00~7BFF]

写入 4Ch 以启用 IAP, 将一个字节写入 ROM [7C00~7DFF]

写入 BAh 以启用 IAP 擦除 ROM [7A00~7BFF] 的 512 字节

写入 BCh 以启用 IAP 擦除 ROM [7C00~7DFF] 的 512 字节

写入其他值以禁用 IAP 写入/页面擦除

C9h.7 **IAPWE (R)**:

0: IAP 写入/页面擦除禁用

1: IAP 写入/页面擦除启用

C9h.6 **IAPTO (R)**:

IAP 超时标志, 当 IAP 超时发生时由硬件设置。当 IAPWE = 0 时由硬件清除。

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX2	WDTE		PWRSV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

F7h.2~1 **IAPTE**: IAP 写看门狗定时器使能

00: 禁用

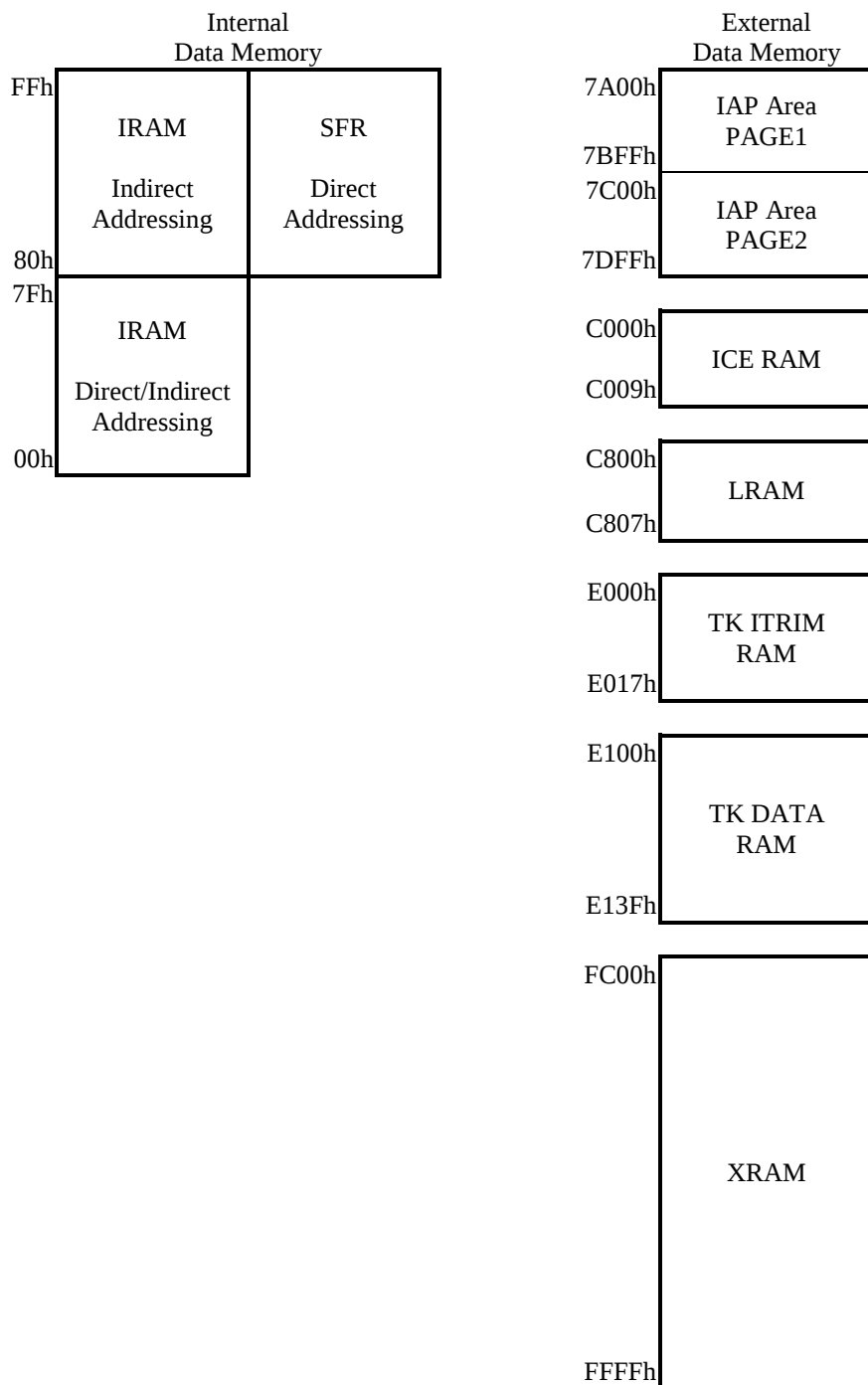
01: 等待 0.8 ms 触发看门狗超时标志, 并退出写入失败状态

10: 等待 3.2ms 触发看门狗超时标志, 并退出写入失败状态

11: 等待 6.4 ms 触发看门狗超时标志, 并退出写入失败状态

2.2 数据存储

作为标准 8051, 该芯片同时具有内部和外部数据存储空间。内部数据存储空间由 256 字节的 IRAM 和 SFR 组成, 可通过丰富的指令集对其进行访问。外部数据存储空间由与程序存储器共享的 1024 字节 XRAM, 8 字节 LCD RAM, 10 字节 ICE RAM, 24 字节 TK ITRIM RAM 和 64 字节 TK DATA RAM, 只能由 MOVX 指令访问。



2.2.1 IRAM

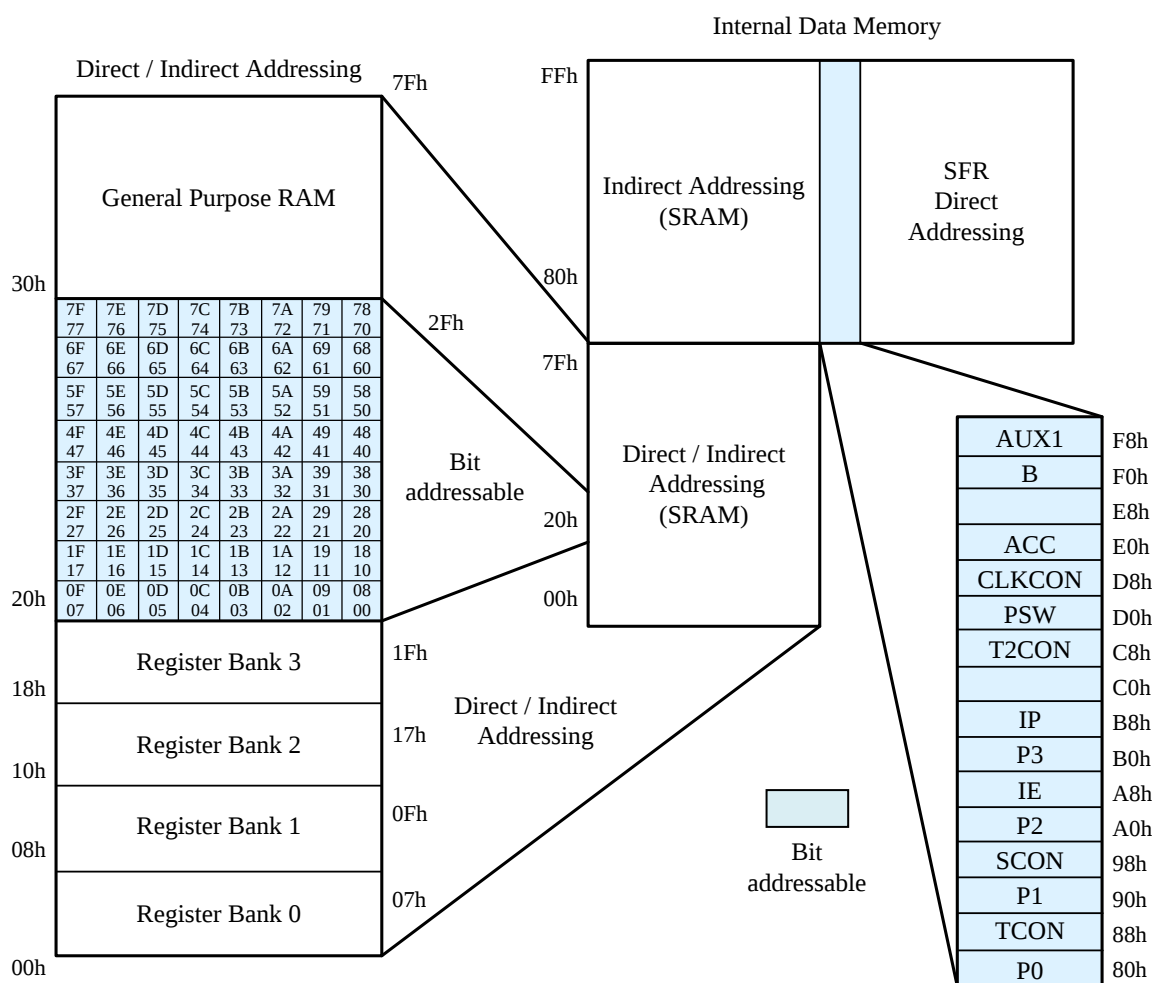
IRAM 位于 8051 内部数据存储空间中。整个 256 字节的 IIRAM 可以通过间接寻址访问，但是只有低 128 字节的 IIRAM 可以通过直接寻址访问。有四个可直接寻址的寄存器组(通过 PSW 切换)，它们从 00h 到 1Fh 占据 IIRAM 空间。地址 20h 至 2Fh 16 字节 IIRAM 空间是位可寻址的。IIRAM 可用作暂存器或程序堆栈。

2.2.2 XRAM

XRAM 位于 8051 外部数据存储器空间(地址从 FC00h 到 FFFFh)。1024 字节 XRAM 只能通过“MOVX”指令存取。

2.2.3 SFRs

可通过特殊功能寄存器（SFR）访问芯片的所有外围功能模块，例如 I/O 端口，定时器和 UART 操作。这些寄存器占据直接数据存储器空间位置的高 128 字节，范围为 80h 至 FFh。有 14 个可位寻址的 SFR（这意味着单个字节内的八个独立位是可寻址的），例如 ACC，B 寄存器，PSW，TCON，SCON 等。其余的 SFR 只能是字节可寻址的。SFR 提供与芯片资源和外围设备的控制和数据交换。TM52 系列微控制器提供具有标准 8051 指令集兼容性的完整二进制代码。除了标准的 8051 SFR，该芯片还实现了额外的 SFR，这些 SFR 用于配置和访问该芯片独有的子系统，例如 ADC / LCD。



	8/0	9/1	A/2	B/3	C/4	D/5	E/6	F/7
F8h	AUX1							
F0h	B	CRCDL	CRCDH	CRCIN		CFGBG	CFGWL	AUX2
E8h		SIADR	SICON	SIRCD1	SITXRCD2			PWRCON
E0h	ACC	MICON	MIDAT			EFTCON	EXA	EXB
D8h	CLKCON	PWM0PRDH	PWM0PRDL	PWM1PRDH	PWM1PRDL	PWM2PRDH	PWM2PRDL	
D0h	PSW	PWM0DH	PWM0DL	PWM1DH	PWM1DL	PWM2DH	PWM2DL	
C8h	T2CON	IAPWE	RCP2L	RCP2H	TL2	TH2	EXA2	EXA3
C0h		TKPINSEL0	TKPINSEL1	TKPINSEL2		ATKCH0	ATKCH1	ATKCH2
B8h	IP	IPH	IP1	IP1H	SPCON	SPSTA	SPDAT	LVDS
B0h	P3	LEDCON	LEDCON2	LEDCON3	TKTMRL	TKCON2		
A8h	IE	INTE1	ADCDL	ADCDH		TKCON	CHSEL	P0ADIE
A0h	P2	PWMCON	P1MODL	P1MODH	P3MODL	P3MODH	PINMOD	TKCHS
98h	SCON	SBUF					PWMOE	PWMCLR
90h	P1	P0OE	P0LOE	P2MOD	OPTION	INTFLG	P1WKUP	SWCMD
88h	TCON	TMOD	TL0	TL1	TH0	TH1	SCON2	SBUF2
80h	P0	SP	DPL	DPH	INTEX	INTEXF	INTPWM	PCON

3. 低电压复位和低电压检测

该芯片提供 LVR 和低电压检测（LVD）功能。CFGWH 可以选择 8 级 LVR，SFR LVDS 可以选择 16 级 LVD。SFR PWRSAV / LVRPD 位也影响 LVR 功能，如下表所示。

操作模式	SFR		CFGWH	低电压复位 (LVR)	功能	备注
	LVRPD	PWRSAV	LVRE			
快钟模式 慢钟模式	0	X	000	ON	LV Reset 2.5V	
	0	X	001	ON	LV Reset 2.7V	
	0	X	010	ON	LV Reset 3.0V	
	0	X	011	ON	LV Reset 3.2V	
	0	X	100	ON	LV Reset 3.4V	
	0	X	101	ON	LV Reset 3.7V	
	0	X	110	ON	LV Reset 3.9V	
	0	X	111	ON	LV Reset 4.2V	
空闲模式 停止模式 暂停模式	0	0	000	ON	LV Reset 2.5V	空闲: 150uA 暂停: 60uA 停止: 56uA
	0	0	001	ON	LV Reset 2.7V	
	0	0	010	ON	LV Reset 3.0V	
	0	0	011	ON	LV Reset 3.2V	
	0	0	100	ON	LV Reset 3.4V	
	0	0	101	ON	LV Reset 3.7V	
	0	0	110	ON	LV Reset 3.9V	
	0	0	111	ON	LV Reset 4.2V	
空闲模式	0	1	XXX	ON	POR 2.4V	130uA
停止模式 暂停模式	0	1	XXX	OFF	Disable	暂停: 11uA 停止: 7.7uA
快钟模式 慢钟模式 空闲模式	1	X	XXX	ON	POR 2.4V	空闲: 130uA
停止模式 暂停模式	1	X	XXX	OFF	Disable	暂停: 11uA 停止: 7.7uA

注：暂停模式会比停止模式多了 SRC 启用的耗电流约 2~4uA。

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX2	WDTE		PWRSAV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

F7h.5 设置 1 可降低空闲/暂停/停止模式下芯片的功耗

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.3 LVRPD: 低电压复位

0: 使能

1: 禁用

SFR BFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDS	LVDIE	LVDO	—	—	LVDS			
R/W	R/W	R	—	—	R/W	R/W	R/W	R/W
Reset	0	0	—	—	0	0	0	0

BFh.7 **LVDIE**: 低压检测中断使能
0: 禁用
1: 启用（注意：EXLVDIE 必须同时为 1 才能生成 LVD 中断）

BFh.6 **LVDO**: 低压检测输出

BFh.3~0 **LVDS**: 低压检测选择（在空闲/暂停/停止模式下自动禁用）

0000: 将 LVD 设置为 2.5V
0001: 将 LVD 设置为 2.6V
0010: 将 LVD 设置为 2.7V
0011: 将 LVD 设置为 2.8V
0100: 将 LVD 设置为 3.0V
0101: 将 LVD 设置为 3.1V
0110: 将 LVD 设置为 3.2V
0111: 将 LVD 设置为 3.3V
1000: 将 LVD 设置为 3.4V
1001: 将 LVD 设置为 3.6V
1010: 将 LVD 设置为 3.7V
1011: 将 LVD 设置为 3.8V
1100: 将 LVD 设置为 3.9V
1101: 将 LVD 设置为 4.0V
1110: 将 LVD 设置为 4.2V
1111: 将 LVD 设置为 4.3V

Flash 7FFFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CFGWH	PROTN	XRSTEN	LVRE			—	MVCLOCKN	FRCPSC

7FFFh.5~3 **LVRE**: 低电压复位功能选择

000: 将 LVR 设置为 2.5V
001: 将 LVR 设置为 2.7V
010: 将 LVR 设置为 3.0V
011: 将 LVR 设置为 3.2V
100: 将 LVR 设置为 3.4V
101: 将 LVR 设置为 3.7V
110: 将 LVR 设置为 3.9V
111: 将 LVR 设置为 4.2V

4. 复位

该芯片有五种类型的复位方法。上电复位 (POR)，外部引脚复位 (XRST)，软件复位 (SWRST)，看门狗定时器复位 (WDTR) 和低电压复位 (LVR)。CFGW 控制复位功能。复位后 SFR 是返回到默认值。

4.1 上电复位

上电复位后，设备停留在复位状态，进行 40 ms 的芯片预热，然后从 Flash 的最后六个字节下载 CFGW 寄存器 (其它复位不会重新加载 CFGW)。上电复位需要 VCC 引脚的电压先放电至接近 VSS 电平，然后再上升超过 2.5V。

4.2 外部引脚复位

外部引脚复位为低电平有效。RSTn 引脚需要保持至少两个 SRC 时钟周期长到芯片可采样。外部引脚复位可以由 CFGW 使能/禁止。

4.3 软件复位

软件复位是通过将数据 56h 写入 SFR 中的 97h 地址来产生。

4.4 看门狗定时器复位

WDT 溢出复位被 SFR F7h 来控制。WDT 使用 SRC 作为计数时基。它在快钟/慢钟模式运行,在空闲/暂停/停止模式下可选运行或停止。看门狗定时器溢出速度可通过 WDTPSC SFR 定义。WDT 由 CLRWDT SFR 或复位清零。

4.5 低电压复位

该芯片提供 8 个低电压复位 (LVR) 选项,用户可由 CFGWH 作出选择。

Flash 7FFFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CFGWH	PROTN	XRSTEN	LVRE			—	MVCLOCKN	FRCPSC

7FFFh.6 **XRSTEN:** 外部引脚复位控制
0: 启用外部引脚复位
1: 禁用外部引脚复位

7FFFh.5~3 **LVRE:** 低电压复位功能选择
000: 将 LVR 设置为 2.5V
001: 将 LVR 设置为 2.7V
010: 将 LVR 设置为 3.0V
011: 将 LVR 设置为 3.2V
100: 将 LVR 设置为 3.4V
101: 将 LVR 设置为 3.7V
110: 将 LVR 设置为 3.9V
111: 将 LVR 设置为 4.2V

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	UART1W	–	WDTPSC		ADCKS		TM3PSC	
R/W	R/W	–	R/W		R/W		R/W	
Reset	0	–	0	0	0	0	0	0

94h.5~4 **WDTPSC:**看门狗定时器预分频时间选择

00:400ms WDT 溢出率

01:200ms WDT 溢出率

10:100ms WDT 溢出率

11:50ms WDT 溢出率

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTFLG	LVDIF	–	TKIF	ADIF	–	–	P1IF	TF3
R/W	R/W	–	R/W	R/W	–	–	R/W	R/W
Reset	0	–	0	0	–	–	0	0

95h.7 **LVDIF:**低电压检测标志

由硬件设置。软件将 7Fh 写入 INTFLG 以清除该标志。

SFR 97h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SWCMD	IAPALL/SWRST							
R/W	W						R/W	R/W
Reset	–						–	0

97h.7~0 **SWRST:**写入 56h 以产生软件复位

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX2	WDTE		PWRSV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

F7h.7~6 **WDTE:** 看门狗定时器复位控制

0x: 看门狗定时器复位关闭

10: 看门狗定时器复位于快钟/慢钟模式下使能,空闲/暂停/停止模式时禁止

11: 看门狗定时器复位始终启用

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.7 **CLRWDT:** 设置以清除看门狗定时器, 硬件自动在一个时钟周期清除它

F8h.3 **LVRPD:** 低压复位功能选择

0: 使能

1: 禁用

5. 时钟电路和工作模式

5.1 时钟电路

该芯片设计有双时钟系统。在运行时,用户可以直接切换从快钟到慢钟或由慢到快。它可以选择除以 1,2,4 或 16 的时钟分频器。快时钟可选用 FXT (快速晶振,1~18 MHz) 或 FRC (18.432MHz)。慢时钟可以选用 SXT (慢速晶振,32 KHz) 或 SRC (慢速内部 RC, 80 KHz)。快钟模式和慢钟模式被定义为快/慢时钟的 CPU 运行速度。

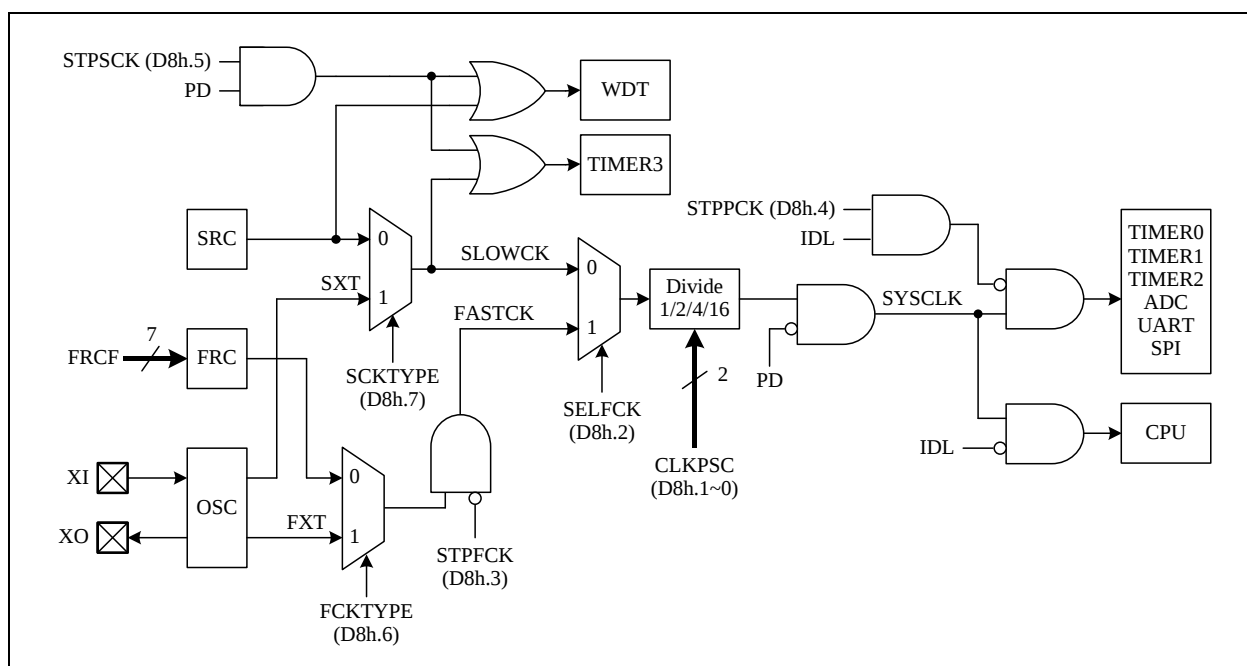
复位后,该设备在慢钟模式 80 KHz 的 SRC 运行。软件应该正确选择安全的芯片运行时钟速率。较高的 V_{CC} 允许芯片在更高的系统时钟频率运行。在典型的情况下,18MHz 的系统时钟频率需要 $V_{CC}>2.5V$ 。

该芯片有两个外部振荡器连接到 XI/XO 引脚。它依赖于外部电路提供时钟信号、频率的稳定,例如一个独立的振荡器,石英晶体或陶瓷谐振器。在快钟模式中,快速振荡器可以使用的范围为 1~18 MHz。在慢钟模式下,慢速振荡器只能使用 32.768 KHz 的时钟频率。

CLKCON SFR 控制系统时钟的正常运行。硬件自动阻断软件异常设置该寄存器。软件只能在快钟模式下改变慢时钟类型,在慢钟模式下改变快时钟类型。千万不要同时写 STPFCK=1 & SELFCK=1。建议在写这个 SFR 时一次只写一个位。

如果使用者想要将 F_{sys} 从慢速时钟切换到 FXT, 用户应该按照以下步骤操作

1. 设置 FCKTYPE (D8h.6)
2. 等待 2ms 直到 FXT 振荡稳定
3. 设置 SELFCK (D8h.2)



时钟结构

该芯片还可以将“系统时钟 2 分频”信号 (CKO) 输出到 P1.4 引脚。CKO 引脚的输出设置由 TCOE SFR 控制 (请参见第 7 节)。

注: 由于 CLKPSC 延迟, 在将慢时钟切换为快时钟之前, 它需要等待 16 个时钟周期 (最大)。另请参阅 AP-TM52XXXX_01S 和 AP-TM52XXXX_02S 有关系统时钟的应用笔记。

SYSCLK	CLKCON (D8h)			
	bit7 SCKTYPE	bit6 FCKTYPE	bit3 STPFCK	bit2 SELFCK
Fast FXT	0/1	1	0	1
Fast FRC	0/1	0	0	1
Slow SXT	1	0/1	0/1	0
Slow SRC	0	0/1	0/1	0
Fast type change	0/1	0 $\leftarrow$ $\rightarrow$ 1	0/1	0
Slow type change	0 $\leftarrow$ $\rightarrow$ 1	0/1	0	1
Stop FRC/FXT	0/1	0/1	0 $\rightarrow$ 1	0
Switch to FRC/FXT	0/1	0/1	0	0 $\rightarrow$ 1
Switch to SRC/SXT	0/1	0/1	0	1 $\rightarrow$ 0

SFR F6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CFGWL	—	FRCF						
R/W	—	R/W						
Reset	—	—	—	—	—	—	—	—

F6h.6~0 **FRCF**: FRC频率调整
00h= 频率最低, 7Fh=频率最高.

SFR D8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCON	SCKTYPE	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	1	0	0	0	1	1

D8h.7 **SCKTYPE**: 慢时钟类型。该位只能在快钟模式 (SELFCK=1) 时改变。

0: SRC, P2.1, P2.0 为 I/O 引脚

1: SXT, P2.1, P2.0 为晶振引脚

D8h.6 **FCKTYPE**: 快时钟类型。该位只能在慢钟模式 (SELFCK=0) 时改变。

0: FRC, P2.1, P2.0 为 I/O 引脚

1: FXT, P2.1, P2.0 为晶振引脚

D8h.5 **STPSCK**: 设为 1, 停止慢钟在停止模式。

D8h.4 **STPPCK**: 设为 1, 停止 UART/Timer0/Timer1/Timer2/ADC 在空闲模式的时钟。

D8h.3 **STPFCK**: 设为 1, 停止快时钟以节省慢钟/空闲模式的电力。

该位只能在慢钟模式时改变。

D8h.2 **SELFCK**: 系统时钟源选择。此位只有当 STPFCK=0 才可以改变。

0: 慢时钟

1: 快时钟

D8h.1~0 **CLKPSC**: 系统时钟分频器, 生效延迟最大为 16 个时钟周期

00: 系统时钟是快/慢时钟除以 16

01: 系统时钟是快/慢时钟除以 4

10: 系统时钟是快/慢时钟除以 2

11: 系统时钟是快/慢时钟除以 1

5.2 操作模式

这个设备有五种操作模式。**快钟模式**被定义为在快时钟速度运行的 CPU。**慢钟模式**被定义为慢时钟速度运行的 CPU。当系统时钟速度较低,功耗较低。

空闲模式通过设置 PCON 中的 IDL 位进入。快或慢时钟都可设置为在空闲模式下的系统时钟源,但慢时钟的省电越好。在空闲模式下,CPU 进入睡眠,而片上外围设备保持活跃。在 CLKCON SFR 中的“STPPCK”位可以设置为进一步降低空闲模式下的电流。如果 STPPCK=1,Timer0/1/2,ADC 和 UART 在空闲模式时停止。较慢的系统时钟频率也有助于节省电流。它可以通过设置 CLKPSC SFR 降低系统时钟频率来实现。空闲模式是通过复位或使能的中断来唤醒。

停止模式是通过设置 PCON 中的 PD 位及 CLKCON 中的 STPSCK 位进入。这种模式在标准的 8051 是所谓的“省电”模式。在停止模式下,除了 WDT 时钟可能开启,其他所有时钟停止。停止模式可以通过复位或引脚唤醒来结束。

暂停模式是通过设置 PCON 中的 PD 位及清除 CLKCON 中的 STPSCK 位进入。在暂停模式下,所有时钟都停止,但如果启用了 Timer3 和 WDT,则它们可能处于开启状态。暂停模式可以通过复位,引脚唤醒或 Timer3 中断来终止。

注: 如果 INTn 引脚是低电平且该唤醒功能启用,则芯片无法进入暂停/停止模式。(INTn=0 and Exn=1,n=0,1,9)

注: 固件必须关闭 Bandgap 以获得最小电流消耗 (VBGOUT=0)

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	SMOD	—	—	—	GF1	GF0	PD	IDL
R/W	R/W	—	—	—	R/W	R/W	R/W	R/W
Reset	0	—	—	—	0	0	0	0

87h.1 **PD:** 停止位,如果 1 进入暂停/停止模式。

87h.0 **IDL:** 空闲位,如果 1 进入空闲模式。

SFR D8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCON	SCKTYPE	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	1	0	0	0	1	1

D8h.7 **SCKTYPE:** 慢时钟类型。该位只能在快钟模式 (SELFCK=1) 时改变。

0: SRC, P2.1, P2.0 为 I/O 引脚

1: SXT, P2.1, P2.0 为晶振引脚

D8h.6 **FCKTYPE:** 快时钟类型。该位只能在慢钟模式 (SELFCK=0) 时改变。

0: FRC, P2.1, P2.0 为 I/O 引脚

1: FXT, P2.1, P2.0 为晶振引脚

D8h.5 **STPSCK:** 设为 1, 停止慢钟在停止模式。

D8h.4 **STPPCK:** 设为 1, 停止 UART/Timer0/Timer1/Timer2/ADC 在空闲模式的时钟。

D8h.3 **STPFCK:** 设为 1, 停止快时钟以节省慢钟/空闲模式的电力。
该位只能在慢钟模式时改变。

D8h.2 **SELFCK:** 系统时钟源选择。此位只有当 STPFCK=0 才可以改变。
0: 慢时钟 1: 快时钟

D8h.1~0 **CLKPSC:** 系统时钟分频器, 生效延迟最大为 16 个时钟周期
00: 系统时钟是快/慢时钟除以 16
01: 系统时钟是快/慢时钟除以 4
10: 系统时钟是快/慢时钟除以 2
11: 系统时钟是快/慢时钟除以 1

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX2	WDTE		PWRSV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	1	1	0

F7h.4 **VBGOUT:** V_{BG} 电压输出至 P3.2

0: 关闭

1: 启用

6. 中断和唤醒

该芯片有 14 源四级中断优先级结构。所有的中断都可以从空闲模式唤醒 CPU,但只有引脚中断可以从暂停/停止模式下唤醒 CPU。每个中断源都有自己的使能控制位。不管它的中断使能控制位是 0 还是 1,中断事件将设置其个别的中断标志。中断向量和标志列表如下。

Vector	Flag	Description
0003	IE0	INT0 外部引脚中断(可以唤醒暂停/停止模式)
000B	TF0	Timer0 中断
0013	IE1	INT1 外部引脚中断(可以唤醒暂停/停止模式)
001B	TF1	Timer1 中断
0023	RI+TI	串口(UART1)中断
002B	TF2+EXF2	Timer2 中断
0033	—	保留为 ICE 模式使用
003B	TF3	Timer3 中断
0043	P1IF	端口 1 外部引脚电平变化中断(可以唤醒暂停/停止模式)
004B	IE2~IE9 LVDIF	INT2~INT9 外部引脚中断(可以唤醒暂停/停止模式) LVD 中断
0053	ADIF TKIF	ADC/触摸按键中断
005B	SPIF+WCOL+MODE	SPI 中断
0063	RI2+TI2	串口(UART2) 中断
006B	MIIF TXDF RCD2F RCD1F	I ² C 中断
0073	PWM0IF PWM1IF PWM2IF	PWM0~2 中断

中断向量和标志

6.1 中断使能和优先级控制

IE 和 INTE1 的 SFR 决定中断是否由 CPU 提供服务。IP,IPH,IP1 和 IP1H 的 SFR 决定中断优先级。中断会被服务,需要相同或更高优先级的中断尚未被服务。如果相同或更高优先级的中断被服务时,新的中断将等待被服务,直到它之前的服务完成。如果较低优先级中断正被服务时,将被停止,开始新的中断服务。当新的中断结束后,被停止的较低优先级的中断才会被完成。

6.2 关于中断子程序的建议

PWM 的周期和占空比为 16 位操作。在读写 PWMxDH, PWMxDL, PWMxPRDH 和 PWMxPRDL 的高字节和低字节时,应避免中断。如果正在读取和写入这些 16 位 SFR 的同时发生中断。这些 SFR 又会在在中断中读写,容易引起读写错误。对于 16 位 PWM 周期和读写占空比,建议仅在主程序中更新数据,或仅在中断中更新数据,以避免可能的错误。

SFR 84h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTEX	EX9	EX8	EX7	EX6	EX5	EX4	EX3	EX2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

84h.7~0 **EX9~EX2:** 外部 INT9~INT2 引脚中断使能和暂停/停止模式唤醒使能。

0: 禁用 INTx 引脚中断和停止模式唤醒

1: 使能 INTx 引脚中断和停止模式唤醒，无论 EA 为 0 还是 1，都可以将 CPU 从停止模式唤醒。（注意：EXLVDIE 必须同时为 1 才能产生 INTx 中断和唤醒）

SFR 96h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1WKUP	P1WKUP							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

96h.7~0 **P1WKUP:**P1.7~P1.0 个别引脚唤醒/中断使能控制

0: 关闭

1: 开启

SFR 9Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMOE	PWM1IE	PWM0IE	—	—	—	PWM2OE	PWM1OE	PWM0OE
R/W	R/W	R/W	—	—	—	R/W	R/W	R/W
Reset	0	0	—	—	—	0	0	0

9Eh.7 **PWM1IE:** PWM1 中断使能

0: 禁用

1: 使能（注意：PWMIE 必须同时为 1 才能产生 PWM 中断）

9Eh.6 **PWM0IE:** PWM0 中断使能

0: 禁用

1: 使能（注意：PWMIE 必须同时为 1 才能产生 PWM 中断）

SFR 9Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCLR	PWM2IE	—	—	—	—	PWM2CLR	PWM1CLR	PWM0CLR
R/W	R/W	—	—	—	—	R/W	R/W	R/W
Reset	0	—	—	—	—	0	0	0

9Fh.7 **PWM2IE:** PWM2 中断使能

0: 禁用

1: 使能（注意：PWMIE 必须同时为 1 才能产生 PWM 中断）

SFR A8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IE	EA	—	ET2	ES	ET1	EX1	ET0	EX0
R/W	R/W	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	—	0	0	0	0	0	0

- A8h.7 **EA**:总中断使能控制
0:禁用所有中断
1:每个中断通过其各个中断控制位使能或禁止
- A8h.5 **ET2**:Timer2 中断使能控制
0:禁用 Timer2 中断
1:允许 Timer2 中断
- A8h.4 **ES**:串口 (UART1) 中断使能控制
0:禁用串口 (UART1) 中断
1:允许串口 (UART1) 中断
- A8h.3 **ET1**:Timer1 中断使能控制
0:禁用 Timer1 中断
1:允许 Timer1 中断
- A8h.2 **EX1**:INT1 引脚中断和暂停/停止模式唤醒使能控制
0:禁用 INT1 引脚中断和暂停/停止模式唤醒
1:允许 INT1 引脚中断和暂停/停止模式唤醒,不管 EA 为 0 或 1,都可从暂停/停止模式下唤醒 CPU。
- A8h.1 **ET0**:Timer0 中断使能
0:禁用 Timer0 中断
1:允许 Timer0 中断
- A8h.0 **EX0**:INT0 引脚中断和暂停/停止模式唤醒使能控制
0:禁用 INT0 引脚中断和暂停/停止模式唤醒
1:允许 INT0 引脚中断和暂停/停止模式唤醒,不管 EA 为 0 或 1,都可从暂停/停止模式下唤醒 CPU。

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE1	PWMIE	I2CE	ES2	SPIE	ADTKIE	EXLVDIE	P1IE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- A9h.7 **PWMIE:** PWM0~PWM2 中断使能
0: 禁用 PWM0~PWM2 中断
1: 允许 PWM0~PWM2 中断
- A9h.6 **I2CE:** I²C (主/从) 中断使能
0: 禁用 I²C interrupt
1: 允许 I²C interrupt
- A9h.5 **ES2:** Serial Port (UART2) 中断使能
0: 禁用 Serial Port (UART2) 中断
1: 允许 Serial Port (UART2) 中断
- A9h.4 **SPIE:** SPI 中断使能
0: 禁用 SPI 中断
1: 允许 SPI 中断
- A9h.3 **ADTKIE:** ADC/触摸按键中断使能控制
0:禁用 ADC/触摸按键
1:允许 ADC/触摸按键
- A9h.2 **EXLVDIE:** 外部 INT2~INT9 和 LVD 中断使能以及暂停/停止模式唤醒使能
0: 禁用 INT2~INT9 引脚中断和暂停/停止模式唤醒
禁用 LVD 中断
1: 使能 INT2~INT9 引脚的中断和暂停/停止模式唤醒, 无论 EA 为 0 还是 1, 都可以将 CPU 从暂停/停止模式唤醒
使能 LVD 中断
- A9h.1 **P1IE:** 端口 1 引脚电平变化中断使能。 该位不会影响端口 1 引脚的暂停/停止模式唤醒功能。
0:禁用端口 1 引脚电平变化中断
1:允许端口 1 引脚电平变化中断
- A9h.0 **TM3IE:** Timer3 中断使能控制
0:禁用 Timer3 中断
1:允许 Timer3 中断

SFR B9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IPH	—	—	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
R/W	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	0	0	0	0	0	0

SFR B8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP	—	—	PT2	PS	PT1	PX1	PT0	PX0
R/W	—	—	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	0	0	0	0	0	0

B9h.5,B8h.5 **PT2H,PT2:** Timer2 中断优先级控制。(PT2H,PT2)=

00:0 级(最低优先级)

01:1 级

10:2 级

11:3 级(最高优先级)

B9h.4,B8h.4 **PSH,PS:** 串口 1 (UART1) 中断优先级控制。定义如上。

B9h.3,B8h.3 **PT1H,PT1:** Timer1 中断优先级控制。定义如上。

B9h.2,B8h.2 **PX1H,PX1:** INT1 引脚中断优先级控制。定义如上。

B9h.1,B8h.1 **PT0H,PT0:** Timer0 中断优先级控制。定义如上。

B9h.0,B8h.0 **PX0H,PX0:** INT0 引脚中断优先级控制。定义如上。

SFR BBh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP1H	PPWMH	PI2CH	PS2H	PSPIH	PADTKIH	PX2_9LVDH	PP1H	PT3H
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR BAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IP1	PPWM	PI2C	PS2	PSPI	PADTKI	PX2_9LVD	PP1	PT3
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

BBh.7,BAh.7 **PPWMH, PPWM:** PWM0~PWM2 中断优先级控制。定义如上

BBh.6,BAh.6 **PI2CH, PI2C:** I2C (Master/Slave) 中断优先级控制。定义如上

BBh.5,BAh.5 **PS2H, PS2:** 串口 2 (UART2) 中断优先级控制。定义如上

BBh.4,BAh.4 **PSPIH, PSPI:** SPI 中断优先级控制。定义如上

BBh.3,BAh.3 **PADTKIH,PADTKI:** ADC/触摸按键中断优先级控制。定义如上。

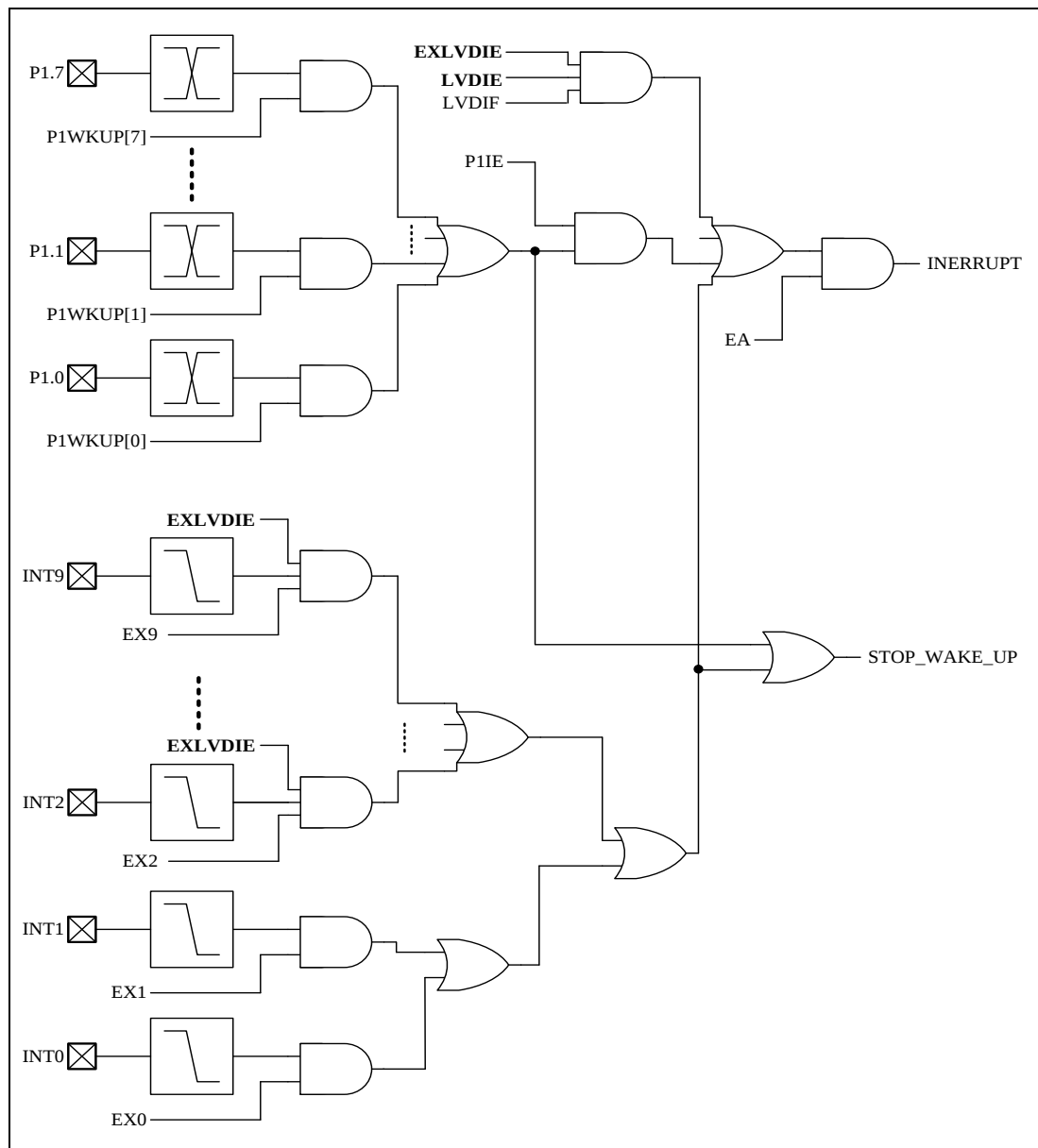
BBh.2,BAh.2 **PX2_9LVDH, PX2_9LVD:** 外部 INT2~INT9 引脚和 LVD 中断优先级控制。定义如上

BBh.1,BAh.1 **PP1H,PP1:** 端口 1 引脚电平变化中断优先级控制。定义如上。

BBh.0,BAh.0 **PT3,PT3:** Timer3 中断优先级控制。定义如上。

6.3 引脚中断和 LVD 中断

引脚中断包括 INT0~INT9 和端口 1 电平变化中断。INT0~INT9 和端口 1 也具有暂停/停止模式唤醒功能。INT0 和 INT1 为 8051 标准触发的下降沿或低电平。INT2~INT9 下降沿触发，并且端口 1 状态改变触发端口 1 电平变化中断。LVD 中断可用于检测 VCC 电压电平并产生中断。



引脚中断和唤醒

注： 如果INTn 引脚为低电平并且使能了唤醒功能，则芯片无法进入暂停/停止模式。(INTn = 0 和 EXn = 1, n = 0~9)

SFR 84h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTEX	EX9	EX8	EX7	EX6	EX5	EX4	EX3	EX2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

84h.7~0 **EX9~EX2:** 外部 INT9~INT2 引脚中断使能和暂停/停止模式唤醒使能

0: 禁用 INTx 引脚中断和暂停/停止模式唤醒

1: 使能 INTx 引脚中断和暂停/停止模式唤醒, 无论 EA 为 0 还是 1, 都可以将 CPU 从暂停/停止模式唤醒。(注意: EXLVDIE 必须同时为 1 才能产生 INTx 中断唤醒)

SFR 85h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTEXF	IE9	IE8	IE7	IE6	IE5	IE4	IE3	IE2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

85h.7~0 **IE9~2:** 外部中断 INT9~INT2 边沿标志。

无论 EXx 是 0 还是 1, 只要检测到 INTx 引脚的下降沿, 就由硬件进行设置。

软件写 0 清除中断标志, 在中断服务程序结束后不自动清除。

SFR 88h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

88h.3 **IE1:** 外部中断 1(INT1 引脚)边沿标志

设置于硬件检测到 INT1 引脚下降沿时,不管 EX1 为 0 或 1。

程序执行中断服务时,它会被自动清除。

88h.2 **IT1:** 外部中断 1 控制位

0:低电平有效(电平触发)的 INT1 引脚

1:下降沿有效(边沿触发)的 INT1 引脚

88h.1 **IE0:** 外部中断 0(INT0 引脚)边沿标志

设置于硬件检测到 INT0 引脚下降沿时,不管 EX0 为 0 或 1。

程序执行中断服务时,它会被自动清除。

88h.0 **IT0:** 外部中断 0 控制位

0:低电平有效(电平触发)的 INT0 引脚

1:下降沿有效(边沿触发)的 INT0 引脚

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTFLG	LVDIF	—	TKIF	ADIF	—	—	P1IF	TF3
R/W	R	—	R/W	R/W	—	—	R/W	R/W
Reset	—	—	0	0	—	—	0	0

95h.7 **LVDIF:** LVD 中断标志

通过软件置 1, 软件可以将 7Fh 写入 INTFLG 以清除该位。。

95h.1 **P1IF:** 端口 1 引脚电平变化中断标志

设置于硬件检测到 P1 引脚的状态变化时,且其相应的中断使能位(P1WKUP)被设置。

程序执行中断服务时,它会被自动清除。

软件也可以写 FDh 到 INTFLG 以清除该标志。(注)

注: 软件可以写 0 清除 INTFLG 中的标志,但写 1 没有任何效果。

SFR 96h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1WKUP	P1WKUP							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

96h.7~0 **P1WKUP**: P1.7~P1.0 引脚唤醒/中断使能控制

0: 禁用

1: 使能

SFR A8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IE	EA	–	ET2	ES	ET1	EX1	ET0	EX0
R/W	R/W	–	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	–	0	0	0	0	0	0

A8h.7 **EA**: 总中断使能控制

0: 禁用所有中断

1: 每个中断通过其各个中断控制位使能或禁止

A8h.2 **EX1**: INT1 引脚中断和暂停/停止模式唤醒使能控制

0: 禁用 INT1 引脚中断和暂停/停止模式唤醒

1: 允许 INT1 引脚中断和暂停/停止模式唤醒, 不管 EA 为 0 或 1, 都可从停止模式下唤醒 CPU。

A8h.0 **EX0**: INT0 引脚中断和暂停/停止模式唤醒使能控制

0: 禁用 INT0 引脚中断和暂停/停止模式唤醒

1: 允许 INT0 引脚中断和暂停/停止模式唤醒, 不管 EA 为 0 或 1, 都可从停止模式下唤醒 CPU。

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE1	PWMIE	I2CE	ES2	SPIE	ADTKIE	EXLVDIE	P1IE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

A9h.2 **EXLVDIE**: 外部 INT2~INT9 和 LVD 中断使能以及暂停/停止模式唤醒使能

0: 禁用 INT2~INT9 引脚中断和暂停/停止模式唤醒

禁用 LVD 中断

1: 使能 INT2~INT9 引脚的中断和暂停/停止模式唤醒, 无论 EA 为 0 还是 1, 都可以将 CPU 从暂停/停止模式唤醒

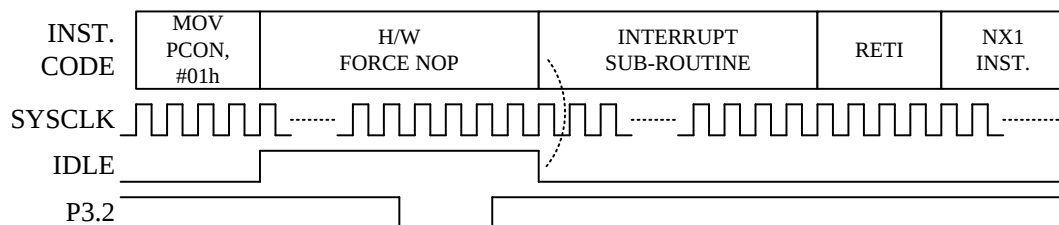
使能 LVD 中断

SFR BFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDS	LVDIE	LVDO	—	—	LVDS			
R/W	R/W	R	—	—	R/W	R/W	R/W	R/W
Reset	0	0	—	—	0	0	0	0

- BFh.7 **LVDIE**: 低压检测中断使能
0: 禁用
1: 启用（注意：EXLVDIE 必须同时为 1 才能生成 LVD 中断）
- BFh.3~0 **LVDS**: 低压检测选择（在空闲/暂停/停止模式下自动禁用）
0000: 将 LVD 设置为 2.5V
0001: 将 LVD 设置为 2.6V
0010: 将 LVD 设置为 2.7V
0011: 将 LVD 设置为 2.8V
0100: 将 LVD 设置为 3.0V
0101: 将 LVD 设置为 3.1V
0110: 将 LVD 设置为 3.2V
0111: 将 LVD 设置为 3.3V
1000: 将 LVD 设置为 3.4V
1001: 将 LVD 设置为 3.6V
1010: 将 LVD 设置为 3.7V
1011: 将 LVD 设置为 3.8V
1100: 将 LVD 设置为 3.9V
1101: 将 LVD 设置为 4.0V
1110: 将 LVD 设置为 4.2V
1111: 将 LVD 设置为 4.3V

6.4 空闲模式唤醒和中断

空闲模式下被启用的中断唤醒,这意味着各个中断使能位(如:EX0)和 EA 位必须都设置为 1 以建立空闲模式唤醒功能。所有被允许的中断(引脚,定时器,ADC,触摸按键,SPI 和 UART),可以将 CPU 从空闲模式唤醒。当空闲被唤醒,立即进入中断服务程序。当中断服务程序返回后,“IDL (PCON.0) 设置后的第一个指令”将被执行。



EA=EX0=1,P3.2 (INT0) 空闲模式唤醒和中断

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	SMOD	—	—	—	GF1	GF0	PD	IDL
R/W	R/W	—	—	—	R/W	R/W	R/W	R/W
Reset	0	—	—	—	0	0	0	0

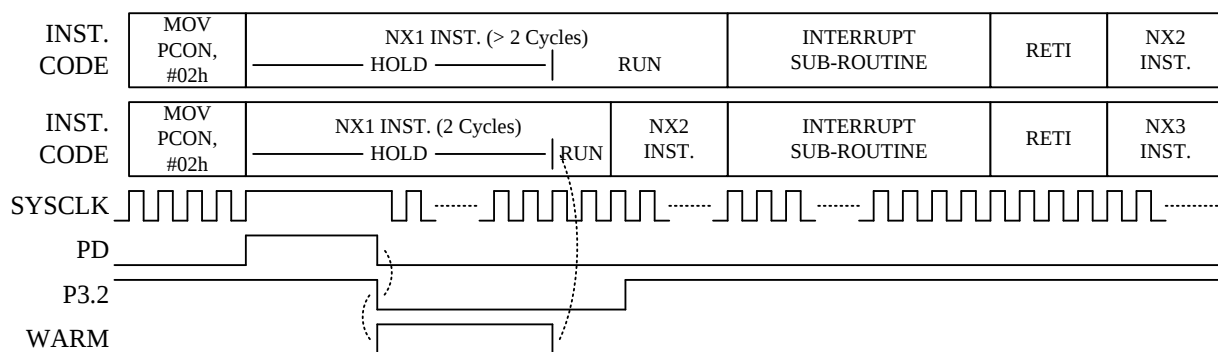
87h.1 PD:停止位,如果 1 进入暂停/停止模式。

87h.0 IDL:空闲位,如果 1 进入空闲模式。

6.5 暂停/停止模式唤醒和中断

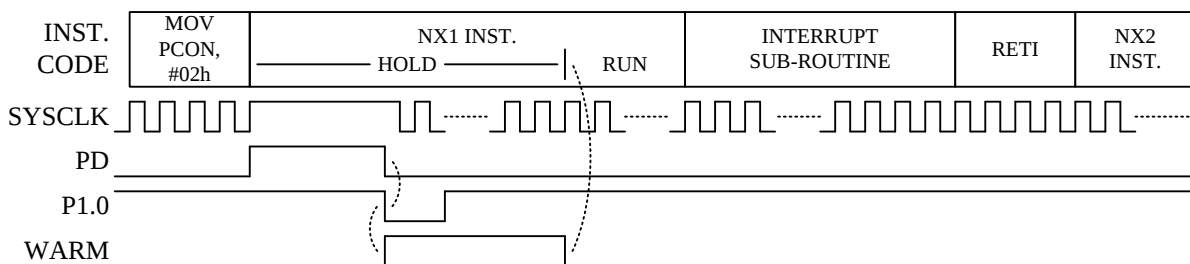
暂停/停止模式唤醒很简单,只要把各个引脚的中断使能位(如:EX0)设置,该引脚唤醒功能启用。设置 EX0~9/EXLVDIE 可以允许 INT0~9 引脚上的暂停/停止模式唤醒功能。设置 P1WKUP 位 7~0 可以启用 P1.7~P1.0 的暂停/停止模式唤醒功能。一旦暂停/停止被唤醒,“PD (PCON.1) 设置后的第一条指令”立即在中断服务之前被执行。中断进入需要 EA=1 (P1WKUP 还需要 P1IE=1) 和该引脚触发状态停留足够长,以被系统时钟采样到。此功能可让 CPU 暂停/停止模式唤醒后,进入或不进入中断子程序。

注: 如果 INTn 引脚是低电平且该唤醒功能启用,则芯片无法进入暂停/停止模式。(INTn=0 and Exn=1, n=0~9)



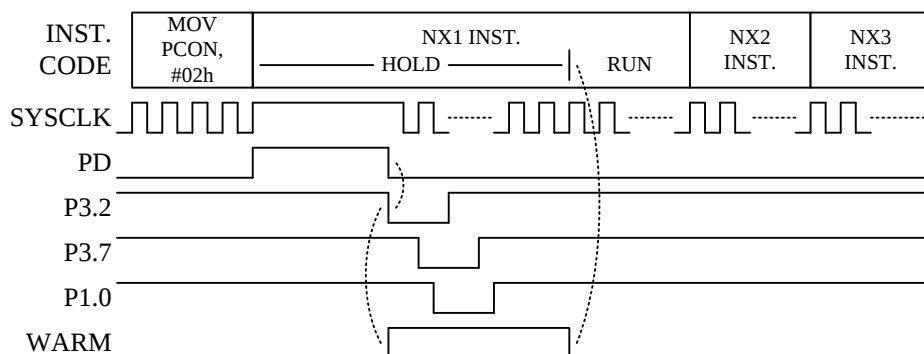
EA=EX0=1

P3.2 (INT0) 预热后被采样, 暂停/模式唤醒和中断



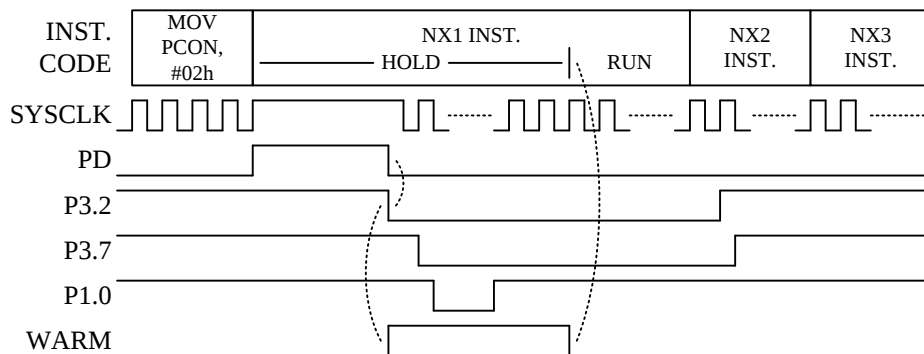
EA=P1IE=P1WKUP=1

P1.0 变化(不需要时钟采样), 暂停/停止模式唤醒和中断



EA=EX0=EXLVDIE=P1WKUP=1, P1IE=0

P3.2/P3.7 脉冲太窄, 暂停/停止模式唤醒, 但没有中断



EX0=EXLVDIE=P1WKUP=P1IE=1, EA=0

暂停/停止模式唤醒, 但没有中断

7. I/O 端口

该芯片总共有 26 多功能 I/O 引脚。所有的 I/O 引脚遵循标准 8051“读-修改-写”功能。读取 SFR 的, 而不是引脚状态的指令, 会读取一个端口或端口位的值, 可能改变它, 然后将它改写到 SFR。(例如: ANL P1,A; INC P2; CPL P3.0)。

7.1 端口 1/端口 2/端口 3

这些引脚可以在四种不同的模式, 如下操作。

模式	端口 1, 端口 2, 端口 3 引脚功能		Px.n SFR 数据	引脚状态	电阻上拉	数位输入
	P3.2~P3.0	其他				
模式 0	伪开漏输出	开漏输出	0	驱动低	N	N
			1	上拉	Y	Y
模式 1	伪开漏输出	开漏输出	0	驱动低	N	N
			1	高阻抗	N	Y
模式 2	CMOS 推挽输出		0	驱动低	N	N
			1	驱动高	N	N
模式 3	替代功能,如 ADC		X (无关)	—	N	N

端口 1,P2.1~P2.0, 端口 3 I/O 引脚菜单

如果一个端口 1, 端口 2 和端口 3 引脚用于施密特触发输入, 软件必须设置 I/O 引脚到模式 0 或模式 1, 并设置相应的端口数据 SFR 为 1 来禁止该引脚的输出驱动电路。

除了 I/O 端口功能外, 每个端口 1, 端口 2 和端口 3 引脚还具有一个或多个替代功能, 例如 LED, ADC 和触摸键。通过将单独的引脚模式控制 SFR 设置为模式 3, 可以激活大多数功能。端口 1/ 端口 3 引脚具有标准的 8051 辅助定义, 例如 INT0 / 1, T0 / 1/2 或 RXD / TXD。 这些引脚功能需要将引脚模式 SFR 设置为模式 0 或模式 1, 并将 P1.n / P3.n SFR 保持为 1。

引脚名称	8051	唤醒	CKO	ADC	TK	LED 正反扫模式	LED 点矩阵模式	其他
P1.7	TXD2	Y			TK10			MISO
P1.6		Y			TK9			PWM2
P1.5		Y		AD9	TK14			
P1.4		Y	CKO	AD8	TK8			
P1.3		Y		AD7	TK7			PWM1
P1.2		Y		AD6	TK6			PWM0
P1.1	T2EX	Y		AD5	TK5			
P1.0	T2	Y	T2O	AD4	TK4			

端口 1 多重菜单

引脚名称	8051	唤醒	CKO	ADC	TK	LED 正反扫模式	LED 点矩阵模式	其他
P3.7	INT2	Y			TK15	LEDS2	LED6	RSTn
P3.6	RXD2	Y			TK11	LEDS5		SCK
P3.5	T1	Y			TK12	LEDS4	LED8	MOSI
P3.4	T0	Y	T0O		TK13	LEDS3	LED7	SS
P3.3	INT1	Y		AD0	TK0			
P3.2	INT0	Y		AD1	TK1			VBGO
P3.1	TXD	Y		AD2	TK2			SDA
P3.0	RXD	Y		AD3	TK3			SCL

端口 3 多重菜单

引脚名称	8051	唤醒	CKO	ADC	TK	LED 正反扫模式	LED 点矩阵模式	其他
P2.1						LEDS1	LED5	XO
P2.0						LEDS0	LED4	XI

端口 2 多重菜单

下面列出了端口 1 / 端口 2 / 端口 3 引脚替代功能所需的 SFR 设置

Alternative Function	Mode	Px.n SFR data	Pin State	Other necessary SFR setting
T0, T1, T2, T2EX, INT0, INT1, INT2	0	1	输入上拉	
	1	1	输入	
RXD, TXD	0	1	输入上拉/伪开漏输出	
	1	1	输入/伪开漏输出	
RXD2,TXD2	0	1	输入上拉/开漏输出	
	1	1	输入/开漏输出	
T0O, T2O, CKO	0	X	时钟开漏输出,上拉	PINMOD
	1	X	时钟开漏输出	
	2	X	时钟输出 (CMOS 推挽)	
VBGO	X	X	带隙电压输出	VBGOUT
LEDS0~LEDS5 LEDC0~LEDC3	X	X	LED 正反扫模式 (BiD) 输出	LEDCON
LED0~LED8	X	X	LED 点矩阵模式 (DMX) 输出	LEDCON3
TK0~TK15	2	1	触摸键 (CMOS 输出高电平)	TKCHS ATKCH0 ATKCH1 ATKCH2
AD0~AD14	3	X	ADC 通道	ADCHS
PWM0~PWM2	0	X	PWM 开漏输出,上拉	PWMOE
	1	X	PWM 开漏输出	
	2	X	PWM 输出 (CMOS 推挽)	
XI, XO	0	1	晶振	CLKCON
I ² C Master SCL	0	X	I ² C 时钟输出 (开漏输出,上拉)	
	1	X	I ² C 时钟输出 (CMOS 推挽)	
I ² C Slave SCL	1	1	I ² C 时钟输入 (高阻抗)	
I ² C Master/Slaver SDA	0	1	I ² C 数据 (上拉)	
SPI Master Mode MISO	1	1	SPI 数据输入	SPCON
SPI Master Mode SCK, MOSI	2	X	SPI 时钟/数据输出 (CMOS 推挽)	
SPI Slave Mode MISO	2	X	SPI 数据输出 (CMOS 推挽)	
SPI Slave Mode SCK, MOSI	1	1	SPI 时钟/数据输入	
SS	1	1	SPI 芯片选择	

对于上表中,“**CMOS 推挽**”引脚意味着它可以吸收和驱动至少 4 mA 的电流。我们不建议使用这种引脚作为输入功能。

一个“**开漏**”引脚意味着它可以吸收至少 4 mA 电流,但只能驱动小电流 (<20μA)。它可以用作输入或输出功能,并且通常需要一个外部上拉电阻。

8051 标准引脚是一个“**伪开漏**”引脚。它可以吸收至少 4 mA 电流于低电平输出,并于输出从低到高时,驱动至少 4 mA 电流 1~2 个时钟周期,然后开为小电流 (<20μA),以维持引脚在高电平。它可以用作输入或输出功能。

注: 对于上述必要的 SFR 设置, LCD/LED 引脚具有最高优先级。因此,如果某个段未使用某个引脚 (例如: 该引脚是 I/O, ADC, TK 和 SPI...), 则软件必须禁用 LCD/LED 功能。

SFR 90h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

90h.7~0 **P1:** 端口 1 数据

SFR A0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	1	1

A0h.1~0 **P2:** 端口 2 数据

SFR B0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P3	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	1	1	1	1	1	1	1

B0h.7~0 **P3:** 端口 3 数据

SFR D8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCON	SCKTYPE	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	1	0	0	0	1	1

D8h.7 **SCKTYPE:** 慢时钟类型。该位只能在快钟模式 (SELFCK=1) 时改变。

0: SRC,P2.1,P2.0 为 I/O 引脚

1: SXT,P2.1,P2.0 为晶振引脚

D8h.6 **FCKTYPE:** 快时钟类型。该位只能在慢钟模式 (SELFCK=0) 时改变。

0: FRC,P2.1,P2.0 为 I/O 引脚

1: FXT,P2.1,P2.0 为晶振引脚

SFR A2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1MODL	P1MOD3		P1MOD2		P1MOD1		P1MOD0	
R/W	R/W		R/W		R/W		R/W	
Reset	0	1	0	1	0	1	0	1

A2h.7~6 **P1MOD3**:P1.3 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P1.3 为 ADC

A2h.5~4 **P1MOD2**:P1.2 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P1.2 为 ADC

A2h.3~2 **P1MOD1**:P1.1 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P1.1 为 ADC

A2h.1~0 **P1MOD0**:P1.0 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P1.0 为 ADC

SFR A3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1MODH	P1MOD7		P1MOD6		P1MOD5		P1MOD4	
R/W	R/W		R/W		R/W		R/W	
Reset	0	1	0	1	0	1	0	1

A3h.7~6 **P1MOD7**:P1.7 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3

A3h.5~4 **P1MOD6**:P1.6 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3

A3h.3~2 **P1MOD5**:P1.5 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P1.5 为 ADC

A3h.1~0 **P1MOD4**:P1.4 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P1.4 为 ADC

SFR A4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P3MODL	P3MOD3		P3MOD2		P3MOD1		P3MOD0	
R/W	R/W		R/W		R/W		R/W	
Reset	0	1	0	1	0	1	0	1

A4h.7~6 **P3MOD3**:P3.3 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P3.3 为 ADC

A4h.5~4 **P3MOD2**:P3.2 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P3.2 为 ADC

A4h.3~2 **P3MOD1**:P3.1 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P3.1 为 ADC

A4h.1~0 **P3MOD0**:P3.0 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3,P3.0 为 ADC

SFR A5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P3MODH	P3MOD7		P3MOD6		P3MOD5		P3MOD4	
R/W	R/W		R/W		R/W		R/W	
Reset	0	0	0	1	0	1	0	1

A5h.7~6 **P3MOD7**:P3.7 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3

A5h.5~4 **P3MOD6**:P3.6 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3

A5h.3~2 **P3MOD5**:P3.5 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3

A5h.1~0 **P3MOD4**:P3.4 引脚控制

00:模式 0
01:模式 1
10:模式 2
11:模式 3

SFR 93h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2MOD	—	—	—	—	P2MOD1		P2MOD0	
R/W	—	—	—	—	R/W		R/W	
Reset	—	—	—	—	0	1	0	1

93h.3~2 **P2MOD1**: P2.1 引脚控制

- 00: 模式 0
- 01: 模式 1
- 10: 模式 2
- 11: 未定义

93h.1~0 **P2MOD0**: P2.0 引脚控制

- 00: 模式 0
- 01: 模式 1
- 10: 模式 2
- 11: 未定义

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PINMOD	—	I2CSEL	TCOE	T2OE	HSNK2EN	HSNK1EN	HSNK0EN	T0OE
R/W	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	0	0	0	0	0	0	0

A6h.5 **TCOE**: 系统时钟信号输出 (CKO) 控制

- 0: 禁止 “系统时钟除以 2” 信号输出到 P1.4
- 1: 允许 “系统时钟除以 2” 信号输出到 P1.4

A6h.4 **T2OE**: Timer2 信号输出 (T2O) 使能

- 0: 禁止 “Timer2 溢出除以 2” 输出到 P1.0
- 1: 允许 “Timer2 溢出除以 2” 输出到 P1.0

A6h.0 **T0OE**: Timer0 信号输出 (T0O) 控制

- 0: 禁止 “Timer0 溢出除以 64” 输出到 P3.4
- 1: 允许 “Timer0 溢出除以 64” 输出到 P3.4

SFR BCh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPCON	SPEN	MSTR	CPOL	CPHA	SSDIS	LSBF	SPCR	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

BCh.7 **SPEN**: SPI 使能

- 0: SPI 禁用
- 1: SPI 使能

BCh.3 **SSDIS**: SS 引脚禁用

- 0: 使能 SS pin
- 1: 禁用 SS pin

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX2	WDTE		PWRSABV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

F7h.4 **VBGOUT**: 带隙电压输出控制

- 0: 禁止
- 1: 带隙电压输出到 P3.2 引脚

SFR 9Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMOE	PWM1IE	PWM0IE	—	—	—	PWM2OE	PWM1OE	PWM0OE
R/W	R/W	R/W	—	—	—	R/W	R/W	R/W
Reset	0	0	—	—	—	0	0	0

- 9Eh.2 **PWM2OE:** PWM2 信号输出使能
0: 禁止 PWM2 信号输出
1: 允许 PWM2 信号输出到 P1.6
- 9Eh.1 **PWM1OE:** PWM1 信号输出使能
0: 禁止 PWM1 信号输出
1: 允许 PWM1 信号输出到 P1.3
- 9Eh.0 **PWM0OE:** PWM0 信号输出使能
0: 禁止 PWM0 信号输出
1: 允许 PWM0 信号输出到 P1.2

SFR B1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LEDCON	LEDEN		LEDPSC		LEDHOLD	LEDBRIT		
R/W	R/W		R/W		R/W	R/W		
Reset	0	0	0	0	0	1	0	0

- B1h.7~6 **LEDEN:** LED 正反扫模式 (BiD)
00: LED 正反扫模式 (BiD) 禁用
01: LED 1/8 占空比 (COM0~3, SEG0~3), LED 引脚的状态将被自动控制
10: LED 1/9 占空比 (COM0~3, SEG0~4), LED 引脚的状态将被自动控制
11: LED 1/10 占空比 (COM0~3, SEG0~5), LED 引脚的状态将被自动控制

SFR B3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LEDCON3	LEDMTEN	LED8EN	LED7EN	LED6EN	LED5EN	LED4EN	LED3EN	LED2EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- B3h.7 **LEDMTEN:** LED 点矩阵模式 (DMX)
0: LED 点矩阵模式 (DMX) 禁用
1: LED 点矩阵模式 (DMX) 使能 及 LED0, LED1 使能
- B3h.6 **LED8EN:** LED 点矩阵模式 (DMX) 引脚使能控制
0: LED8 禁用
1: LED8 使能
- B3h.5 **LED7EN:** LED 点矩阵模式 (DMX) 引脚使能控制
0: LED7 禁用
1: LED7 使能
- B3h.4 **LED6EN:** LED 点矩阵模式 (DMX) 引脚使能控制
0: LED6 禁用
1: LED6 使能
- B3h.3 **LED5EN:** LED 点矩阵模式 (DMX) 引脚使能控制
0: LED5 禁用
1: LED5 使能
- B3h.2 **LED4EN:** LED 点矩阵模式 (DMX) 引脚使能控制
0: LED4 禁用
1: LED4 使能
- B3h.1 **LED3EN:** LED 点矩阵模式 (DMX) 引脚使能控制
0: LED3 禁用
1: LED3 使能
- B3h.0 **LED2EN:** LED 点矩阵模式 (DMX) 引脚使能控制
0: LED2 禁用
1: LED2 使能

7.2 端口 0

这些引脚与 TK, ADC 和 LCD / LED 共享。如果将 Port0 定义为 I/O 引脚, 则可以将其用作 CMOS 推挽输出或施密特触发器输入。当 SFR 位 P0OE.n = 0 和 P0.n = 1 时, 使能该引脚的上拉功能。

端口 0 引脚功能	P0OE.n	P0.n SFR 数据	引脚状态	电阻上拉	数位输入
输入	0	0	高阻抗	N	Y
	0	1	上拉	Y	Y
CMOS 推挽输出	1	0	驱动低	N	N
	1	1	驱动高	N	N

端口 0 I/O 引脚菜单

引脚名称	唤醒	ADC	TK	LCD	LED 正反扫模式	LED 点矩阵模式
P0.7		AD12	TK19	LCDC7		
P0.6		AD14	TK18	LCDC6		
P0.5		AD13	TK17	LCDC5		
P0.4			TK16	LCDC4		
P0.3			CLD	LCDC3	LEDC3	LED3
P0.2				LCDC2	LEDC2	LED2
P0.1				LCDC1	LEDC1	LED1
P0.0				LCDC0	LEDC0	LED0

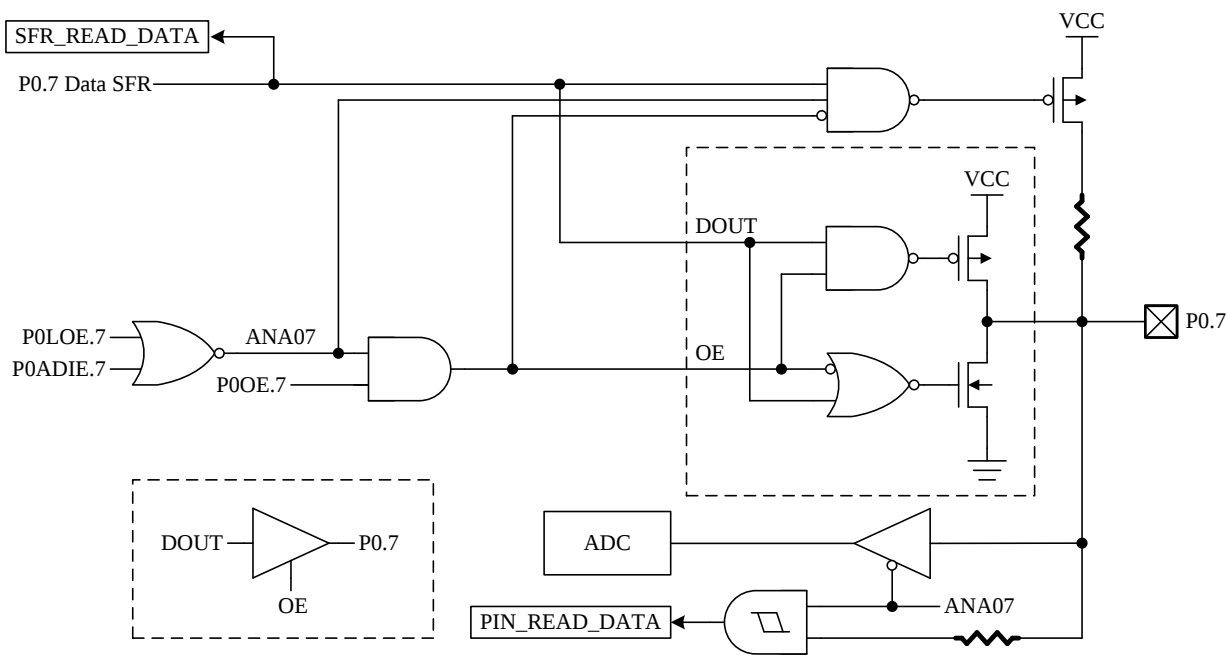
端口 0 多重菜单

端口 0 引脚的替代功能所需的 SFR 设置如下。

替代功能	PxOE.n	Px.n SFR 数据	引脚状态	其他必要的 SFR 设定
LEDC0~ LEDC3	X	X	LED 正反扫模式 (BiD) 输出	LEDCON
LED0~ LED3	X	X	LED 点矩阵模式 (DMX) 输出	LEDCON3
LCDC0~ LCDC7	X	X	LCD 1/2 偏压输出	P0LOE
AD12~AD14	X	X	ADC 输入	P0ADIE
CLD	1	0	触控按键电荷收集电容器连接引脚	TKXCAP
TK16~TK19	1	1	触摸键输入 (CMOS 推挽输出高)	TKCHS

端口 0 替代功能模式设置

注: P0LOE 及 P0ADIE 的优先权高于 P0OE。



P0.7 引脚结构

SFR 80h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

80h.7~0 **P0:** 端口 0 数据还控制 P0.n 引脚的上拉功能。如果 P0.n SFR 数据为“1”且对应的 P0OE.n = 0（输入模式），则启用上拉。

SFR 91h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0OE	P0OE							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

91h.7~0 **P0OE:** 端口 0 CMOS 推挽输出使能控制

0: 禁用

1: 使能

SFR 92h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0LOE	P0LOE							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

92h.7~0 **P0LOE:** 端口 0 LCD 1/2 偏置输出使能控制

0: 禁用

1: 使能

SFR AFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0ADIE	P0ADIE			—	—	—	—	—
R/W	R/W			—	—	—	—	—
Reset	0	0	0	—	—	—	—	—

AFh.7~5 **P0ADIE**: ADC 通道输入使能
 000: P0.7~P0.5 为数字输入
 1xx: P0.7 是 ADC 输入
 x1x: P0.6 是 ADC 输入
 xx1: P0.5 是 ADC 输入

SFR B1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LEDCON	LEDEN		LEDPSC		LEDHOLD	LEDBRIT		
R/W	R/W		R/W		R/W	R/W		
Reset	0	0	0	0	0	1	0	0

B1h.7~6 **LEDEN**: LED 正反扫模式 (BiD)
 00: LED 正反扫模式 (BiD) 禁用
 01: LED 1/8 占空比 (COM0~3, SEG0~3), LED 引脚的状态将被自动控制
 10: LED 1/9 占空比 (COM0~3, SEG0~4), LED 引脚的状态将被自动控制
 11: LED 1/10 占空比 (COM0~3, SEG0~5), LED 引脚的状态将被自动控制

SFR B3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LEDCON3	LEDMTEN	LED8EN	LED7EN	LED6EN	LED5EN	LED4EN	LED3EN	LED2EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

B3h.7 **LEDMTEN**: LED 点矩阵模式 (DMX)
 0: LED 点矩阵模式 (DMX) 禁用
 1: LED 点矩阵模式 (DMX) 使能 及 LED0, LED1 使能

B3h.6 **LED8EN**: LED 点矩阵模式 (DMX) 引脚使能控制
 0: LED8 禁用
 1: LED8 使能

B3h.5 **LED7EN**: LED 点矩阵模式 (DMX) 引脚使能控制
 0: LED7 禁用
 1: LED7 使能

B3h.4 **LED6EN**: LED 点矩阵模式 (DMX) 引脚使能控制
 0: LED6 禁用
 1: LED6 使能

B3h.3 **LED5EN**: LED 点矩阵模式 (DMX) 引脚使能控制
 0: LED5 禁用
 1: LED5 使能

B3h.2 **LED4EN**: LED 点矩阵模式 (DMX) 引脚使能控制
 0: LED4 禁用
 1: LED4 使能

B3h.1 **LED3EN**: LED 点矩阵模式 (DMX) 引脚使能控制
 0: LED3 禁用
 1: LED3 使能

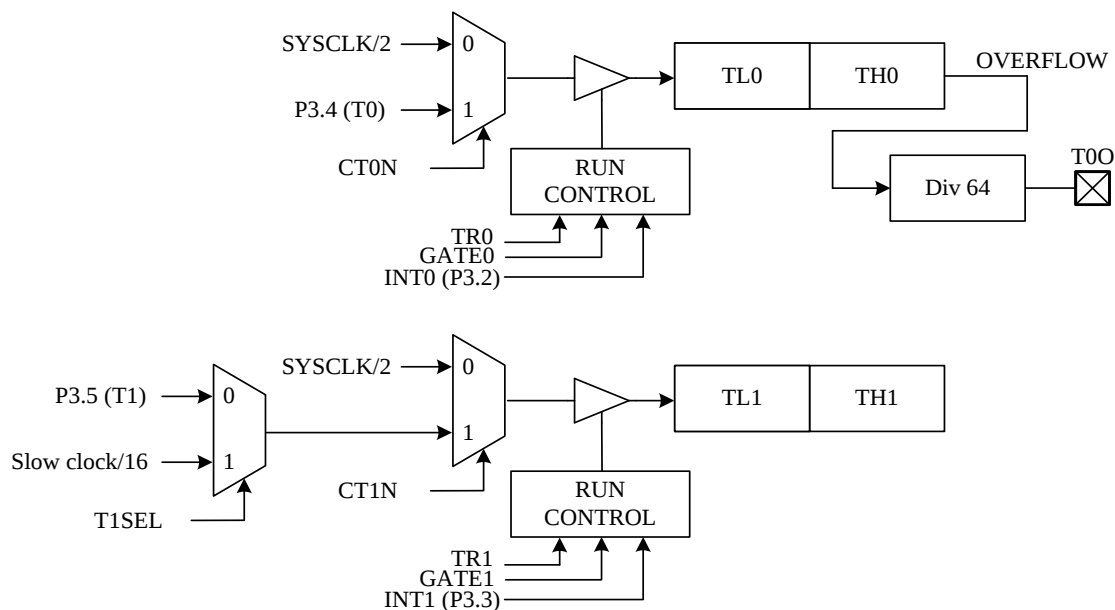
B3h.0 **LED2EN**: LED 点矩阵模式 (DMX) 引脚使能控制
 0: LED2 禁用
 1: LED2 使能

8. 定时器

Timer0,Timer1 和 Timer2 设置为标准的 8051 兼容的定时器/计数器。相较于传统的 12T 8051,该芯片的 Timer0/1/2 使用 2 个系统时钟周期的时间基本单元。也就是说,在定时器模式下,这些定时器以每一个“2 个系统时钟”率增加;在计数器模式下,T0/T1/T2 引脚输入脉冲必须大于 2 个系统时钟以便该设备可以辨识。除了标准 8051 定时器功能,T0O 引脚输出“Timer0 溢出除以 64”的信号,而 T2O 引脚输出“Timer2 溢出除以 2”的信号。当时基是 SXT,Timer3 被设置为一个实时时钟计数。

8.1 Timer0/1

TCON 和 TMOD 用于设置操作模式,并控制 Timer0/1 的运行和中断产生,定时器/计数器的值存储在两个成对的 8 位寄存器(TL0,TH0,和 TL1,TH1)。



Timer0 and Timer1 结构

SFR 88h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- 88h.7 **TF1**:Timer1 溢出标志
当定时器/计数器 1 溢出时由硬件设置。
当 CPU 转向进入中断服务程序时由硬件清零。
- 88h.6 **TR1**:Timer1 运行控制
0:Timer1 停止
1:Timer1 运行
- 88h.5 **TF0**:Timer0 溢出标志
当定时器/计数器 0 溢出时由硬件设置。
当 CPU 转向进入中断服务程序时由硬件清零。
- 88h.4 **TR0**:Timer0 运行控制
0:Timer0 停止
1:Timer0 运行

SFR 89h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMOD	GATE1	CT1N	TMOD1		GATE0	CT0N	TMOD0	
R/W	R/W	R/W	R/W		R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

- 89h.7 **GATE1**:Timer1 门控位
0:当 TR1 位设置时 Timer1 使能
1:只有当 INT1 引脚为高,TR1 位设置时 Timer1 使能
- 89h.6 **CT1N**:Timer1 计数器/定时器选择位
0:定时器模式,Timer1 的数据以 2 个系统时钟周期率增加
1:计数器模式,Timer1 的数据在 T1 引脚的下降沿时增加
- 89h.5~4 **TMOD1**:Timer1 模式选择
00:8 位定时器/计数器 (TH1) 和 5 位预分频器 (TL1)
01:16 位定时器/计数器
10:8 位自动重载定时器/计数器 (TL1),溢出时从 TH1 重新装载。
11:Timer1 停止
- 89h.3 **GATE0**:Timer0 门控位
0:当 TR0 位设置时 Timer0 使能
1:只有当 INT0 引脚为高,TR0 位设置时 Timer0 使能
- 89h.2 **CT0N**:Timer0 计数器/定时器选择位
0:定时器模式,Timer0 的数据以 2 个系统时钟周期率增加
1:计数器模式,Timer0 的数据在 T0 引脚的下降沿时增加
- 89h.1~0 **TMOD0**:Timer0 模式选择
00:8 位定时器/计数器 (TH0) 和 5 位预分频器 (TL0)
01:16 位定时器/计数器
10:8 位自动重载定时器/计数器 (TL0),溢出时从 TH0 重新装载。
11:TL0 是一个 8 位定时器/计数器。TH0 是一个 8 位定时器/计数器,使用 Timer1 的 TR1 和 TF1 位。

SFR 8Ah	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TL0	TL0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Ah.7~0 **TL0**:Timer0 数据的低字节

SFR 8Bh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TL1	TL1							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Bh.7~0 **TL1**:Timer1 数据的低字节

SFR 8Ch	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH0	TH0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Ch.7~0 **TH0**:Timer0 数据的高字节

SFR 8Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH1	TH1							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

8Dh.7~0 **TH1**:Timer1 数据的高字节

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	—	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	—	R/W	R/W	R/W	R/W	R/W
Reset	0	0	—	0	0	0	0	0

F8h.1 **T1SEL**: Timer1 计数器模式 (CT1N = 1) 输入选择

0: P3.5 (T1) 脚 (8051 标准)

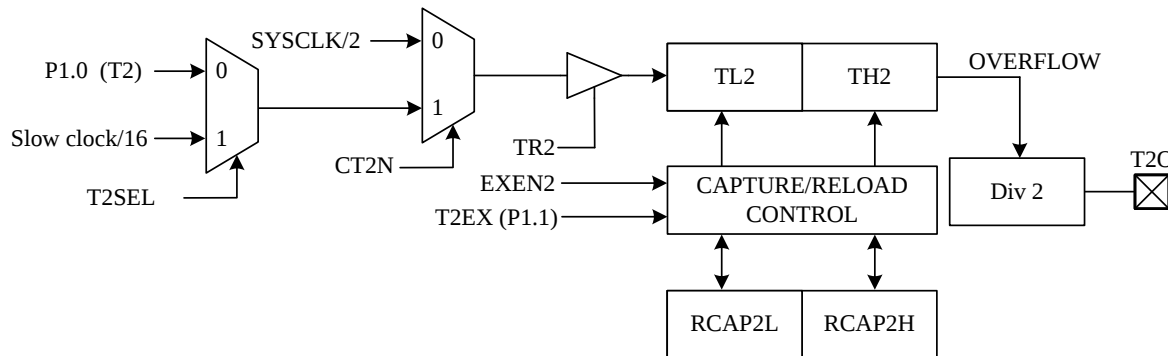
1: 慢时钟除以 16 (SLOWCLK/16)

注: 另请参阅第 6 章的有关 Timer0/1 中断使能和优先级的更多信息。

注: 同时参阅第 7 章关于 T0O 引脚输出设置的详细信息。

8.2 Timer2

Timer2 通过 TCON2 寄存器存储在 TL2 和 TH2 的定时器/计数器 2 低和高字节和存储在 RCAP2L 和 RCAP2H 的 Timer2 重载/捕获寄存器的高和低字节来控制。



Timer2 结构

SFR C8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2N	CPRL2N
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

C8h.7 **TF2**:Timer2 溢出标志

当定时器/计数器 2 溢出时由硬件设置,除非 RCLK=1 或 TCLK=1。此位必须由软件清零。

C8h.6 **EXF2**:T2EX 中断引脚下降沿标志

如果 EXEN2=1,当捕获或重载是由 T2EX 引脚的下降沿跳变引起时被设置。该位必须由软件清零。

C8h.5 **RCLK**:UART 接收时钟控制位

0:模式 1 或 3 时使用 Timer1 溢出作为串行端口接收时钟

1:模式 1 或 3 时使用 Timer2 溢出作为串行端口接收时钟

C8h.4 **TCLK**:UART 发送时钟控制位

0:模式 1 或 3 时使用 Timer1 溢出作为串行端口发送时钟

1:模式 1 或 3 时使用 Timer2 溢出作为串行端口发送时钟

C8h.3 **EXEN2**:T2EX 引脚使能

0:T2EX 引脚禁用

1:T2EX 引脚使能,如果 RCLK=TCLK=0,当检测出 T2EX 引脚的下降沿跳变,这引起捕获或重载

C8h.2 **TR2**:Timer2 运行控制

0:Timer2 停止

1:Timer2 运行

C8h.1 **CT2N**:Timer2 计数器/定时器选择位

0:定时器模式,Timer2 的数据以 2 个系统时钟周期率增加

1:计数器模式,Timer2 的数据在 T2 引脚的下降沿时增加

C8h.0 **CPRL2N**:Timer2 捕捉/重载控制位

0:重载模式,如果 EXEN2=1 当 Timer2 溢出或 T2EX 引脚上的下降沿跳变则自动重载

1:捕捉模式,如果 EXEN2=1 在 T2EX 引脚上的下降沿跳变则捕捉

如果 RCLK=1 或 TCLK=1 时,CPRL2N 被忽略,Timer2 溢出时定时器被强制自动重载

SFR CAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCP2L	RCP2L							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CAh.7~0 **RCP2L**:Timer2 重载/捕获数据的低字节

SFR CBh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
RCP2H	RCP2H							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CBh.7~0 **RCP2H**:Timer2 重载/捕获数据的高字节

SFR CCh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TL2	TL2							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CCh.7~0 **TL2**:Timer2 数据的低字节

SFR CDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TH2	TH2							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

CDh.7~0 **TH2**:Timer2 数据的高字节

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.2 **T2SEL**: Timer2 计数器模式 (CT2N = 1) 输入选择

- 0: P1.0 (T2) 引脚 (8051 标准)
- 1: 慢时钟除以 16 (SLOWCLK / 16)

F8h.1 **T1SEL**: Timer1 计数器模式 (CT1N = 1) 输入选择

- 0: P3.5 (T1) 脚 (8051 准)
- 1: 慢时钟除以 16 (SLOWCLK / 16)

注: 另请参阅第6章的有关Timer2 中断使能和优先级的更多信息。

注: 同时参阅第7章关于T2O 引脚输出设置的详细信息。

8.3 Timer3

该芯片的 Timer3 作为时基计数器,周期性地产生中断。它会产生一个中断标志位 (TF3) 当时钟除以 32768,16384,8192 或 128 取决于 TM3PSC 位。Timer3 的时钟源为慢时钟 (SRC 或 SXT)。当时钟源为 SXT 即是理想的实时时钟 (RTC) 功能。

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	UART1W	–	WDTPSC		ADCKS		TM3PSC	
R/W	R/W	–	R/W		R/W		R/W	
Reset	0	–	0	0	0	0	0	0

94h.1~0 **TM3PSC**:Timer3 中断率控制选择

00:中断率是 32768 慢时钟周期

01:中断率是 16384 慢时钟周期

10:中断率是 8192 慢时钟周期

11:中断率是 128 慢时钟周期

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTFLG	LVDIF	–	TKIF	ADIF	–	–	P1IF	TF3
R/W	R	–	R/W	R/W	–	–	R/W	R/W
Reset	–	–	0	0	–	–	0	0

95h.0 **TF3**:Timer3 中断标志

当 Timer3 到达 TM3PSC 设置周期时由硬件设置。

当程序执行中断服务程序时被自动清除。

软件也可以写 FEh 到 INTFLG 清除该标志。(注 2)

注 2: 软件可以写 0 清除 INTFLG 中的标志,但写 1 没有任何效果。

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.6 **CLRTM3**:设置以清除 Timer3, 硬件会在下一个时钟周期自动清除此设置

注: 另请参阅第 6 章的有关 Timer3 中断使能和优先级的更多信息。

8.4 T0O 和 T2O 输出控制

该装置可以产生各种频率的波形引脚输出 (CMOS 推挽格式) 供蜂鸣器使用。T0O 波形由 Timer0 溢出除以 64 产生,T2O 波形由 Timer2 溢出除以 2 产生。用户可以设置定时器自动重装速度以控制自己的频率。设置 T0OE 和 T2OE SFR 可输出这些波形。

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PINMOD	–	I2CSEL	TCOE	T2OE	HSNK2EN	HSNK1EN	HSNK0EN	T0OE
R/W	–	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	0	0	0	0	0	0	0

A6h.4 **T2OE**:Timer2 信号输出 (T2O) 使能

0:禁止 Timer2 溢出除以 2 输出到 P1.0

1:允许 Timer2 溢出除以 2 输出到 P1.0

A6h.0 **T0OE**:Timer0 信号输出 (T0O) 控制

0:禁止“Timer0 溢出除以 64”输出到 P3.4

1:允许“Timer0 溢出除以 64”输出到 P3.4

9. UART

该芯片具有二个 UART，分别为 UART1 和 UART2。

UART1 是标准的 8051 全双工 UART，UART1 使用 SCON 和 SBUF 的 SFR。SCON 是控制寄存器，SBUF 是数据寄存器。数据被写入到 SBUF 用于传输，而 SBUF 被读取时，可获得接收数据。接收到的数据和发送数据寄存器是完全独立的。

UART2 使用 SCON2 和 SBUF2 的 SFR。SCON2 是控制寄存器，SBUF2 是数据寄存器。数据被写入到 SBUF 用于传输，而 SBUF 被读取时，可获得接收数据。接收到的数据和发送数据寄存器是完全独立的。UART2 支持 UART 的大部分功能，但不支持模式 0 和模式 2，也不支持 Timer2 和单线 UART 模式。另一方面，对于 UART2，不使用 SMOD 选项。始终启用 UART2 双波特率。

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCON	SMOD	—	—	—	GF1	GF0	PD	IDL
R/W	R/W	—	—	—	R/W	R/W	R/W	R/W
Reset	0	—	—	—	0	0	0	0

87h.7 **SMOD**:UART1 双波特率控制位
0:禁止 UART1 双波特率控制位
1:允许 UART1 双波特率控制位

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	UART1W	TKFJMP	WDTPSC		ADCKS		TM3PSC	
R/W	R/W	R/W	R/W		R/W		R/W	
Reset	0	0	0	0	0	0	0	0

94h.7 **UART1W**:一线 UART1 模式使能,TXD/RXD 都使用 P3.1 脚
0:禁止一线 UART1 模式
1:允许一线 UART1 模式

SFR 98h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

98h.7~6 **SM0,SM1**:串行端口模式选择位 0,1
00:模式 0:8 位移位寄存器,波特率= $F_{SYSCLK}/2$
01:模式 1:8 位 UART1,波特率可变
10:模式 2:9 位 UART1,波特率= $F_{SYSCLK}/32$ 或/64
11:模式 3:9 位 UART1,波特率可变

98h.5 **SM2**:串行端口模式选择位 2
SM2 通过一线串行方式实现多机通信并修改上述如下。在模式 2 和 3,当 SM2 设置,如果接收到的第九位数据为 0,那么接收中断不会产生。在模式 1 中,除非有效的停止位被接收,接收中断不会产生。在模式 0 中,SM2 应为 0。

98h.4 **REN**:UART1 接收使能
0:禁止接收
1:允许接收

98h.3 **TB8**:发送位 8,在模式 2 和 3 为发送第九位

98h.2 **RB8**:接收位 8,包含模式 2 和 3 的接收第九位,如果 SM2 = 0,为模式 1 停止位

98h.1 **TI**:发送中断标志

由硬件设置在模式 0 第 8 位的结束时,或在其他模式中停止位的开始时。必须通过软件清零。

98h.0 **RI**:接收中断标志

由硬件设置在模式 0 第 8 位的结束时,或在其他模式下停止位的取样点。必须通过软件清零。

SFR 99h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SBUF	SBUF							
R/W	R/W							
Reset	—	—	—	—	—	—	—	—

99h.7~0 **SBUF:UART1** 发送和接收数据。发送写入该位置数据和接收从该位置读取数据,但路径是独立的。

SFR 8Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SCON2	SM	—	—	REN2	TB82	RB82	TI2	RI2
R/W	R/W	—	—	R/W	R/W	R/W	R/W	R/W
Reset	0	—	—	0	0	0	0	0

8Eh.7 **SM:** UART2 串行端口模式选择位

0: 模式 1: 8 位 UART2, 波特率可变

1: 模式 3: 9 位 UART2, 波特率可变

(UART2 不支持模式 0 /模式 2)

8Eh.4 **REN2:** UART2 接收使能

0: 禁止接收

1: 允许接收

8Eh.3 **TB82:** 传输位 8, 即在模式 3 中要传输的第 9 位

8Eh.2 **RB82:** 接收位 8, 包含在模式 3 中接收到的第 9 位

8Eh.1 **TI2:** 发送中断标志

在模式 1 和 3 中, 在停止位的开头通过硬件进行设置。必须通过软件进行清除。

8Eh.0 **RI2:** 接收中断标志

在模式 1 和 3 中, 在停止位的采样点由硬件设置。必须由软件清除。

SFR 8Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SBUF2	SBUF2							
R/W	R/W							
Reset	—	—	—	—	—	—	—	—

8Fh.7~0 **SBUF2:** UART2 发送和接收数据。发送写入该位置数据和接收从该位置读取数据,但路径是独立的。

F_{SYSCLK} 表示系统时钟频率。

- 模式 0: (UART2 不支持)
波特率= $F_{\text{SYSCLK}}/2$
- 模式 1,3:如果使用 Timer1 自动重载模式
波特率= $(\text{SMOD}+1) \times F_{\text{SYSCLK}}/(32 \times 2 \times (256 - \text{TH1}))$
- 模式 1,3:如果使用 Time2 (UART2 不支持)
波特率=Timer2 overflow rate/16= $F_{\text{SYSCLK}}/(32 \times (65536 - \text{RCP2H}, \text{RCP2L}))$
- 模式 2: (UART2 不支持)
波特率= $(\text{SMOD}+1) \times F_{\text{SYSCLK}}/64$

注: 另请参阅第 6 章的有关 UART 中断使能和优先级的更多信息。

注: 同时参阅第 8 章有关 Timer2 如何控制 UART 时钟的详细信息。

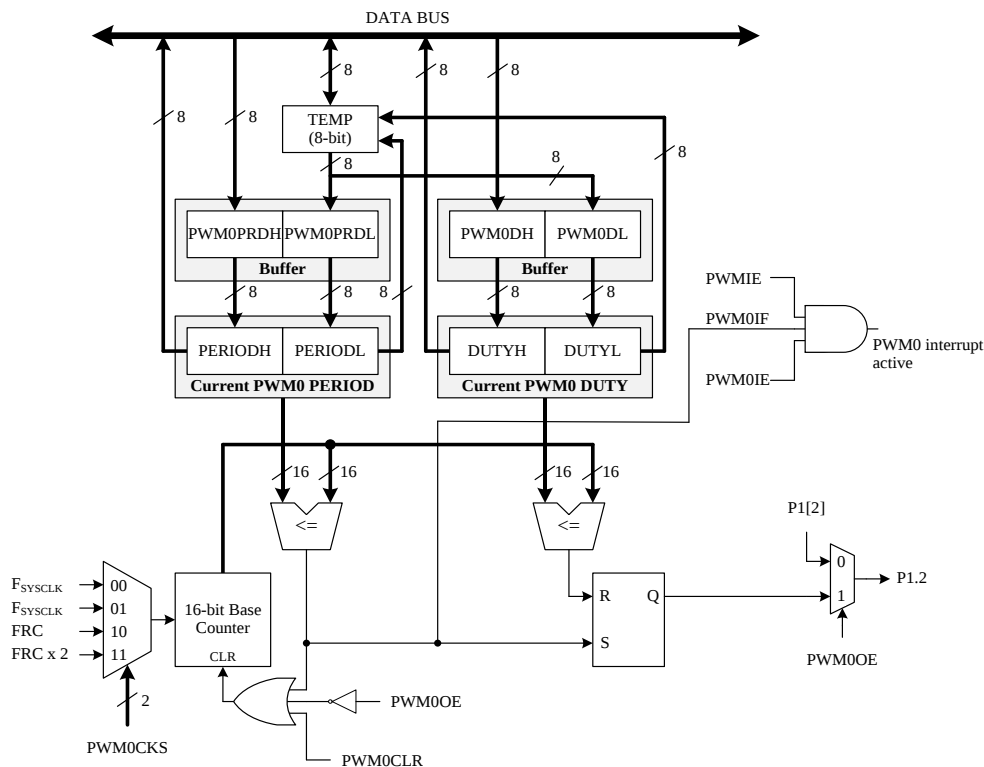
10. PWMs

该芯片具有三个独立的 16 位 PWM 模块 PWM0, PWM1 和 PWM2。PWM0~2 具有相同的操作结构。下面以 PWM0 为例进行说明。PWM 可以基于 PWM 时钟生成具有 65536 占空比分辨率的变化频率波形。PWM 时钟可以选择 FRC 双倍频率 (FRC x 2), FRC 或 F_{SYSC}CLK 作为其时钟源。引脚模式 SFR 控制 PWM 输出波形格式。模式 1 使 PWM 开漏输出, 而模式 2 使 PWM CMOS 推挽输出。(请参阅第 7 节)

16 位 PWM0PRD, PWM0D 寄存器均具有低字节和高字节结构。高字节可以直接访问, 但低字节只能通过内部 8 位缓冲器访问, 必须以特定方式对这些寄存器对进行读写。需要注意的重要一点是, 只有在执行对其相应的高字节的写或读操作时, 才与 8 位缓冲区及其相关的低字节进行数据传输。**简而言之, 先写低字节, 再写高字节。首先读取高字节, 然后读取低字节。**

PWM0OE 位用于选择 PWM0 的输出。如果 PWM0OE 被清除, 则 PWM0 将被清除并停止, 否则 PWM0 正在运行。PWM0CLR 位具有相同的功能。当 PWM0CLR 位置 1 时, 将清除并保持 PWM0, 否则 PWM0 正在运行。PWM0 的结构如下所示。可以通过写 PWM0DH 和 PWM0DL 来更改 PWM0 占空比。每当 16 位基数计数器与 16 位 PWM0 占空比寄存器 {PWM0DH, PWM0DL} 匹配时, PWM0 输出信号就会复位为低电平。可以通过将周期值写入 PWM0PRDH 和 PWM0PRDL 寄存器来设置 PWM0 周期。写入 PWM0D 或 PWM0PRD 寄存器后, 新值将立即保存到其自己的缓冲区中。硬件将在当前周期结束时或清除 PWM0 时更新这些值。PWM0~2 有一个相应的中断标志, 在周期结束时产生一个中断标志。

PWMxDH, PWMxDL, PWMxPRDH 或 PWMxPRDL 为 16 位操作, 程序在高字节和低字节写入和读取时, 应该避免发生中断。假如在读写这些 16 位的 SFR 其间发生中断。而中断内又对这些 SFR 做读写。则容易造成读写的错误。16 位 PWM period 与 duty 的读写, 建议只在主程式内更新数据, 或只在中断内更新数据, 以避免可能的错误产生。



PWM0 结构

SFR 86h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTPWM	—	—	—	—	—	PWM2IF	PWM1IF	PWM0IF
R/W	—	—	—	—	—	R/W	R/W	R/W
Reset	—	—	—	—	—	0	0	0

86h.2 **PWM2IF:**
0: 软件写入 0 清除
1: 周期结束由硬件设置

86h.1 **PWM1IF:**
0: 软件写入 0 清除
1: 周期结束由硬件设置

86h.0 **PWM0IF:**
0: 软件写入 0 清除
1: 周期结束由硬件设置

SFR 9Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMOE	PWM1IE	PWM0IE	—	—	—	PWM2OE	PWM1OE	PWM0OE
R/W	R/W	R/W	—	—	—	R/W	R/W	R/W
Reset	0	0	—	—	—	0	0	0

9Eh.7 **PWM1IE:** PWM1 中断使能
0: 禁用
1: 使能（注意：PWMIE 必须同时为 1 才能产生 PWM 中断）

9Eh.6 **PWM0IE:** PWM0 中断使能
0: 禁用
1: 使能（注意：PWMIE 必须同时为 1 才能产生 PWM 中断）

9Eh.2 **PWM2OE:**
0: 禁用 1: PWM2 使能并将信号输出到 P1.6 引脚

9Eh.1 **PWM1OE:**
0: 禁用 1: PWM1 使能并将信号输出到 P1.3 引脚

9Eh.0 **PWM0OE:**
0: 禁用 1: PWM0 使能并将信号输出到 P1.2 引脚

SFR 9Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCLR	PWM2IE	—	—	—	—	PWM2CLR	PWM1CLR	PWM0CLR
R/W	R/W	—	—	—	—	R/W	R/W	R/W
Reset	0	—	—	—	—	0	0	0

9Fh.7 **PWM2IE:** PWM2 中断使能
0: 禁用
1: 允许（注意：PWMIE 必须同时为 1 才能产生 PWM 中断）

9Fh.2 **PWM2CLR:**
0: PWM2 正在运行 1: PWM2 被清除并保持

9Fh.1 **PWM1CLR:**
0: PWM1 正在运行 1: PWM1 被清除并保持

9Fh.0 **PWM0CLR:**
0: PWM0 正在运行 1: PWM0 被清除并保持

SFR A1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCON	—	—	PWM2CKS		PWM1CKS		PWM0CKS	
R/W	—	—	R/W		R/W		R/W	
Reset	—	—	1	0	1	0	1	0

A1h.5~4 **PWM2CKS**:PWM2 时钟源

00:F_{SYSCLK}
01:F_{SYSCLK}
10:FRC
11:FRC x 2

A1h.3~2 **PWM1CKS**:PWM1 时钟源

00:F_{SYSCLK}
01:F_{SYSCLK}
10:FRC
11:FRC x 2

A1h.1~0 **PWM0CKS**:PWM0 时钟源

00:F_{SYSCLK}
01:F_{SYSCLK}
10:FRC
11:FRC x 2

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE1	PWMIE	I2CE	ES2	SPIE	ADTKIE	EXLVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

A9h.6 **PWMIE**: PWM0~2 中断使能

0: 禁止 PWM0~2 中断
1: 使能 PWM0~2 中断

SFR D1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0DH	PWM0DH							
R/W	R/W							
Reset	1	0	0	0	0	0	0	0

D1h.7~0 **PWM0DH**: PWM0 占空比高字节
 写顺序: 先写 PWM0DL, 然后再写 PWM0DH
 读顺序: 先读 PWM0DH, 然后再读 PWM0DL

SFR D2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0DL	PWM0DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D2h.7~0 **PWM0DL**: 占空比低字节
 写顺序: 先写 PWM0DL, 然后再写 PWM0DH
 读顺序: 先读 PWM0DH, 然后再读 PWM0DL

SFR D3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1DH	PWM1DH							
R/W	R/W							
Reset	1	0	0	0	0	0	0	0

D3h.7~0 **PWM1DH**: PWM1 占空比高字节
 写顺序: 先写 PWM1DL, 然后再写 PWM1DH
 读顺序: 先读 PWM1DH, 然后再读 PWM1DL

SFR D4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1DL	PWM1DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D4h.7~0 **PWM1DL**: PWM1 占空比低字节
 写顺序: 先写 PWM1DL, 然后再写 PWM1DH
 读顺序: 先读 PWM1DH, 然后再读 PWM1DL

SFR D5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2DH	PWM2DH							
R/W	R/W							
Reset	1	0	0	0	0	0	0	0

D5h.7~0 **PWM2DH**: PWM2 占空比高字节
 写顺序: 先写 PWM2DL, 然后再写 PWM2DH
 读顺序: 先读 PWM2DH, 然后再读 PWM2DL

SFR D6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2DL	PWM2DL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D6h.7~0 **PWM2DL**: PWM2 占空比低字节
 写顺序: 先写 PWM2DL, 然后再写 PWM2DH
 读顺序: 先读 PWM2DH, 然后再读 PWM2DL

SFR D9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0PRDH	PWM0PRDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

D9h.7~0 **PWM0PRDH**: PWM0 周期高字节
 写顺序: 先写 PWM0PRDL, 然后再写 PWM0PRDH
 读顺序: 先读 PWM0PRDH, 然后再读 PWM0PRDL

SFR DAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0PRDL	PWM0PRDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

DAh.7~0 **PWM0PRDL**: PWM0 周期低字节
 写顺序: 先写 PWM0PRDL, 然后再写 PWM0PRDH
 读顺序: 先读 PWM0PRDH, 然后再读 PWM0PRDL

SFR DBh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PRDH	PWM1PRDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

DBh.7~0 **PWM1PRDH**: PWM1 周期高字节
 写顺序: 先写 PWM1PRDL, 然后再写 PWM1PRDH
 读顺序: 先读 PWM1PRDH, 然后再读 PWM1PRDL

SFR DCh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1PRDL	PWM1PRDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

DCh.7~0 **PWM1PRDL**: PWM1 周期低字节
 写顺序: 先写 PWM1PRDL, 然后再写 PWM1PRDH
 读顺序: 先读 PWM1PRDH, 然后再读 PWM1PRDL

SFR DDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2PRDH	PWM2PRDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

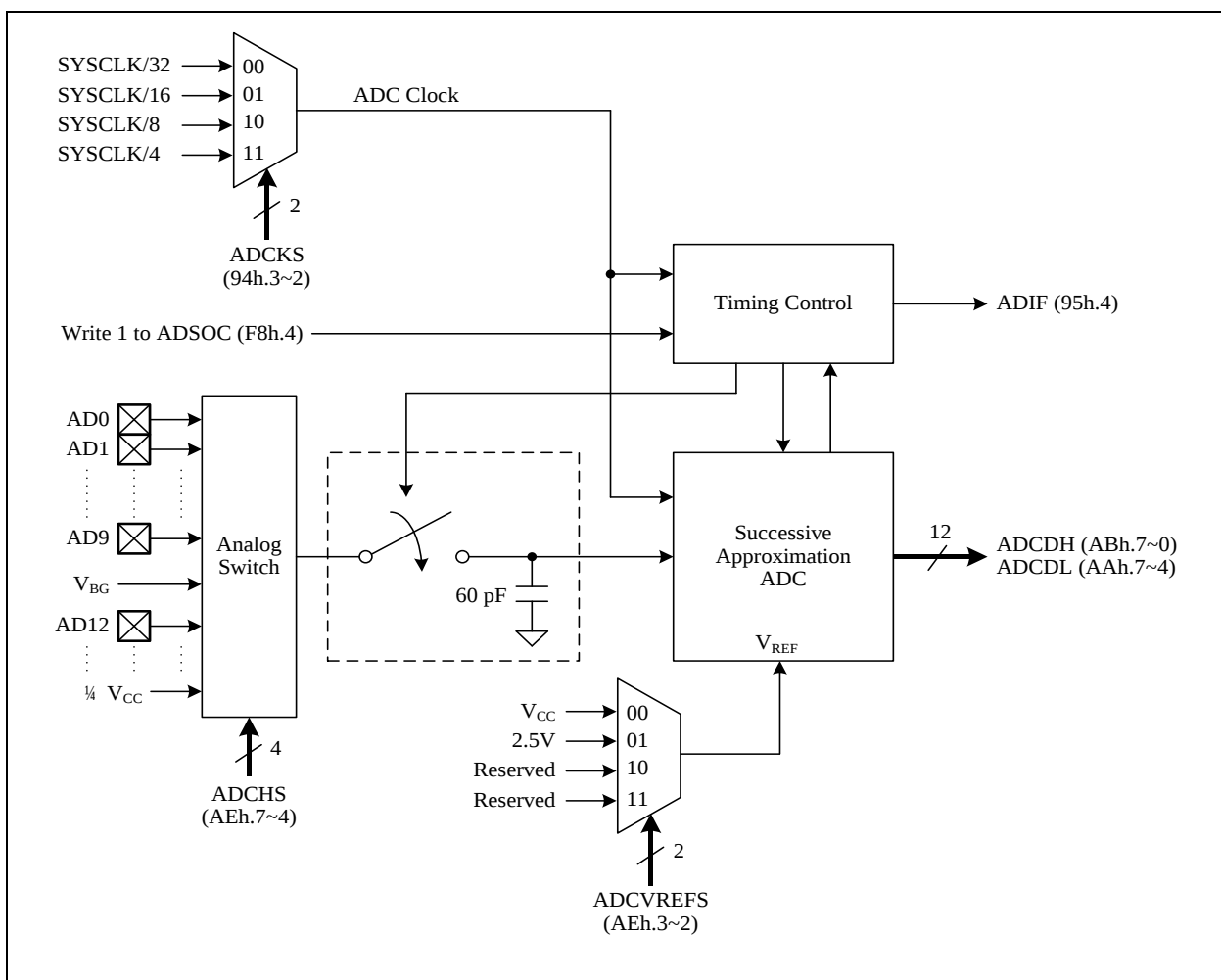
DDh.7~0 **PWM2PRDH**: PWM2 周期高字节
 写顺序: 先写 PWM2PRDL, 然后再写 PWM2PRDH
 读顺序: 先读 PWM2PRDH, 然后再读 PWM2PRDL

SFR DEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2PRDL	PWM2PRDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

DEh.7~0 **PWM2PRDL**: PWM2 周期低字节
 写顺序: 先写 PWM2PRDL, 然后再写 PWM2PRDH
 读顺序: 先读 PWM2PRDH, 然后再读 PWM2PRDL

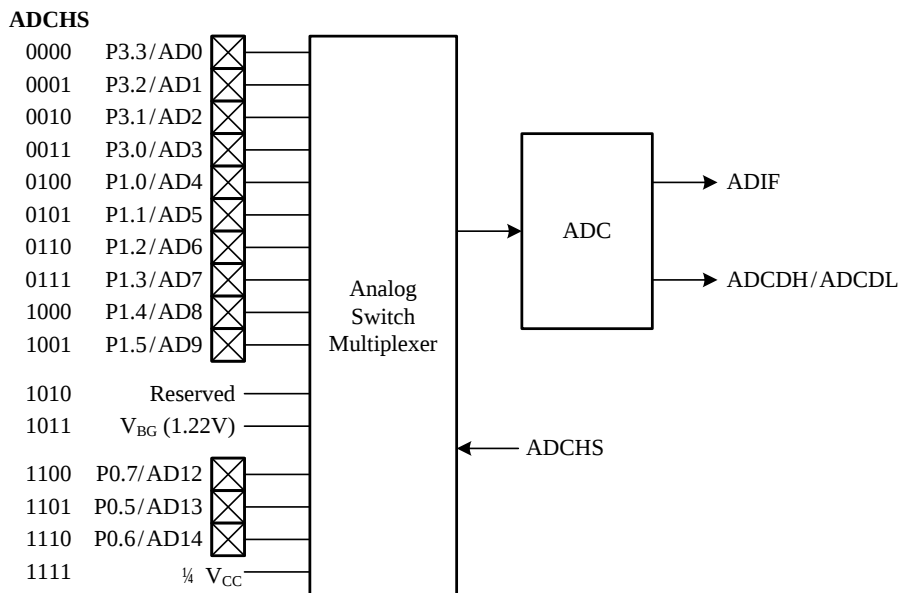
11. ADC

该芯片提供了一个 12 位 ADC 包括了 16 通道模拟输入多路复用器,控制寄存器,时钟发生器,12 位逐次逼近寄存器和输出数据寄存器。使用 ADC,首先要设置 ADCKS 位选择合适的 ADC 时钟频率,它必须小于 1 MHz。然后,设置 ADSOC 位启动 ADC 转换,在转换结束硬件将自动清除它。转换结束后硬件将设置 ADIF 位,当 ADC 中断使能,并产生一个中断。ADIF 位可以通过写 0 到该位或写 1 至 ADSOC 清零。因为 ADC 通道和触摸按键通道共享, ADC 的通道选择和触摸按键通道选择应避免设置在同一引脚,以避免 ADC/TK 通道互相干扰,而影响输入灵敏度。模拟输入电平必须保持从 V_{SS} 到 V_{CC} 的范围内。可以从以下二个电压中选择 ADC 的 V_{REF} : V_{CC} 和 2.5V。ADCHS 選擇到 V_{BG} 時, ADCVREFS 必需設置為 V_{CC} , 否則 ADC 轉換無效。



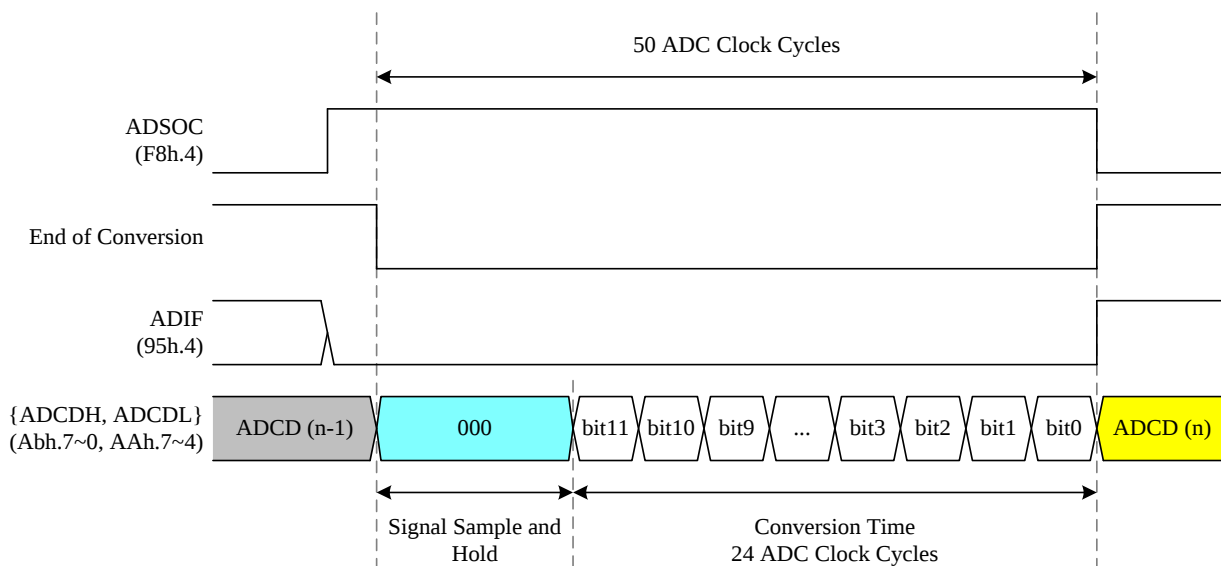
11.1 ADC 通道

12 位 ADC,一共有 16 个通道,指定为 AD0~AD9, AD12~AD14, V_{BG} 和 $1/4V_{CC}$ 。ADC 通道通过模拟开关多路复用器连接到模拟输入引脚。模拟开关多路复用器由 ADCHS 寄存器控制。 V_{BG} 是内部基准电压 1.22V。当 ADC 通道选择到 V_{BG} 时, V_{BG} 生成器将自动启用。通过设置 SFR VBGEN = 1 始终使能 V_{BG} 生成器, 用户可以获得更稳定的 V_{BG} 电压。



11.2 ADC 转换时间

转换时间是 ADC 转换电压所需要的时间。该 ADC 转换每个位需要两个 ADC 时钟周期,以及多个时钟周期进行输入电压采样和保持。一共需要 50 个 ADC 时钟周期以执行完全转化。当转换时间结束,ADIF 中断标志由硬件设置,12 位 A/D 转换结果被加载到 ADCDH 和 ADCDL 寄存器。



SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPTION	UART1W	–	WDTOSC		ADCKS		TM3PSC	
R/W	R/W	–	R/W		R/W		R/W	
Reset	0	–	0	0	0	0	0	0

94h.3~2 **ADCKS**:ADC 时钟频率选择

00:F_{SYSClk}/32

01:F_{SYSClk}/16

10:F_{SYSClk}/8

11:F_{SYSClk}/4

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTFLG	LVDIF	–	TKIF	ADIF	–	–	P1IF	TF3
R/W	R	–	R/W	R/W	–	–	R/W	R/W
Reset	–	–	0	0	–	–	0	0

95h.4 **ADIF**:ADC 中断标志

于转换结束时由硬件设置。软件写入 EFh 到 INTFLG 或设置 ADSOC 位来清除该标志。

注: S/W 可以写 0 清除 INTFLG 中的标志, 但写 1 没有任何效果。

SFR AAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCDL	ADCDL				–	–	–	–
R/W	R				–	–	–	–
Reset	–	–	–	–	–	–	–	–

Aah.7~4 **ADCDL**: ADC 数据位 3~0

SFR ABh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCDH	ADCDH							
R/W	R							
Reset	–	–	–	–	–	–	–	–

ABh.7~0 **ADCDH**:ADC 数据位 11~4

SFR AEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CHSEL	ADCHS				ADCVREFS		VBGEN	—
R/W	R/W				R/W	R/W	R/W	—
Reset	1	1	1	1	0	0	0	—

AEh.7~4 **ADCHS**: ADC 通道选择

0000: AD0 (P3.3)
0001: AD1 (P3.2)
0010: AD2 (P3.1)
0011: AD3 (P3.0)
0100: AD4 (P1.0)
0101: AD5 (P1.1)
0110: AD6 (P1.2)
0111: AD7 (P1.3)
1000: AD8 (P1.4)
1001: AD9 (P1.5)
1010: 保留
1011: V_{BG} (内部基准电压源)
1100: AD12 (P0.7)
1101: AD13 (P0.5)
1110: AD14 (P0.6)
1111: $1/4 V_{CC}$ (内部基准电压源)

AEh.3~2 **ADCVREFS**: ADC 参考电压

00: V_{CC}
01: 2.5V
10: 保留
11: 保留

AEh.1 **VBGEN**: 强制 V_{BG} 生成器启用

0: V_{BG} 生成器自动启用和禁用
1: 强制 V_{BG} 发生器启用包括在空闲模式下, 但在暂停 / 停止模式下禁用。

注: F/W 必须关闭 Bandgap 以获得微小电流 (ADCHS ≠ 1011b)

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.4 **ADSOC**: 启动 ADC 转换

设置 ADSOC 位启动 ADC 转换, ADSOC 位将由硬件于转换结束时被清除。软件也可以写 0 清除该标志。

12. 触摸按键 (FTK)

触摸键提供了一种简单，可靠的方法来实现手指触摸检测。在按键扫描操作期间，设备支持 20 个通道的触摸键检测。

要使用触摸键，用户应正确设置。有两种方法可以将 IO 设置为触摸通道。将 SFR PxMODx 设置为 11b 或将 SFR TKPINSEL0~2 设置为将 IO 强制为触摸通道。如果设置了 TKPINSEL0~2，则相应的 IO 引脚将被固定为触摸通道，并且将不再受 PxNMODx 的影响。

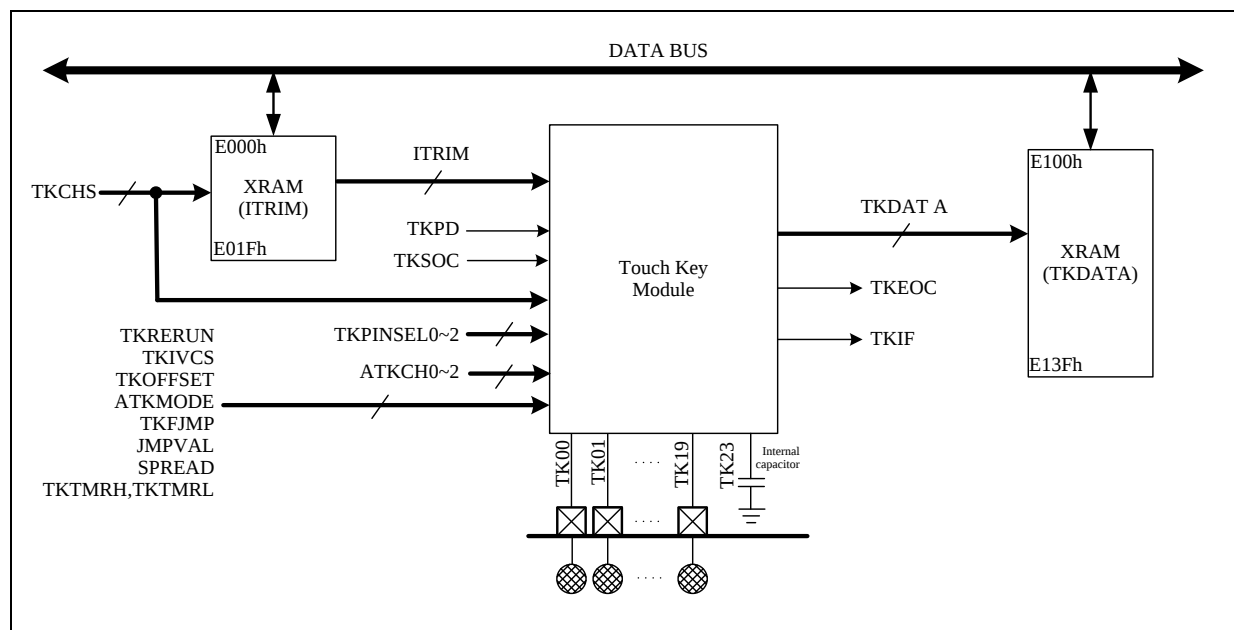
TKPINSEL	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSEL0	TK07	TK06	TK05	TK04	TK03	TK02	TK01	TK00
TKPINSEL1	TK15	TK14	TK13	TK12	TK11	TK10	TK09	TK08
TKPINSEL2					TK19	TK18	TK17	TK16

设置 TKPINSEL0~2 将 IO 固定为触摸通道

在触摸模式下，用户分配 TKPD = 0 来打开触摸模块，然后将 TKSOC 位置 1 以开始触摸键转换，转换结束时 TKSOC 位可以自动清零。但是，如果 F_{SYSCCLK} 太慢，由于时钟采样率，硬件可能无法清除 TKSOC。TKEOC = 0 表示正在进行转换。TKEOC = 1 表示转换完成，并且触摸键计数结果存储在 XRAM 中。在 TKEOC = 1 之后，用户必须等待至少 50 μs 才能进行下一次转换。但是，如果 TKRERUN = 1，则将始终转换触摸，因此无需为每次转换设置 TKSOC。减少/增加 TKTMR 可以减少/增加 TKDATA 以适应系统条件。

FTK 具有内部内置的参考电容器来模拟 KEY 行为。设置 TKCHS = 17h 并开始扫描可以获取内部参考电容器 (TKCAP) 的触摸数据计数。由于内部电容器不受水或手机的影响，因此对于比较环境背景噪声很有用。设置 TKFJMP，可以通过硬件控制自动更改触摸键时钟的频率。这可能有助于提高抗噪能力。

ITRIM 是触摸通道参考电压微调的 7 位数据。E000h.6~0 为 TK00 参考电压微调。E001h.6~0 是 TK01 参考电压微调。E017h.6~0 是 TKCAP (TK23) 参考电压微调等。用户可以使用 ITRIM 为不同的触摸通道获得相似的参考电压



FTK 结构

SFR ATKCH0~2 用于指定扫描触摸通道，并且每个位都映射到触摸引脚。触摸扫描将从低位扫描到高位。如果 ATKMODE = 0, 则 TK 最多可以扫描 21 个通道, TK00~TK19 和 TKCAP(TK23), 每个通道扫描一次。如果 ATKMODE = 1, 则 TK 最多可以扫描 16 个通道, 每个通道扫描两次。如果 ATKMODE = 2, 则 TK 最多可以扫描 8 个通道, 每个通道被扫描 4 次。如果 ATKMODE = 3, 则 TK 最多可以扫描 4 个通道, 每个通道被扫描 8 次。

例如：

条件 ATKMODE = 0, 扫描 TK16/TK14/TK08/TK07/TK06/TK02

⇒ TKPINSEL2=0000_0001, TKPINSEL1=0100_0001, TKPINSEL0=1100_0100

⇒ ATKCH2=0000_0001, ATKCH1=0100_0001, ATKCH0=1100_0100

⇒ TKCHS=0x02 (指定第一个扫描通道)

存储在 XRAM 中的触摸数据的排列如下。

XRAM	
E100h	TK00 DATAL
E101h	TK00 DATAH
E102h	TK01 DATAL
E103h	TK01 DATAH
...	
E126h	TK19 DATAL
E127h	TK19 DATAH
...	
E12Eh	TK23 DATAL
E12Fh	TK23 DATAH
...	
E13Fh	

触摸扫描结果为 14 位数据，分别为 DATAH 6 位与 DATAL 8 位，DATAH/L 必需依照顺序读取，才能得到正确的 14 位数据：**首先读取低字节 (DATAL)，然后读取高字节 (DATAH)**

条件 ATKMODE = 1, 扫描 TK16/TK14/TK08/TK07/TK06/TK02

⇒ TKPINSEL2=0000_0001, TKPINSEL1=0100_0001, TKPINSEL0=1100_0100

⇒ ATKCH2=0000_0001, ATKCH1=0100_0001, ATKCH0=1100_0100

⇒ TKCHS=0x02 (指定第一个扫描通道)

存储在 XRAM 中的触摸数据的排列如下。

XRAM	
E100h	TK02 1 st DATAL
E101h	TK02 1 st DATAH
E102h	TK02 2 nd DATAL
E103h	TK02 2 nd DATAH
E104h	TK06 1 st DATAL
E105h	TK06 1 st DATAH
E106h	TK06 2 nd DATAL
E107h	TK06 2 nd DATAH
	...
E114h	TK16 1 st DATAL
E115h	TK16 1 st DATAH
E116h	TK16 2 nd DATAL
E117h	TK16 2 nd DATAH
	...
E13Fh	

触摸扫描结果为 14 位数据, 分别为 DATAH 6 位与 DATAL 8 位, DATAH/L 必需依照顺序读取, 才能得到正确的 14 位数据: **首先读取低字节 (DATAL), 然后读取高字节 (DATAH)**

条件 ATKMODE = 2, 扫描 TK16/TK14/TK08/TK07/TK06/TK02

⇒ TKPINSEL2=0000_0001, TKPINSEL1=0100_0001, TKPINSEL0=1100_0100

⇒ ATKCH2=0000_0001, ATKCH1=0100_0001, ATKCH0=1100_0100

⇒ TKCHS=0x02 (指定第一个扫描通道)

存储在 XRAM 中的触摸数据的排列如下。

XRAM	
E100h	TK02 1 st DATAL
E101h	TK02 1 st DATAH
E102h	TK02 2 nd DATAL
E103h	TK02 2 nd DATAH
E104h	TK02 3 rd DATAL
E105h	TK02 3 rd DATAH
E106h	TK02 4 th DATAL
E107h	TK02 4 th DATAH
E108h	TK06 1 st DATAL
E109h	TK06 1 st DATAH
E10Ah	TK06 2 nd DATAL
E10Bh	TK06 2 nd DATAH
E10Ch	TK06 3 rd DATAL
E10Dh	TK06 3 rd DATAH
E10Eh	TK06 4 th DATAL
E10Fh	TK06 4 th DATAH
	...
E128h	TK16 1 st DATAL
E129h	TK16 1 st DATAH
E12Ah	TK16 2 nd DATAL
E12Bh	TK16 2 nd DATAH
E12Ch	TK16 3 rd DATAL
E12Dh	TK16 3 rd DATAH
E12Eh	TK16 4 th DATAL
E12Fh	TK16 4 th DATAH
	...
E13Fh	

触摸扫描结果为 14 位数据, 分别为 DATAH 6 位与 DATAL 8 位, DATAH/L 必需依照顺序读取, 才能得到正确的 14 位数据: **首先读取低字节 (DATAL), 然后读取高字节 (DATAH)**

条件 ATKMODE = 3, 扫描 TK08/TK07/TK06/TK02

⇒ TKPINSEL2=0000_0000, TKPINSEL1=0000_0001, TKPINSEL0=1100_0100

⇒ ATKCH2=0000_0000, ATKCH1=0000_0001, ATKCH0=1100_0100

⇒ TKCHS=0x02 (指定第一个扫描通道)

存储在 XRAM 中的触摸数据的排列如下。

XRAM	
E100h	TK02 1 st DATAL
E101h	TK02 1 st DATAH
E102h	TK02 2 nd DATAL
E103h	TK02 2 nd DATAH
E104h	TK02 3 rd DATAL
E105h	TK02 3 rd DATAH
E106h	TK02 4 th DATAL
E107h	TK02 4 th DATAH
E108h	TK02 5 th DATAL
E109h	TK02 5 th DATAH
E10Ah	TK02 6 th DATAL
E10Bh	TK02 6 th DATAH
E10Ch	TK02 7 th DATAL
E10Dh	TK02 7 th DATAH
E10Eh	TK02 8 th DATAL
E10Fh	TK02 8 th DATAH
	...
E130h	TK08 1 st DATAL
E131h	TK08 1 st DATAH
E132h	TK08 2 nd DATAL
E133h	TK08 2 nd DATAH
E134h	TK08 3 rd DATAL
E135h	TK08 3 rd DATAH
E136h	TK08 4 th DATAL
E137h	TK08 4 th DATAH
E138h	TK08 5 th DATAL
E139h	TK08 5 th DATAH
E13Ah	TK08 6 th DATAL
E13Bh	TK08 6 th DATAH
E13Ch	TK08 7 th DATAL
E13Dh	TK08 7 th DATAH
E13Eh	TK08 8 th DATAL
E13Fh	TK08 8 th DATAH

触摸扫描结果为 14 位数据, 分别为 DATAH 6 位与 DATAL 8 位, DATAH/L 必需依照顺序读取, 才能得到正确的 14 位数据: **首先读取低字节 (DATAL), 然后读取高字节 (DATAH)**

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTFLG	LVDIF	–	TKIF	ADIF	–	–	P1IF	TF3
R/W	R	–	R/W	R/W	–	–	R/W	R/W
Reset	–	–	0	0	–	–	0	0

95h.5 **TKIF:** 触摸键中断标志

如果 F_{SYSCLK} 足够快，则在触摸按键转换结束时通过硬件设置。软件将 DFh 写入 INTFLG 或将 TKSOC 设置 1 以清除该标志。

SFR ADh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON	TKPD	TKEOC	TKRERUN	TKIVCS	TKXCAP	TKOFFSET	ATKMODE	
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Reset	1	1	0	0	0	0	0	0

ADh.7 **TKPD:** 触摸键开关

0: 触摸键启用

1: 触摸键禁用

ADh.6 **TKEOC:** 触摸键转换标志结束时，在 TKSOC = 1 之后 TKEOC 可能有 3uS 的延迟，因此 F / W 必须等待足够的时间才能轮询该标志。

0: 表示正在进行转换

1: 表示转换完成

ADh.5 **TKRERUN:** 触摸键自动重新启动，无需重新设置 TKSOC 即可重新启动触摸转换器。

0: 自动重启禁用。每次触摸转换都需要执行一次 TKSOC

1: 自动重启启用。TKSOC 执行一次后，触摸键将连续转换而无需重新执行 TKSOC

ADh.4 **TKIVCS:** 触摸键操作电压选择

0: VCHG=2.8V; VINT=1.4V

1: VCHG=3.6V; VINT=1.8V

ADh.3 **TKXCAP:** 触摸键外部电容器选择

0: 禁用触摸键外部电容器

1: 启用触摸键外部电容器

ADh.2 **TKOFFSET:** 非扫描触摸键的状态

0: 连接到 V_{SS}

1: 连接到交流屏蔽，连接到 V_{SS} @ EOC

ADh.1~0 **ATKMODE:** 触摸键扫描模式

00: 触摸键扫描方法，每个通道扫描 1 次，最多 21 个触摸通道

01: 触摸键扫描方法，每个通道扫描 2 次，最多 16 个触摸通道

10: 触摸键扫描方法，每个通道扫描 4 次，最多 8 个触摸通道

11: 触摸键扫描方法，每个通道扫描 8 次，最多 4 个触摸通道

注意：有关触摸键中断使能和优先级的更多信息，另请参阅第 6 节。

SFR B4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKTMRL	TKTMRL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

B4h.7~0 **TKTMRL**: 触摸键扫描长度位 7~0 调整。

00: 最短 FF: 最长

SFR B5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCON2	TKFJMP	JMPVAL		SPREAD	TKTMRH			
R/W	R/W	R/W		R/W	R/W			
Reset	0	0	0	0	0	0	0	0

B5h.7 **TKFJMP**: 内部触摸键时钟频率自动调整选项

0: 禁用

1: 启用

B5h.6~5 **JMPVAL**: 触摸键时钟频率微调, 仅在 TKFJMP = 0 中可用

00 = 频率最低, 11 = 频率最高

B5h.4 **TKFJMP**: 内部触摸键时钟频率自动调整选项

0: 禁用

1: 启用

B5h.3~0 **TKTMRH**: 触摸键扫描长度 11~8 调节。

0000: 最短 1111: 最长

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

F8h.5 **TKSOC**: 触摸键转换开始

设置为1即可开始触摸键转换。如果F_{SYSCCLK}足够快, 该位将由硬件清除。软件也可以写0清除该标志。

SFR A7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKCHS	—	—	—	TKCHS				
R/W	—	—	—	R/W				
Reset	—	—	—	1	1	1	1	1

A7h.4~0 **TKCHS:** 触摸按键通道选择

00000: TK0 (P3.3)
 00001: TK1 (P3.2)
 00010: TK2 (P3.1)
 00011: TK3 (P3.0)
 00100: TK4 (P1.0)
 00101: TK5 (P1.1)
 00110: TK6 (P1.2)
 00111: TK7 (P1.3)
 01000: TK8 (P1.4)
 01001: TK9 (P1.6)
 01010: TK10 (P1.7)
 01011: TK11 (P3.6)
 01100: TK12 (P3.5)
 01101: TK13 (P3.4)
 01110: TK14 (P1.5)
 01111: TK15 (P3.7)
 10000: TK16 (P0.3)
 10001: TK17 (P0.5)
 10010: TK18 (P0.6)
 10011: TK19 (P0.7)
 10111: TKCAP 内部参考电容通道

SFR C1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSEL0	TKPINSEL0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C1h.7 TK07 引脚固定为触摸键通道：0：禁用 1：启用
 C1h.6 TK06 引脚固定为触摸键通道：0：禁用 1：启用
 C1h.5 TK05 引脚固定为触摸键通道：0：禁用 1：启用
 C1h.4 TK04 引脚固定为触摸键通道：0：禁用 1：启用
 C1h.3 TK03 引脚固定为触摸键通道：0：禁用 1：启用
 C1h.2 TK02 引脚固定为触摸键通道：0：禁用 1：启用
 C1h.1 TK01 引脚固定为触摸键通道：0：禁用 1：启用
 C1h.0 TK00 引脚固定为触摸键通道：0：禁用 1：启用

SFR C2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSEL1	TKPINSEL1							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C2h.7 TK15 引脚固定为触摸键通道：0：禁用 1：启用
 C2h.6 TK14 引脚固定为触摸键通道：0：禁用 1：启用
 C2h.5 TK13 引脚固定为触摸键通道：0：禁用 1：启用
 C2h.4 TK12 引脚固定为触摸键通道：0：禁用 1：启用
 C2h.3 TK11 引脚固定为触摸键通道：0：禁用 1：启用
 C2h.2 TK10 引脚固定为触摸键通道：0：禁用 1：启用
 C2h.1 TK09 引脚固定为触摸键通道：0：禁用 1：启用
 C2h.0 TK08 引脚固定为触摸键通道：0：禁用 1：启用

SFR C3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TKPINSEL2	TKPINSEL2							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C3h.7~4 保留
 C3h.3 TK19 引脚固定为触摸键通道：0：禁用 1：启用
 C3h.2 TK18 引脚固定为触摸键通道：0：禁用 1：启用
 C3h.1 TK17 引脚固定为触摸键通道：0：禁用 1：启用
 C3h.0 TK16 引脚固定为触摸键通道：0：禁用 1：启用

SFR C5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCH0	ATKCH0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C5h.7 TK07 扫描启用: 0: 禁用 1: 启用
 C5h.6 TK06 扫描启用: 0: 禁用 1: 启用
 C5h.5 TK05 扫描启用: 0: 禁用 1: 启用
 C5h.4 TK04 扫描启用: 0: 禁用 1: 启用
 C5h.3 TK03 扫描启用: 0: 禁用 1: 启用
 C5h.2 TK02 扫描启用: 0: 禁用 1: 启用
 C5h.1 TK01 扫描启用: 0: 禁用 1: 启用
 C5h.0 TK00 扫描启用: 0: 禁用 1: 启用

SFR C6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCH1	ATKCH1							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

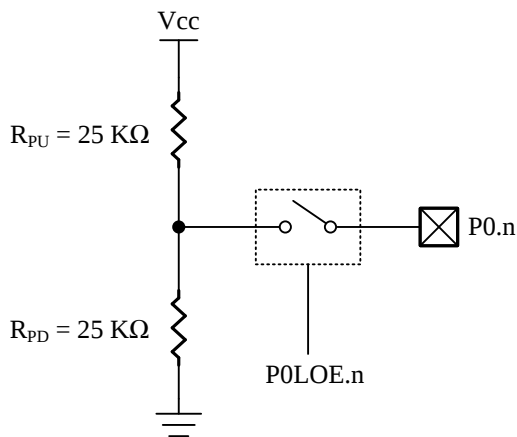
C6h.7 TK15 扫描启用: 0: 禁用 1: 启用
 C6h.6 TK14 扫描启用: 0: 禁用 1: 启用
 C6h.5 TK13 扫描启用: 0: 禁用 1: 启用
 C6h.4 TK12 扫描启用: 0: 禁用 1: 启用
 C6h.3 TK11 扫描启用: 0: 禁用 1: 启用
 C6h.2 TK10 扫描启用: 0: 禁用 1: 启用
 C6h.1 TK09 扫描启用: 0: 禁用 1: 启用
 C6h.0 TK08 扫描启用: 0: 禁用 1: 启用

SFR C7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATKCH2	ATKCH2							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

C7h.7 TKCAP (TK23) 内部参考电容通道扫描使能: 0: 禁用 1: 启用
 C7h.6~4 保留
 C7h.3 TK19 扫描启用: 0: 禁用 1: 启用
 C7h.2 TK18 扫描启用: 0: 禁用 1: 启用
 C7h.1 TK17 扫描启用: 0: 禁用 1: 启用
 C7h.0 TK16 扫描启用: 0: 禁用 1: 启用

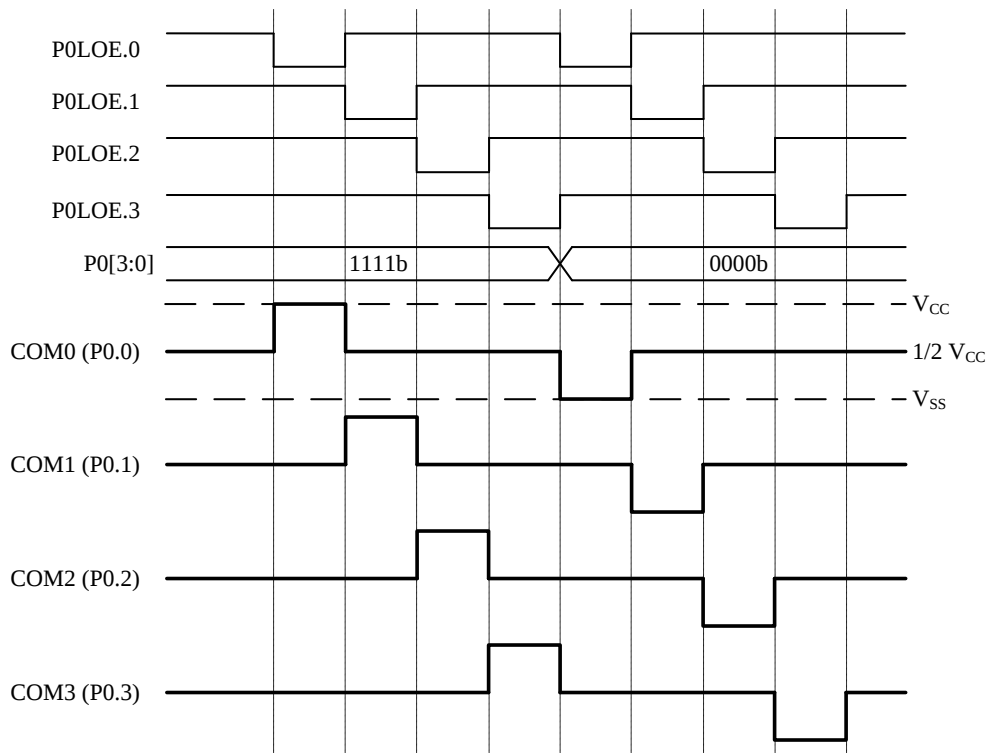
13. 软件控制的 LCD 驱动器

该芯片支持软件控制的方法来驱动 LCD。它能够通过 8 个公共端 (COM) 和 18 个段 (SEG) 驱动 144 点 (最大) 的 LCD 面板。P0.0~P0.7 用于公共引脚 COM0~COM7, 其他引脚用于段引脚。当 P0.0~P0.7 的 P0LOE = 1 时, COM0~COM7 能够驱动 1/2 偏压。请参考下图。



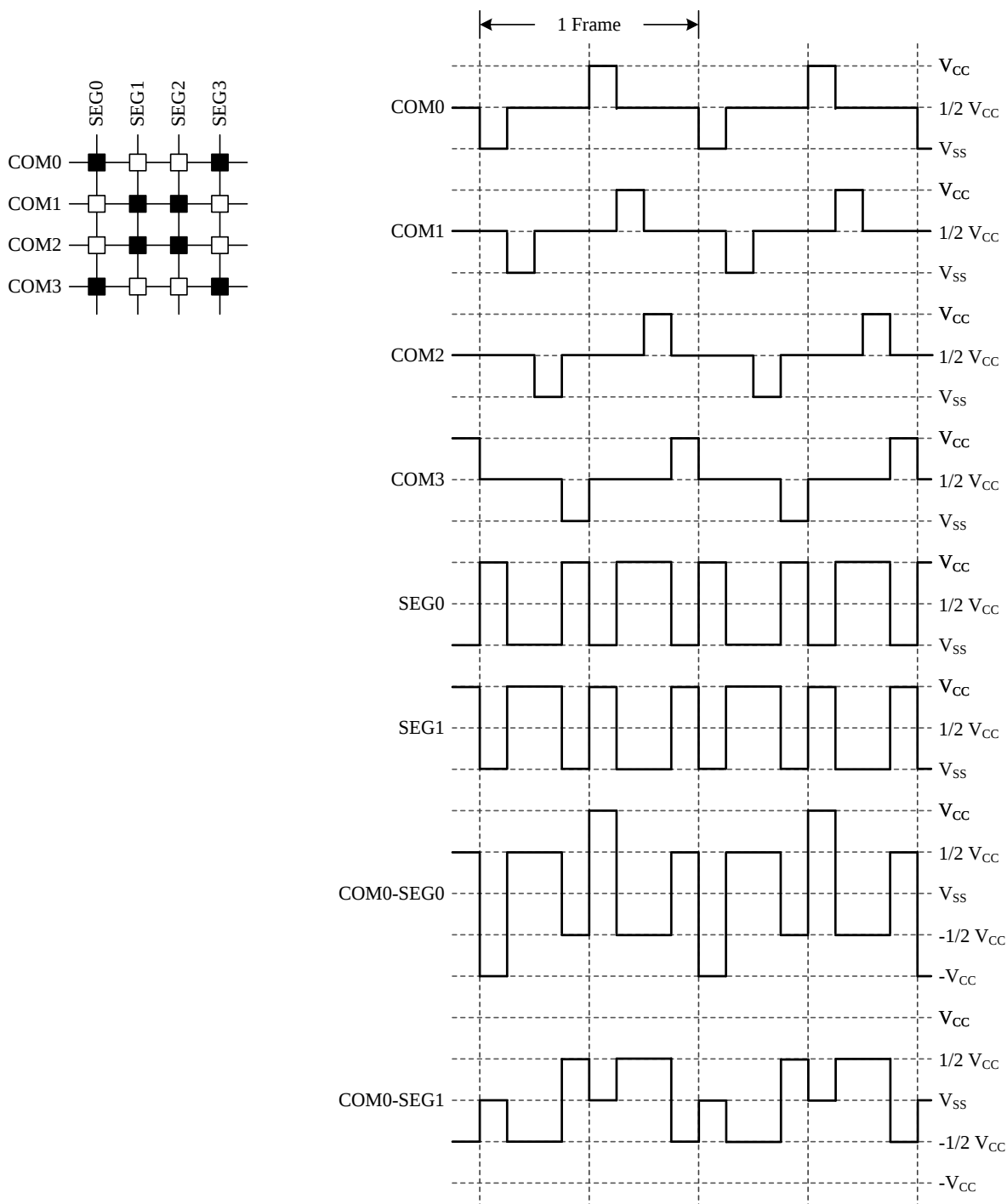
LCD COM0~7 Circuit

在 COM 引脚上输出的任何重复波形的频率都可以用来表示 LCD 帧速率。下图显示了一个 LCD 框架。



软件控制的 LCD COM0~3 扫描

1/4 占空比,1/2 偏压输出波形



SFR 92h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P0LOE	P0LOE							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

92h.7~0 P0LOE: P0.7~P0.0 LCD 1/2 偏置输出使能控制

0: 禁用

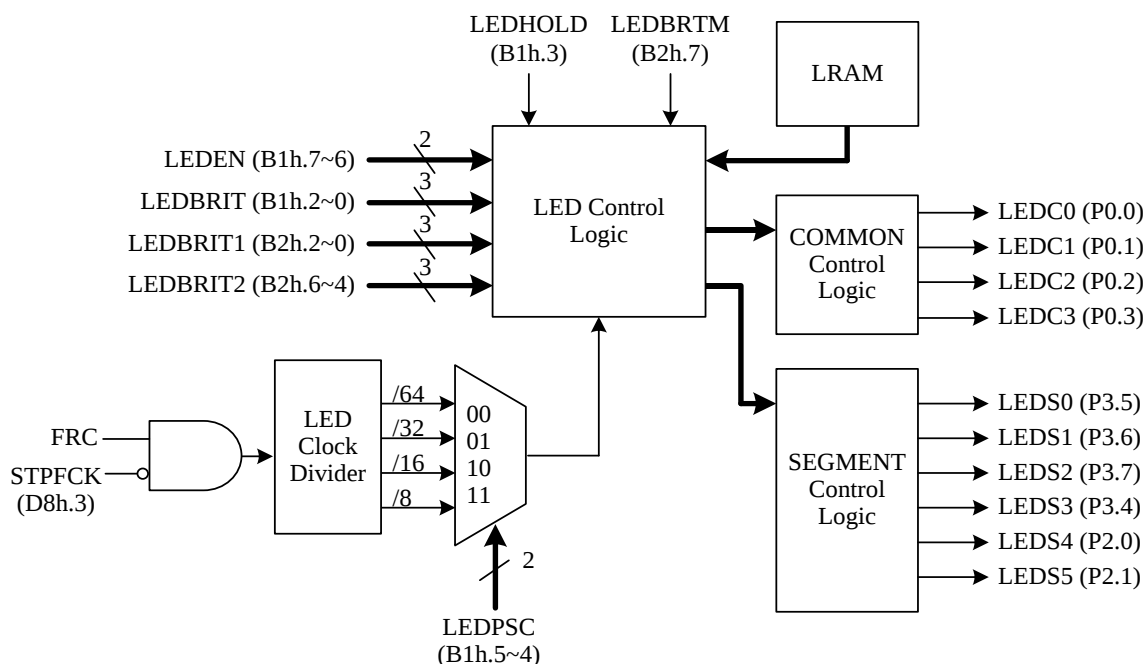
1: 使能

14. LED 控制器/驱动器

该模块可以配置两种驱动模式：LED 正反扫模式 (BiD) 和 LED 点矩阵模式 (DMX)。通过寄存器配置，它仅同时支持一种操作模式。

14.1 LED 正反扫模式 (BiD)

当 LED BiD 模式使用相同数量的引脚时，可以驱动比传统矩阵模式更多的 LED 像素。在此模式下，它最多提供 10 个引脚 (LEDC0~C3, LEDS0~S5) 来驱动 48 个像素的 LED 模块。所有 10 个引脚均具有较高的灌电流，可直接驱动 LED。该 LED 控制器还为所有 10 引脚提供 3 组 8 级亮度调节。为了避免公共信号发生变化时 LED 闪烁，该芯片提供了死区时间控制。在死区时间段中，段引脚将输出短时的无效信号，而不是立即更改信号。要开始 LED 扫描，只需设置 LEDEN。然后，硬件将自动控制引脚模式。还可通过设置 LEDHOLD 提供扫描暂停功能。

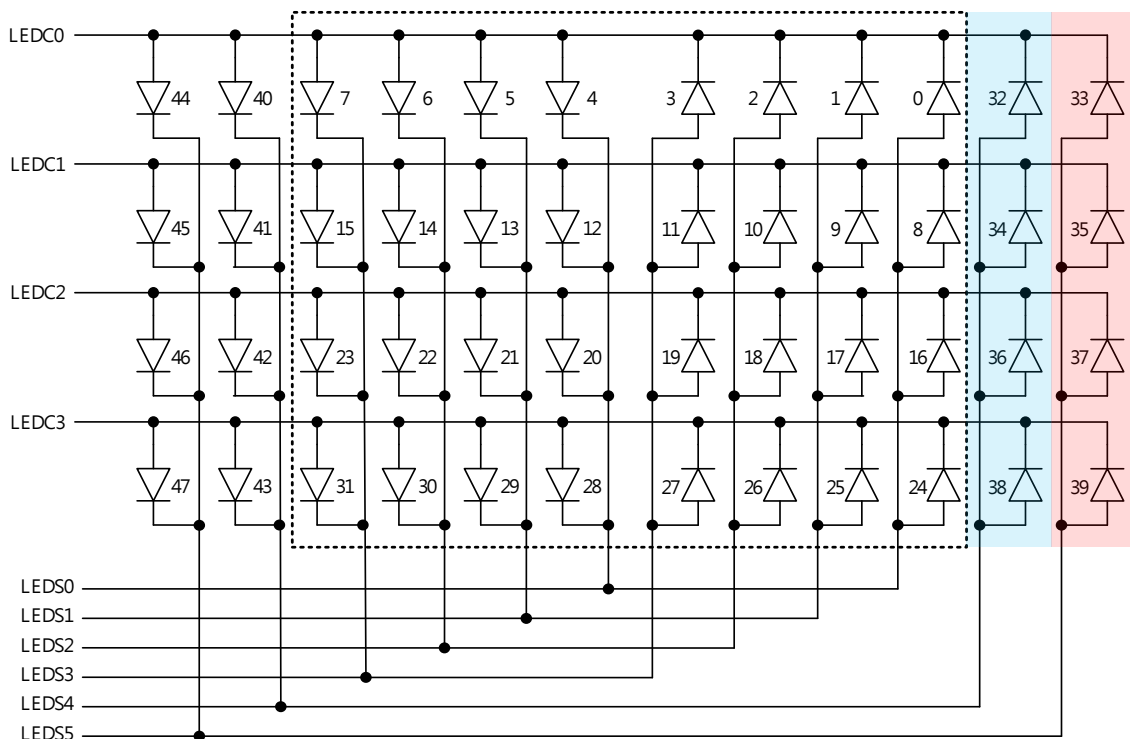


LEDEN	占空比	矩阵选择	最大驱动像素点
0	Disable	-	-
1	1/8	4COM x 4SEG	32 (4x4x2)
2	1/9	4COM x 5SEG	40 (4x5x2)
3	1/10	4COM x 6SEG	48 (4x6x2)

LRAM Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
C800h	SEG3-COM0+	SEG2-COM0+	SEG1-COM0+	SEG0-COM0+	COM0-SEG3+	COM0-SEG2+	COM0-SEG1+	COM0-SEG0+
C801h	SEG3-COM1+	SEG2-COM1+	SEG1-COM1+	SEG0-COM1+	COM1-SEG3+	COM1-SEG2+	COM1-SEG1+	COM1-SEG0+
C802h	SEG3-COM2+	SEG2-COM2+	SEG1-COM2+	SEG0-COM2+	COM2-SEG3+	COM2-SEG2+	COM2-SEG1+	COM2-SEG0+
C803h	SEG3-COM3+	SEG2-COM3+	SEG1-COM3+	SEG0-COM3+	COM3-SEG3+	COM3-SEG2+	COM3-SEG1+	COM3-SEG0+
C804h	COM3-SEG5+	COM3-SEG4+	COM2-SEG5+	COM2-SEG4+	COM1-SEG5+	COM1-SEG4+	COM0-SEG5+	COM0-SEG4+
C805h	SEG5-COM3+	SEG5-COM2+	SEG5-COM1+	SEG5-COM0+	SEG4-COM3+	SEG4-COM2+	SEG4-COM1+	SEG4-COM0+

LRAM Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
C800h	7	6	5	4	3	2	1	0
C801h	15	14	13	12	11	10	9	8
C802h	23	22	21	20	19	18	17	16
C803h	31	30	29	28	27	26	25	24
C804h	39	38	37	36	35	34	33	32
C805h	47	46	45	44	43	42	41	40

LED 正反扫模式对应 LRAM 显示配置表



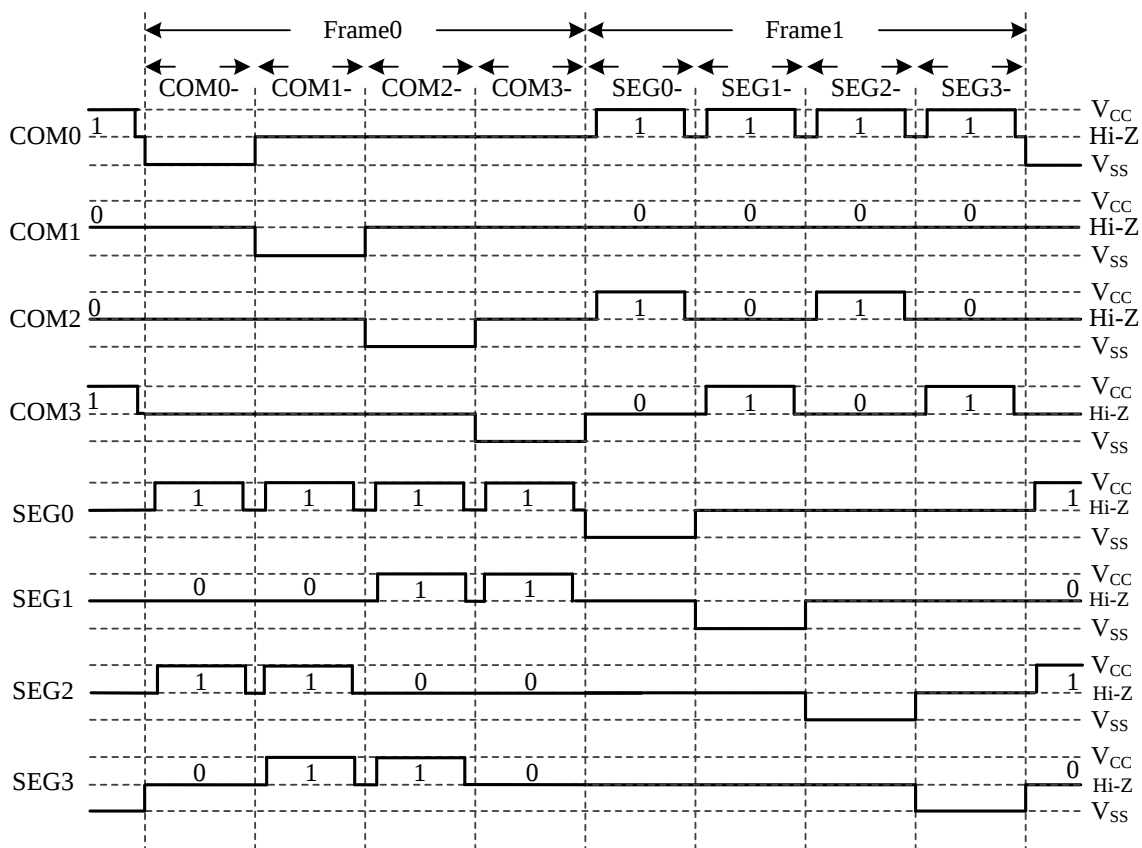
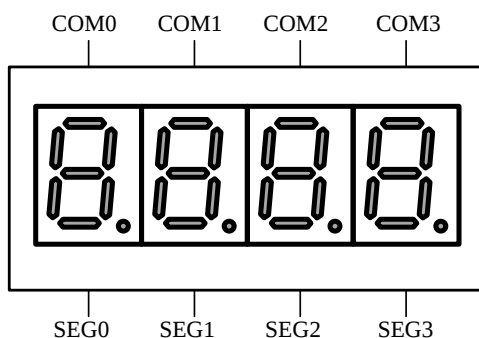
LED 4COM x 6SEG 正反扫模式对应像素点

注: LED 像素点 0~31, 40~47 的亮度由 LEDBRIT (B1h.2~0) 控制。

LED 像素点 32, 34, 36, 38 的亮度由 LEDBRIT1 (B2h.2~0) 控制。

LED 像素点 33, 35, 37, 39 的亮度由 LEDBRIT2 (B2h.6~4) 控制。

应用电路：4COM x 4SEG (1/8 占空比)



◇ Example:

```

MOV    DPTR,#0C800h    ; LEDRAM0
MOV    A,#0FFh
MOVX   @DPTR, A        ; C800h = FFh

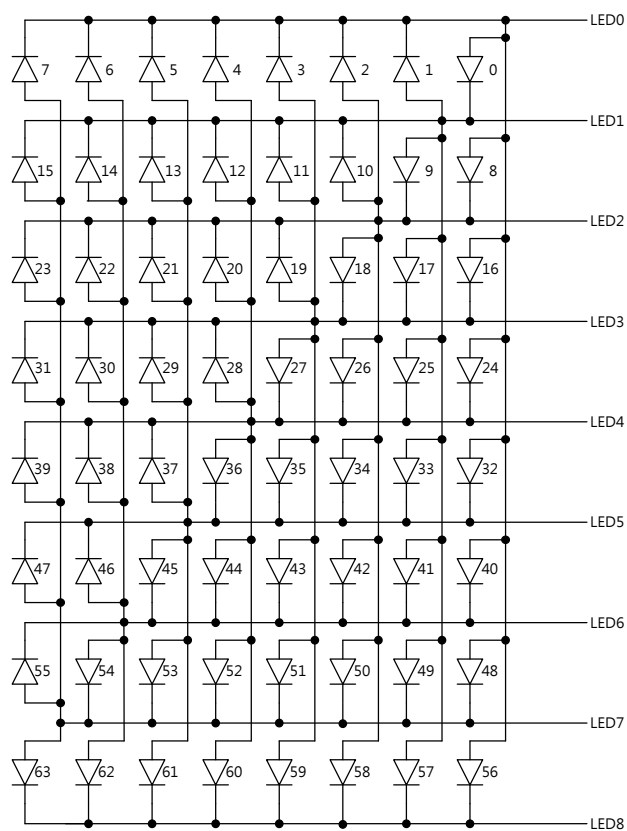
MOV    LEDCON,#056h    ; LED duty = 1/8
                        ; LEDPSC = FRC/32
                        ; Brightness=6
    
```

14.2 LED 点矩阵模式(DMX)

如果 LEDMTEN = 1, 则将启用 LED 点矩阵模式。LED 点矩阵是通用的 8 * 8 点矩阵。对应 LED0 ~LED8 端口, 最多可配置 8x8 = 56 个 LED 点进行驱动, LED 的对应位置在下图的 8 * 8 点矩阵中标记地址, XRAM 中的显示配置对应于照明 相应地址的状态(1 表示点亮, 0 表示不点亮)。支持多达 64 灯的 LED 驱动器。使用 LEDCON3 选择点矩阵 4 * 5、5 * 6、6 * 7、7 * 8 或 8 * 8, 相应的 LED 地址保持不变。可以通过 LCDBRIT2 设置 LED 的亮度。设置为 1111b 时, 它是最高亮度。另外, LEDBRITM 用于设置亮度和均匀度位。当 LEDBRITM = 0 时, 可以获得更好的显示均匀性。当 LEDBRITM = 1 时, 可以获得更好的显示亮度。LED 点矩阵模式的 SEG 信号也有支持死区时间, 避免造成 LED 的闪烁, 亦可通过设置 LEDHOLD 提供扫描暂停功能。

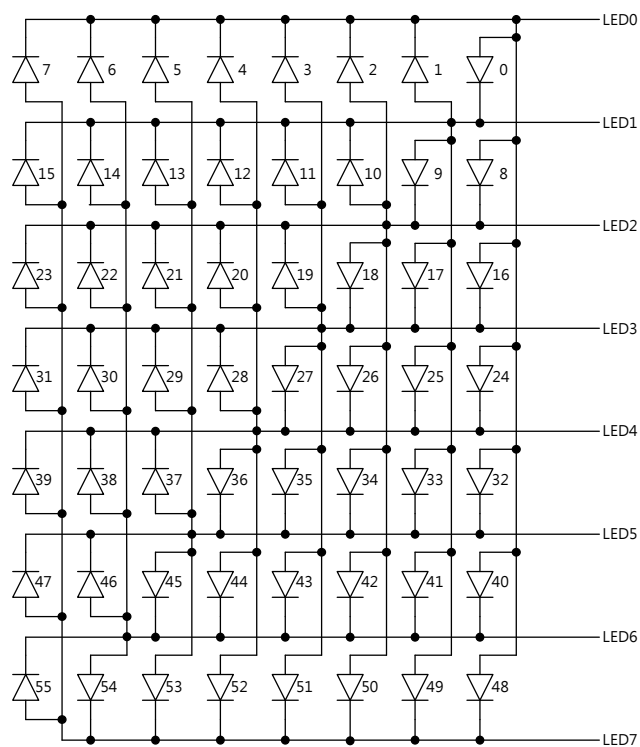
XRAM Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
C800h	7	6	5	4	3	2	1	0
C801h	15	14	13	12	11	10	9	8
C802h	23	22	21	20	19	18	17	16
C803h	31	30	29	28	27	26	25	24
C804h	39	38	37	36	35	34	33	32
C805h	47	46	45	44	43	42	41	40
C806h	55	54	53	52	51	50	49	48
C807h	63	62	61	60	59	58	57	56

LED 点矩阵模式对应的显示配置表

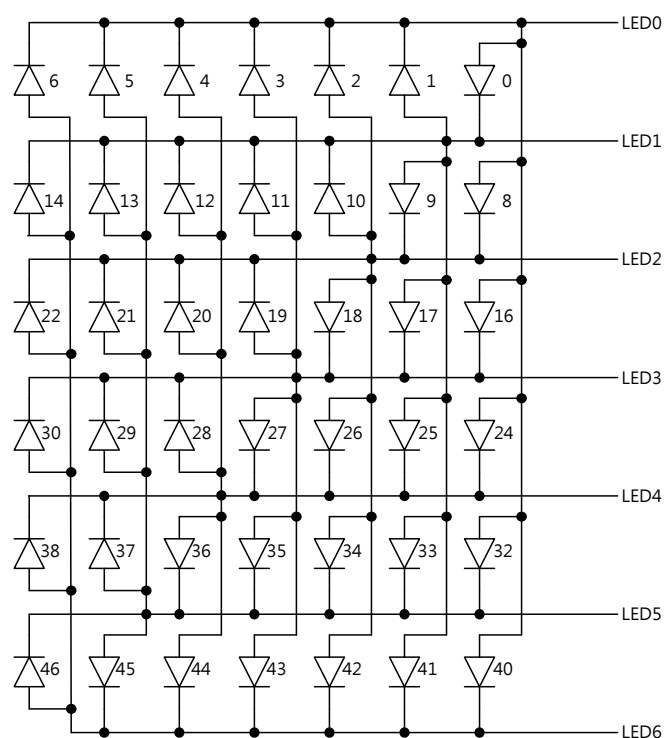


LED 8*8 点矩阵对应像素点

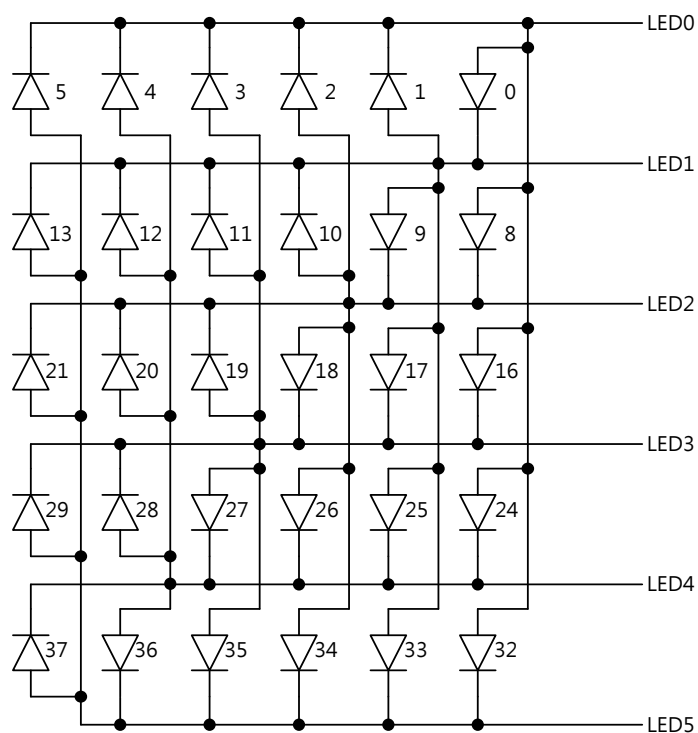
注: LED 像素点 0~63 的亮度由 LEDBRIT2 (B2h.6~4) 控制



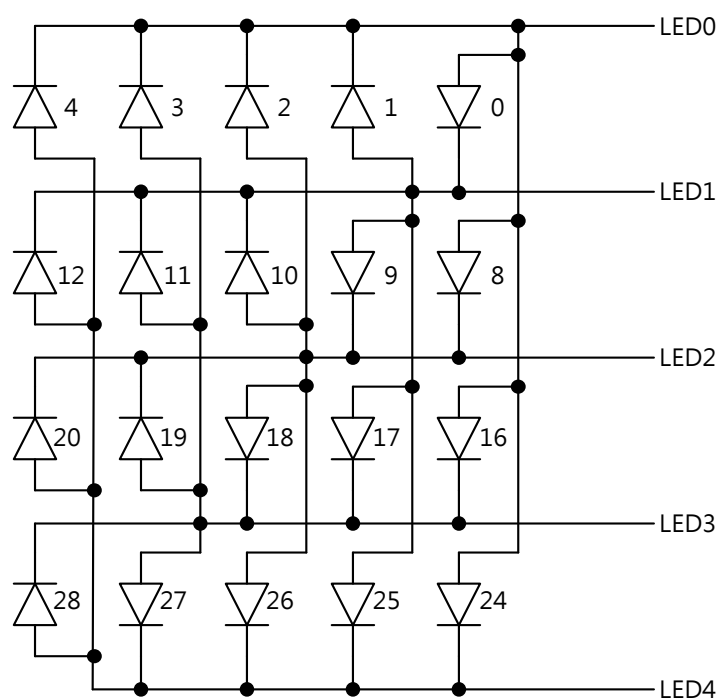
LED 7*8 点矩阵对应像素点



LED 6*7 点矩阵对应像素点



LED 5*6 点矩阵对应像素点



LED 4*5 点矩阵对应像素点

SFR B1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LEDCON	LEDEN		LEDPSC		LEDHOLD	LEDBRIT		
R/W	R/W		R/W		R/W	R/W		
Reset	0	0	0	0	0	1	1	1

B1h.7~6 **LEDEN**: LED 正反扫模式使能和占空比选择

- 00: LED 正反扫模式关闭
- 01: LED 1/8占空比(4COM x 4SEG)
- 10: LED 1/9占空比(4COM x 5SEG)
- 11: LED 1/10占空比(4COM x 6SEG)

B1h.5~4 **LEDPSC**: LED 时钟预分频器选择

- 00: LED时钟选择FRC除以64
- 01: LED时钟选择FRC除以32
- 10: LED时钟选择FRC除以16
- 11: LED时钟选择FRC除以8

B1h.3 **LEHOLD**: LED 暂停功能

- 0: 释放以运行LED扫描
- 1: 暂停 LED 扫描, 所有 LED 引脚状态均为 Hi-Z

B1h.2~0 **LEDBRIT**:

LED 正反扫模式: LED 编号 0~31, 40~47 亮度控制

- 000: 等级 0 (最暗的)
- ...
- 111: 等级 7 (最亮的)

SFR B2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LEDCON2	LEDBRITM	LEDBRIT2			—	LEDBRIT1		
R/W	R/W	R/W			—	R/W		
Reset	0	1	1	1	—	1	1	1

B2h.7 **LEDBRITM**: 亮度模式控制

- 0: 均匀亮度模式
- 1: 亮度增强模式

B2h.6~4 **LEDBRIT2**:

LED 正反扫模式: LED 像素点 33, 35, 37, 39 亮度控制

LED 点矩阵模式: LED 像素点 0~63 亮度控制

- 000: 等级 0 (最暗的)
- ...
- 111: 等级 7 (最亮的)

B2h.2~0 **LEDBRIT1**:

LED 正反扫模式: LED 数字 32、34、36、38 亮度控制

- 000: 等级 0 (最暗的)
- ...
- 111: 等级 7 (最亮的)

SFR B3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LEDCON3	LEDMTEN	LED8EN	LED7EN	LED6EN	LED5EN	LED4EN	LED3EN	LED2EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- B3h.7 **LEDMTEN**: LED 点矩阵模式使能
0: LED 点矩阵模式关闭
1: LED 点矩阵模式使能且 LED0, LED1 使能
- B3h.6 **LED8EN**: LED 点矩阵模式引脚使能控制
0: LED8 关闭
1: LED8 使能
- B3h.5 **LED7EN**: LED 点矩阵模式引脚使能控制
0: LED7 关闭
1: LED7 使能
- B3h.4 **LED6EN**: LED 点矩阵模式引脚使能控制
0: LED6 关闭
1: LED6 使能
- B3h.3 **LED5EN**: LED 点矩阵模式引脚使能控制
0: LED5 关闭
1: LED5 使能
- B3h.2 **LED4EN**: LED 点矩阵模式引脚使能控制
0: LED4 关闭
1: LED4 使能
- B3h.1 **LED3EN**: LED 点矩阵模式引脚使能控制
0: LED3 关闭
1: LED3 使能
- B3h.0 **LED2EN**: LED 点矩阵模式引脚使能控制
0: LED2 关闭
1: LED2 使能

SFR D8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CLKCON	SCKTYPE	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	1	0	0	0	1	1

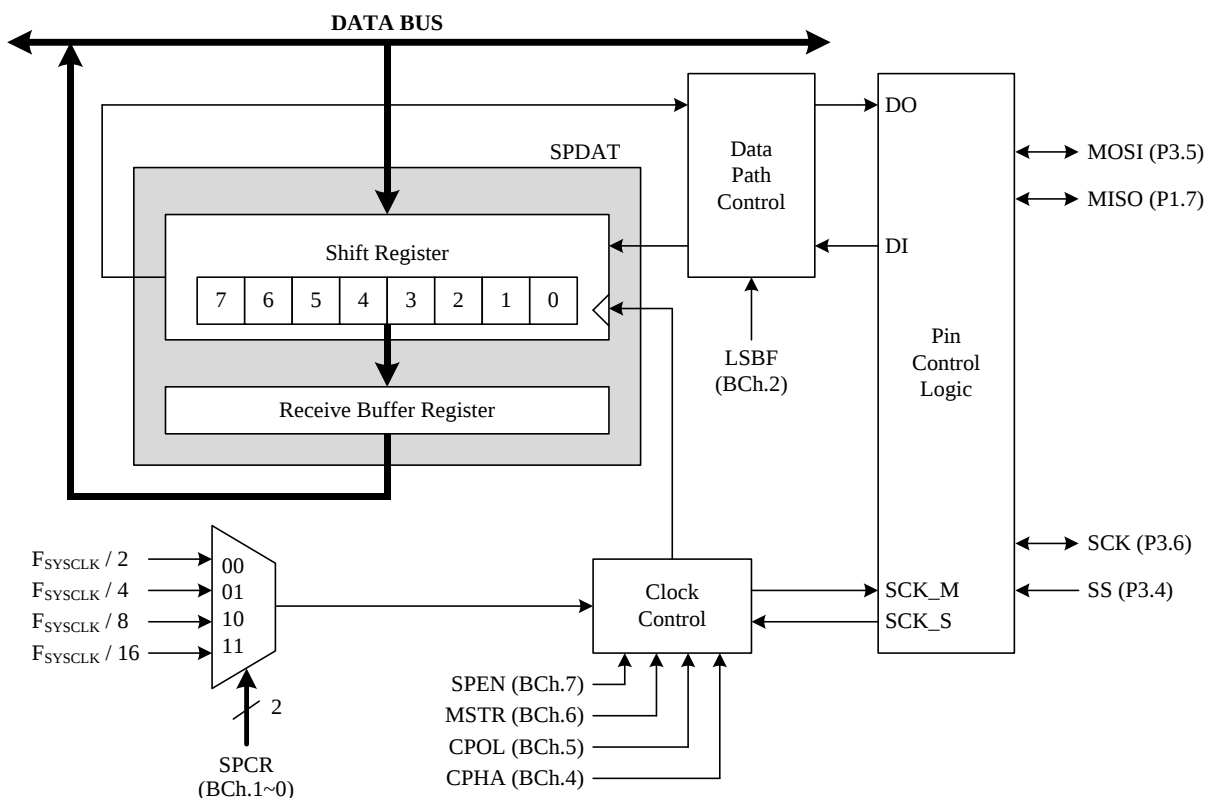
- D8h.3 **STPFCK**: 设置为 1 将停止快速时钟以在慢速/空闲模式下进行节能。该位只能在慢速模式下更改。

15. 串行外围设备接口（SPI）

串行外围界面 (SPI) 模块能够全双工,同步的进行 MCU 和外围设备之间的串行通信。外围设备可以是其它的 MCU,ADC,传感器,或闪存存储器等。SPI 运行在可达系统时钟除以二之时钟速率。韧体可以读取状态标志,或者操作可被中断驱动。下图显示了 SPI 系统框图。

SPI 模块的功能包括:

- 主或从模式操作
- 3 线或 4 线模式操作
- 全双工操作
- 可编程的传输波特率
- 单接收缓冲区
- 串行时钟相位和极性选项
- MSB 优先或 LSB 优先移动可选



SPI Function Pin	PxMODx	Px.n SFR data	Pin State
主模式, MISO	1	1	SPI 数据输入
主模式, SCK, MOSI	2	X	SPI 时钟/数据输出 (CMOS 推挽)
从模式, MISO	2	X	SPI 数据输出 (CMOS 推挽)
从模式, SCK, MOSI	1	1	SPI 时钟/数据输入
SS	1	1	SPI 芯片选择

SPI 端口模式设定

SPI 使用的四个信号说明如下。MOSI 信号是主设备的输出和从设备的输入。该信号是当 SPI 工作在主模式时的输出, SPI 处于从模式下为输入。MISO 信号是从设备的输出和主设备的输入。当 SPI 工作在主模式时的输入, SPI 处于从模式下为输出。数据是通过设置 LSBF 位决定优先传输最高位 (MSB) 或最低位 (LSB)。SCK 信号是主设备的输出和输入到从设备。它是用来同步主设备和从设备的 MOSI 和 MISO 线的数据。SPI 在主模式下提供 8 种可编程的时钟频率的 SCK 信号。SS 信号是低电平有效的从机选择引脚。在 4 线从模式下, 该信号没有被选中 ($SS = 1$) 时会被忽略。当在 SPCON 中的 SSDIS 设置时, 在主或从模式下 SS 都会被忽略。在从模式和 SSDIS 被清零, 当 SS 为低电平, SPI 有效运行。在多重从模式下, 在同一时间只有一个从设备会被选择, 以避免对 MISO 线发生总线冲突。在主模式和 SSDIS 清零, 当 SS 为低电平, 在 SPSTA 中的 MODF 会被设置。对于多重主模式, 使能 SS 线, 以避免 MOSI 和 SCK 线被多个主设备驱动。

主模式

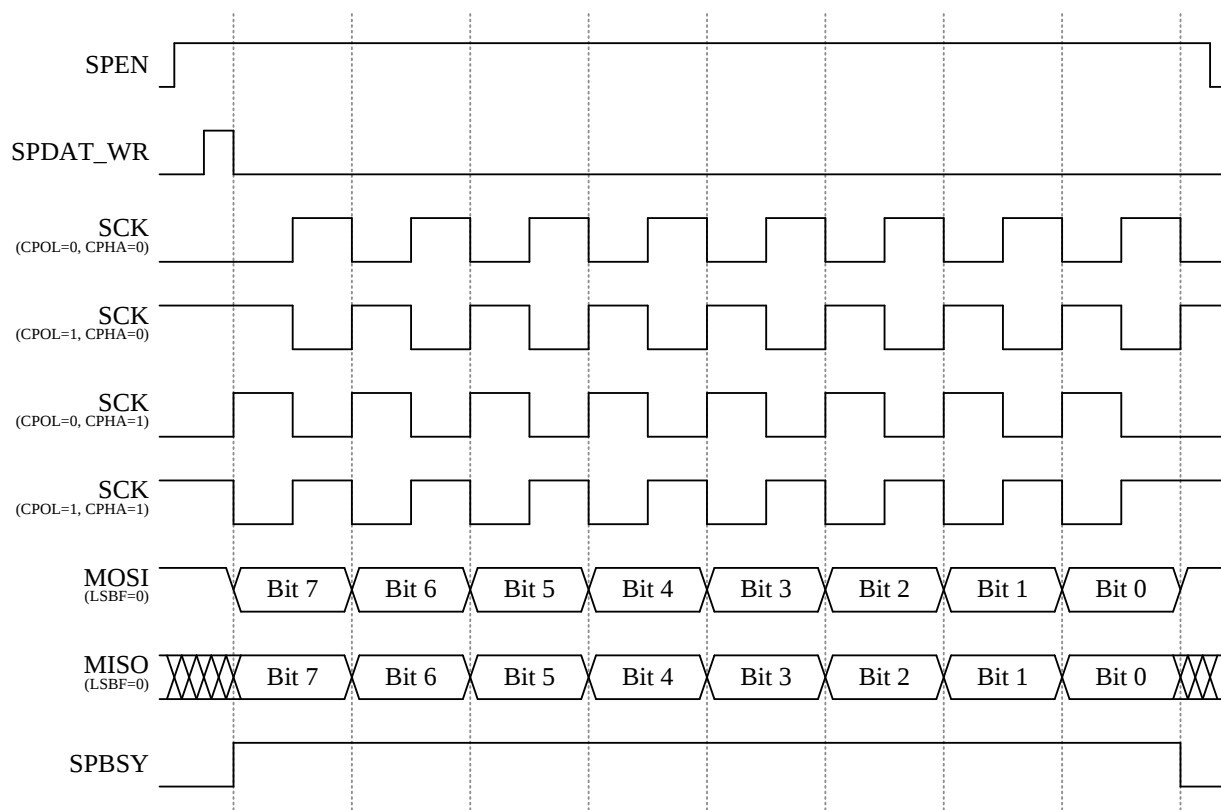
SPI 通过设置 SPCON 中的 MSTR 位以工作在主控模式。要开始传送, 先将数据写到 SPDAT。如果 SPBSY 位被清除, 数据将被转移到移位寄存器, 并开始移出到 MOSI 线上。在同时, 数据从 MISO 线移入从设备。当传输结束时在 SPSTA 中的 SPIF 位变为被设置, 则接收的数据被写入到接收缓冲器, 并将 SPSTA 中的 RCVBF 位设置。为了防止溢出条件, 软件必须在下一个字节进入移位寄存器之前先读取 SPDAT。当数据写入 SPDAT 时 SPBSY 位将会被设置以开始传送, 并在主控模式的第八个 SCK 周期结束时被清零。

从模式

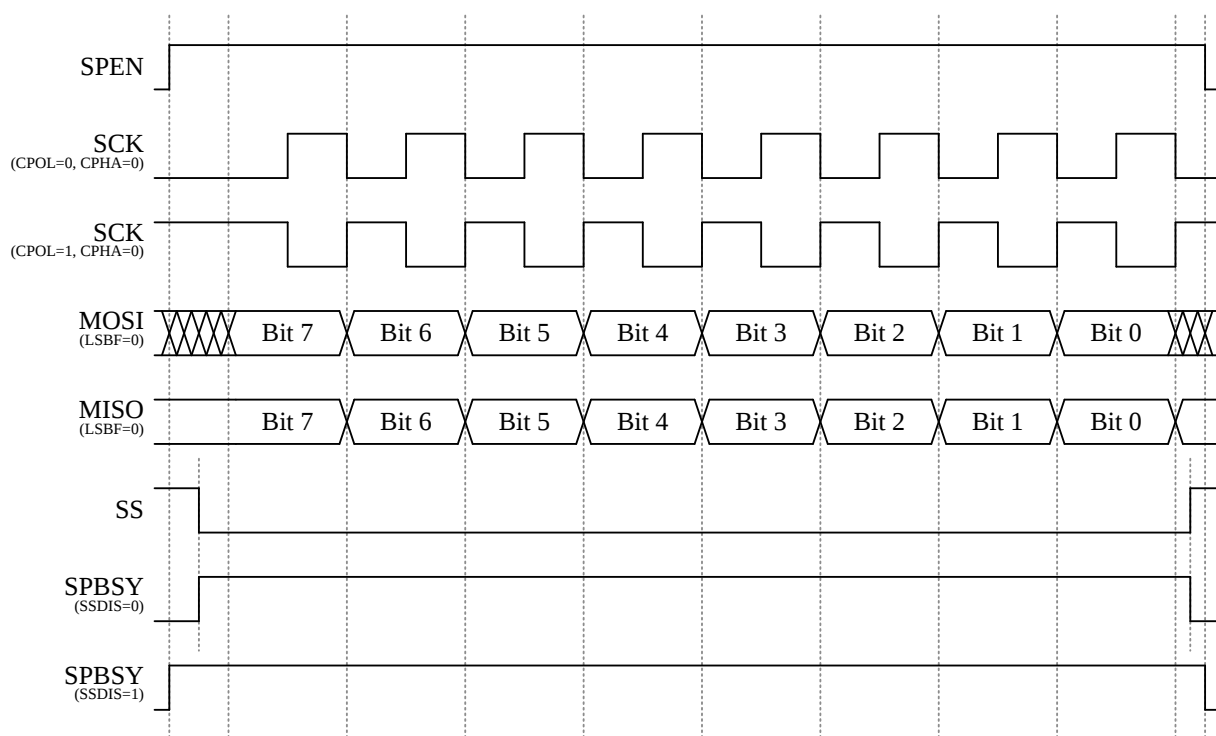
SPI 通过清除 SPCON 中的 MSTR 位成为从模式。如果 SSDIS 清零, 则传输开始当 SS 变低并保持低直到数据传送结束。如果 SSDIS 设置, 在 SPCON 中的 SPEN 位被设置时, 传输将会开始, 并不在意 SS。主设备的数据将通过 MOSI 线转移到移位寄存器, 并从移位寄存器移出到 MISO 线。当一个字节进入移位寄存器时, 如果 RCVBF 被清除, 数据将被传输到接收缓冲器。如果 RCVBF 设置, 较新的接收数据将不会传送到接收缓冲器而且 RCVOVF 位会被设置。当一个字节进入移位寄存器后, SPIF 和 RCVBF 位会被设置。为了防止溢出条件, 软件必须在下一个字节进入移位寄存器之前先读取 SPDAT 或写 0 到 RCVBF。在从模式下允许的最大 SCK 频率为 $F_{SYSCLK} / 4$ 。在从模式下, 当 SSDIS 位清零, SPBSY 位指的是 SS 引脚, SSDIS 位设置时则指向 SPEN 位。

串行时钟

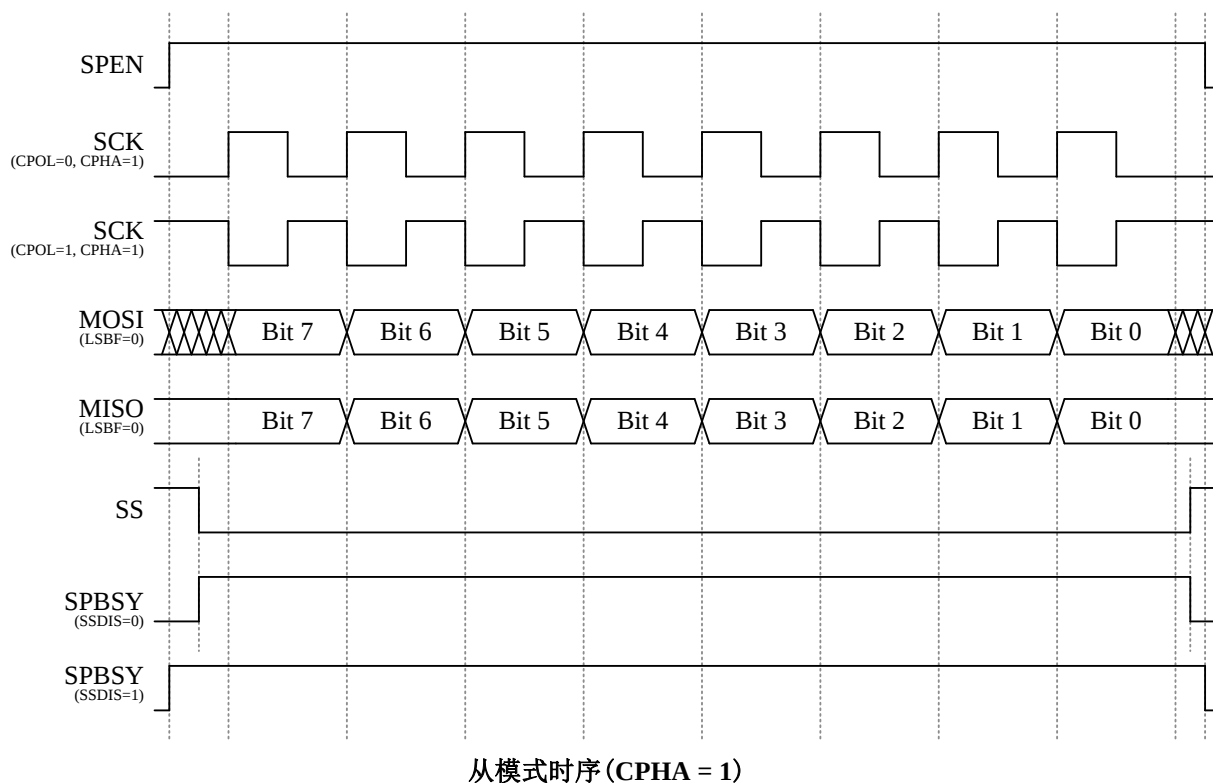
SPI 通过设置 SPCON 寄存器的 CPOL 和 CPHA 位, 产生四个时钟类型。CPOL 位定义在 SPI 空闲状态的 SCK 的水平。在 CPOL 位被清零时空闲状态的 SCK 电平为低的, CPOL 位设置时是高电平。CPHA 位定义用于采样和数据移位的边沿。在 CPHA 位被清除时 SPI 在 SCK 周期的第一个边沿采样数据和在 SCK 周期的第二个边沿移动数据。CPHA 位被设置时在 SCK 期间的第二边沿的 SPI 采样数据和 SCK 周期的第一个边沿移位数据。下图显示了在主从模式的详细时间。SPEN 位被设置前主设备和从设备必须被配置为使用相同的时钟类型。SPCR 控制主模式的串行时钟频率。在从模式时, 此寄存器被忽略。在主模式下 SPI 时钟可以选择为系统时钟除以 2, 4, 8 或 16。



主模式时序



从模式时序 (CPHA = 0)



在主和从模式中, **SPIF** 位是在数据传输结束时由硬件设置,如果 **SPI** 中断使能,并产生一个中断。程序执行中断服务程序时,**SPIF** 位被自动清除。软件也可以写 0 清除该标志。如果在 **SPBSY** 设置时将数据写入 **SPDAT**,**WCOL** 位将被硬件设置,如果 **SPI** 中断使能,并产生一个中断。当这种情况发生时,数据写入到 **SPDAT** 将被忽略,并且移位寄存器将不会被写入。写 0 到该位或当 **SPBSY** 被清除时重写 **SPDAT** 数据将会清除 **WCOL** 标志。在主模式中,当 **SSDIS** 被清零,**SS** 引脚被拉低,**MODE** 位会被设置,如果 **SPI** 中断使能,中断就会产生。当 **MODE** 位被设置,**SPCON** 中的 **SPEN** 和 **MSTR** 将被硬件清零。写 0 到 **MODE** 位将清除该标志。

SFR BCh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPCON	SPEN	MSTR	CPOL	CPHA	SSDIS	LSBF	SPCR	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Reset	0	0	0	0	0	0	0	0

- BCh.7 **SPEN**:SPI 使能
0:SPI 关闭 1:SPI 使能
- BCh.6 **MSTR**:主模式使能
0:从模式 1:主模式
- BCh.5 **CPOL**:SPI 时钟极性
0:SCK 在空闲状态是低电平
1:SCK 在空闲状态是高电平
- BCh.4 **CPHA**:SPI 时钟相位
0:在 SCK 周期的第一个边沿采样数据
1:在 SCK 周期的第二个边沿采样数据
- BCh.3 **SSDIS**:SS 引脚禁用
0:使能 SS 引脚 1:禁用 SS 引脚
- BCh.2 **LSBF**:LSB 优先
0:MSB 优先 1:LSB 优先
- BCh.1~0 **SPCR**:SPI 时钟速率
00:F_{SYSCLK}/2 01:F_{SYSCLK}/4 10:F_{SYSCLK}/8 11:F_{SYSCLK}/16

SFR BDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPSTA	SPIF	WCOL	MODF	RCVOVF	RCVBF	SPBSY	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R	—	—
Reset	0	0	0	0	0	0	—	—

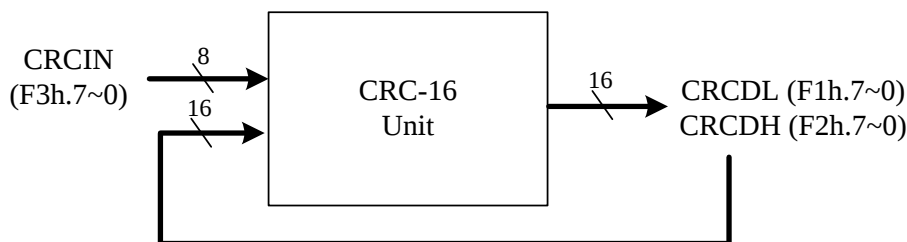
- BDh.7 **SPIF**:SPI 中断标志
这是由硬件在数据传输结束时设定。中断产生时由硬件清零。写 0 到该位将清除该标志。
- BDh.6 **WCOL**:写入冲突中断标志
由硬件设置,如果 SPBSY = 1 时写入数据到 SPDAT。写 0 到该位或 SPBSY = 0 时重写 SPDAT 数据将清除该标志。
- BDh.5 **MODF**:模式故障中断标志
在主模式时,当 SSDIS = 0,SS 引脚被拉低时通过硬件设置。写 0 到该位将清除该标志。当该位被设置,SPCON 中的 SPEN 和 MSTR 将被硬件清零。
- BDh.4 **RCVOVF**:接收缓冲区溢出标志
在数据传输结束和 RCVBF = 1 时将通过硬件设置。写 0 到该位或读取 SPDAT 寄存器将清除该标志。
- BDh.3 **RCVBF**:接收缓冲器满标志
在数据传输结束时将通过硬件设置。写 0 到该位或读取 SPDAT 寄存器将清除该标志。
- BDh.2 **SPBSY**:SPI 忙碌标志(唯读)
当 SPI 传输正在进行时,由硬件设置。

SFR BEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SPDAT	SPDAT							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- BEh.7~0 **SPDAT**:SPI 发送和接收之数据
SPDAT 寄存器被用于发送和接收数据。写数据到 SPDAT 会放置数据到移位寄存器,并开始在主模式下传输。读 SPDAT 则存回接收缓冲器的内容。

16. 循环冗余校验码(CRC)

此芯片支持 16 位的循环冗余校验功能。循环冗余校验（CRC）计算单元是一种错误检测技术测试算法，用于验证数据传输或存储数据的正确性。CRC 计算采用 8 位数据流或数据块作为输入，并产生 16 位输出余数。数据流由同一生成多项式计算。



CRC 框图

CRC 生成器提供了基于 CRC-16-IBM 多项式的 16 位 CRC 结果计算。在这个 CRC 生成器中，只有一个多项式(如下)可用于数值计算，它不支持其他任何多项式的 16 位 CRC 计算。对 CRCIN 寄存器的每次写入操作，将被创建存储在 CRCDH 和 CRCDL 寄存器中的前一个 CRC 值的组合中，这将需要一个 MCU 指令周期来计算。

IBM 的 CRC-16 多项式表示 (Modbus) : $X^{16} + X^{15} + X^2 + 1$

SFR F1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCDL	CRCDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

F1h.7~0 **CRCDL**: 16 位 CRC 校验和数据位 7~0

SFR F2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCDH	CRCDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

F2h.7~0 **CRCDL**: 16 位 CRC 校验和数据位 15~8

SFR F3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCIN	CRCIN							
W	W							
Reset	—	—	—	—	—	—	—	—

F3h.7~0 **CRCIN**: CRC 输入数据寄存器

17. 乘法器和除法器

该芯片提供的乘法器和除法器具有以下功能。 8 位操作与标准 8051 完全兼容。

- 8 位×8 位= 16 位（标准 8051）
- 8 位÷8 位= 8 位，余数 8 位（标准 8051）
- 16 位×16 位= 32 位
- 16 位÷16 位= 16 位，余数 16 位
- 32 位÷16 位= 32 位，余数 16 位

无论是 8 位/ 16 位/ 32 位操作，都可以通过 MUL AB 和 DIV AB 指令轻松执行。对于 16 位/ 32 位乘法和除法运算，还有额外的 SFR EXA / EXA2 / EXA3 / EXB。对于 8 位乘法器/除法器操作，请确保 SFR 位 MULDIV16 = 0 和 DIV2 = 0。

对于 16 位乘法器操作被乘数，乘数和乘积如下。 16 位乘法器需要 16 个系统时钟周期才能执行。

条件	SFR bit MULDIV16=1 and DIV32=0			
乘法	字节3	字节2	字节1	字节0
被乘数	-	-	EXA	A
乘数	-	-	EXB	B
乘积	EXB	B	A	EXA
OV	乘积 (EXB or B) !=0			

对于 16 位除法器操作被除数，除数，商，余数如下。 16 位除法器需要 16 个系统时钟周期来执行。

条件	SFR bit MULDIV16=1 and DIV32=0			
除法	字节3	字节2	字节1	字节0
被除数	-	-	EXA	A
除数	-	-	EXB	B
商	-	-	A	EXA
余数	-	-	B	EXB
OV	除数 EXB = B =0			

对于 32 位÷16 位运算被除数，除数，商，余数的读取方式如下。 32 位除法器需要 32 个系统时钟周期来执行。

条件	SFR bit MULDIV16=1 and DIV32=1			
除法	字节3	字节2	字节1	字节0
被除数	EXA3	EXA2	EXA	A
除数	-	-	EXB	B
商	A	EXA	EXA2	EXA3
余数	-	-	B	EXB
OV	除数 EXB=B=0			

SFR CEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXA2	EXA2							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

CEh.7~0 **EXA2:** 扩充累加器 2

SFR CFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXA3	EXA3							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

CFh.7~0 **EXA3:** 扩充累加器 3

SFR E6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXA	EXA							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

E6h.7~0 **EXA:** 扩充累加器

SFR E7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EXB	EXB							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

E7h.7~0 **EXB:** 扩充 B 寄存器

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX2	WDTE		PWRSAB	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

F7h.3 **DIV32:**

仅在 MULDIV16 = 1 时有效

0: 指令 DIV 作为 16/16 位除法运算

1: 指令 DIV 作为 32/16 位除法运算

F7h.0 **MULDIV16:**

0: 指令 MUL / DIV 为 8 * 8, 8/8 操作

1: 指令 MUL / DIV 为 16 * 16、16 / 16 或 32/16 操作

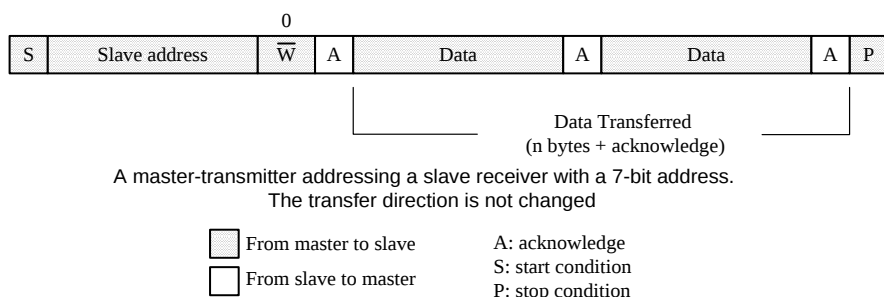
ARITHMETIC				
Mnemonic	Description	byte	cycle	opcode
MUL AB	Multiply A by B	1	8/16	A4
DIV AB	Divide A by B	1	8/16/32	84

18. 主 I²C 接口

主 I²C 接口传输模式:

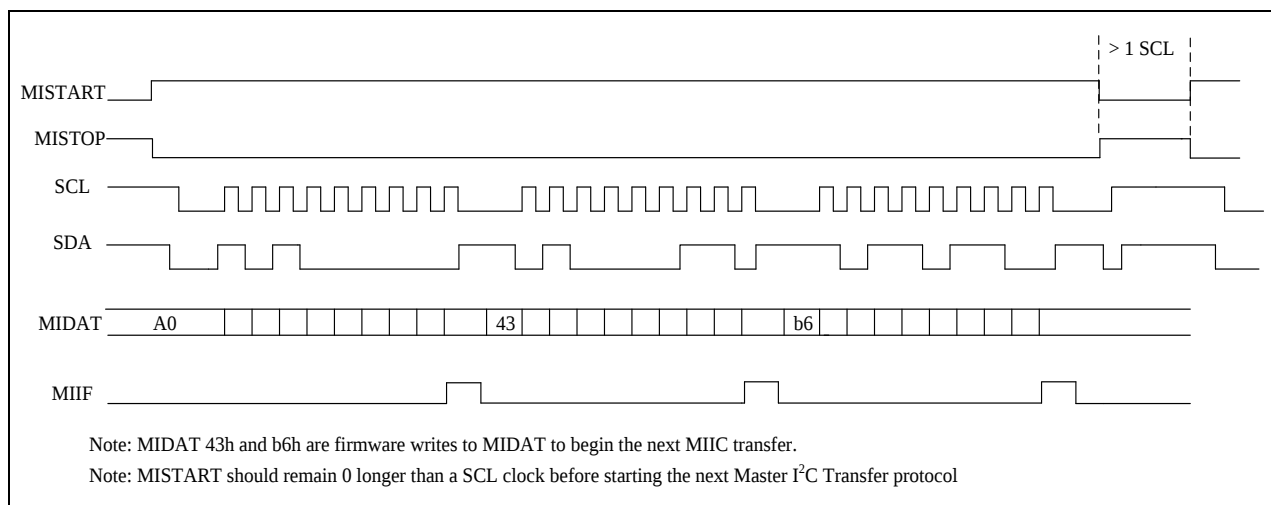
一开始先时将从机地址和方向位写入 MIDAT 并设置 MISTART。在 MISTART 设置之后, 将发送 7 位从地址和一位方向位到从机。等待直到 MIIF 转换为 1 时, 代表地址和方向位传输完成, 用户应清除 MIIF 并写入数据到 MIDAT 以开始第一次数据传输。当 MIIF 转换为 1 时, 代表数据传输到从机完成。用户可以再次写入数据到 MIDAT 以将开始下一次数据传输到从机。设置 MISTOP 以完成传送模式。

在数据传输时, MISTART 必须保持为 1。并在最终数据发送/接收之后, 设置 MISTOP 以完成发送/接收协议。在重新启动主 I²C 传送接收协议之前, MISTART 应保持为 0, 且等待时间大于 SCL 时钟后才可进行下一次传送接收。SCL 时钟可通过 MICR 进行调整。



主 I²C 传送流程:

- (1) 将从机地址和方向位写入 MIDAT
- (2) 清除 MISTOP 并设置 MISTART 以启动 I²C 传输
- (3) 等到 MIIF 转换为 1 时(根据用户要求发出中断), 清除 MIIF
- (4) 将数据写入 MIDAT 以开始下一次传输(MISTART 必须保持为 1)
- (5) 等到 MIIF 转换为 1 时(根据用户要求发出中断), 清除 MIIF, 循环(4)~(5)进行下一次传输
- (6) 清除 MISTART, 设置 MISTOP 以停止 I²C 传输



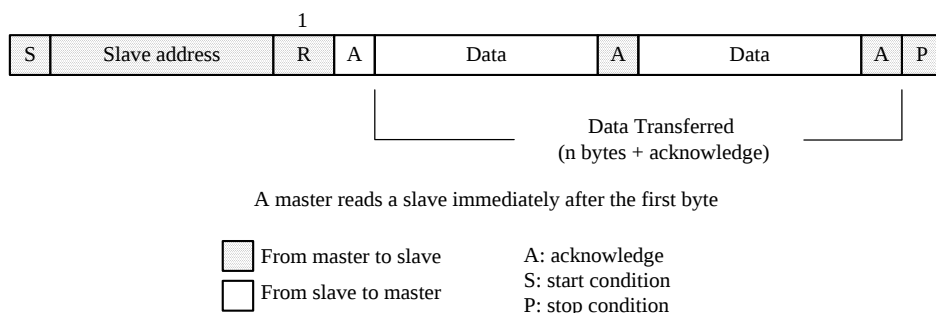
主发送时序

注: 在开始下一个主 I²C 协议之前, MISTART 应该保持 0 比 SCL 周期更长。

I²C 主接口接收模式:

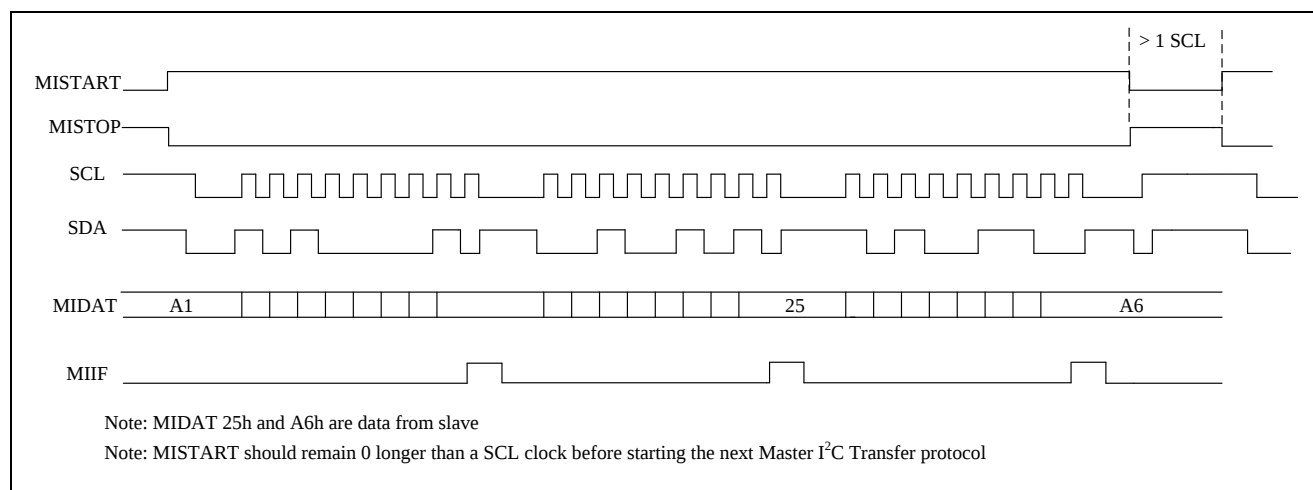
一开始先将从机地址和方向位写入 MIDAT 并设置 MISTART。在 MISTART 设置之后, 将发送 7 位从机地址和一位方向位到从机。当 MIIF 转换为 1 时, 代表地址和方向位传输完成。用户应清除 MIIF 并读取 MIDAT 以开始第一次接收数据 (此时尚未完成接收数据, 应丢弃读入的 MIDAT)。当 MIIF 转换为 1 时, 代表对从机接收的数据已完成。用户可以读取 MIDAT 以得到接收数据, 硬件会同时开始下一次接收。设置 MISTOP 以完成接收模式。

在数据传输时, MISTART 必须保持为 1。并在最终数据发送/接收之后, 设置 MISTOP 以完成发送/接收协议。在重新启动主 I²C 传送接收协议之前, MISTART 应保持为 0, 且等待时间大于 SCL 时钟后才可进行下一次传送接收。SCL 时钟可通过 MICR 进行调整。



主 I²C 接收流程:

- (1) 将从机地址和方向位写入 MIDAT
- (2) 清除 MISTOP 并设置 MISTART 以启动 I²C 传输
- (3) 等到 MIIF 转换为 1 时 (根据用户要求发出中断), 清除 MIIF
- (4) 读取 MIDAT 以开始第一次接收数据 (此时尚未完成接收数据, 应丢弃读入的 MIDAT)
- (5) 等到 MIIF 转换为 1 时 (根据用户要求发出中断), 清除 MIIF
- (6) 读取 MIDAT 以得到接收数据, 循环 (5)~(6) 进行下一次接收
- (7) 设置 MISTOP 以停止 I²C 传输



主接收时序

I ² C Function Pin	P3modx	P3.n SFR data	Pin State
I ² C 主 SCL	0	X	时钟输出（开漏输出）
	2	X	时钟输出（CMOS 推挽）
I ² C 主/从 SDA	0	1	数据（上拉）

主 I²C 的引脚模式设置

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PINMOD	—	I2CSEL	TCOE	T2OE	HSNK2EN	HSNK1EN	HSNK0EN	T0OE
R/W	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	0	0	0	0	0	0	0

A6h.6 **I2CSEL**: I²C 引脚选择
0: SCL/SDA = P3.4/P3.5
1: SCL/SDA = P3.0/P3.1

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE1	PWMIE	I2CE	ES2	SPIE	ADTKIE	EXLVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

A9h.6 **I2CE**: I²C 中断使能
0: 禁用 I²C 中断
1: 使能 I²C 中断

SFR E1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MICON	MIEN	MIACKO	MIIF	MIACKI	MISTART	MISTOP	MICR	
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	1	0	0

E1h.7 **MIEN**: 主 I²C 使能
0: 禁用
1: 使能

E1h.6 **MIACKO**: 当主 I²C 接收数据时，向 I²C 总线发送 ACK
0: ACK 到从机
1: NACK 到从机

E1h.5 **MIIF**: 主 I²C 中断标志
当主 I²C 发送或接收一个字节完成时，由硬件设置。向该位写入"0"将清除该标志

E1h.4 **MIACKI**: 当主 I²C 传输时，ACK 来自 I²C 总线（只读）
0: 收到 ACK
1: 收到 NACK

E1h.3 **MISTART**: 主 I²C 启动位
1: 启动 I²C 总线传输

E1h.2 **MISTOP**: 主 I²C 停止位
1: 发送停止信号以停止 I²C 总线

E1h.1~0 **MICR**: 主 I²C 时钟频率选择
00: F_{sys}/4（例如，如果 F_{sys}=16MHz，I²C 时钟为 4 MHz）
01: F_{sys}/16（例如，如果 F_{sys}=16MHz，I²C 时钟为 1 MHz）
10: F_{sys}/64（例如，如果 F_{sys}=16MHz，I²C 时钟为 250 KHz）
11: F_{sys}/256（例如，如果 F_{sys}=16MHz，I²C 时钟为 62.5 KHz）

SFR E2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MIDAT	MIDAT							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

E2h.7~0 **MIDAT**: 主 I²C 数据移位寄存器

(W): 在开始条件之后和停止条件之前, 写入该寄存器将恢复向 I²C 总线的传输

(R): 在开始条件之后和停止条件之前, 读取该寄存器将恢复从 I²C 总线的接收

SFR EAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SICON	MIIE	TXDIE	RCD2IE	RCD1IE	—	TXDF	RCD2F	RCD1F
R/W	R/W	R/W	R/W	R/W	—	R/W	R/W	R/W
Reset	0	0	0	0	—	1	0	0

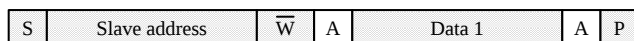
EAh.7 **MIIE**: I²C 主中断使能

0: 禁用

1: 使能

19. 从机 I²C 接口

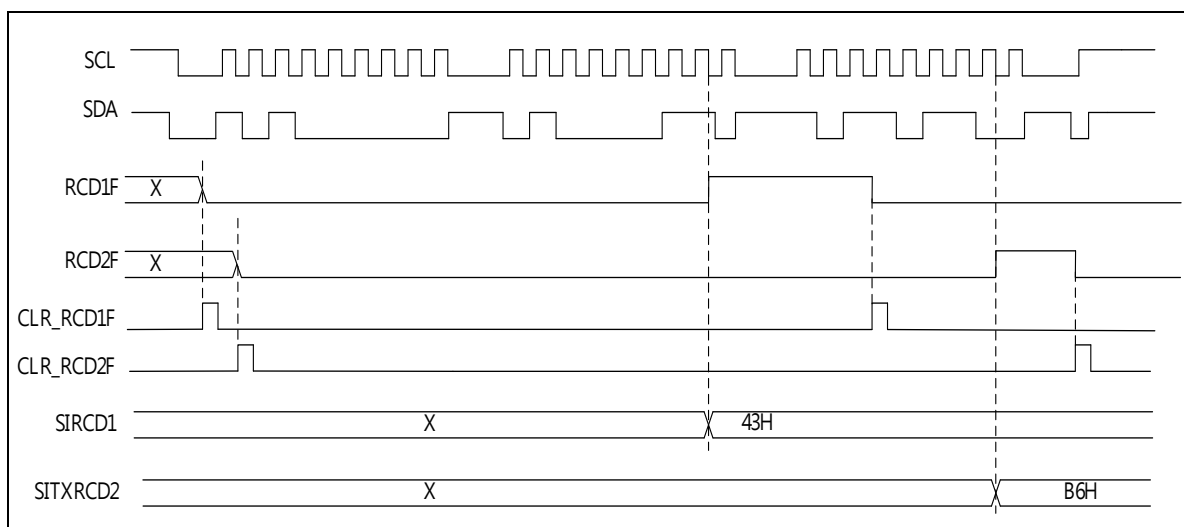
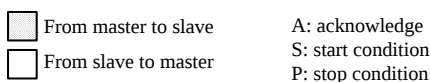
该芯片提供如下的从机 I²C 接口接收协议。从机 I²C 模块允许在启动条件之后每次接收一个或两个字节的数。在接收 DATA1 之前，请注意 RCD1F 必须为 0。在 DATA1 接收完成之后，RCD1F 将被转换为 1，并且将根据用户的请求发出中断。用户可以使用固件清除 RCD1F，然后再次接收下一个 DATA1。用户可以将 RCD1F 写入 0 来清除 RCD1F。DATA2 和 RCD2F 的运行方式与 DATA1 和 RCD1 相同。DATA1 或 DATA2 接收完成后，主控端应重新启动传输协议以传输下一个 DATA1 和 DATA2。



Slave I²C Receive Byte protocol

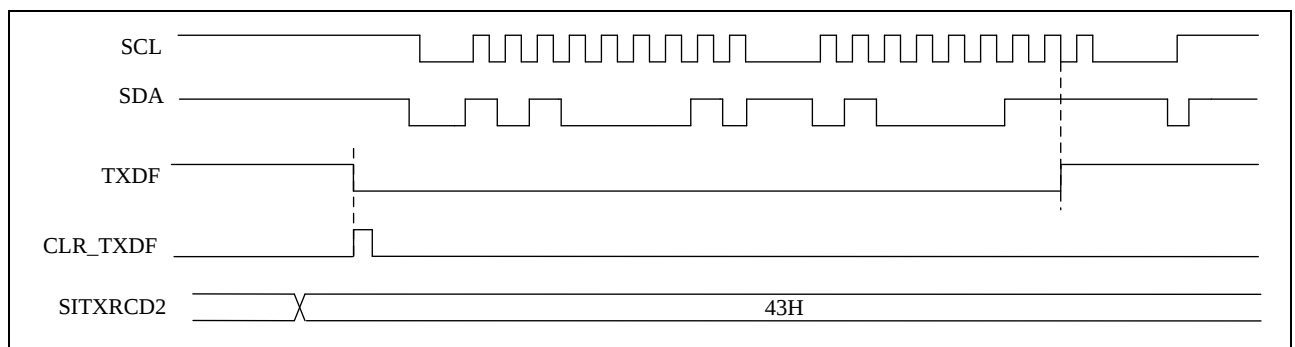
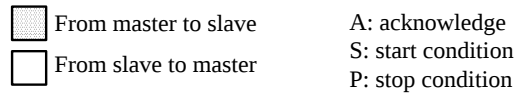


Slave I²C Receive Two Byte protocol



从机接收时序

该芯片提供以下从机设备 I²C 接口传输协议。从 I²C 模块允许在启动条件之后每次发送一个字节数据。在发送数据之前，请注意 TXDF 必须为 0。在数据发送完成之后，TXDF 将转换为 1，并根据用户的请求发出中断。用户可以使用固件清除 TXDF，然后再次传输下一个数据。用户可以将 TXDF 写入 0 来清除 TXDF。每次传送完成后，主控端应重新启动传输协议以传输下一笔数据。


Slave I²C Transmit protocol


从机发送时序

I ² C 功能引脚	P3MODx	P3.n SFR data	引脚状态
I ² C 从 SCL	1	1	时钟输入(高阻抗)
I ² C 主/从 SDA	0	1	数据(上拉)

从 I²C 的引脚模式设置

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE1	PWMIE	I2CE	ES2	SPIE	ADTKIE	EXLVDIE	PCIE	TM3IE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

A9h.6 I2CE: I²C 中断使能

0: 禁用 I²C 中断

1: 使能 I²C 中断

SFR E9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SIADR	SA							SIEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	1	1	0	0	1	0	0

E9h.7~1 SA: 从机分配的 I²C 地址

E9h.0 SIEN: 从机 I²C 使能

0: 禁用

1: 使能

SFR EAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SICON	MIIE	TXDIE	RCD2IE	RCD1IE	—	TXDF	RCD2F	RCD1F
R/W	R/W	R/W	R/W	R/W	—	R/W	R/W	R/W
Reset	0	0	0	0	—	1	0	0

- EAh.6 **TXDIE**: 从机 I²C 发送完成中断使能
0: 禁用
1: 使能
- EAh.5 **RCD2IE**: 从机 I²C DATA2 (SITXRCD2) 接收完成中断使能
0: 禁用
1: 使能
- EAh.4 **RCD1IE**: 从机 I²C DATA1 (SIRCD1) 接收完成中断使能
0: 禁用
1: 使能
- EAh.2 **TXDF**: 从机 I²C 传输完成中断标志
从机 I²C 传输完成时由硬件设置, 写 0 清除
- EAh.1 **RCD2F**: 从机 I²C DATA2 (SITXRCD2) 接收完成中断标志
从机 I²C DATA2 (SITXRCD2) 接收完成后由硬件置位, 写 0 清除
- EAh.0 **RCD1F**: 从机 I²C DATA1 (SIRCD1) 接收完成中断标志
从机 I²C DATA1 (SITXRCD1) 接收完成后由硬件置位, 写 0 清除

SFR EBh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SIRCD1	SIRCD1							
R/W	R	R	R	R	R	R	R	R
Reset	—	—	—	—	—	—	—	—

- EBh.7~0 **SIRCD1**: 从机 I²C 数据接收寄存器 1 (DATA1)

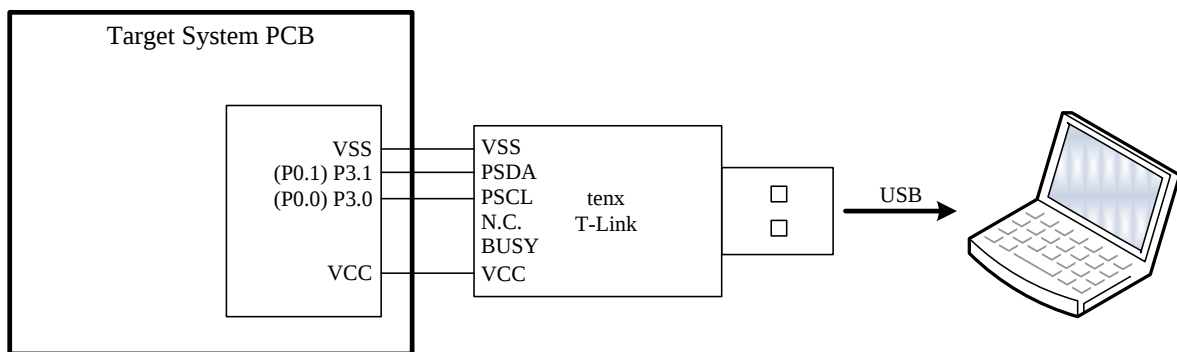
SFR ECh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SITXRCD2	SITXRCD2							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	—	—	—	—	—	—	—	—

- ECh.7~0 **SITXRCD2**: 从机 I²C 发送和接收数据寄存器
(读): 从机 I²C 数据接收寄存器 2 (DATA2)
(写): 从机 I²C 数据传输寄存器 (TXD)

20. 在线仿真器 (ICE) 模式

该设备可以支持在线仿真模式。要使用 ICE 模式，用户只需将 P3.0 和 P3.1 引脚连接到 tenx 专有 EV 模块。好处是用户可以仿真整个系统，而无需更改板载目标设备。但是 ICE 模式有一些限制，如下所示。

1. 设备必须处于未保护状态。
2. 设备的 P3.0/P3.1 引脚必须在输入模式下工作 (P3MOD0 = 0/1 and P3MOD1=0/1)。
3. Tenx EV 模块占用了程序存储器的寻址空间 6C00h~6FFFh 和 0033h~003Ah。因此，用户程序无法访问这些空间。
4. 无法模拟 T-Link 通信引脚的功能。
5. P3.0 和 P3.1 引脚可以替换为 P0.0 和 P0.1 (仅仿真时可以替换)。
6. V_{DD} 电平由 T-Link 模块控制。



32K Bytes program memory	
0000h	Reset / Interrupt Vector
006Fh	
0070h	User Code area
6BFFh	
6C00h	ICE mode reserve area
6FFFh	
7A00h	User Code area
7DFFh	User Code or IAP area (EEPROM like)
	User Code area
7FF0h	CRC16L
7FF1h	CRC16H
7FF2h	
7FFEh	tenx reserve area
7FFFh	CFGWH

TM52eF1375A

SFR & CFGW 映像

Adr	RST	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
80h	0000-0000	P0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
81h	0000-0111	SP	SP							
82h	0000-0000	DPL	DPL							
83h	0000-0000	DPH	DPH							
84h	0000-0000	INTEX	EX9	EX8	EX7	EX6	EX5	EX4	EX3	EX2
85h	0000-0000	INTEXF	IE9	IE8	IE7	IE6	IE5	IE4	IE3	IE2
86h	xxxx-x000	INTPWM	—	—	—	—	—	PWM2IF	PWM1IF	PWM0IF
87h	0xxx-0000	PCON	SMOD	—	—	—	GF1	GF0	PD	IDL
88h	0000-0000	TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
89h	0000-0000	TMOD	GATE1	CT1N	TMOD1		GATE0	CT0N	TMOD0	
8Ah	0000-0000	TL0	TL0							
8Bh	0000-0000	TL1	TL1							
8Ch	0000-0000	TH0	TH0							
8Dh	0000-0000	TH1	TH1							
8Eh	0100-0000	SCON2	SM	—	—	REN2	TB82	RB82	TI2	RI2
8Fh	xxxx-xxxx	SBUF2	SBUF2							
90h	1111-1111	P1	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
91h	0000-0000	P0OE	P0OE							
92h	0000-0000	P0LOE	P0LOE							
93h	0000-0101	P2MOD	—	—	—	—	P2MOD1		P2MOD0	
94h	0000-0000	OPTION	UART1W	—	WDTPSC		ADCKS		TM3PSC	
95h	xx00-x000	INTFLG	LVDIF	—	TKIF	ADIF	—	—	P1IF	TF3
96h	0000-0000	P1WKUP	P1WKUP							
97h	xxxx-xx00	SWCMD	IAPEN / SWRST / WDTO							
98h	0000-0000	SCON	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
99h	xxxx-xxxx	SBUF	SBUF							
9Eh	0000-0000	PWMOE	PWM1IE	PWM0IE	—	—	—	PWM2OE	PWM1OE	PWM0OE
9Fh	0x00-0000	PWMCLR	PWM2IE	—	—	—	—	PWM2CLR	PWM1CLR	PWM0CLR
A0h	0000-0011	P2	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
A1h	xx10-1010	PWMCON	—	—	PWM2CKS		PWM1CKS		PWM0CKS	
A2h	0101-0101	P1MODL	P1MOD3		P1MOD2		P1MOD1		P1MOD0	
A3h	0101-0101	P1MODH	P1MOD7		P1MOD6		P1MOD5		P1MOD4	
A4h	0101-0101	P3MODL	P3MOD3		P3MOD2		P3MOD1		P3MOD0	
A5h	0001-0101	P3MODH	P3MOD7		P3MOD6		P3MOD5		P3MOD4	
A6h	0000-0000	PINMOD	—	I2CSEL	TCOE	T2OE	HSNK2EN	HSNK1EN	HSNK0EN	T0OE
A7h	xxx1-1111	TKCHS	—			TKCHS				
A8h	0x00-0000	IE	EA	—	ET2	ES	ET1	EX1	ET0	EX0
A9h	xx00-0000	INTE1	PWMIE	I2CE	ES2	SPIE	ADTKIE	EXLVDIE	P1IE	TM3IE
AAh	xxxx-xxxx	ADCDL	ADCDL				—			
ABh	xxxx-xxxx	ADCDH	ADCDH							
ADh	1100-0100	TKCON	TKPD	TKEOC	TKRERUN	TKIVCS	TKXCAP	TKOFFSET	ATKMODE	
A Eh	1111-000x	CHSEL	ADCHS				ADCVREFS		VBGEN	
AFh	000x-xxxx	P0ADIE	P0ADIE			—	—	—	—	—
B0h	1111-1111	P3	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
B1h	0000-x100	LEDCON	LEDEN		LEDPSC		LEDHOLD	LEDBRIT		
B2h	0100-x100	LEDCON2	LEDBRITM	LEDBRIT2			—	LEDBRIT1		
B3h	0000-0000	LEDCON3	LEDMTEN	LED8EN	LED7EN	LED6EN	LED5EN	LED4EN	LED3EN	LED2EN
B4h	0000-0000	TKTMRL	TKTMRL							
B5h	0000-0000	TKCON2	TKFJMP	JMPVAL		SPREAD	TKTMRH			
B8h	xx00-0000	IP	—	—	PT2	PS	PT1	PX1	PT0	PX0
B9h	xx00-0000	IPH	—	—	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
BAh	xx00-0000	IP1	—	—	PS2	PSPI	PADTKI	PX2_9LVD	PP1	PT3
BBh	xx00-0000	IP1H	—	—	PS2H	PSPIH	PADTKIH	PX2_9LVDH	PP1H	PT3H

Adr	RST	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BCh	0000-0000	SPCON	SPEN	MSTR	CPOL	CPHA	SSDIS	LSBF	SPCR	
BDh	0000-0xxx	SPSTA	SPIF	WCOL	MODF	RCVOVF	RCVBF	SPBSY	–	–
BEh	0000-0000	SPDAT	SPDAT							
BFh	0xxx-0000	LVDS	LVDIE	LVDO	–	–	LVDS			
C1h	0000-0000	TKPINSEL0	TKPINSEL0							
C2h	0000-0000	TKPINSEL1	TKPINSEL1							
C3h	0000-0000	TKPINSEL2	TKPINSEL2							
C5h	0000-0000	ATKCH0	ATKCH0							
C6h	0000-0000	ATKCH1	ATKCH1							
C7h	0000-0000	ATKCH2	ATKCH2							
C8h	0000-0000	T2CON	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2N	CPRL2N
C9h	00xx-xxxx	IAPWE	IAPWE / IAPTO							
CAh	0000-0000	RCP2L	RCP2L							
CBh	0000-0000	RCP2H	RCP2H							
CCh	0000-0000	TL2	TL2							
CDh	0000-0000	TH2	TH2							
CEh	0000-0000	EXA2	EXA2							
CFh	0000-0000	EXA3	EXA3							
D0h	0000-0000	PSW	CY	AC	F0	RS1	RS0	OV	F1	P
D1h	1000-0000	PWM0DH	PWM0DH							
D2h	0000-0000	PWM0DL	PWM0DL							
D3h	1000-0000	PWM1DH	PWM1DH							
D4h	0000-0000	PWM1DL	PWM1DL							
D5h	1000-0000	PWM2DH	PWM2DH							
D6h	0000-0000	PWM2DL	PWM2DL							
D8h	00x0-0011	CLKCON	SCKTYPE	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
D9h	1111-1111	PWM0PRDH	PWM0PRDH							
DAh	1111-1111	PWM0PRDL	PWM0PRDL							
DBh	1111-1111	PWM1PRDH	PWM1PRDH							
DCh	1111-1111	PWM1PRDL	PWM1PRDL							
DDh	1111-1111	PWM2PRDH	PWM2PRDH							
DEh	1111-1111	PWM2PRDL	PWM2PRDL							
E0h	0000-0000	ACC	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
E1h	000x-0100	MICON	MIEN	MIACKO	MIIF	MIACKI	MISTART	MISTOP	MICR	
E2h	0000-0000	MIDAT	MIDAT							
E5h	0000-0000	EFTCON	EFT2CS	EFT1CS	EFT1S		EFTSLOW	EFTWCPU	EFTWOUT	CKHLDE
E6h	0000-0000	EXA	EXA							
E7h	0000-0000	EXB	EXB							
E9h	0110-1000	SIADR	SA							SIEN
EAh	0000-x100	SICON	MIIE	TXDIE	RCD2IE	RCD1IE	–	TXDF	RCD2F	RCD1F
EBh	xxxx-xxxx	SIRCD1	SIRCD1							
ECh	xxxx-xxxx	SITXRCD2	SITXRCD2							
EFh	0000-0000	PWRCON	–	–	–	AVPULL	WARMTIME	ENVPULL	PWRIDLE	PWSLOW
F0h	0000-0000	B	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
F1h	1111-1111	CRCDL	CRCDL							
F2h	1111-1111	CRCDH	CRCDH							
F3h	0000-0000	CRCIN	CRCIN							
F5h	xxxx-xxxx	CFGBG	–	–	–	BGTRIM				
F6h	xxxx-xxxx	CFGWL	–	FRCF						
F7h	0000-1110	AUX2	WDTE		PWRSV	VBGOUT	DIV32	IAPTE		MULDIV16
F8h	0000-0000	AUX1	CLRWDT	CLRTM3	TKSOC	ADSOC	LVRPD	T2SEL	T1SEL	DPSEL

Flash Address	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
7FFh	CFGWH	PROTN	XRSTEN	LVRE			–	MVCLOCKN	FRCPSC

SFR & CFGW 说明

下表列出所有的 SFR,特地保留原始英文,以供使用者交互参考。

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
80h	P0	7~0	P0	R/W	00h	Port0 data
81h	SP	7~0	SP	R/W	07h	Stack Point
82h	DPL	7~0	DPL	R/W	00h	Data Point low byte
83h	DPH	7~0	DPH	R/W	00h	Data Point high byte
84h	INTEX	7	EX9	R/W	0h	External INT9~INT2 pin Interrupt enable and Halt/Stop mode wake up enable. 0: Disable INTx pin Interrupt and Halt/Stop mode wake up 1: Enable INTx pin Interrupt and Halt/Stop mode wake up, it can wake up CPU from Halt/Stop mode no matter EA is 0 or 1 (note: EXLVDIE must be 1 at the same time to generate INTx interrupt and wake up)
		6	EX8	R/W	0h	
		5	EX7	R/W	0h	
		4	EX6	R/W	0h	
		3	EX5	R/W	0h	
		2	EX4	R/W	0h	
		1	EX3	R/W	0h	
		0	EX2	R/W	0h	
85h	INTEXF	7	IE9	R/W	0h	INT9 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		6	IE8	R/W	0h	INT8 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		5	IE7	R/W	0h	INT7 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		4	IE6	R/W	0h	INT6 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		3	IE5	R/W	0h	INT5 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		2	IE4	R/W	0h	INT4 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		1	IE3	R/W	0h	INT3 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		0	IE2	R/W	0h	INT2 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
86h	INTPWM	2	PWM2IF	R/W	0h	PWM2 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		1	PWM1IF	R/W	0h	PWM1 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		0	PWM0IF	R/W	0h	PWM0 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
87h	PCON	7	SMOD	R/W	0	Set 1 to enable UART1 double baud rate
		3	GF1	R/W	0	General purpose flag bit
		2	GF0	R/W	0	General purpose flag bit
		1	PD	R/W	0	Power down control bit, set 1 to enter Halt/Stop mode
		0	IDL	R/W	0	Idle control bit, set 1 to enter Idle mode
88h	TCON	7	TF1	R/W	0	Timer1 overflow flag Set by H/W when Timer/Counter 1 overflows. Cleared by H/W when CPU vectors into the interrupt service routine.
		6	TR1	R/W	0	Timer1 run control. 1: timer runs; 0: timer stops
		5	TF0	R/W	0	Timer0 overflow flag Set by H/W when Timer/Counter 0 overflows. Cleared by H/W when CPU vectors into the interrupt service routine.
		4	TR0	R/W	0	Timer0 run control. 1:timer runs; 0:timer stops
		3	IE1	R/W	0	External Interrupt 1 (INT1 pin) edge flag Set by H/W when an INT1 pin falling edge is detected. Cleared by H/W when CPU vectors into the interrupt service routine.
		2	IT1	R/W	0	External Interrupt 1 control bit 0: Low level active (level triggered) for INT1 pin 1: Falling edge active (edge triggered) for INT1 pin
		1	IE0	R/W	0	External Interrupt 0 (INT0 pin) edge flag Set by H/W when an INT0 pin falling edge is detected. Cleared by H/W when CPU vectors into the interrupt service routine.
		0	IT0	R/W	0	External Interrupt 0 control bit 0: Low level active (level triggered) for INT0 pin 1: Falling edge active (edge triggered) for INT0 pin

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
89h	TMOD	7	GATE1	R/W	0	Timer1 gating control bit 0: Timer1 enable when TR1 bit is set 1: Timer1 enable only while the INT1 pin is high and TR1 bit is set
		6	CT1N	R/W	0	Timer1 Counter/Timer select bit 0: Timer mode, Timer1 data increases at 2 System clock cycle rate 1: Counter mode, Timer1 data increases at T1 pin's negative edge
		5~4	TMOD1	R/W	00	Timer1 mode select 00: 8-bit timer/counter (TH1) and 5-bit prescaler (TL1) 01: 16-bit timer/counter 10: 8-bit auto-reload timer/counter (TL1). Reloaded from TH1 at overflow. 11: Timer1 stops
		3	GATE0	R/W	0	Timer0 gating control bit 0: Timer0 enable when TR0 bit is set 1: Timer0 enable only while the INT0 pin is high and TR0 bit is set
		2	CT0N	R/W	0	Timer0 Counter/Timer select bit 0: Timer mode, Timer0 data increases at 2 System clock cycle rate 1: Counter mode, Timer0 data increases at T0 pin's negative edge
		1~0	TMOD0	R/W	00	Timer0 mode select 00: 8-bit timer/counter (TH0) and 5-bit prescaler (TL0) 01: 16-bit timer/counter 10: 8-bit auto-reload timer/counter (TL0). Reloaded from TH0 at overflow. 11: TL0 is an 8-bit timer/counter. TH0 is an 8-bit timer/counter using Timer1's TR1 and TF1 bits.
8Ah	TL0	7~0	TL0	R/W	00h	Timer0 data low byte
8Bh	TL1	7~0	TL1	R/W	00h	Timer1 data low byte
8Ch	TH0	7~0	TH0	R/W	00h	Timer0 data high byte
8Dh	TH1	7~0	TH1	R/W	00h	Timer1 data high byte
8Eh	SCON2	7	SM	R/W	0	UART2 Serial port mode select bit 0: Mode1: 8 bit UART2, Baud Rate is variable 1: Mode3: 9 bit UART2, Baud Rate is variable
		4	REN2	R/W	0	UART2 reception enable 0: Disable reception 1: Enable reception
		3	TB82	R/W	0	Transmit Bit 8, the ninth bit to be transmitted in Mode3
		2	RB82	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode3
		1	TI2	R/W	0	Transmit interrupt flag Set by H/W at the beginning of the stop bit in Mode 1 & 3. Must be cleared by S/W.
		0	RI2	R/W	0	Receive interrupt flag Set by H/W at the sampling point of the stop bit in Mode 1 & 3. Must be cleared by S/W.
8Fh	SBUF2	7~0	SBUF2	R/W	–	UART2 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.
90h	P1	7~0	P1	R/W	FFh	Port1 data
91h	P0OE	7~0	P0OE	R/W	00h	Port0 CMOS Push-Pull output enable control 0: Disable 1: Enable
92h	P0LOE	7~0	P0LOE	R/W	00h	Port0 LCD 1/2 bias output enable control 0: Disable 1: Enable
93h	P2MOD	3~2	P2MOD1	R/W	01	P2.1 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: not defined
		1~0	P2MOD0	R/W	01	P2.0 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: not defined

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
94h	OPTION	7	UART1W	R/W	0	Set 1 to enable one wire UART1 mode, both TXD/RXD use P3.1 pin.
		5~4	WDTPSC	R/W	00	Watchdog Timer pre-scalar time select 00: 480ms WDT overflow rate 01: 240ms WDT overflow rate 10: 120ms WDT overflow rate 11: 60ms WDT overflow rate
		3~2	ADCKS	R/W	00	ADC clock rate select 00: F _{SYSClk} /32 01: F _{SYSClk} /16 10: F _{SYSClk} /8 11: F _{SYSClk} /4
		1~0	TM3PSC	R/W	00	Timer3 Interrupt rate 00: Timer3 Interrupt rate is 32768 Slow clock cycle 01: Timer3 Interrupt rate is 16384 Slow clock cycle 10: Timer3 Interrupt rate is 8192 Slow clock cycle 11: Timer3 Interrupt rate is 128 Slow clock cycle
95h	INTFLG	7	LVDIF	R	–	Low Voltage Detect flag Set by H/W when a low voltage occurs.
		5	TKIF	R/W	0	Touch Key Interrupt Flag Set by H/W at the end of TK conversion if SYSClk is fast enough. S/W writes DFh to INTFLG or sets the TKSOC bit to clear this flag.
		4	ADIF	R/W	0	ADC interrupt flag Set by H/W at the end of ADC conversion. S/W writes EFh to INTFLG or sets the ADSOC bit to clear this flag.
		1	P1IF	R/W	0	Port1 pin change Interrupt flag Set by H/W when a Port1 pin state change is detected and its interrupt enable bit is set (P1WKUP). P1IE does not affect this flag's setting. It is cleared automatically when the program performs the interrupt service routine. S/W can write FDh to INTFLG to clear this bit.
		0	TF3	R/W	0	Timer3 Interrupt Flag Set by H/W when Timer3 reaches TM3PSC setting cycles. It is cleared automatically when the program performs the interrupt service routine. S/W can write FEh to INTFLG to clear this bit.
96h	P1WKUP	7~0	P1WKUP	R/W	00h	P1.7~P1.0 pin individual Wake-up/Interrupt enable control 0: Disable; 1: Enable.
97h	SWCMD	7~0	SWRST	W		Write 56h to generate S/W Reset
		7~0	IAPEN	W		Write 65h to set IAPEN control flag; Write other value to clear IAPEN flag. It is recommended to clear it immediately after IAP access.
		1	WDTO	R	0	WatchDog Time-Out flag
		0	IAPEN	R	0	Flag indicates Flash memory sectors can be accessed by IAP or not. This bit combines with MVCLOCK to define the accessible IAP area.

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
98h	SCON	7	SM0	R/W	0	UART1 Serial port mode select bit 0, 1 (SM0, SM1) = 00: Mode0: 8 bit shift register, Baud Rate= $F_{SYSCLK}/2$ 01: Mode1: 8 bit UART1, Baud Rate is variable 10: Mode2: 9 bit UART1, Baud Rate= $F_{SYSCLK}/32$ or $/64$ 11: Mode3: 9 bit UART1, Baud Rate is variable
		6	SM1	R/W	0	
		5	SM2	R/W	0	
						Serial port mode select bit 2 SM2 enables multiprocessor communication over a single serial line and modifies the above as follows. In Modes 2 & 3, if SM2 is set then the received interrupt will not be generated if the received ninth data bit is 0. In Mode 1, the received interrupt will not be generated unless a valid stop bit is received. In Mode 0, SM2 should be 0.
		4	REN	R/W	0	Set 1 to enable UART1 Reception
		3	TB8	R/W	0	Transmitter bit 8, ninth bit to transmit in Modes 2 and 3
		2	RB8	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode 2 and 3 or the stop bit is Mode 1 if SM2=0
		1	TI	R/W	0	Transmit Interrupt flag Set by H/W at the end of the eighth bit in Mode 0, or at the beginning of the stop bit in other modes. Must be cleared by S/W
		0	RI	R/W	0	Receive Interrupt flag Set by H/W at the end of the eighth bit in Mode 0, or at the sampling point of the stop bit in other modes. Must be cleared by S/W.
99h	SBUF	7~0	SBUF	R/W	–	UART1 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.
9Eh	PWMOE	7	PWM1IE	R/W	0	PWM1 Interrupt Enable. 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		6	PWM0IE	R/W	0	PWM0 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		2	PWM2OE	R/W	0	PWM2 enable and signal output to P1.6 pin 0: disable 1: enable
		1	PWM1OE	R/W	0	PWM1 enable and signal output to P1.3 pin 0: disable 1: enable
		0	PWM0OE	R/W	0	PWM0 enable and signal output to P1.2 pin 0: disable 1: enable
9Fh	PWMCLR	7	PWM2IE	R/W	0	PWM2 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		2	PWM2CLR	R/W	0	PWM2 clear enable 0: PWM2 is running 1: PWM2 is cleared and held
		1	PWM1CLR	R/W	0	PWM1 clear enable 0: PWM1 is running 1: PWM1 is cleared and held
		0	PWM0CLR	R/W	0	PWM0 clear enable 0: PWM0 is running 1: PWM0 is cleared and held
A0h	P2	7~0	P2	R/W	03h	P2 data

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
A1h	PWMCON	5~4	PWM2CKS	R/W	10	PWM2 clock source 00: F _{SYSCLK} 01: F _{SYSCLK} 10: FRC 11: FRC x 2
		3~2	PWM1CKS	R/W	10	PWM1 clock source 00: F _{SYSCLK} 01: F _{SYSCLK} 10: FRC 11: FRC x 2
		1~0	PWM0CKS	R/W	10	PWM0 clock source 00: F _{SYSCLK} 01: F _{SYSCLK} 10: FRC 11: FRC x 2
A2h	P1MODL	7~6	P1MOD3	R/W	01	P1.3 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P1.3 is ADC input
		5~4	P1MOD2	R/W	01	P1.2 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P1.2 is ADC input
		3~2	P1MOD1	R/W	01	P1.1 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P1.1 is ADC input
		1~0	P1MOD0	R/W	01	P1.0 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P1.0 is ADC input
A3h	P1MODH	7~6	P1MOD7	R/W	01	P1.7 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3
		5~4	P1MOD6	R/W	01	P1.6 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3
		3~2	P1MOD5	R/W	01	P1.5 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P1.5 is ADC input
		1~0	P1MOD4	R/W	01	P1.4 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P1.4 is ADC input
A4h	P3MODL	7~6	P3MOD3	R/W	01	P3.3 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P3.3 is ADC input
		5~4	P3MOD2	R/W	01	P3.2 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P3.2 is ADC input
		3~2	P3MOD1	R/W	01	P3.1 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P3.1 is ADC input
		1~0	P3MOD0	R/W	01	P3.0 Pin Control 00: Mode0; 01: Mode1; 10: Mode2 11: Mode3, P3.0 is ADC input
A5h	P3MODH	7~6	P3MOD7	R/W	00	P3.7 Pin Control 00: Mode0; 01: Mode1; 10: Mode2; 11: Mode3
		5~4	P3MOD6	R/W	01	P3.6 Pin Control 00: Mode0; 01: Mode1; 10: Mode2; 11: Mode3
		3~2	P3MOD5	R/W	01	P3.5 Pin Control 00: Mode0; 01: Mode1; 10: Mode2; 11: Mode3
		1~0	P3MOD4	R/W	01	P3.4 Pin Control 00: Mode0; 01: Mode1; 10: Mode2; 11: Mode3

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
A6h	PINMOD	6	I2CSEL	R/W	0	I2C Pin Select 0: SCL/SDA = P3.4/P3.5 1: SCL/SDA = P3.0/P3.1
		5	TCOE	R/W	0	Set 1 to enable "System clock divided by 2" (CKO) output to P1.4 pin
		4	T2OE	R/W	0	Set 1 to enable "Timer2 overflow divided by 2" (T2O) output to P1.0 pin
		3	HSNK2EN	R/W	0	Pin H-sink enable (Group 2: P10~P17) 0: Group 2 High-sink disable 1: Group 2 High-sink enable
		2	HSNK1EN	R/W	0	Pin H-sink enable (Group 1: P04 ~P07, P30 ~P33) 0: Group 1 High-sink disable 1: Group 1 High-sink enable
		1	HSNK0EN	R/W	0	Pin H-sink enable (Group 0: P00~P03, P20, P21, P34~P37) 0: Group 0 High-sink disable 1: Group 0 High-sink enable
		0	T0OE	R/W	0	Set 1 to enable "Timer0 overflow divided by 64" (T0O) output to P3.4 pin
A7h	TKCHS	4~0	TKCHS	R/W	1Fh	Specify the first Touch Key channel 00000: TK0 (P3.3) 00001: TK1 (P3.2) 00010: TK2 (P3.1) 00011: TK3 (P3.0) 00100: TK4 (P1.0) 00101: TK5 (P1.1) 00110: TK6 (P1.2) 00111: TK7 (P1.3) 01000: TK8 (P1.4) 01001: TK9 (P1.6) 01010: TK10 (P1.7) 01011: TK11 (P3.6) 01100: TK12 (P3.5) 01101: TK13 (P3.4) 01110: TK14 (P1.5) 01111: TK15 (P3.7) 10000: TK16 (P0.3) 10001: TK17 (P0.5) 10010: TK18 (P0.6) 10011: TK19 (P0.7) 10111: TK reference capacitor
A8h	IE	7	EA	R/W	0	Global interrupt enable control. 0: Disable all Interrupts. 1: Each interrupt is enabled or disabled by its own interrupt control bit.
		5	ET2	R/W	0	Set 1 to enable Timer2 interrupt
		4	ES	R/W	0	Set 1 to enable Serial Port (UART1) Interrupt
		3	ET1	R/W	0	Set 1 to enable Timer1 Interrupt
		2	EX1	R/W	0	Set 1 to enable external INT1 pin Interrupt & Halt/Stop mode wake up capability
		1	ET0	R/W	0	Set 1 to enable Timer0 Interrupt
		0	EX0	R/W	0	Set 1 to enable external INT0 pin Interrupt & Halt/Stop mode wake up capability
A9h	INTE1	7	PWMIE	R/W	0	Set 1 to enable PWM0~PWM2 interrupt
		6	I2CE	R/W	0	Set 1 to enable I ² C (master/slave) interrupt
		5	ES2	R/W	0	Set 1 to enable Serial Port (UART2) interrupt
		4	SPIE	R/W	0	Set 1 to enable SPI interrupt
		3	ADTKIE	R/W	0	Set 1 to enable ADC/Touch Key Interrupt
		2	EXLVDIE	R/W	0	Set 1 to enable external INT2~INT9 pin Interrupt, Halt/Stop mode wake up capability and LVD interrupt.
		1	P1IE	R/W	0	Set 1 to enable Port1 Pin Change Interrupt
		0	TM3IE	R/W	0	Set 1 to enable Timer3 Interrupt
AAh	ADC DL	7~4	ADC DL	R	—	ADC data bit 3~0
ABh	ADC DH	7~0	ADC DH	R	—	ADC data bit 11~4

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
ADh	TKCON	7	TKPD	R/W	1	Touch Key Power Down 0: Touch Key enable; 1: Touch Key disable
		6	TKEOC	R	1	Touch Key end of conversion flag 0: Indicates conversion is in progress 1: Indicates conversion is finished
		5	TKRERUN	R/W	0	Touch Key Auto re-start, doesn't need to set TKSOC again to restart TK converter. 0: Auto re-start disable. TKSOC needs to be executed once for each TK conversion 1: Auto re-start enable. After TKSOC is executed once, TK will be converted continuously without re-executing TKSOC
		4	TKIVCS	R/W	0	Touch Key internal voltage control select 0: $V_{CHG}=2.8V$; $V_{INT}=1.4V$ 1: $V_{CHG}=3.6V$; $V_{INT}=1.8V$
		3	TKXCAP	R/W	0	Touch Key external capacitor select 0: disable Touch Key external capacitor 1: enable Touch Key external capacitor
		2	TKOFFSET	R/W	0	status of non-scan Touch Key 0: connect to V_{SS} 1: connect to AC shielding , connect to $V_{SS}@EOC$
		1~0	ATKMODE	R/W	00	Touch Key Scan Mode 00: TK scan method, each channel scan 1 time, max 21 TK channels 01: TK scan method, each channel scan 2 times, max 16 TK channels 10: TK scan method, each channel scan 4 times, max 8 TK channels 11: TK scan method, each channel scan 8 times, max 4 TK channels
AEh	CHSEL	7~4	ADCHS	R/W	1111	ADC channel select 0000: AD0 (P3.3) 0001: AD1 (P3.2) 0010: AD2 (P3.1) 0011: AD3 (P3.0) 0100: AD4 (P1.0) 0101: AD5 (P1.1) 0110: AD6 (P1.2) 0111: AD7 (P1.3) 1000: AD8 (P1.4) 1001: AD9 (P1.5) 1010: Reserved 1011: V_{BG} (Internal Bandgap Reference Voltage) 1100: AD12 (P0.7) 1101: AD13 (P0.5) 1110: AD14 (P0.6) 1111: $1/4 V_{CC}$
		3~2	ADCVREFS	R/W	00	ADC reference voltage 00: V_{CC} 01: 2.5V 1x: Reserved
		1	VBGEN	R/W	0	force V_{BG} generator enable 0: V_{BG} generator is automatically enable and disable 1: Force V_{BG} generator enable included in Idle mode but disabled in Halt/Stop mode
AFh	P0ADIE	7~5	P0ADIE	R/W	000	ADC channel input enable 000: P0.7~P0.4 are digital input 1xx: P0.7 is ADC input x1x: P0.6 is ADC input xx1: P0.5 is ADC input
B0h	P3	7~0	P3	R/W	FFh	Port3 data

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B1h	LEDCON	7~6	LEDEN	R/W	00	LED BiD mode enable and duty select 00: LED BiD mode disable 01: LED 1/8 duty (4COM x 4SEG) 10: LED 1/9 duty (4COM x 5SEG) 11: LED 1/10 duty (4COM x 6SEG)
		5~4	LEDPSC	R/W	00	LED clock prescaler select 00: LED clock is FRC divided by 64 01: LED clock is FRC divided by 32 10: LED clock is FRC divided by 16 11: LED clock is FRC divided by 8
		3	LEDHOLD	R/W	0	LED clock hold 0: LED scan 1: LED clock hold
		2~0	LEDBRIT	R/W	111	BiD mode: LED number 0~31, 40~47 brightness control 000: Level 0 (Darkest) ... 111: Level 7 (Brightest)
B2h	LEDCON2	7	LEDBRITM	R/W	0	Brightness smooth control 0: Uniform brightness mode 1: Brightness enhancement mode
		6~4	LEDBRIT2	R/W	111	BiD matrix mode: LED number 33, 35, 37, 39 brightness control DMX mode: LED number 0~63 brightness control 000: Level 0 (Darkest) ... 111: Level 7 (Brightest)
		2~0	LEDBRIT1	R/W	111	BiD mode: LED number 32, 34, 36, 38 brightness control 000: Level 0 (Darkest) ... 111: Level 7 (Brightest)
B3h	LEDCON3	7	LEDMTEN	R/W	0	LED DMX mode enable 0: disable 1: enable
		6	LED8EN	R/W	0	LED DMX mode pin enable control 0: LED8 disable 1: LED8 enable
		5	LED7EN	R/W	0	LED DMX mode pin enable control 0: LED7 disable 1: LED7 enable
		4	LED6EN	R/W	0	LED DMX mode pin enable control 0: LED6 disable 1: LED6 enable
		3	LED5EN	R/W	0	LED DMX mode pin enable control 0: LED5 disable 1: LED5 enable
		2	LED4EN	R/W	0	LED DMX mode pin enable control 0: LED4 disable 1: LED4 enable
		1	LED3EN	R/W	0	LED DMX mode pin enable control 0: LED3 disable 1: LED3 enable
		0	LED2EN	R/W	0	LED DMX mode pin enable control 0: LED2 disable 1: LED2 enable
B4h	TKTMRL	7~0	TKTMRL	R/W	FFh	Touch Key Scan length bit 7~0 adjustment. 00: shortest, FF: longest

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B5h	TKCON2	7	TKFJMP	R/W	0	Internal Touch Key clock frequency auto adjust option 0: Disable 1: Enable
		6~5	JMPVAL	R/W	0	Touch Key Clock frequency fine tune , only available in TKFJMP=0 00=frequency slowest, 11=frequency fastest
		4	SPREAD	R/W	0	Touch Key spread spectrum 0: Disable 1: Enable
		3~0	TKTMRH	R/W	0	Touch Key Scan length 11~8 adjustment. 0000: shortest, 1111: longest
B8h	IP	5	PT2	R/W	0	Timer2 Interrupt Priority Low bit
		4	PS	R/W	0	Serial Port (UART1) Interrupt Priority Low bit
		3	PT1	R/W	0	Timer1 Interrupt Priority Low bit
		2	PX1	R/W	0	External INT1 Pin Interrupt Priority Low bit
		1	PT0	R/W	0	Timer0 Interrupt Priority Low bit
		0	PX0	R/W	0	External INT0 Pin Interrupt Priority Low bit
B9h	IPH	5	PT2H	R/W	0	Timer2 Interrupt Priority High bit
		4	PSH	R/W	0	Serial Port (UART1) Interrupt Priority High bit
		3	PT1H	R/W	0	Timer1 Interrupt Priority High bit
		2	PX1H	R/W	0	External INT1 Pin Interrupt Priority High bit
		1	PT0H	R/W	0	Timer0 Interrupt Priority High bit
		0	PX0H	R/W	0	External INT0 Pin Interrupt Priority High bit
BAh	IP1	7	PPWM	R/W	0	PWM Interrupt Priority Low bit
		6	PI2C	R/W	0	I2C Interrupt Priority Low bit
		5	PS2	R/W	0	Serial Port (UART2) interrupt priority low bit
		4	PSPI	R/W	0	SPI interrupt priority low bit
		3	PADTKI	R/W	0	ADC/Touch Key Interrupt Priority Low bit
		2	PX2_9LVD	R/W	0	External INT2~INT9 Pin Interrupt Priority Low bit
		1	PP1	R/W	0	Port1 pin change Interrupt Priority Low bit
		0	PT3	R/W	0	Timer3 Interrupt Priority Low bit
BBh	IP1H	7	PPWMH	R/W	0	PWM Interrupt Priority High bit
		6	PI2CH	R/W	0	I2C Interrupt Priority High bit
		5	PS2H	R/W	0	Serial Port (UART2) interrupt priority high bit
		4	PSPIH	R/W	0	SPI interrupt priority high bit
		3	PADTKIH	R/W	0	ADC/Touch Key Interrupt Priority High bit
		2	PX2_9LVDH	R/W	0	External INT2~INT9 Pin Interrupt Priority High bit
		1	PP1H	R/W	0	Port1 Interrupt Priority High bit
		0	PT3H	R/W	0	Timer3 Interrupt Priority High bit

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
BCh	SPCON	7	SPEN	R/W	0	SPI enable 0: SPI disable 1: SPI enable
		6	MSTR	R/W	0	Master mode enable 0: Slave mode 1: Master mode
		5	CPOL	R/W	0	SPI clock polarity 0: SCK is low in idle state 1: SCK is high in idle state
		4	CPHA	R/W	0	SPI clock phase 0: Data sample on first edge of SCK period 1: Data sample on second edge of SCK period
		3	SSDIS	R/W	0	SS pin disable 0: Enable SS pin 1: Disable SS pin
		2	LSBF	R/W	0	LSB first 0: MSB first 1: LSB first
		1~0	SPCR	R/W	00	SPI clock rate 00: F _{SYSCLK} /2 01: F _{SYSCLK} /4 10: F _{SYSCLK} /8 11: F _{SYSCLK} /16
BDh	SPSTA	7	SPIF	R/W	0	SPI interrupt flag This is set by H/W at the end of a data transfer. Cleared by H/W when an interrupt is vectored into. Writing 0 to this bit will clear this flag.
		6	WCOL	R/W	0	Write collision interrupt flag Set by H/W if write data to SPDAT when SPBSY is set. Write 0 to this bit or rewrite data to SPDAT when SPBSY is cleared will clear this flag.
		5	MODF	R/W	0	Mode fault interrupt flag Set by H/W when SSDIS is cleared and SS pin is pulled low in Master mode. Write 0 to this bit will clear this flag. When this bit is set, the SPEN and MSTR in SPCON will be cleared by H/W.
		4	RCVOVF	R/W	0	Received buffer overrun flag Set by H/W at the end of a data transfer and RCVBF is set. Write 0 to this bit or read SPDAT register will clear this flag.
		3	RCVBF	R/W	0	Receive buffer full flag Set by H/W at the end of a data transfer. Write 0 to this bit or read SPDAT register will clear this flag.
		2	SPBSY	R	0	SPI busy flag Set by H/W when a SPI transfer is in progress.
BEh	SPDAT	7~0	SPDAT	R/W	0	SPI transmit and receive data The SPDAT register is used to transmit and receive data. Writing data to SPDAT place the data into shift register and start a transfer when in master mode. Reading SPDAT returns the contents of the receive buffer.

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
BFh	LVDS	7	LVDIE	R/W	0	Low Voltage Detect interrupt enable 0: Disable 1: Enable (note: EXLVDIE must be 1 at the same time to generate LVD interrupt)
		6	LVDO	R	-	Low Voltage Detect output
		3~0	LVDS	R/W	0	Low Voltage Detect select (Auto disable in Idle/Halt/Stop mode) 0000: Set LVD at 2.5V 0001: Set LVD at 2.6V 0010: Set LVD at 2.7V 0011: Set LVD at 2.8V 0100: Set LVD at 3.0V 0101: Set LVD at 3.1V 0110: Set LVD at 3.2V 0111: Set LVD at 3.3V 1000: Set LVD at 3.4V 1001: Set LVD at 3.6V 1010: Set LVD at 3.7V 1011: Set LVD at 3.8V 1100: Set LVD at 3.9V 1101: Set LVD at 4.0V 1110: Set LVD at 4.2V 1111: Set LVD at 4.3V
C1h	TKPINSEL0	7~0	TKPINSEL0	R/W	00	Touch Key TK7~TK0 Channel Select 0: Normal IO 1: Touch Key
C2h	TKPINSEL1	7~0	TKPINSEL1	R/W	00	Touch Key TK15~TK8 Channel Select 0: Normal IO 1: Touch Key
C3h	TKPINSEL2	7~0	TKPINSEL2	R/W	00	Touch Key TK23~TK16 Channel Select 0: Normal IO 1: Touch Key
C5h	ATKCH0	7~0	ATKCH0	R/W	00	Auto Touch Key TK7~TK0 Channel Select 0: Disable auto scan 1: Enable auto scan
C6h	ATKCH1	7~0	ATKCH1	R/W	00	Auto Touch Key TK15~TK8 Channel Select 0: Disable auto scan 1: Enable auto scan
C7h	ATKCH2	7~0	ATKCH2	R/W	00	Auto Touch Key TK23~TK16 Channel Select 0: Disable auto scan 1: Enable auto scan

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
C8h	T2CON	7	TF2	R/W	0	Timer2 overflow flag Set by H/W when Timer/Counter 2 overflows unless RCLK=1 or TCLK=1. This bit must be cleared by S/W.
		6	EXF2	R/W	0	T2EX interrupt pin falling edge flag Set when a capture or a reload is caused by a negative transition on T2EX pin if EXEN2=1. This bit must be cleared by S/W.
		5	RCLK	R/W	0	UART receive clock control bit 0: Use Timer1 overflow as receive clock for serial port in mode 1 or 3 1: Use Timer2 overflow as receive clock for serial port in mode 1 or 3
		4	TCLK	R/W	0	UART transmit clock control bit 0: Use Timer1 overflow as transmit clock for serial port in mode 1 or 3 1: Use Timer2 overflow as transmit clock for serial port in mode 1 or 3
		3	EXEN2	R/W	0	T2EX pin enable 0: T2EX pin disable 1: T2EX pin enable, it cause a capture or reload when a negative transition on T2EX pin is detected if RCLK=TCLK=0
		2	TR2	R/W	0	Timer2 run control 0: timer stops 1: timer runs
		1	CT2N	R/W	0	Timer2 Counter/Timer select bit 0: Timer mode, Timer2 data increases at 2 System clock cycle rate 1: Counter mode, Timer2 data increases at T2 pin's negative edge
		0	CPRL2N	R/W	0	Timer2 Capture/Reload control bit 0: Reload mode, auto-reload on Timer2 overflows or negative transitions on T2EX pin if EXEN2=1. 1: Capture mode, capture on negative transitions on T2EX pin if EXEN2=1. If RCLK=1 or TCLK=1, CPRL2N is ignored and timer is forced to auto-reload on Timer2 overflow.
C9h	IAPWE	7~0	IAPWE	W	—	Write 4Ah to enable one byte IAP write to ROM[7A00~7BFF] Write 4Ch to enable one byte IAP write to ROM[7C00~7DFF] Write BAh to enable ERASE 512 byte of ROM[7A00~7BFF] Write BCh to enable ERASE 512 byte of ROM[7C00~7DFF] Write other value to disable IAP write
		7	IAPWE	R	0	Flag indicates Flash memory can be written by IAP or not 0: IAP Write/Erase disable 1: IAP Write/Erase enable
C9h	IAPWE	6	IAPTO	R	0	IAP (or EEPROM write) Time-Out flag Set by H/W when IAP (or EEPROM write) Time-out occurs. Cleared by H/W when IAPWE=0 (or EEPWE=0).
CAh	RCP2L	7~0	RCP2L	R/W	00h	Timer2 reload/capture data low byte
CBh	RCP2H	7~0	RCP2H	R/W	00h	Timer2 reload/capture data high byte
CCh	TL2	7~0	TL2	R/W	00h	Timer2 data low byte
CDh	TH2	7~0	TH2	R/W	00h	Timer2 data high byte
CEh	EXA2	7~0	EXA2	R/W	00h	Expansion accumulator 2
CFh	EXA3	7~0	EXA3	R/W	00h	Expansion accumulator 3
D0h	PSW	7	CY	R/W	0	ALU carry flag
		6	AC	R/W	0	ALU auxiliary carry flag
		5	F0	R/W	0	General purpose user-definable flag
		4	RS1	R/W	0	Register Bank Select bit 1
		3	RS0	R/W	0	Register Bank Select bit 0
		2	OV	R/W	0	ALU overflow flag
		1	F1	R/W	0	General purpose user-definable flag
		0	P	R/W	0	Parity flag
D1h	PWM0DH	7~0	PWM0DH	R/W	80h	PWM0 duty high byte write sequence: PWM0DL then PWM0DH read sequence: PWM0DH then PWM0DL
D2h	PWM0DL	7~0	PWM0DL	R/W	00h	PWM0 duty low byte write sequence: PWM0DL then PWM0DH read sequence: PWM0DH then PWM0DL

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
D3h	PWM1DH	7~0	PWM1DH	R/W	80h	PWM1 duty high byte write sequence: PWM1DL then PWM1DH read sequence: PWM1DH then PWM1DL
D4h	PWM1DL	7~0	PWM1DL	R/W	00h	PWM1 duty low byte write sequence: PWM1DL then PWM1DH read sequence: PWM1DH then PWM1DL
D5h	PWM2DH	7~0	PWM2DH	R/W	80h	PWM2 duty high byte write sequence: PWM2DL then PWM2DH read sequence: PWM2DH then PWM2DL
D6h	PWM2DL	7~0	PWM2DL	R/W	00h	PWM2 duty low byte write sequence: PWM2DL then PWM2DH read sequence: PWM2DH then PWM2DL
D8h	CLKCON	7	SCKTYPE	R/W	0	Slow clock Type. This bit can be changed only in Fast mode (SELFCK=1) 0: SRC 1: SXT, P2.0 and P2.1 are crystal pins
		6	FCKTYPE	R/W	0	Fast clock type. This bit can be changed only in Slow mode (SELFCK=0). 0: FRC 1: FXT, P2.0 and P2.1 are crystal pins, oscillator gain is high for FXT
		5	STPSCK	R/W	1	Set 1 to stop SRC clock in PDOWN mode
		4	STPPCK	R/W	0	Set 1 to stop UART/Timer0/1/2 clock in Idle mode for current reducing.
		3	STPFCK	R/W	0	Set 1 to stop Fast clock for power saving in Slow/Idle mode. This bit can be changed only in Slow mode.
		2	SELFCK	R/W	0	System clock select. This bit can be changed only when STPFCK=0. 0: Slow clock 1: Fast clock
		1~0	CLKPSC	R/W	11	System clock prescaler. Effective after 16 clock cycles (Max.) delay. 00: System clock is Fast/Slow clock divided by 16 01: System clock is Fast/Slow clock divided by 4 10: System clock is Fast/Slow clock divided by 2 11: System clock is Fast/Slow clock divided by 1
D9h	PWM0PRDH	7~0	PWM0PRDH	R/W	FFh	PWM0 period high byte write sequence: PWM0PRDL then PWM0PRDH read sequence: PWM0PRDH then PWM0PRDL
DAh	PWM0PRDL	7~0	PWM0PRDL	R/W	FFh	PWM0 period low byte write sequence: PWM0PRDL then PWM0PRDH read sequence: PWM0PRDH then PWM0PRDL
DBh	PWM1PRDH	7~0	PWM1PRDH	R/W	FFh	PWM1 period high byte write sequence: PWM1PRDL then PWM1PRDH read sequence: PWM1PRDH then PWM1PRDL
DCh	PWM1PRDL	7~0	PWM1PRDL	R/W	FFh	PWM1 period low byte write sequence: PWM1PRDL then PWM1PRDH read sequence: PWM1PRDH then PWM1PRDL
DDh	PWM2PRDH	7~0	PWM2PRDH	R/W	FFh	PWM2 period high byte write sequence: PWM2PRDL then PWM2PRDH read sequence: PWM2PRDH then PWM2PRDL
DEh	PWM2PRDL	7~0	PWM2PRDL	R/W	FFh	PWM2 period low byte write sequence: PWM2PRDL then PWM2PRDH read sequence: PWM2PRDH then PWM2PRDL
E0h	ACC	7~0	ACC	R/W	00h	Accumulator

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
E1h	MICON	7	MIEN	R/W	0	Master I ² C enable 0: disable 1: enable
		6	MIACKO	R/W	0	When Master I ² C receive data, send acknowledge to I ² C Bus 0: ACK to slave device 1: NACK to slave device
		5	MIIF	R/W	0	Master I ² C Interrupt flag 0: write 0 to clear it 1: Master I ² C transfer one byte complete
		4	MIACKI	R	–	When Master I ² C transfer, acknowledgement form I ² C bus (read only) 0: ACK received 1: NACK received
		3	MISTART	R/W	0	Master I ² C Start bit 1: start I ² C bus transfer
		2	MISTOP	R/W	1	Master I ² C Stop bit 1: send STOP signal to stop I ² C bus
		1~0	MICR	R/W	00	Master I ² C (SCL) clock frequency selection 00: Fsys/4 (ex. If Fsys=16MHz, I ² C clock is 4M Hz) 01: Fsys/16 (ex. If Fsys=16MHz, I ² C clock is 1M Hz) 10: Fsys/64 (ex. If Fsys=16MHz, I ² C clock is 250K Hz) 11: Fsys/256 (ex. If Fsys=16MHz, I ² C clock is 62.5K Hz)
E2h	MIDAT	7~0	MIDAT	R/W	00	Master I ² C data shift register (W): After Start and before Stop condition, write this register will resume transmission to I ² C bus (R): After Start and before Stop condition, read this register will resume receiving from I ² C bus
E5h	ETTCN	7	EFT2CS	R/W	0	EFT2 Detector enable 0: Disable 1: Enable
		6	EFT1CS	R/W	0	EFT1 Detector enable 0: Disable 1: Enable
		5~4	EFT1S	R/W	0	EFT1 Detector sensitivity adjustment
		3	EFTSLOW	R/W	0	Force System clock to Slow clock while EFT detected 0: Disable 1: Enable
		2	EFTWCPU	R/W	0	CPU enter Wait state while EFT detected 0: Disable 1: Enable
		1	EFTWOUT	R/W	0	EFTWAIT output to pin 0: P00 = normal I/O 1: P00 = EFTWAIT
		0	CKHLDE	R/W	00	Clock hold enable 0: Disable 0: Enable
E6h	EXA	7~0	EXA	R/W	00h	Expansion accumulator
E7h	EXB	7~0	EXB	R/W	00h	Expansion B register
E9h	SIADR	7~1	SA	R/W	64h	Slave I ² C address assigned
		0	SIEN	R/W	0	Slave I ² C enable 0: disable 1: enable

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
EAh	SICON	7	MIIE	R/W	0	I ² C Master interrupt enable 0: disable 1: enable
		6	TXDIE	R/W	0	Slave I ² C transmission completed interrupt enable 0: disable 1: enable
		5	RCD2IE	R/W	0	Slave I ² C DATA2(SITXRCD2) reception completed interrupt enable 0: disable 1: enable
		4	RCD1IE	R/W	0	Slave I ² C DATA1(SIRCD1) reception completed interrupt enable 0: disable 1: enable
		2	TXDF	R/W	1	Slave I ² C transmission completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I ² C transmission complete
		1	RCD2F	R/W	0	Slave I ² C DATA2(SITXRCD2) reception completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I ² C DATA2(SITXRCD2) reception complete enable
		0	RCD1F	R/W	0	Slave I ² C DATA1(SIRCD1) reception completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I ² C DATA1(SIRCD1) reception complete
EBh	SIRCD1	7~0	SIRCD1	R	—	Slave I ² C data receive register1 (DATA1)
ECh	SITXRCD2	7~0	SITXRCD2	R/W	—	Slave I ² C transmit and receive data register Read: Slave I ² C data receive register2 (DATA2) Write: Slave I ² C data transmission register (TXD)
EFh	PWRCON	4	AVPULL	R/W	0	Auto turn-on V _{PULL} when Slow modeto Fast mode 0: disable 1: enable
		3	WARMTIME	R/W	0	Warm up time after Halt/Slow mode 0: 64 Clock 1: 128 Clock
		2	ENVPULL	R/W	0	Power control, force V _{PULL} enable 0: disable 1: enable
		1	PWRIDLE	R/W	0	Power control, V _{PULL} control at Idle mode 0: VDD = LDO @ Idle mode 1: VDD = V _{PULL} @ Idle mode
		0	PWRSLOW	R/W	0	Power control, V _{PULL} control at Slow mode 0: VDD = LDO @ Slow modle 1: VDD = V _{PULL} @ Slow mode
F0h	B	7~0	B	R/W	00h	B register
F1h	CRCDL	7~0	CRCDL	R/W	FFh	16-bit CRC data bit 7~0
F2h	CRCDH	7~0	CRCDH	R/W	FFh	16-bit CRC data bit 15~8
F3h	CRCIN	7~0	CRCIN	W	—	CRC input data
F5h	CFGBG	3~0	BGTRIM	R/W	—	V _{BG} trimming value (Chip Reserved)
F6h	CFGWL	6~0	FRCF	R/W	—	FRC frequency adjustment 00h: lowest frequency 7Fh: highest frequency

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
F7h	AUX2	7~6	WDTE	R/W	—	Watchdog Timer Reset control 0x: WDT disable 10: WDT enable in Fast/Slow mode, disable in Idle/Halt/Stop mode 11: WDT always enable
		5	PWRSV	R/W	—	Set 1 to reduce the chip's power consumption at Idle/Halt/Stop Mode.
		4	VBGOUT	R/W	0	Bandgap voltage output control 0: P3.2 as normal I/O 1: Bandgap voltage output to P3.2 pin
		3	DIV32	R/W	0	only active when MULDIV16 =1 0: instruction DIV as 16/16 bit division operation 1: instruction DIV as 32/16 bit division operation
		2~1	IAPTE	R/W	00	IAP watchdog timer enable 00: Disable 01: wait 0.8 ms trigger watchdog time-out flag 10: wait 3.2 ms trigger watchdog time-out flag 11: wait 6.4 ms trigger watchdog time-out flag
		0	MULDIV16	R/W	0	0: instruction MUL/DIV as 8*8, 8/8 operation 1: instruction MUL/DIV as 16*16, 16/16 or 32/16 operation
F8h	AUX1	7	CLRWDT	R/W	0	Set 1 to clear WDT, H/W auto clear it at next clock cycle
		6	CLRTM3	R/W	0	Set 1 to clear Timer3, HW auto clear it at next clock cycle.
		5	TKSOC	R/W	0	Touch Key Start of Conversion Set 1 to start Touch Key conversion. If SYSCLK is fast enough, this bit will be cleared by H/W at the end of conversion. S/W can also write 0 to clear this flag.
		4	ADSOC	R/W	0	ADC Start of Conversion Set 1 to start ADC conversion. Cleared by H/W at the end of conversion. S/W can also write 0 to clear this flag.
		3	LVRPD	R/W	0	Low Voltage Reset function select 0: enable LVR 1: disable LVR
		2	T2SEL	R/W	0	Timer2 counter mode (CT2N=1) input select 0: P1.0 (T2) pin (8051 standard) 1: Slow clock divide by 16 (SLOWCLK/16)
		1	T1SEL	R/W	0	Timer1 counter mode (CT1N=1) input select 0: P3.5 (T1) pin (8051 standard) 1: Slow clock divide by 16 (SLOWCLK/16)
		0	DPSEL	R/W	0	Active DPTR Select

Adr	Flash	Bit#	Bit Name	Description
7FFFh	CFGWH	7	PROTN	Flash Code Protect, 0=Protect
		6	XRSTEN	External Pin Reset enable, 0=enable.
		5~3	LVRE	Low Voltage Reset function select 000: Set LVR at 2.52V 001: Set LVR at 2.74V 010: Set LVR at 2.99V 011: Set LVR at 3.23V 100: Set LVR at 3.48V 101: Set LVR at 3.72V 110: Set LVR at 3.96V 111: Set LVR at 4.2V
		1	MVCLOCKN	If 0, the MOVC & MOVX instruction's accessibility to MOVC-Lock area is limited.
		0	FRCPSC	FRC frequency select 0: 9.216 MHz 1: 18.432 MHz

指令集

指令都是 1,2 或 3 个字节长如“字节”列所示。每条指令需要 2~8 个系统时钟周期来执行如“周期”列中所示。

ARITHMETIC				
Mnemonic	Description	byte	cycle	opcode
ADD A, Rn	Add register to A	1	2	28-2F
ADD A, dir	Add direct byte to A	2	2	25
ADD A, @Ri	Add indirect memory to A	1	2	26-27
ADD A, #data	Add immediate to A	2	2	24
ADDC A, Rn	Add register to A with carry	1	2	38-3F
ADDC A, dir	Add direct byte to A with carry	2	2	35
ADDC A, @Ri	Add indirect memory to A with carry	1	2	36-37
ADDC A, #data	Add immediate to A with carry	2	2	34
SUBB A, Rn	Subtract register from A with borrow	1	2	98-9F
SUBB A, dir	Subtract direct byte from A with borrow	2	2	95
SUBB A, @Ri	Subtract indirect memory from A with borrow	1	2	96-97
SUBB A, #data	Subtract immediate from A with borrow	2	2	94
INC A	Increment A	1	2	04
INC Rn	Increment register	1	2	08-0F
INC dir	Increment direct byte	2	2	05
INC @Ri	Increment indirect memory	1	2	06-07
DEC A	Decrement A	1	2	14
DEC Rn	Decrement register	1	2	18-1F
DEC dir	Decrement direct byte	2	2	15
DEC @Ri	Decrement indirect memory	1	2	16-17
INC DPTR	Increment data pointer	1	4	A3
MUL AB	Multiply A by B	1	8 / 16	A4
DIV AB	Divide A by B	1	8/16/32	84
DA A	Decimal Adjust A	1	2	D4

LOGICAL				
Mnemonic	Description	byte	cycle	opcode
ANL A, Rn	AND register to A	1	2	58-5F
ANL A, dir	AND direct byte to A	2	2	55
ANL A, @Ri	AND indirect memory to A	1	2	56-57
ANL A, #data	AND immediate to A	2	2	54
ANL dir, A	AND A to direct byte	2	2	52
ANL dir, #data	AND immediate to direct byte	3	4	53
ORL A, Rn	OR register to A	1	2	48-4F
ORL A, dir	OR direct byte to A	2	2	45
ORL A, @Ri	OR indirect memory to A	1	2	46-47
ORL A, #data	OR immediate to A	2	2	44
ORL dir, A	OR A to direct byte	2	2	42
ORL dir, #data	OR immediate to direct byte	3	4	43
XRL A, Rn	Exclusive-OR register to A	1	2	68-6F
XRL A, dir	Exclusive-OR direct byte to A	2	2	65
XRL A, @Ri	Exclusive-OR indirect memory to A	1	2	66-67
XRL A, #data	Exclusive-OR immediate to A	2	2	64
XRL dir, A	Exclusive-OR A to direct byte	2	2	62
XRL dir, #data	Exclusive-OR immediate to direct byte	3	4	63
CLR A	Clear A	1	2	E4
CPL A	Complement A	1	2	F4
SWAP A	Swap Nibbles of A	1	2	C4
RL A	Rotate A left	1	2	23

LOGICAL				
Mnemonic	Description	byte	cycle	opcode
RLC A	Rotate A left through carry	1	2	33
RR A	Rotate A right	1	2	03
RRC A	Rotate A right through carry	1	2	13

DATA TRANSFER				
Mnemonic	Description	byte	cycle	opcode
MOV A, Rn	Move register to A	1	2	E8-EF
MOV A, dir	Move direct byte to A	2	2	E5
MOV A, @Ri	Move indirect memory to A	1	2	E6-E7
MOV A, #data	Move immediate to A	2	2	74
MOV Rn, A	Move A to register	1	2	F8-FF
MOV Rn, dir	Move direct byte to register	2	4	A8-AF
MOV Rn, #data	Move immediate to register	2	2	78-7F
MOV dir, A	Move A to direct byte	2	2	F5
MOV dir, Rn	Move register to direct byte	2	4	88-8F
MOV dir, dir	Move direct byte to direct byte	3	4	85
MOV dir, @Ri	Move indirect memory to direct byte	2	4	86-87
MOV dir, #data	Move immediate to direct byte	3	4	75
MOV @Ri, A	Move A to indirect memory	1	2	F6-F7
MOV @Ri, dir	Move direct byte to indirect memory	2	4	A6-A7
MOV @Ri, #data	Move immediate to indirect memory	2	2	76-77
MOV DPTR, #data	Move immediate to data pointer	3	4	90
MOVC A, @A+DPTR	Move code byte relative DPTR to A	1	4	93
MOVC A, @A+PC	Move code byte relative PC to A	1	4	83
MOVB A, @Ri	Move external data (A8) to A	1	4	E2-E3
MOVB A, @DPTR	Move external data (A16) to A	1	4	E0
MOVB @Ri, A	Move A to external data (A8)	1	4	F2-F3
MOVB @DPTR, A	Move A to external data (A16)	1	4	F0
PUSH dir	Push direct byte onto stack	2	4	C0
POP dir	Pop direct byte from stack	2	4	D0
XCH A, Rn	Exchange A and register	1	2	C8-CF
XCH A, dir	Exchange A and direct byte	2	2	C5
XCH A, @Ri	Exchange A and indirect memory	1	2	C6-C7
XCHD A, @Ri	Exchange A and indirect memory nibble	1	2	D6-D7

BOOLEAN				
Mnemonic	Description	byte	cycle	opcode
CLR C	Clear carry	1	2	C3
CLR bit	Clear direct bit	2	2	C2
SETB C	Set carry	1	2	D3
SETB bit	Set direct bit	2	2	D2
CPL C	Complement carry	1	2	B3
CPL bit	Complement direct bit	2	2	B2
ANL C, bit	AND direct bit to carry	2	4	82
ANL C, /bit	AND direct bit inverse to carry	2	4	B0
ORL C, bit	OR direct bit to carry	2	4	72
ORL C, /bit	OR direct bit inverse to carry	2	4	A0
MOV C, bit	Move direct bit to carry	2	2	A2
MOV bit, C	Move carry to direct bit	2	4	92

BRANCHING				
Mnemonic	Description	byte	cycle	opcode
ACALL addr 11	Absolute jump to subroutine	2	4	11-F1
LCALL addr 16	Long jump to subroutine	3	4	12
RET	Return from subroutine	1	4	22
RETI	Return from interrupt	1	4	32
AJMP addr 11	Absolute jump unconditional	2	4	01-E1
LJMP addr 16	Long jump unconditional	3	4	02
SJMP rel	Short jump (relative address)	2	4	80
JC rel	Jump on carry = 1	2	4	40
JNC rel	Jump on carry = 0	2	4	50
JB bit, rel	Jump on direct bit = 1	3	4	20
JNB bit, rel	Jump on direct bit = 0	3	4	30
JBC bit, rel	Jump on direct bit = 1 and clear	3	4	10
JMP @A+DPTR	Jump indirect relative DPTR	1	4	73
JZ rel	Jump on accumulator = 0	2	4	60
JNZ rel	Jump on accumulator $\neq$ 0	2	4	70
CJNE A, dir, rel	Compare A, direct, jump not equal relative	3	4	B5
CJNE A, #data, rel	Compare A, immediate, jump not equal relative	3	4	B4
CJNE Rn, #data, rel	Compare register, immediate, jump not equal relative	3	4	B8-BF
CJNE @Ri, #data, rel	Compare indirect, immediate, jump not equal relative	3	4	B6-B7
DJNZ Rn, rel	Decrement register, jump not zero relative	2	4	D8-DF
DJNZ dir, rel	Decrement direct byte, jump not zero relative	3	4	D5

MISCELLANEOUS				
Mnemonic	Description	byte	cycle	opcode
NOP	No operation	1	2	00

在上表中,如 E8-EF 中的指令操作码(十六进制)指示用于一个连续的块的 8 个不同的寄存器,寄存器编号,由其相应的操作码的最低 3 位定义。码的不连续的块,如 11-F1(举例),用于绝对跳转和调用,码的前 3 位用于指示目的地址的顶部 3 位。

电气特性

1. 最大绝对额定值 ($T_A=25^{\circ}\text{C}$)

参数	额定值	单位
电源电压	$V_{SS}-0.3 \sim V_{SS}+5.5$	V
输入电压	$V_{SS}-0.3 \sim V_{CC}+0.3$	
输出电压	$V_{SS}-0.3 \sim V_{CC}+0.3$	
每 1 引脚高电位输出电流	-25	mA
全部引脚高电位输出电流	-80	
每 1 引脚低电位输出电流	+30	
全部引脚低电位输出电流	+150	
最大工作电压	5.5	V
工作温度	$-40 \sim +105$	$^{\circ}\text{C}$
储存温度	$-65 \sim +150$	

2. DC 特性 ($T_A=25^{\circ}\text{C}$, $V_{CC}=2.5\text{V} \sim 5.5\text{V}$)

参数	符号	条件		最小值	典型值	最大值	单位
工作电压	V_{CC}	$F_{SYSCLK}=18.432\text{ MHz}$		2.5	—	5.5	V
输入高电压	V_{IH}	所有输入	$V_{CC}=5\text{V}$	$0.6V_{CC}$	—	—	V
			$V_{CC}=3\text{V}$	$0.6V_{CC}$	—	—	V
输入低电压	V_{IL}	所有输入	$V_{CC}=5\text{V}$	—	—	$0.2V_{CC}$	V
			$V_{CC}=3\text{V}$	—	—	$0.2V_{CC}$	V
I/O 端口 拉电流	I_{OH}	所有输出	$V_{CC}=5\text{V}$, $V_{OH}=0.9V_{CC}$	5.5	11	—	mA
			$V_{CC}=3\text{V}$, $V_{OH}=0.9V_{CC}$	2.5	5	—	
I/O 端口 灌电流	I_{OL}	所有输出 $HSNKxEN=1$	$V_{CC}=5\text{V}$, $V_{OL}=0.1V_{CC}$	60	80	—	mA
			$V_{CC}=3\text{V}$, $V_{OL}=0.1V_{CC}$	15	30	—	
		所有输出 $HSNKxEN=0$	$V_{CC}=5\text{V}$, $V_{OL}=0.1V_{CC}$	20	40	—	
			$V_{CC}=3\text{V}$, $V_{OL}=0.1V_{CC}$	10	20	—	
电源电流	I_{DD}	快钟 $V_{CC}=5\text{V}$	$FRC=18.432\text{ MHz}$	—	3.2	—	mA
		快钟 $V_{CC}=3\text{V}$	$FRC=18.432\text{ MHz}$	—	3	—	
		慢钟	$V_{CC}=5\text{V}$	—	190	—	μA
			$V_{CC}=3\text{V}$	—	160	—	
		空闲 $PWRSV=0$	$SRC, V_{CC}=5\text{V}$	—	150	—	
			$SRC, V_{CC}=3\text{V}$	—	135	—	
		空闲 $PWRSV=1$	$V_{CC}=5\text{V}$	—	132	—	
			$V_{CC}=3\text{V}$	—	125	—	
		暂停 $PWRSV=0$	$V_{CC}=5\text{V}$	—	60	—	
			$V_{CC}=3\text{V}$	—	50	—	
		暂停 $PWRSV=1$	$V_{CC}=5\text{V}$	—	11	—	
			$V_{CC}=3\text{V}$	—	4	—	
系统时钟频率	F_{SYSCLK}	$V_{CC} > LVR_{TH}$	$V_{CC}=2.5\text{V}$	—	—	18.432	MHz
				—	—	—	
LVR 参考电压	V_{LVR}	$T_A=25^{\circ}\text{C}$		—	4.2	—	V
				—	3.9	—	
				—	3.7	—	
				—	3.4	—	
				—	3.2	—	
				—	3.0	—	
				—	2.7	—	
LVR 滞后电压	V_{HYST}	$T_A=25^{\circ}\text{C}$		—	± 0.1	—	V
				—	± 0.1	—	

参数	符号	条件		最小值	典型值	最大值	单位
LVD 参考电压	V_{LVD}	$T_A=25^{\circ}\text{C}$		—	4.3	—	V
				—	4.2	—	
				—	4.0	—	
				—	3.9	—	
				—	3.8	—	
				—	3.7	—	
				—	3.6	—	
				—	3.4	—	
				—	3.3	—	
				—	3.2	—	
				—	3.1	—	
				—	3.0	—	
				—	2.8	—	
				—	2.7	—	
				—	2.6	—	
				—	2.5	—	
低电压 检测时间	t_{LVR}	$T_A=25^{\circ}\text{C}$		100	—	—	μs
上拉电阻	R_P	$V_{IN}=0\text{V}$	$V_{CC}=5\text{V}$	—	35	—	$\text{K}\Omega$
			$V_{CC}=3\text{V}$		55		

3. 时钟时序 ($T_A = -40^{\circ}\text{C} \sim +105^{\circ}\text{C}$)

参数	条件	最小值	典型值	最大值	单位
内部 RC 频率	25°C, $V_{CC}=5.0\text{V}$	-1%	18.432	+1%	MHz
	-40°C ~ 105°C, $V_{CC}=5.0\text{V}$	-2%	18.432	+1.5%	
	25°C, $V_{CC}=2.5\text{V} \sim 5.0\text{V}$	-2%	18.432	+1%	
	-40°C ~ 105°C, $V_{CC}=2.5 \sim 5.0\text{V}$	-5%	18.432	+2%	

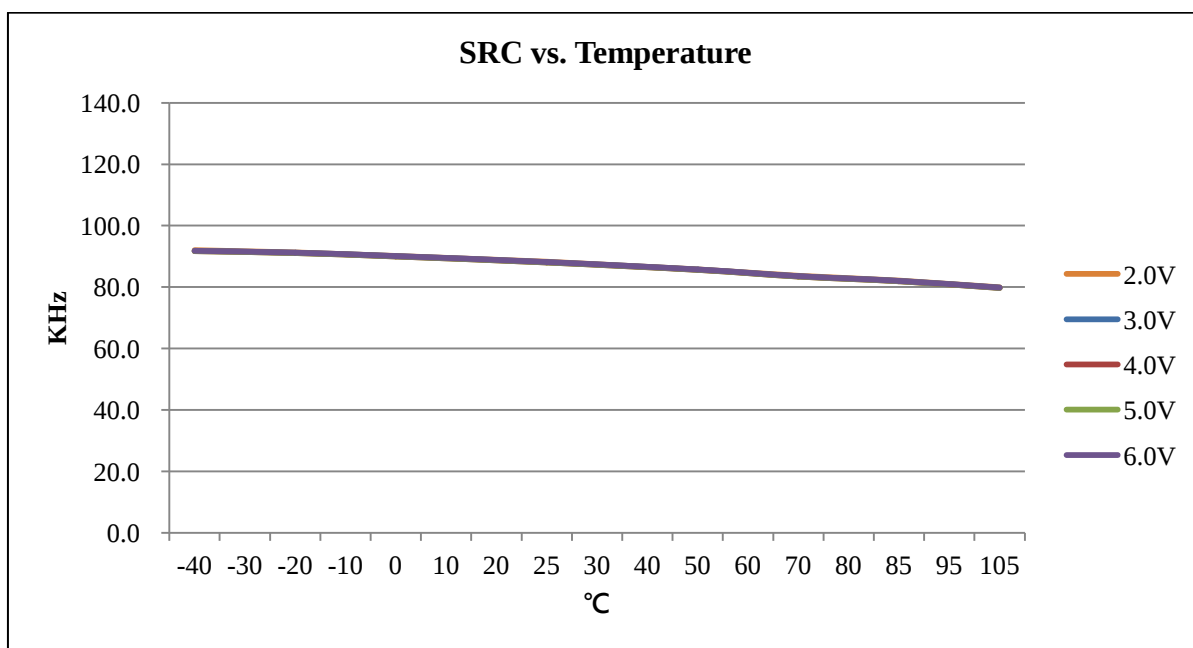
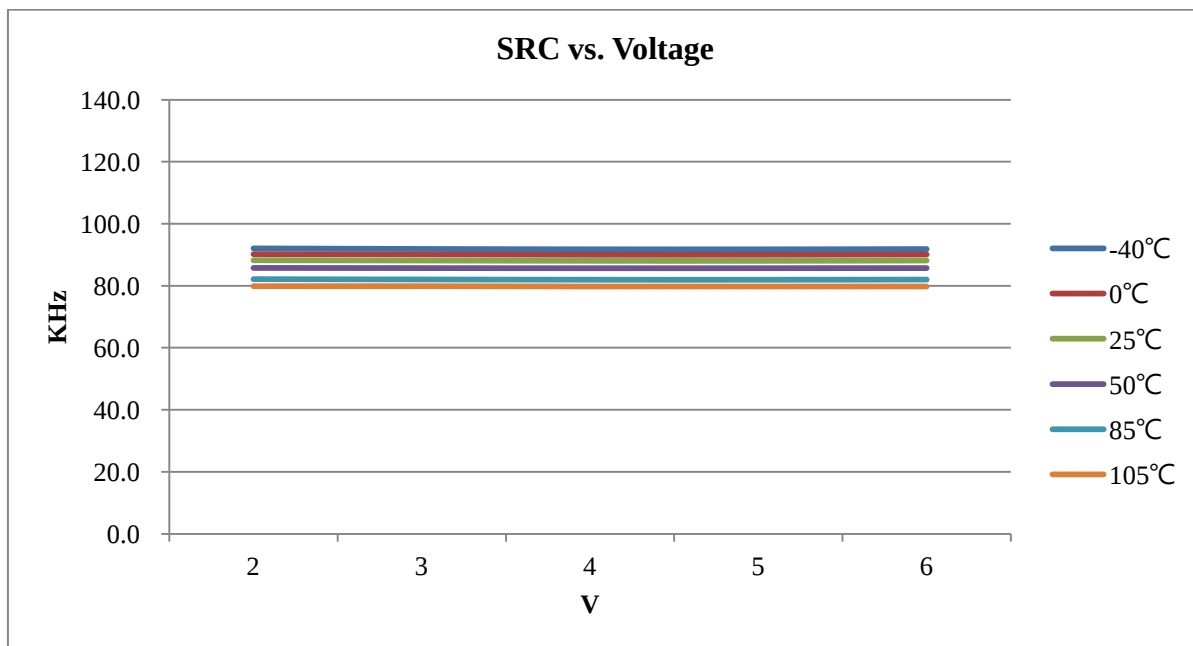
4. 复位时序特性 ($T_A = -40^{\circ}\text{C} \sim +105^{\circ}\text{C}$)

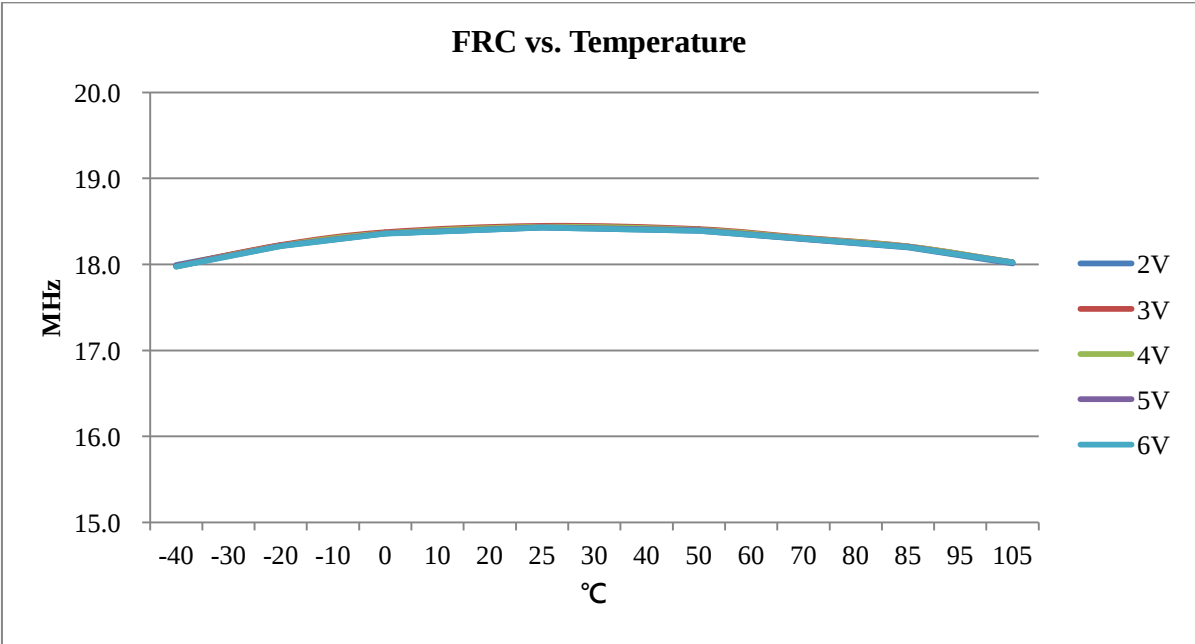
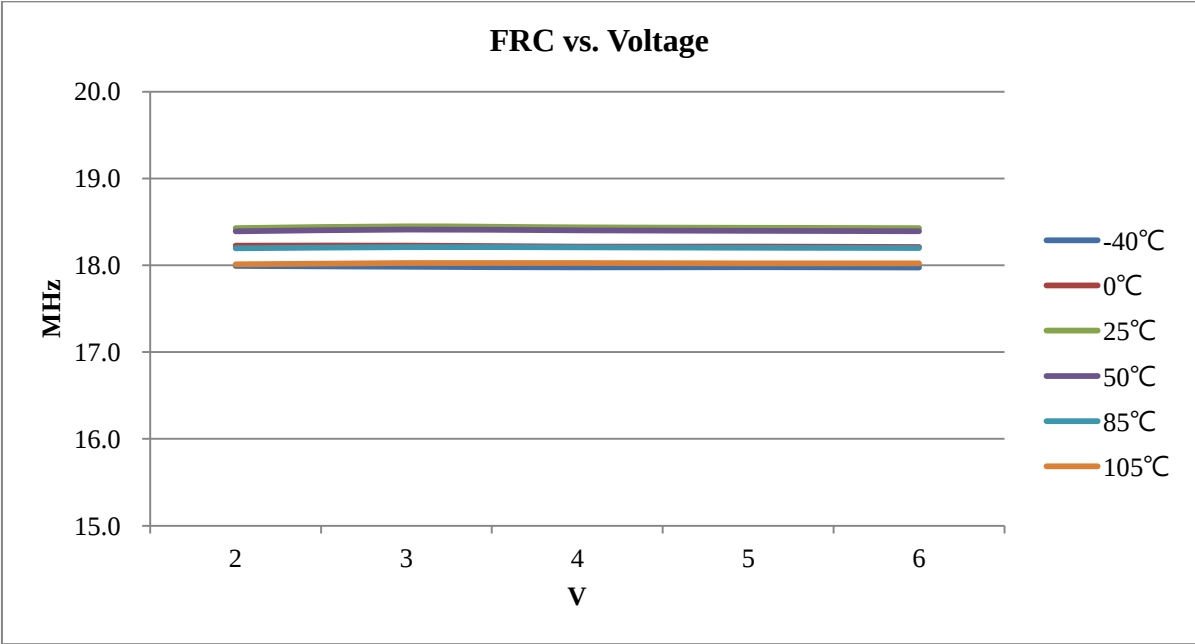
参数	条件	最小值	典型值	最大值	单位
RESET 输入低电平宽度	$V_{CC}=5.0\text{V} \pm 10\%$	30	—	—	μs
WDT 唤醒时间	$V_{CC}=5.0\text{V}$, WDT_PSC=11	—	52	—	ms
	$V_{CC}=3.0\text{V}$, WDT_PSC=11	—	57	—	

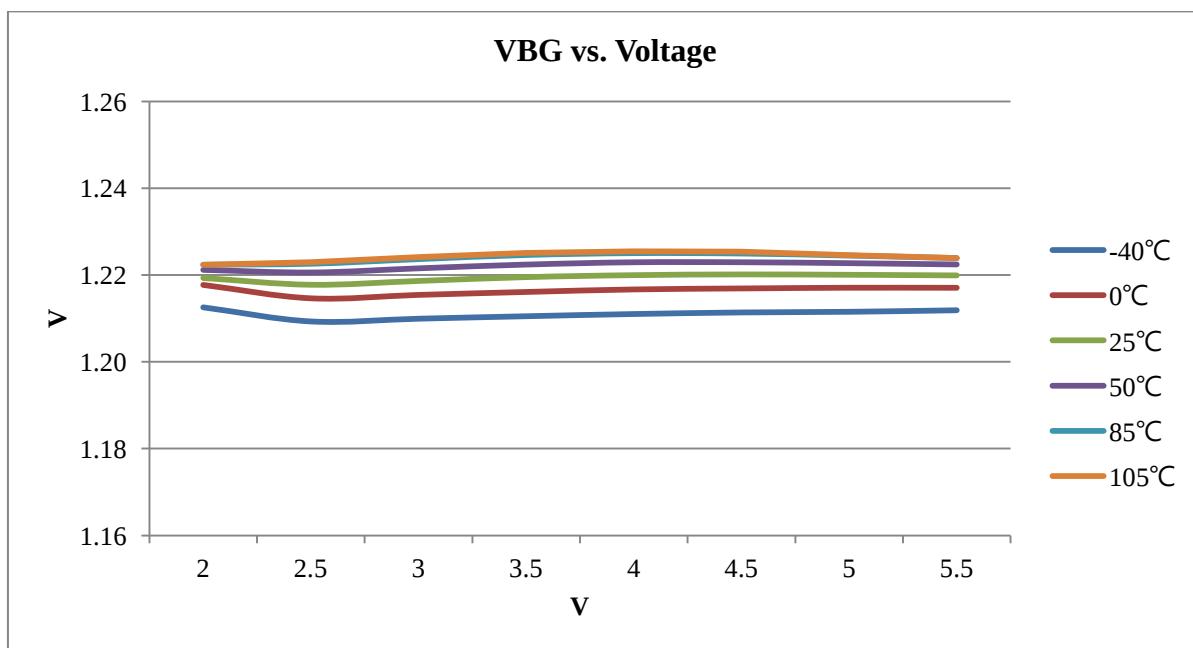
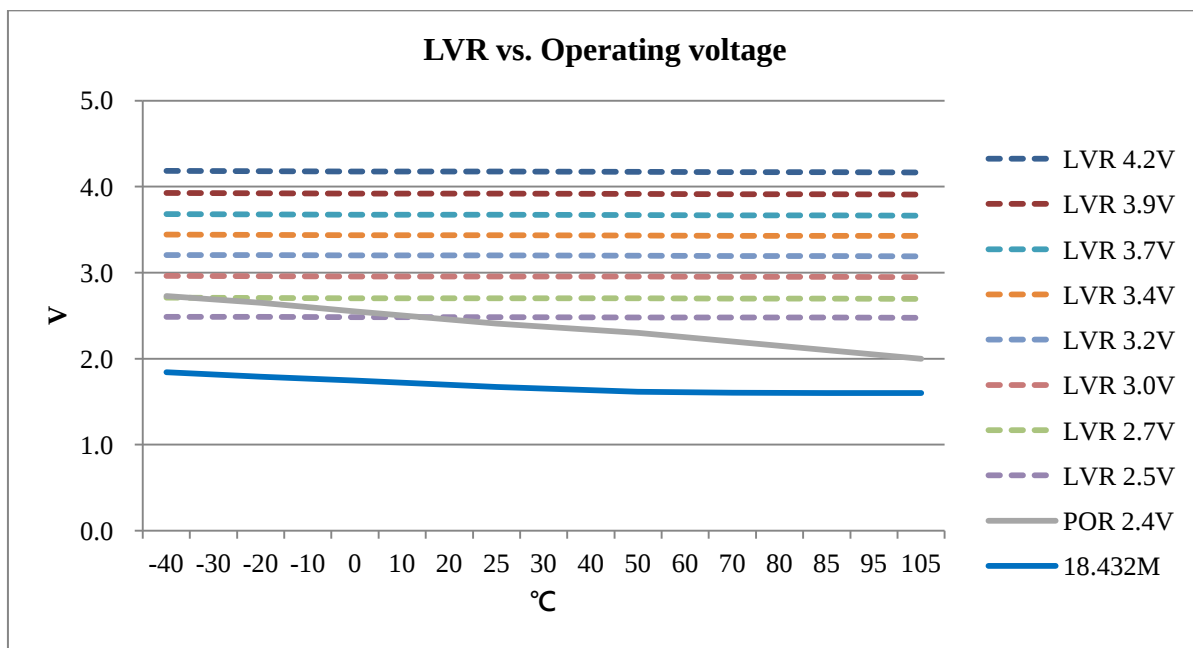
5. ADC 电气特性 ($T_A = 25^{\circ}\text{C}$, $V_{CC} = 3.0\text{V} \sim 5.5\text{V}$, $V_{SS} = 0\text{V}$)

参数	条件		最小值	典型值	最大值	单位
总绝对误差	$V_{CC}=5.12\text{V}$, $V_{SS}=0\text{V}$		—	± 2.5	± 4	LSB
积分非线性误差			—	± 3.2	± 5	
最大输入时钟 (f_{ADC})	信号驱动源阻抗 ($R_s < 10\text{K ohm}$)		—	—	2	MHz
	信号驱动源阻抗 ($R_s < 20\text{K ohm}$)		—	—	1	
	信号驱动源阻抗 ($R_s < 50\text{K ohm}$)		—	—	0.5	
	信号来号是 V_{BG} (ADCHS=1011b)		—	—	0.5	
转换时间	$F_{\text{ADC}} = 1\text{MHz}$		—	50	—	μs
带隙基准电压 (V_{BG})	—	$V_{CC}=3\text{V} \sim 5.5\text{V}$ -40°C ~ 105°C	-1.5%	1.22	+1.5%	V
ADC 参考电压 (V_{ADC})	ADCVREFS=1	$V_{CC}=3\text{V} \sim 5.5\text{V}$ 40°C ~ 105°C	-1.5%	2.5	+1.5%	
VCC / 4 基准电压 ($V_{1/4}$)	—	$V_{CC}=5\text{V}$, 25°C	-0.8%	1.26	+0.8%	
		$V_{CC}=3.6\text{V}$, 25°C	-0.8%	0.907	+0.8%	
输入电压	—		V_{SS}	—	V_{CC}	

6. 特性曲线图







封装说明

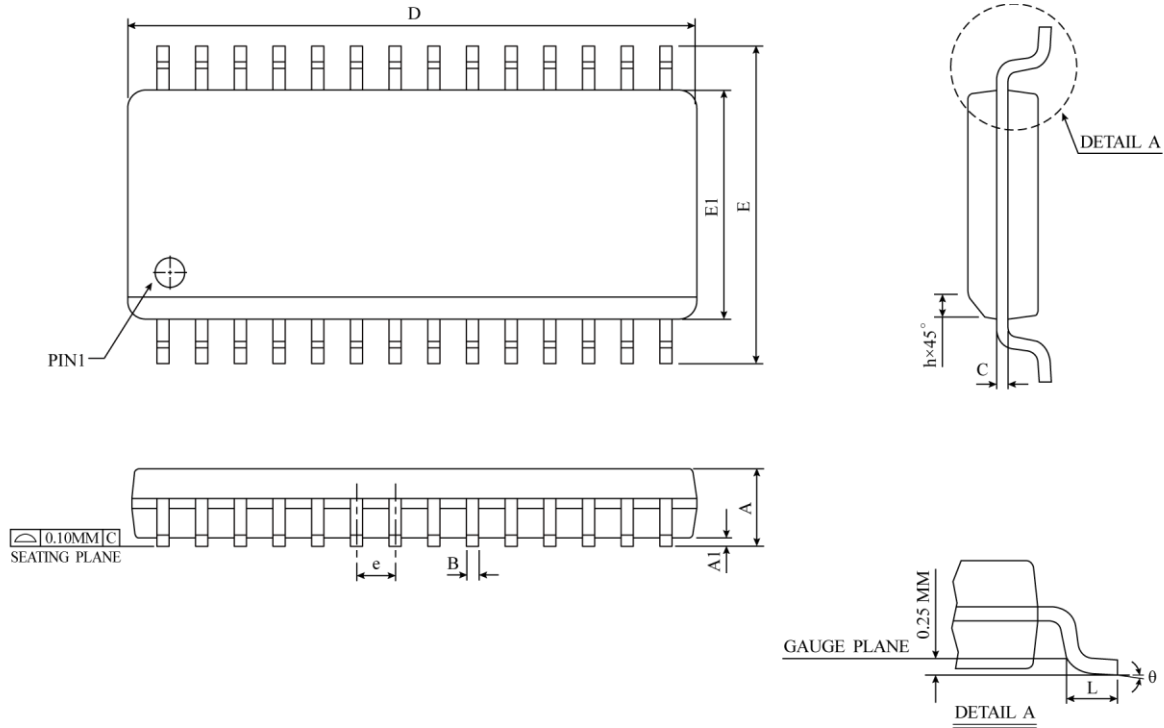
请注意，此处提供的包装信息仅供参考。由于此信息经常更新，因此用户可以联系销售人员以咨询最新的包装信息和库存。

订购须知

Ordering number	Package
TM52eF1375A-MTP	Wafer/Dice blank chip
TM52eF1375A-COD	Wafer/Dice with code
TM52eF1375A-MTP-23	SOP 28-pin (300 mil)
TM52eF1375A-MTP-21	SOP 20-pin (300 mil)
TM52eF1375A-MTP-16	SOP 16-pin (150 mil)

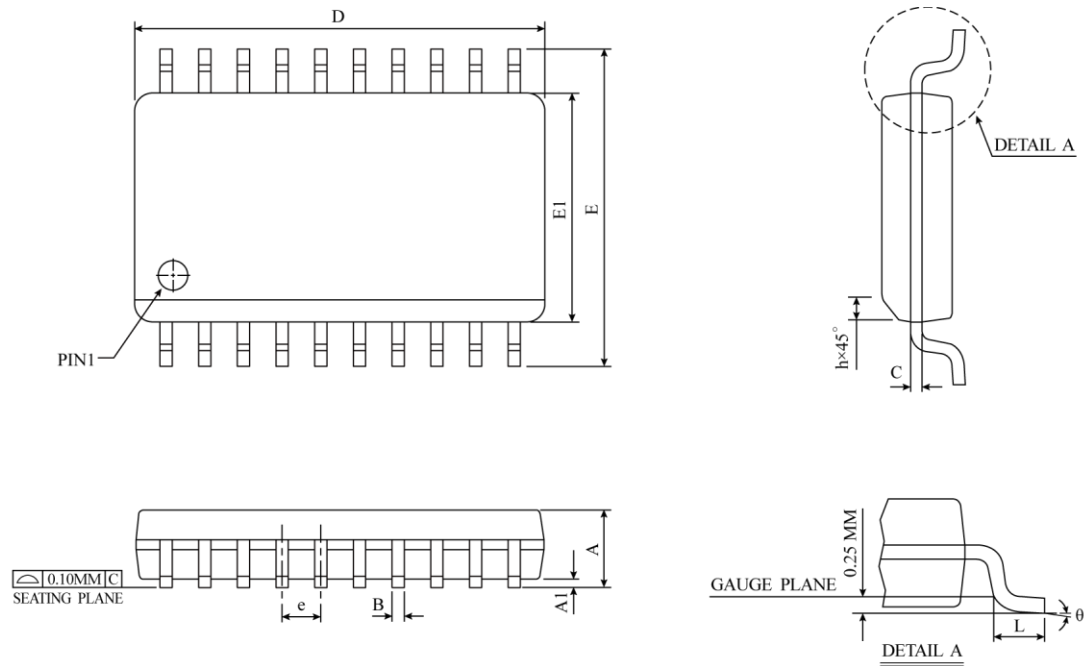
包装信息

SOP-28 引脚（300mil）包装尺寸



SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.35	2.50	2.65	0.0926	0.0985	0.1043
A1	0.10	0.20	0.30	0.0040	0.0079	0.0118
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.23	0.28	0.32	0.0091	0.0108	0.0125
D	17.70	17.90	18.10	0.6969	0.7047	0.7125
E	10.00	10.33	10.65	0.3940	0.4425	0.4910
E1	7.40	7.50	7.60	0.2914	0.2953	0.2992
e	1.27 BSC			0.050 BSC		
h	0.25	0.50	0.75	0.0100	0.0195	0.0290
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-013 (AE)					

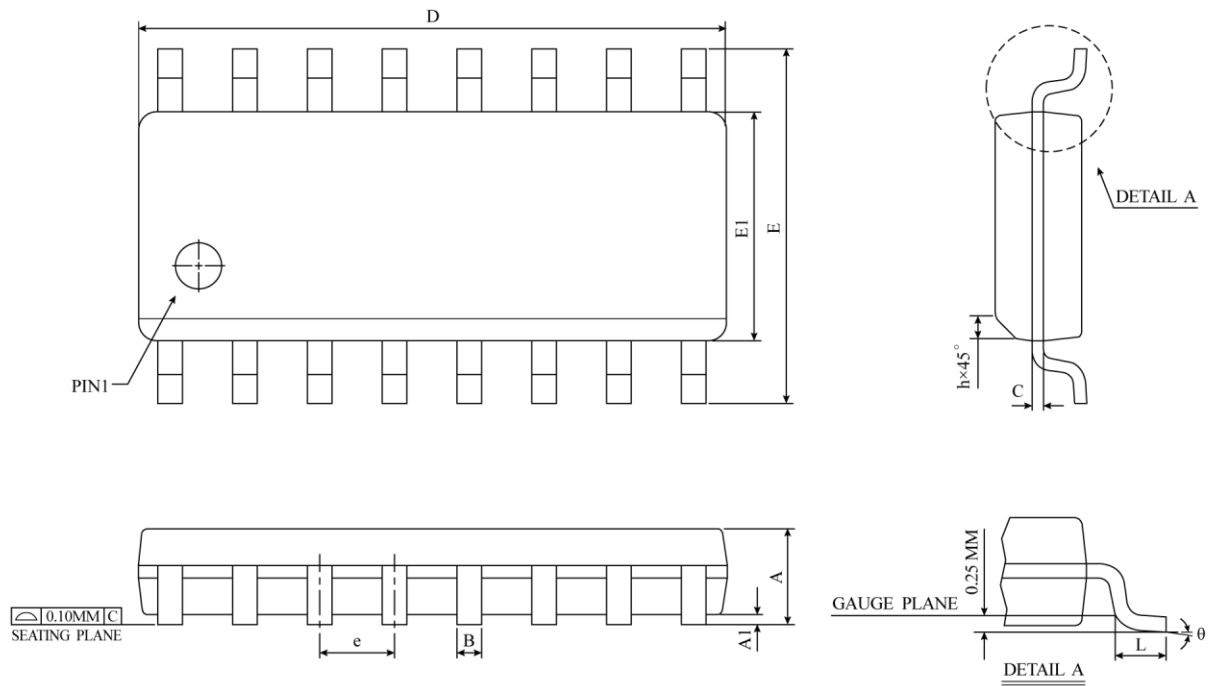
△ * NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.

SOP-20 引脚 (300mil) 包装尺寸


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.35	2.50	2.65	0.0926	0.0985	0.1043
A1	0.10	0.20	0.30	0.0040	0.0079	0.0118
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.23	0.28	0.32	0.0091	0.0108	0.0125
D	12.60	12.80	13.00	0.4961	0.5040	0.5118
E	10.00	10.33	10.65	0.3940	0.4425	0.4910
E1	7.40	7.50	7.60	0.2914	0.2953	0.2992
e	1.27 BSC			0.050 BSC		
h	0.25	0.50	0.75	0.0100	0.0195	0.0290
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-013 (AC)					

△ * NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.

SOP-16 引脚 (150mil) 包装尺寸



SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	1.35	1.55	1.75	0.0532	0.0610	0.0688
A1	0.10	0.18	0.25	0.0040	0.0069	0.0098
B	0.33	0.42	0.51	0.0130	0.0165	0.0200
C	0.19	0.22	0.25	0.0075	0.0087	0.0098
D	9.80	9.90	10.00	0.3859	0.3898	0.3937
E	5.80	6.00	6.20	0.2284	0.2362	0.2440
E1	3.80	3.90	4.00	0.1497	0.1536	0.1574
e	1.27 BSC			0.050 BSC		
h	0.25	0.38	0.50	0.0099	0.0148	0.0196
L	0.40	0.84	1.27	0.0160	0.0330	0.0500
θ	0°	4°	8°	0°	4°	8°
JEDEC	MS-012 (AC)					

△ * NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.