



十速

**TM52eF0C86/85**

**规格书**

**Rev 0.91**

**(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. 如果 INTn 引脚为低电平并且使能了唤醒功能，则芯片无法进入停止/暂停模式。（INTn = 0 和 EXn = 1, n = 0~1）
2. 如果 PCIF (POIF~P5IF) 為高電平，芯片不能進入停止/暫停模式。
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	Oct, 2022	新颁。
V0.91	Oct, 2022	1.工作溫度。 2. IAP 寫入次數。 3.UART 描述。

# 目录

使用注意事项.....	2
修改纪录.....	3
TM52 系列 eF0xxx 家族.....	7
概述.....	8
系统框图.....	8
基本功能.....	9
IC 引脚图.....	13
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 程序和 Information 存储器.....	20
2.2 IAP 启动升级.....	21
2.3 IAP 操作.....	23
2.4 FLASH ICP 模式.....	26
2.5 数据存储器.....	28
2.5.1 IRAM.....	29
2.5.2 XRAM.....	29
2.5.3 SFRs.....	29
3. 低电压复位和低电压检测.....	31
4. 复位.....	33
4.1 上电复位.....	33
4.2 外部引脚复位.....	33
4.3 软件复位.....	33
4.4 看门狗定时器复位.....	33
4.5 低电压复位.....	33
5. 时钟电路和工作模式.....	35

5.1 时钟电路 .....	35
5.2 操作模式 .....	37
6. 中断和唤醒.....	39
6.1 中断使能和优先级控制 .....	39
6.2 关于中断子程序的建议 .....	39
6.3 引脚中断和 LVD 中断 .....	43
6.4 空闲模式唤醒和中断 .....	46
6.5 停止/暂停模式唤醒和中断 .....	46
7. I/O 端口 .....	48
7.1 端口 0~端口 5.....	48
8. 定时器.....	54
8.1 Timer0/1 .....	54
8.2 Timer2 .....	56
8.3 Timer3 .....	58
8.4 T0O 和 T2O 输出控制 .....	58
9. UARTs .....	59
10. PWMs .....	63
11. ADC .....	70
11.1 ADC 通道.....	71
11.2 ADC 转换时间.....	71
12. 触摸按键(FTK) .....	73
13. LCD / LED 控制器/驱动器.....	86
13.1 LED 矩阵模式 (MX, Matrix) .....	86
13.2 LED 点矩阵模式 (DMX, Dot Matrix) .....	87
13.3 LCD 模式 .....	91
14. 串行外围设备接口 (SPI) .....	97
15. 循环冗余校验码(CRC).....	102
16. 乘法器和除法器.....	103
17. 主 I <sup>2</sup> C 接口 .....	105
18. 从机 I <sup>2</sup> C 接口 .....	108
19. 在线仿真器(ICE)模式 .....	111
<b>SFR &amp; CFGW 映像 .....</b>	<b>112</b>
<b>SFR &amp; CFGW 说明 .....</b>	<b>114</b>
<b>指令集.....</b>	<b>134</b>

电气特性.....	137
1. 最大绝对额定值.....	137
2. DC 特性.....	138
3. 时钟时序.....	140
4. 复位时序特性.....	140
5. ADC 电气特性.....	140
3. 特性曲线图.....	141
封装说明.....	144

**TM52 系列 eF0xxx 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)	4K~32K 有 IAP, ISP,ICP	256 ~ 1024	SXT SRC FXT FRC	快钟 慢钟 空闲 停止 暂停	8051标准		15-bit	16 阶	8 阶

注:IAP,ISP 只针对 Flash 类型的程序存储器

**家族成员特性**

P/N	程序存储器	RAM 字节	IO 引脚	PWM	SAR ADC	触摸按键	LCD	LED	串口
TM52-eF1716 TM52-eF1732	Flash 16KB 32KB	1280	30	16-bit x3 8-bit x3	12-bit 16-ch	20-ch	8com	BiD 4Cx6S	SPI UARTx2 I <sup>2</sup> C
TM52-eF1374 TM52-eF1375	Flash 16KB 32KB	1280	26	16-bit x3	12-bit 16-ch	20-ch	8com	BiD 4Cx6S DMX 8x8	SPI UARTx2 I <sup>2</sup> C
TM52-eF0C85 TM52-eF0C86	Flash 32KB 64KB	4352	42	16-bit x9	12-bit 44-ch	21-ch x 2	4Cx20S ~ 8Cx16S	MX 8x8 DMX 7x8	SPI UARTx3 I <sup>2</sup> C

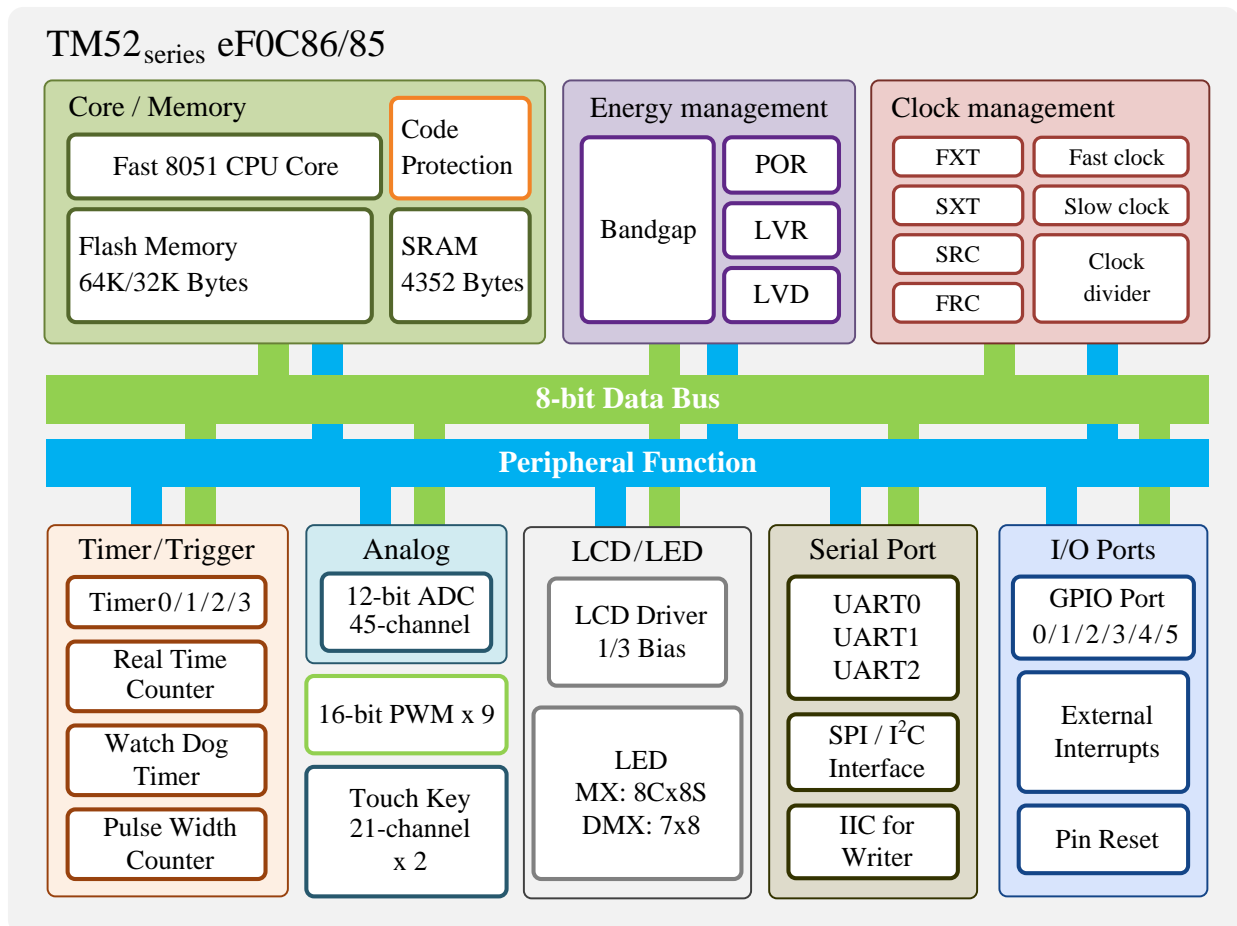
P/N	工作电压	工作电流					最大系统时钟(Hz)			
		快钟 FRC	慢钟 SRC	空闲 SRC	停止	暂停	SXT	SRC	FXT	FRC
TM52-F1716 TM52-F1732	2.5~5.5V	3.5mA	0.18mA	0.15 mA	7uA@5V 1.4uA@3V	11uA@5V 4uA@3V	32K	80K	16M	14.7456M
TM52-eF1374 TM52-eF1375	2.3~5.5V	4mA	0.22mA	0.2mA	10uA@5V 4uA@3V	13uA@5V 6uA@3V	32K	80K	18M	18.432M
TM52-eF0C85 TM52-eF0C86	2.3~5.5V	3.5mA	0.2mA	0.18mA	11uA@5V 4uA@3V	14uA@5V 6uA@3V	32K	80K	16M	18.432M

## 概述

TM52系列eF0C86/85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-eF0C86/85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64K/32K字节的Flash程序存储器,4352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,9组16位脉冲宽度调制器,45通道的12位模数转换器(ADC),2组21通道触摸按键,I<sup>2</sup>C/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





## 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
  - 64K 字节 FLASH 程序存储器(TM52eF0C86)
  - 32K 字节 FLASH 程序存储器(TM52eF0C85)
  - 支持 IAP
  - 程序码保护功能
  - 最低擦除 100K 次
  - 最低 10 年数据保留
3. 总计 4352 字节 SRAM(IRAM+XRAM)
  - 256 字节 IRAM 在 8051 内部数据存储器区
  - 4096 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 1~16 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
  - 时钟源可选择慢时钟、FRC/512、慢时钟/2 或 FRC/1024
  - 中断期可选时钟除以 65536/16384/4096/1024 选项
7. UARTs
  - UART0, 8051 标准 UART
  - UART1, 仅支持 mode1 和 mode3
  - UART2, 仅支持 mode1 和 mode3

## 8. 九个 16 位的 PWM 有周期调整

- 三个 16 位 PWM，具有独立的周期调整 (PWM0~PWM2)
- 六个 16 位 PWM，具有共享的周期调整 (PWM30~PWM35)
- PWM2 带死区控制

## 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. 2 组 21 通道触摸键（FTK）

## 12. 12 位 ADC,具有 42 个通道的外部引脚输入和 2 通道内部参考电压

- 内部基准电压源 (VBG): 1.22V @  $V_{CC}=5V\sim 3V$ , 25°C
- 内部基准电压源:  $1/4V_{CC}$

## 13. LCD 控制器/驱动器

- 4x20~8x16 LCD 驱动器
- 1/3 LCD 偏压

## 14. LED 控制器/驱动器

- 矩阵模式 (MX) : 8 \* 8, 16 引脚, 最多 64 点
- 点矩阵模式 (DMX) : 7 \* 8、6 \* 7、5 \* 6、4 \* 5, 8 引脚, 最高 56 点

## 15. 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
- 端口 0/1/2/3/4/5 引脚电平变化中断
- UART0/UART1/UART2 TX/RX 中断
- ADC /触摸键中断
- SPI 中断
- I<sup>2</sup>C 中断
- PWM0/PWM1/PWM2/PWM3 中断

## 16. 引脚中断能将停止/暂停模式下的 CPU 唤醒

- 每个引脚可以定义为中断和唤醒引脚（通过引脚更改）

## 17. 最大 42 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（60mA @ VCC = 5V, VOL = 0.1VCC）
- LED 模式下 COM 引脚均具有高灌电流（120mA @ VCC = 5V, VOL = 0.1VCC）

## 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 级低电压检测

- 4.38V/4.3V/4.14V/4.06V/3.9V/3.82V/3.66V/3.58V/  
3.42V/3.34V/3.18V/3.1V/2.94V/2.86V/2.7V/2.62V

## 21. 8 级低电压复位

- 3.92V/3.68V/3.44V/3.20V/2.96V/2.72V/2.48V/2.24V

## 22. 5 种电源工作模式

- 快钟模式/慢钟模式/空闲模式/停止模式/暂停模式

## 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, P3.4/P3.5 引脚, 或 P0.2/P0.3 引脚
- 与 ICP 编程引脚共享

## 26. 烧录器接口

- 使用 P3.0/P3.1 引脚

**27. 工作电压**

- $V_{CC} = 2.3V \sim 5.5V$  @FSYSCLK=18.432MHz
- $I_{CC} < 10\mu A$  @Vcc=3V, Halt mode, LCD 启用, 0.5S 唤醒(时计电流)
- $I_{CC} = 11\mu A$  @Stop mode, VCC=5V
- $I_{CC} = 4\mu A$  @Stop mode, V<sub>CC</sub>=3V
- $I_{CC} = 180\mu A$  @Idle mode, V<sub>CC</sub>=5V

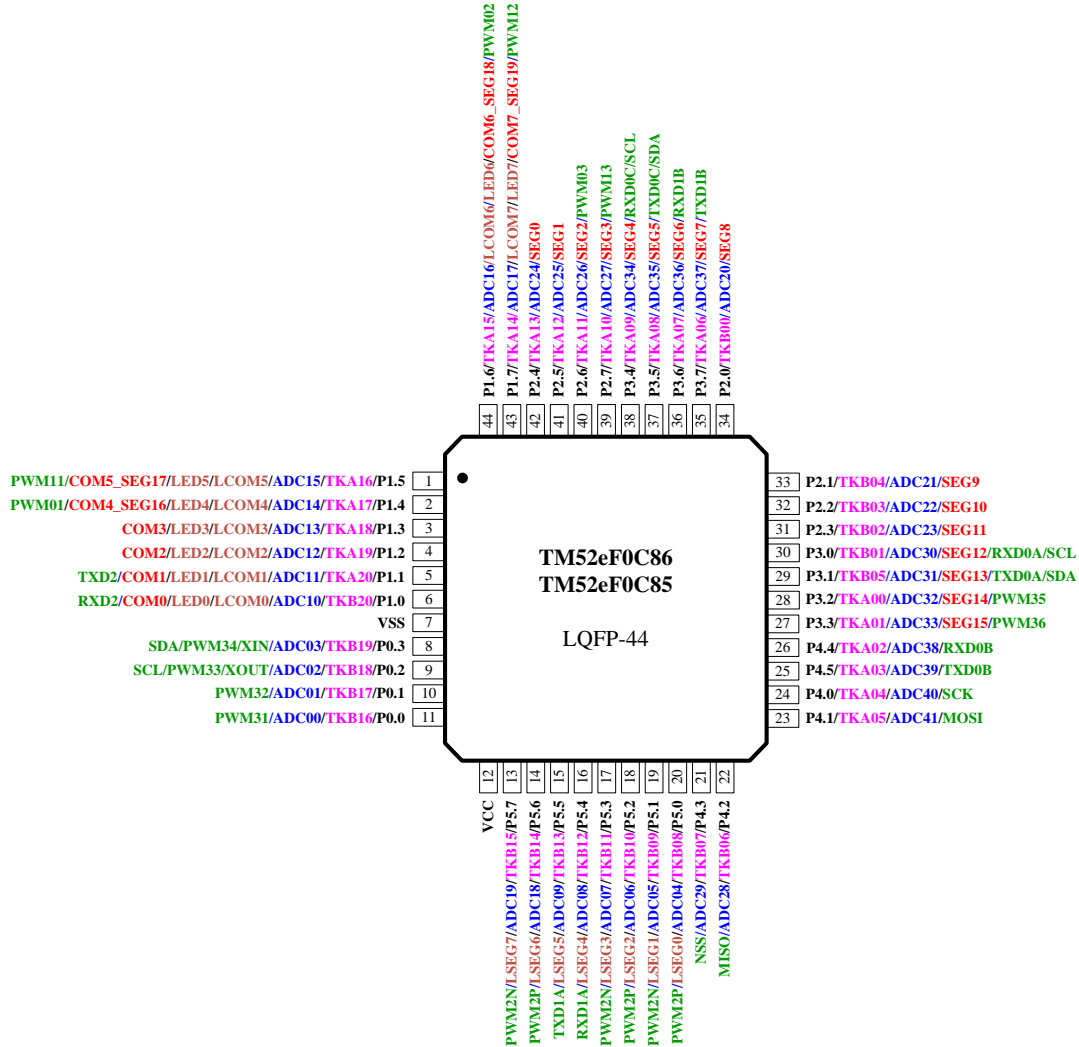
**28. 工作温度范围**

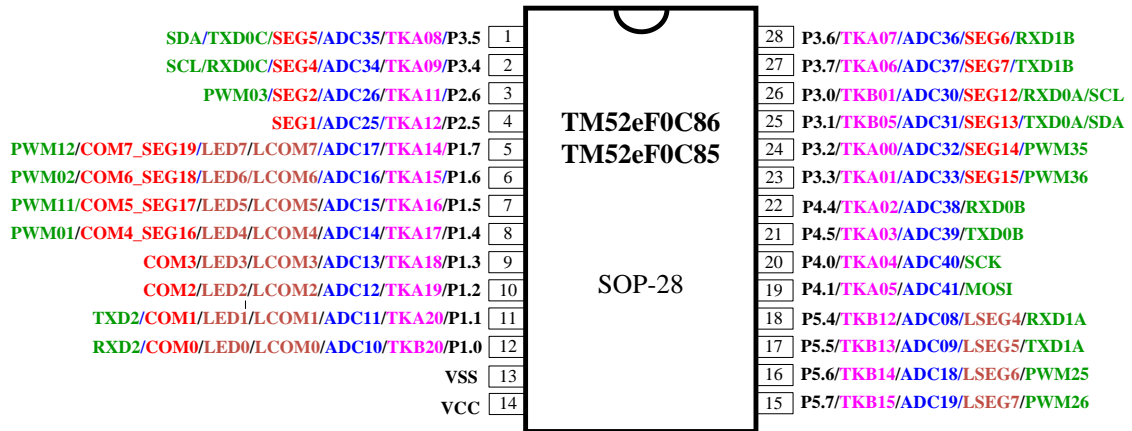
- $-40^{\circ}C \sim +105^{\circ}C$

**29. 封装类型**

- 44-pin LQFP
- SOP-28

IC 引脚图





## 引脚描述

Name	In/Out	Pin Description
P0/P1/P2/P3/P4/P5	I/O	施密特触发器输入, CMOS 推挽输出或“漏极开路”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/停止/暂停模式唤醒输入
RXD0A, TXD0A RXD0B, TXD0B RXD0C, TXD0C	I/O	UART0 接收和发送数据引脚
RXD1A, TXD1A RXD1B, TXD1B	I/O	UART1 接收和发送数据引脚
RXD2, TXD2	I/O	UART2 接收和发送数据引脚
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
VBGO	O	带隙电压输出
PWM0 PWM1 PWM2P, PWM2N PWM30~PWM35	O	16 位 PWM 输出
AD0~AD41	I	ADC 输入
TKA00~TKA20 TKB00~TKB20	I	触摸键输入
XCAPA, XCAPB	I	触摸按键模块 A/B 电荷收集电容器连接引脚
COM0~COM7	O	LCD COM 输出
SEG0~SEG19	O	LCD segment 输出
LCOM0~LCOM7	O	LED 矩阵模式 COM 输出
LSEG0~LSEG7	O	LED 矩阵模式 segment 输出
LED0~LED7	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 <sup>2</sup> C SCL
SDA	I/O	I <sup>2</sup> 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	唤醒	外部中断	LCD	LED 矩阵	LED 点矩阵	ADC	触摸按键	UART	PWM	SPI	I <sup>2</sup> C	Other
LQFP-44															
1	P15	I/O	Hi-Z	•	•	•	•	•	•	•	•	•			
2	P14	I/O	Hi-Z	•	•	•	•	•	•	•		•			
3	P13	I/O	Hi-Z	•	•	•	•	•	•	•					
4	P13	I/O	Hi-Z	•	•	•	•	•	•	•					
5	P11	I/O	Hi-Z	•	•	•	•	•	•	•	•				
6	P10	I/O	Hi-Z	•	•	•	•	•	•	•	•				T2O
7	VSS	P													
8	P03	I/O	Hi-Z	•	•				•	•		•		•	Crystal
9	P02	I/O	Hi-Z	•	•				•	•		•		•	Crystal
10	P01	I/O	Hi-Z	•	•				•	•		•			
11	P00	I/O	Hi-Z	•	•				•	•		•			
12	VCC	P													
13	P57	I/O	Hi-Z	•	•		•		•	•		•			Reset
14	P56	I/O	Hi-Z	•	•		•		•	•		•			
15	P55	I/O	Hi-Z	•	•		•		•	•	•				
16	P54	I/O	Hi-Z	•	•		•		•	•	•				
17	P53	I/O	Hi-Z	•	•		•		•	•		•			
18	P52	I/O	Hi-Z	•	•		•		•	•		•			
19	P51	I/O	Hi-Z	•	•		•		•	•		•			
20	P50	I/O	Hi-Z	•	•		•		•	•		•			
21	P43	I/O	Hi-Z	•	•				•	•			•		
22	P42	I/O	Hi-Z	•	•				•	•			•		
23	P41	I/O	Hi-Z	•	•				•	•			•		
24	P40	I/O	Hi-Z	•	•				•	•			•		
25	P45	I/O	Hi-Z	•	•				•	•	•				
26	P44	I/O	Hi-Z	•	•				•	•	•				
27	P33	I/O	Hi-Z	•	•	•			•	•		•			INT1
28	P32	I/O	Hi-Z	•	•	•			•	•		•			INT0
29	P31	I/O	Hi-Z	•	•	•			•	•	•			•	
30	P30	I/O	Hi-Z	•	•	•			•	•	•			•	
31	P23	I/O	Hi-Z	•	•	•			•	•					
32	P22	I/O	Hi-Z	•	•	•			•	•					
33	P21	I/O	Hi-Z	•	•	•			•	•					
34	P20	I/O	Hi-Z	•	•	•			•	•					
35	P37	I/O	Hi-Z	•	•	•			•	•	•				
36	P36	I/O	Hi-Z	•	•	•			•	•	•				
37	P35	I/O	Hi-Z	•	•	•			•	•	•			•	
38	P34	I/O	Hi-Z	•	•	•			•	•	•			•	T0O
39	P27	I/O	Hi-Z	•	•	•			•	•		•			
40	P26	I/O	Hi-Z	•	•	•			•	•		•			
41	P25	I/O	Hi-Z	•	•	•			•	•					
42	P24	I/O	Hi-Z	•	•	•			•	•					
43	P17	I/O	Hi-Z	•	•	•	•	•	•	•		•			
44	P16	I/O	Hi-Z	•	•	•	•	•	•	•		•			



## 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
<b>ACC</b>	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</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
<b>SP</b>	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
<b>DPL</b>	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
<b>DPH</b>	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
<b>AUX1</b>	CLRWDT	CLRTM3	TKSOC	ADSOC	–	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.0 **DPSEL**:活动 DPTR 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			指令	标志		
	C	OV	AC		C	OV	AC
ADD	X	X	X	CLR C	0		
ADDC	X	X	X	CPL C	X		
SUBB	X	X	X	ANL C,bit	X		
MUL	0	X		ANL C,/bit	X		
DIV	0	X		ORL C,bit	X		
DA	X			ORL C,/bit	X		
RRC	X			MOV C,bit	X		
RLC	X			CJNE	X		
SETB C	1						

“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)

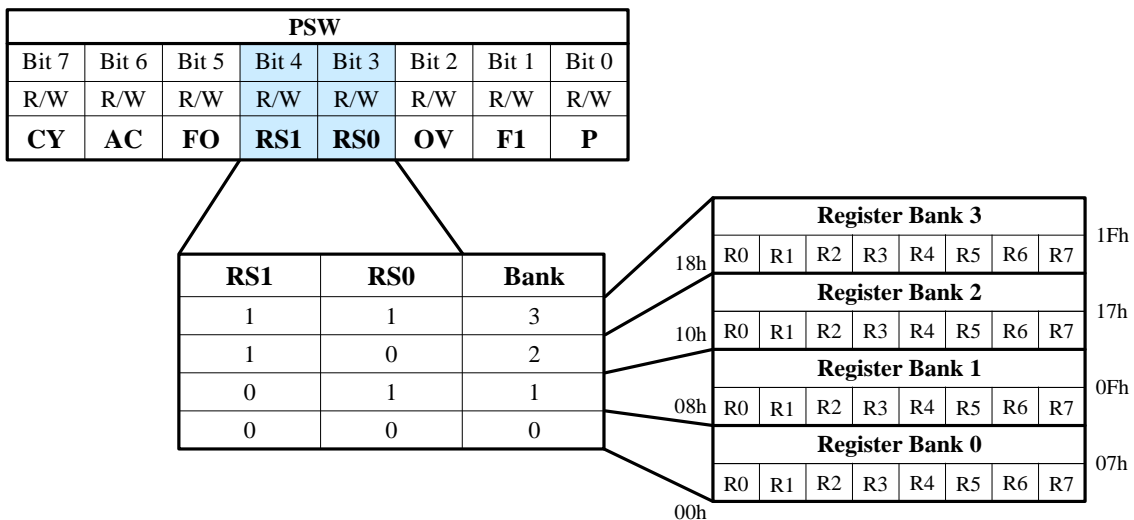
10:存储区 2(10h~17h)

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

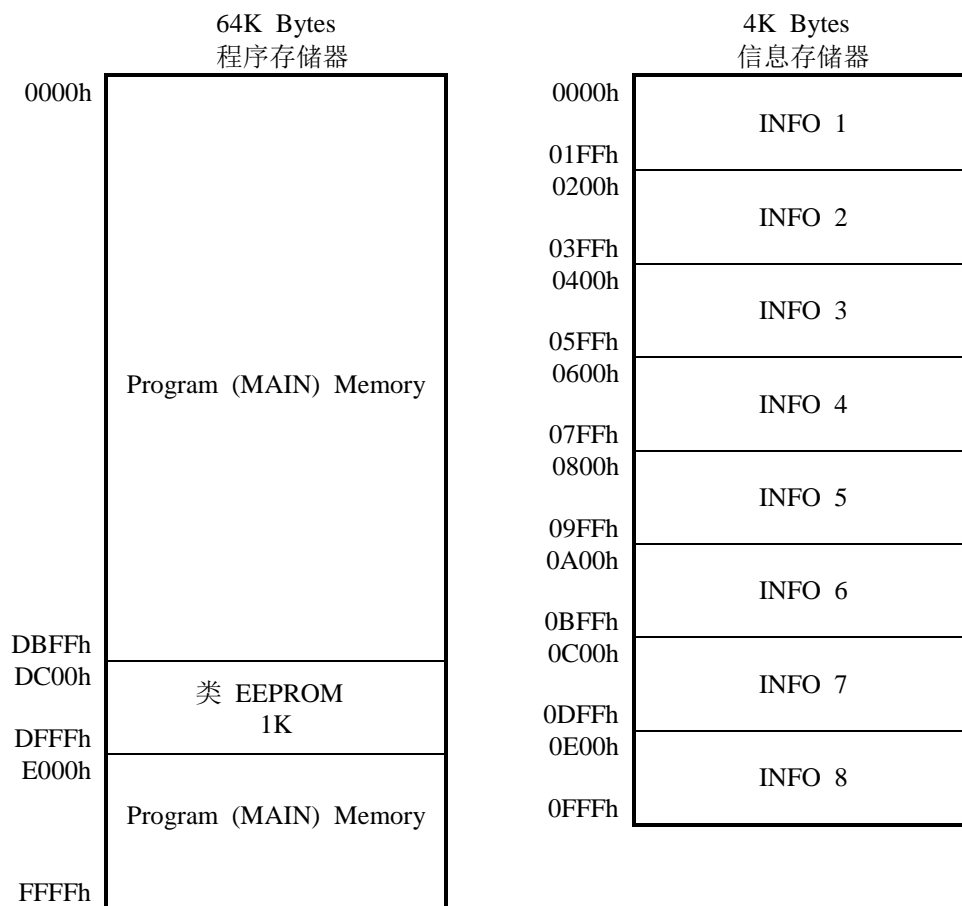
D0h.0 **P**:奇偶标志。由硬件于每个指令周期设置/清零来表示在累加器“1”位之奇/偶数。



## 2. 存储器

### 2.1 程序和 Information 存储器

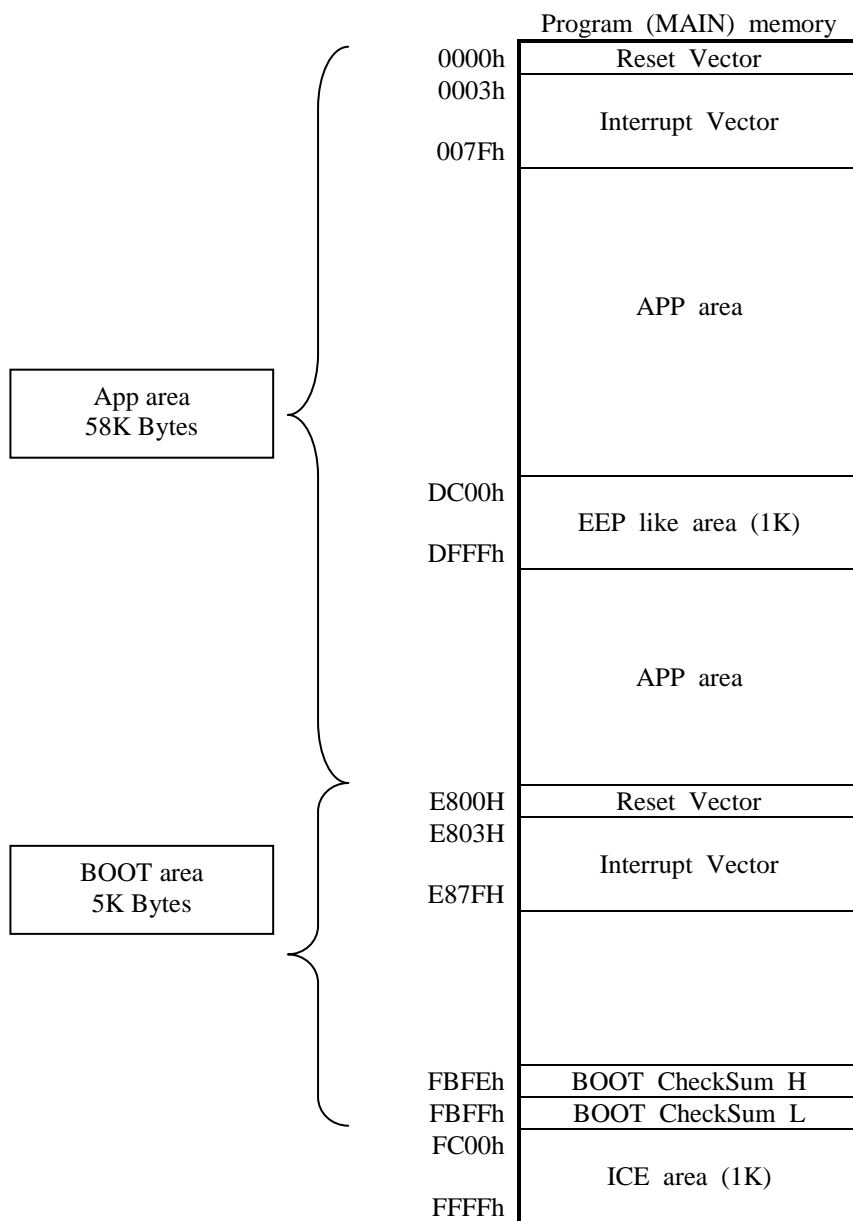
该芯片有一个 64K Bytes 的程序存储器,可分为 128 页和 4K Byte 的信息存储器,可分为 8 页。类 EEPROM 区域 (DC00h~DFFFh) 为 1024 字节。对于程序存储器和信息存储器,一页为 512Bytes。Flash 仅提供页面擦除和字节写入功能。如果有使用 ICE 的功能, FC00h~FFFFh 为 ICE 保留区域。信息存储器 INFO1~4 为系统保留, INFO5~8 可供用户擦除和写入。程序存储器区域 READ 只需通过“MOVC”指令即可实现。信息存储器区域(INFO)的读取可以通过“MOVX A,@DPTR”指令简单实现。



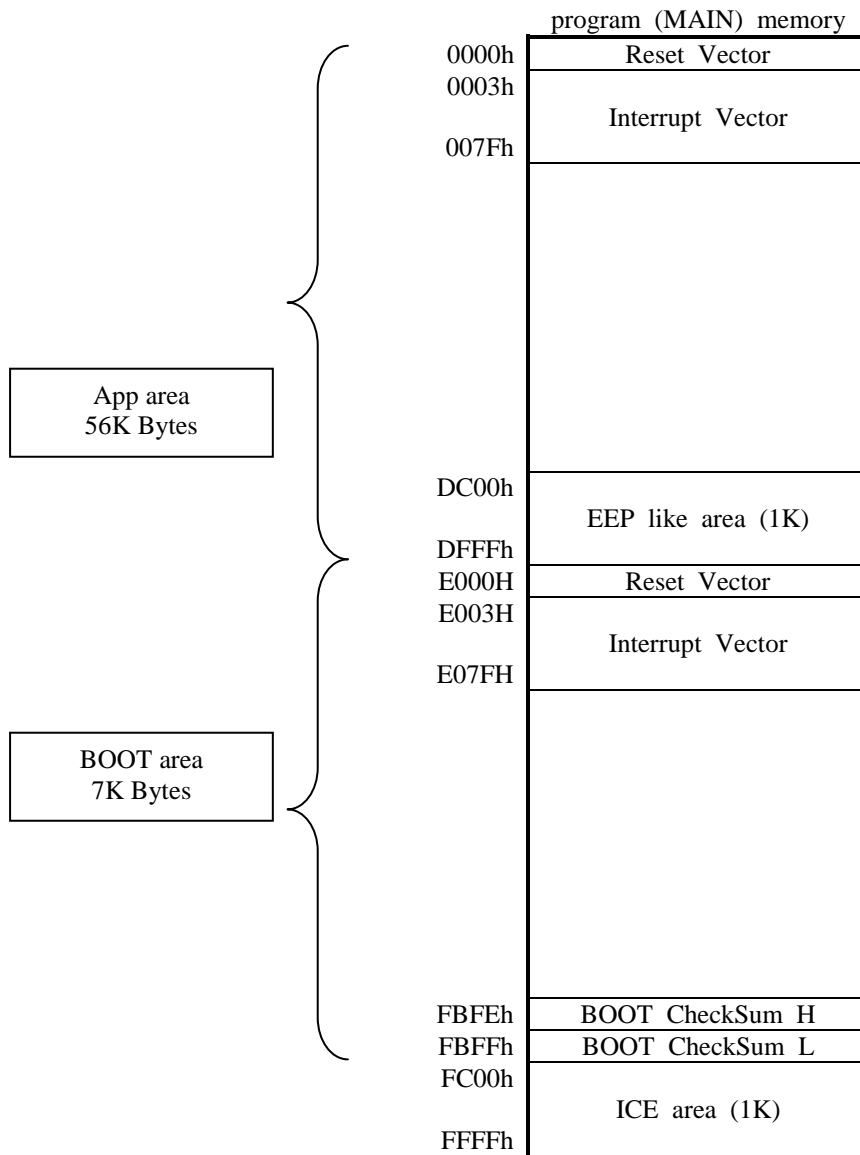
## 2.2 IAP 启动升级

Flash 支持 IAP BOOT 升级功能。此时可将程序存储器分为 BOOT 区与 APP 区,BOOT 区存放被保护的更新程序代码,APP 区则存放可被改写的程序代码。设置 SFR RSTV 以决定复位后的复位/中断向量(软件复位、WDT 复位或引脚复位)。BOOT 区有自己的存储写保护,BOOT 区的大小通过设置 CFGWH[1:0]来配置,选择 5K、7K 或无 BOOT 区。

如果 CFGWH[1:0]=00b, 未设置 BOOT 区域, 则上电复位/中断向量为 0000h。  
 如果 CFGWH[1:0]=10b, 设置 7K BOOT 区域, 则上电复位/中断向量为 E000h。  
 如果 CFGWH[1:0]=11b, 设置 5K BOOT 区域, 则上电复位/中断向量为 E800h。



64K Bytes Program (MAIN) memory with 5K boot area



64K Bytes Program (MAIN) memory with 7K boot area

### 2.3 IAP 操作

eF1386 具有“应用程序中” IAP 功能，允许软件在 CPU 运行期间擦除/写入数据到闪存。使用 IAP 写入时，需要先擦除再写入字节。擦除后，每个地址只能写入一次。Flash 仅提供页面擦除和字节写入功能。在写入 IAP 之前，用户应先关闭 LVR。

闪存 IAP 写入只需通过“MOVX @DPTR, A”指令即可实现。闪存 IAP 写入需要更高的 VCC 电压，VCC>2.5V。

当 IAP 写/擦除时，SFR IAPWE/SWCMD 需要遵循以下设置

#### SFR IAPWE

写入 E2h 和 4Ch 以启用 MAIN（类 EEP / APP）区域字节写入

写入 E2h 和 BAh 以启用 MAIN（类 EEP / APP）区域页擦除

写入 A1h 和 4Ch 使能 INFO5~8 区域字节写入

写入 A1h 和 BAh 使能 INFO5~8 区域页擦除

#### SFR SWCMD

写入 65h 和 A7h 以启用 IAP MAIN (APP) 区域写入/擦除

写/擦除范围	類 EEP 區(MAIN)		APP 區 (when Boot area 7K)		APP 區 (when Boot area 5K)		信息存储器 INFO 5~8	
	写	擦除	写	擦除	写	擦除	写	擦除
	DC00h-DFFFh		0000h-DBFFh		0000h-DBFFh E000h-E7FFh		0800h-0FFFh	
SWCMD 设置	-		65, A7 僅在 Boot 区 E000~FFFFh 時使能生效		65, A7 僅在 Boot 区 E800~FFFFh 時使能生效		-	
IAPWE 设置	E2, 4C	E2, BA	E2, 4C	E2, BA	E2, 4C	E2, BA	A1, 4C	A1, BA

IAP 写/擦除使能条件

## 擦除类 EEP 区域 DC00h~DDFFh

```

; IAP example code
; need 2.5V < VCC < 5.5V
ORL    LVRCON, #10h      ; Disable LVR
MOV    DPTR, #DC00h     ; DPTR=DC00h=target IAP address
MOV    IAPWE, #E2h      ;
MOV    IAPWE, #BAh      ;
MOVX   @DPTR, A         ; write any data to DC00h~DDFFh
                          ; to page erase EEP-like area
                          ; from DC00h to DDFFh
                          ; 2ms H/W writing time, CPU wait
MOV    IAPWE, #00h      ; IAP write disable, immediately after IAP write
ANL    LVRCON, #0EFh    ; Enable LVR
    
```

## 擦除 APP 区域 0000h~01FFh

```

; IAP example code
; need 2.5V < VCC < 5.5V
ORL    LVRCON, #10h      ; Disable LVR
MOV    DPTR, #0000h     ; DPTR=0000h=target IAP address
MOV    IAPWE, #E2h      ;
MOV    IAPWE, #BAh      ;
MOV    SWCMD, #65h      ; Only takes effect at boot area
MOV    SWCMD, #A7h      ; Only takes effect at boot area
MOVX   @DPTR, A         ; write any data to 0000h~01FFh
                          ; to page erase APP area from 0000h~01FFh
                          ; 2ms H/W writing time, CPU wait
MOV    IAPWE, #00h      ; IAP write disable, immediately after IAP write
MOV    SWCMD, #00h      ; IAP write disable, immediately after IAP write
ANL    LVRCON, #0EFh    ; Enable LVR
    
```

## 擦除 INFO5 区域 0800h~09FFh

```

; IAP example code
; need 2.5V < VCC < 5.5V
ORL    LVRCON, #10h      ; Disable LVR
MOV    DPTR, #0800h     ; DPTR=0800h=target IAP address
MOV    IAPWE, #A1h      ;
MOV    IAPWE, #BAh      ;
MOVX   @DPTR, A         ; write any data to 0800h~09FF
                          ; to page erase INFO5 area from 0800h~09FFh
                          ; 2ms H/W writing time, CPU wait
MOV    IAPWE, #00h      ; IAP write disable, immediately after IAP write
ANL    LVRCON, #0EFh    ; Enable LVR
    
```



## 写入类 EEP 区域 DC00h

```

; IAP example code
; need 2.5V < VCC < 5.5V
ORL    LVRCON, #10h      ; Disable LVR
MOV    DPTR, #DC00h     ; DPTR=DC00h=target IAP address
MOV    IAPWE, #E2h      ;
MOV    IAPWE, #4Ch      ;
MOV    A, #55h          ;
MOVX   @DPTR, A         ; write 55h to EEP-like area DC00h
MOV    IAPWE, #00h      ; IAP write disable, immediately after IAP write
ANL    LVRCON, #0EFh    ; Enable LVR
    
```

## 写入 APP 区 0000h

```

; IAP example code
; need 2.5V < VCC < 5.5V
ORL    LVRCON, #10h      ; Disable LVR
MOV    DPTR, #0000h     ; DPTR=0000h=target IAP address
MOV    IAPWE, #E2h      ;
MOV    IAPWE, #4Ch      ;
MOV    SWCMD, #65h      ; Only takes effect at boot area
MOV    SWCMD, #A7h      ; Only takes effect at boot area
MOV    A, #55h          ;
MOVX   @DPTR, A         ; write 55h to APP area 0000h
MOV    IAPWE, #00h      ; IAP write disable, immediately after IAP write
MOV    SWCMD, #00h      ; IAP write disable, immediately after IAP write
ANL    LVRCON, #0EFh    ; Enable LVR
    
```

## 写入 INFO5 区域 0800h

```

; IAP example code
; need 2.5V < VCC < 5.5V
ORL    LVRCON, #10h      ; Disable LVR
MOV    DPTR, #0800h     ; DPTR=0800h=target IAP address
MOV    IAPWE, #A1h      ;
MOV    IAPWE, #4Ch      ;
MOV    A, #55h          ;
MOVX   @DPTR, A         ; write 55h to INFO5 area 0800h
MOV    IAPWE, #00h      ; IAP write disable, immediately after IAP write
ANL    LVRCON, #0EFh    ; Enable LVR
    
```

## 读取 INFO5 区域 0800h

```

; IAP example code
; need 2.5V < VCC < 5.5V
MOV    DPTR, #0800h     ; DPTR=0800h=target IAP address
MOVX   A, @DPTR        ; Read data of INFO5 area 0800h
    
```

## 读取主区域 0800h

```

; IAP example code
; need 2.5V < VCC < 5.5V
MOV    DPTR, #0800h     ; DPTR=0800h=target IAP address
CLR    A
MOVX   A, @A+DPTR      ; Read data of Main area 0800h
    
```

## 2.4 FLASH ICP 模式

FLASH 存储器可以通过 tenx 专用的烧录器(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

INFO2 0200h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CFGWH	PROTN	XRSTEN	-			-	BOOTV	

0200h.1~0 **BOOTV**: 上电复位向量选择  
 00: 0x0000  
 01: 0x0000  
 10: 0xE000 (BOOT 区域 7K 字节)  
 11: 0xE800 (BOOT 区域 5K 字节, 默认)

SFR 97h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
<b>SWCMD</b>	IAPEN/SWRST								
	-						WDTO	IAPEN	
R/W	W						R	R	
Reset	-						0	0	

97h.7~0 **IAPEN (W)**:  
 写入 65h 和 A7h 以启用 APP 区域 IAP 写入/擦除 (仅在 Boot 区时设置才能生效)  
 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
<b>IAPWE</b>	IAPWE							
	IAPWE	IAPTO	-					
R/W	R	R	W					
Reset	0	0	-					

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

写入E2h 和4Ch 以启用 IAP APP 区域写入  
 写入E2h 和BAh 以启用 IAP APP 区域擦除  
 写入A1h 和4Ch 以启用 IAP INFO5~8 区域写入  
 写入A1h 和BAh 以启用 IAP INFO5~8 区域擦除  
 写入其他值以禁用 IAP 写/擦除。 建议IAP访问后立即清除

**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 EEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>BOOTV</b>	-	-	-	-	-	RSTV	BOOTVR	
R/W	-	-	-	-	-	R/W	R	R
Reset	-	-	-	-	-	1	0	0

**EEh.2 RSTV: 改变复位向量 (上电复位时默认 1, 其余复位不会更改使用者设置)**

0: 复位向量 = 0x0000  
 1: 复位向量 = 0xE800 or 0xE000 (由 BOOTV 决定)

**EEh.1~0 BOOTVR: 上电复位向量选择。 只读。 从 CFGWH.BOOTV 加载**

00: 0x0000  
 01: 0x0000  
 10: 0xE000 (BOOT 区 7K 字节)  
 11: 0xE800 (BOOT 区 5K 字节)

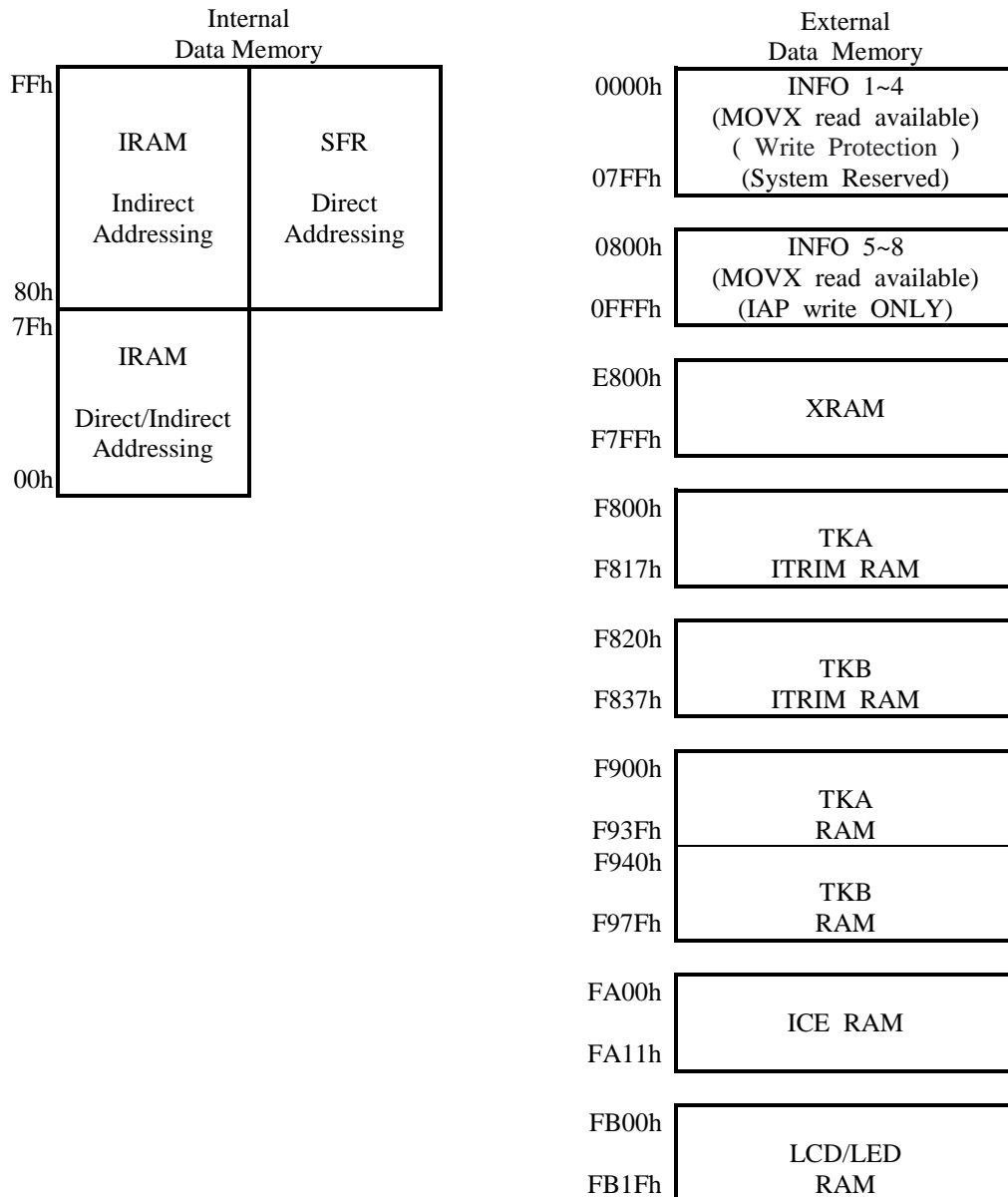
SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSAV	VBGOUT	DIV32	IAPTE		MULDIV16
R/W	R/W	R/W	R/W	R/W	R/W	R/W		R/W
Reset	0	0	0	0	0	0	0	0

**F7h.2~1 IAPTE: IAP 写看门狗定时器使能**

00: 禁用  
 01: 等待 0.8mS 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 6.4mS 触发看门狗超时标志, 并退出写入失败状态

## 2.5 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 4096 字节 XRAM, 6 字节 LCD RAM, 10 字节 ICE RAM 與 TK RAM 只能由 MOVX 指令访问。INFO1~8 是闪存信息存储器, 但在使用中被视为外部数据存储器。



### 2.5.1 IRAM

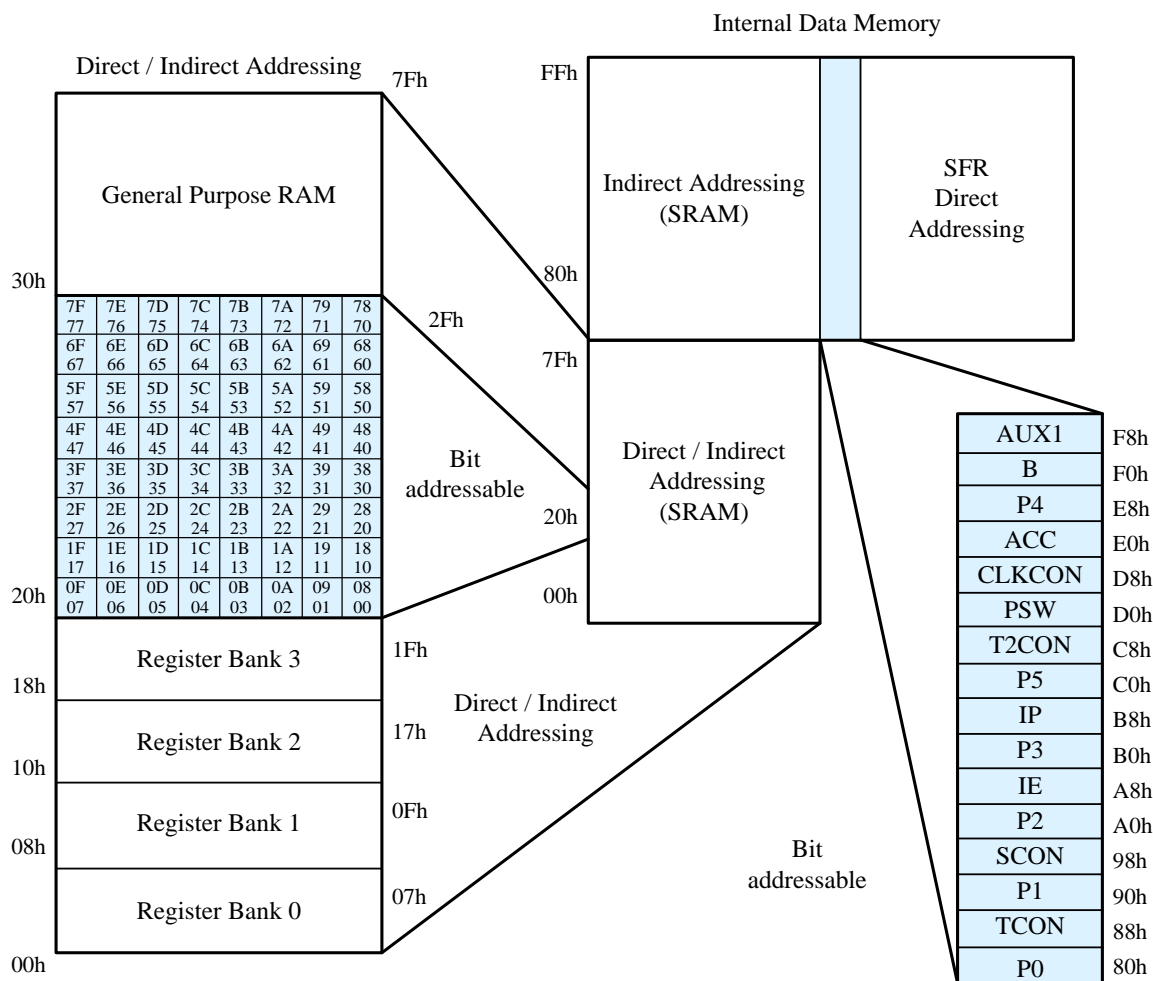
IRAM 位于 8051 内部数据存储空间中。整个 256 字节的 IRAM 可以通过间接寻址访问，但是只有低 128 字节的 IRAM 可以通过直接寻址访问。有四个可直接寻址的寄存器组(通过 PSW 切换)，它们从 00h 到 1Fh 占据 IRAM 空间。地址 20h 至 2Fh 16 字节 IRAM 空间是位可寻址的。IRAM 可用作暂存器或程序堆栈。

### 2.5.2 XRAM

XRAM 位于 8051 外部数据存储器空间(地址从 E800h 到 F7FFh)。4096 字节 XRAM 只能通过“MOVX”指令存取。

### 2.5.3 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 and 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	P4	SIADR	SICON	SIRCD1	SITXRCD2		BOOTV	PWRCON
E0h	ACC	MICON	MIDAT			EFTCON	EXA	EXB
D8h	CLKCON	PWMPRDH	PWMPRDL			UART1CON	UART0CON	TKPINSB2
D0h	PSW	PWMDH	PWMDL			UART2CON	LVRCON	TKPINSB1
C8h	T2CON	IAPWE	RCP2L	RCP2H	TL2	TH2	EXA2	EXA3
C0h	P5	TKPinsa0	TKPinsa1	TKPinsa2	TKPINSB0	ATKCHA0	ATKCHA1	ATKCHA2
B8h	IP	IPH	IP1	IP1H	SPCON	SPSTA	SPDAT	LVDCON
B0h	P3	LXDCON	LXDCON2		TKTMRL	TKCON2	ATKCHB1	ATKCHB0
A8h	IE	INTE1	ADC DL	ADC DH	TKCHSB	TKCON	CHSEL	ATKCHB2
A0h	P2	PWMCON	PINMOD10	PINMOD32	PINMOD54	PINMOD76	PINMODE	TKCHSA
98h	SCON	SBUF	SCON1	SBUF1	TKCON3	PWM2CON	PWMIDX	PWMEN
90h	P1	PORTIDX			OPTION	INTFLG	INTPIN	SWCMD
88h	TCON	TMOD	TL0	TL1	TH0	TH1	SCON2	SBUF2
80h	P0	SP	DPL	DPH		INTPORT	INTPWM	PCON

### 3. 低电压复位和低电压检测

该芯片提供 LVR 和低压检测 (LVD) 功能。SFR LVRCON 可以选择 8 级 LVR，SFR LVDCON 可以选择 16 级 LVD。SFR PWRSVAV 位也影响 LVR 功能，如下表所示。

操作模式	SFR			低电压复位 (LVR)	功能	电流消耗
	LVRPD	PWRSVAV	LVR5			
快钟模式 慢钟模式	0	X	000	ON	LV Reset 2.24V	
	0	X	001	ON	LV Reset 2.48V	
	0	X	010	ON	LV Reset 2.72V	
	0	X	011	ON	LV Reset 2.96V	
	0	X	100	ON	LV Reset 3.2V	
	0	X	101	ON	LV Reset 3.44V	
	0	X	110	ON	LV Reset 3.68V	
	0	X	111	ON	LV Reset 3.92V	
空闲模式 暂停模式 停止模式	0	0	000	ON	LV Reset 2.24V	空闲:180uA@5V 暂停: 58uA@5V 停止: 55uA@5V
	0	0	001	ON	LV Reset 2.48V	
	0	0	010	ON	LV Reset 2.72V	
	0	0	011	ON	LV Reset 2.96V	
	0	0	100	ON	LV Reset 3.2V	
	0	0	101	ON	LV Reset 3.44V	
	0	0	110	ON	LV Reset 3.68V	
	0	0	111	ON	LV Reset 3.92V	
空闲模式	0	1	XXX	ON	POR 2.3V	165uA@5V
	1	0	XXX			
暂停模式 停止模式	0	1	XXX	OFF	-	暂停: 14uA@5V 停止: 11uA@5V
	1	0	XXX			

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
AUX2	WDTE		PWRSVAV	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 **PWRSVAV:** 省电模式控制设
- 0: 不省电
  - 1: 省电，在 IDLE/HALT/STOP 模式下禁用 LVR

SFR BFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LVDCON</b>	LVDM	LVDO	LVDDBS	LVDPD	LVDS			
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

- BFh.7 LVDM:** 低压检测模式  
 0: 当  $VCC < LVDS$  时  $LVDIF = 1$  和  $LVDO = 1$   
 1: 当  $VCC > LVDS$  时  $LVDIF = 1$  和  $LVDO = 1$
- BFh.6 LVDO:** 低压检测输出
- BFh.5 LVDDBS:** 低压检测去抖  
 0: 禁用 1: 使能
- BFh.4 LVDPD:** 低压检测选择  
 0: 使能 1: 禁用
- BFh.3~0 LVDS:** 低压检测选择  
 0000: 将 LVD 设置为 2.52V  
 0001: 将 LVD 设置为 2.62V  
 0010: 将 LVD 设置为 2.74V  
 0011: 将 LVD 设置为 2.86V  
 0100: 将 LVD 设置为 2.99V  
 0101: 将 LVD 设置为 3.1V  
 0110: 将 LVD 设置为 3.23V  
 0111: 将 LVD 设置为 3.35V  
 1000: 将 LVD 设置为 3.48V  
 1001: 将 LVD 设置为 3.6V  
 1010: 将 LVD 设置为 3.72V  
 1011: 将 LVD 设置为 3.84V  
 1100: 将 LVD 设置为 3.96V  
 1101: 将 LVD 设置为 4.08V  
 1110: 将 LVD 设置为 4.2V  
 1111: 将 LVD 设置为 4.32V

SFR D6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LVRCON</b>	SXTGAIN		–	LVRPD	–	LVRS		
R/W	R/W	R/W	–	R/W	–	R/W	R/W	R/W
Reset	0	0	–	0	–	0	0	0

- D6h.4 LVRPD:** 低电压复位功能选择  
 0: 使能 1: 禁用
- D6h.2~0 LVRS:** 低电压复位功能选择  
 000: 将 LVR 设置为 2.24V  
 001: 将 LVR 设置为 2.48V  
 010: 将 LVR 设置为 2.72V  
 011: 将 LVR 设置为 2.96V  
 100: 将 LVR 设置为 3.20V  
 101: 将 LVR 设置为 3.44V  
 110: 将 LVR 设置为 3.68V  
 111: 将 LVR 设置为 3.92V



## 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 40mS 的芯片预热,然后从 Flash 的最后两个字节下载 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 2.3V。

### 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/停止/暂停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 LVR 功能。CFGWH 可以选择 8 级 LVR。

SFR D6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVRCON	SXTGAIN		–	LVRPD	–	LVRS		
R/W	R/W	R/W	–	R/W	–	R/W	R/W	R/W
Reset	0	0	–	0	–	0	0	0

D6h.4 **LVRPD:** 低电压复位功能选择

- 0: 使能
- 1: 禁用

D6h.2~0 **LVRS:** 低电压复位功能选择

- 000: 将 LVR 设置为 2.24V
- 001: 将 LVR 设置为 2.48V
- 010: 将 LVR 设置为 2.72V
- 011: 将 LVR 设置为 2.96V
- 100: 将 LVR 设置为 3.20V
- 101: 将 LVR 设置为 3.44V
- 110: 将 LVR 设置为 3.68V
- 111: 将 LVR 设置为 3.92V

SFR 94h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>OPTION</b>	TM3CKS		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.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
<b>INTFLG</b>	LVDIF	-	TKIFA	ADIF	-	-	PCIF	TF3
R/W	R	-	R/W	R/W	-	-	R/W	R/W
Reset	-	-	0	0	-	-	0	0

95h.7 **LVDIF**:低电压检测标志

由硬件设置。S/W 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
<b>SWCMD</b>	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
<b>AUX2</b>	WDTE		PWRSVAV	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/停止/暂停模式时禁止

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
<b>AUX1</b>	CLRWDT	CLRMTM3	TKSOCA	ADSOC	-	TKSOCB	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.7 **CLRWDT**: 设置以清除看门狗定时器,H/W 自动在一个时钟周期清除它

## 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,1~16 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。

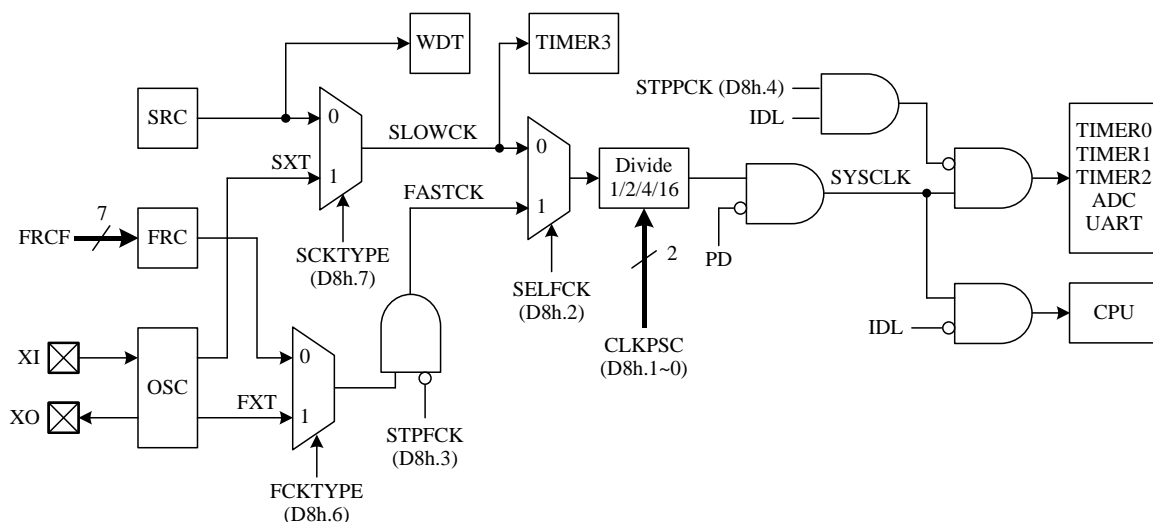
复位后,该设备在慢钟模式 80 KHz 的 SRC 运行。S/W 应该正确选择安全的芯片运行时钟速率。较高的  $V_{CC}$  允许芯片在更高的系统时钟频率运行。在典型的情况下,18MHz 的系统时钟频率需要  $V_{CC}>2.3V$ 。

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 1~16 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。H/W 自动阻断 S/W 异常设置该寄存器。S/W 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)



**Note:** 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-TM52XXXXX\_01S 和 AP-TM52XXXXX\_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 ← → 1	0/1	0
Slow type change	0 ← → 1	0/1	0	1
Stop FRC/FXT	0/1	0/1	0 → 1	0
Switch to FRC/FXT	0/1	0/1	0	0 → 1
Switch to SRC/SXT	0/1	0/1	0	1 → 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  
1:SXT

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

0:FRC  
1:FXT

D8h.5 **STPSCK**: 设置 1 以在 PD=1 后停止慢时钟 (Halt/Stop 模式控制)

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

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

这个设备有四种操作模式。**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,则芯片无法进入停止/暂停模式。(INTn=0 and Exn=1,n=0,1,9)

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PCON</b>	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 进入停止/暂停模式。

87h.0 **IDL:**空闲位,如果 1 进入空闲模式。

SFR F7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRSVAV	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:** VBG 电压输出至 P3.2

0: 关闭

1: 启用, 应设置附加条件 VBGGEN=1 (AEh.1)

SFR D8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CLKCON</b>	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 1:SXT
- D8h.6 **FCKTYPE**:快时钟类型。该位只能在慢钟模式(SELFCK=0)时改变。  
0:FRC 1:FXT
- D8h.5 **STPSCK**:设置 1 以在 PD=1 后停止慢时钟 (Halt/Stop 模式控制)
- 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  
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

## 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停止模式下唤醒 CPU。暂停模式可由 Time3 和引脚中断唤醒。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(可以唤醒停止/暂停模式)
000B	TF0	Timer0 中断
0013	IE1	INT1 外部引脚中断(可以唤醒停止/暂停模式)
001B	TF1	Timer1 中断
0023	RI+TI	串口(UART)中断
002B	TF2+EXF2	Timer2 中断
0033	-	保留为 ICE 模式使用
003B	TF3	Timer3 中断
0043	PCIF P0IF~P5IF PIN0IF~PIN7IF	Port0~PORT5 外部引脚电平变化中断(可以唤醒停止/暂停模式)
004B	LVDIF	LVD 中断
0053	ADIF TKIFA TKIFB	ADC/触摸按键 A/ B 中断
005B	SPIF+WCOL+MODF	SPI 中断
0063	RI1+TI1 RI2+TI2	串口(UART1/UART2) 中断
006B	MIF TXDF, RCD2F, RCD1F	I <sup>2</sup> C 中断
0073	PWM0IF, PWM1IF PWM2IF, PWM3IF	PWM0~3 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

在进入中断程序时,除了传统已熟知应执行 PUSH, POP 的 SFR A 或 PSW 外,有一些用来索引的 SFR 也应该加入 PUSH POP 的行列,如 PORTIDX 和 PWMIDX,以避免中断前后写读这些 SFR 可能造成不一致的问题。另外 PWMDH, PWMDL, PWMPRDH 或 PWMPRDL 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,建议只在主程序内更新数据,或只在中断内更新数据,以避免可能的错误产生。

SFR A8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IE</b>	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(UART0)中断使能控制  
 0:禁用串口(UART0)中断  
 1:允许串口(UART0)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 引脚中断和停止/暂停模式唤醒使能控制  
 0:禁用 INT1 引脚中断和停止/暂停模式唤醒  
 1:允许 INT1 引脚中断和停止/暂停模式唤醒,不管 EA 为 0 或 1,都可从停止/暂停模式下唤醒 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 引脚中断和停止/暂停模式唤醒使能控制  
 0:禁用 INT0 引脚中断和停止/暂停模式唤醒  
 1:允许 INT0 引脚中断和停止/暂停模式唤醒,不管 EA 为 0 或 1,都可从停止/暂停模式下唤醒 CPU。



SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE1</b>	PWMIE	I2CE	ES2	SPIE	ADTKIE	LVDIE	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.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:** I<sup>2</sup>C (master/slave) 中断使能  
 0: 禁用 I<sup>2</sup>C interrupt  
 1: 允许 I<sup>2</sup>C interrupt
- A9h.5 **ES2:** Serial Port (UART1/UART2) 中断使能  
 0: 禁用 Serial Port (UART1/UART2) 中断  
 1: 允许 Serial Port (UART1/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 **LVDIE:** LVD 中断使能  
 0: 禁用 LVD interrupt  
 1: 允许 LVD interrupt.
- A9h.1 **PCIE:** Port0~Port5 引脚电平变化中断使能  
 0: 禁用 Port0~Port5 引脚电平变化中断  
 1: 允许 Port0~Port5 引脚电平变化中断
- A9h.0 **TM3IE:** Timer3 中断使能控制和暂停模式唤醒  
 0:禁用 Timer3 中断和暂停模式唤醒  
 1:允许 Timer3 中断和暂停模式唤醒, 无论 EA 是 0 还是 1, 它都可以从 Halt 模式唤醒 CPU。

SFR B9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IPH</b>	–	–	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
<b>IP</b>	–	–	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(UART)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
<b>IP1H</b>	PPWMH	PI2CH	PS2H	PSPIH	PADTKIH	PLVDH	PPCH	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
<b>IP1</b>	PPWM	PI2C	PS2	PSPI	PADTKI	PLVD	PPC	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**: Serial Port (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。

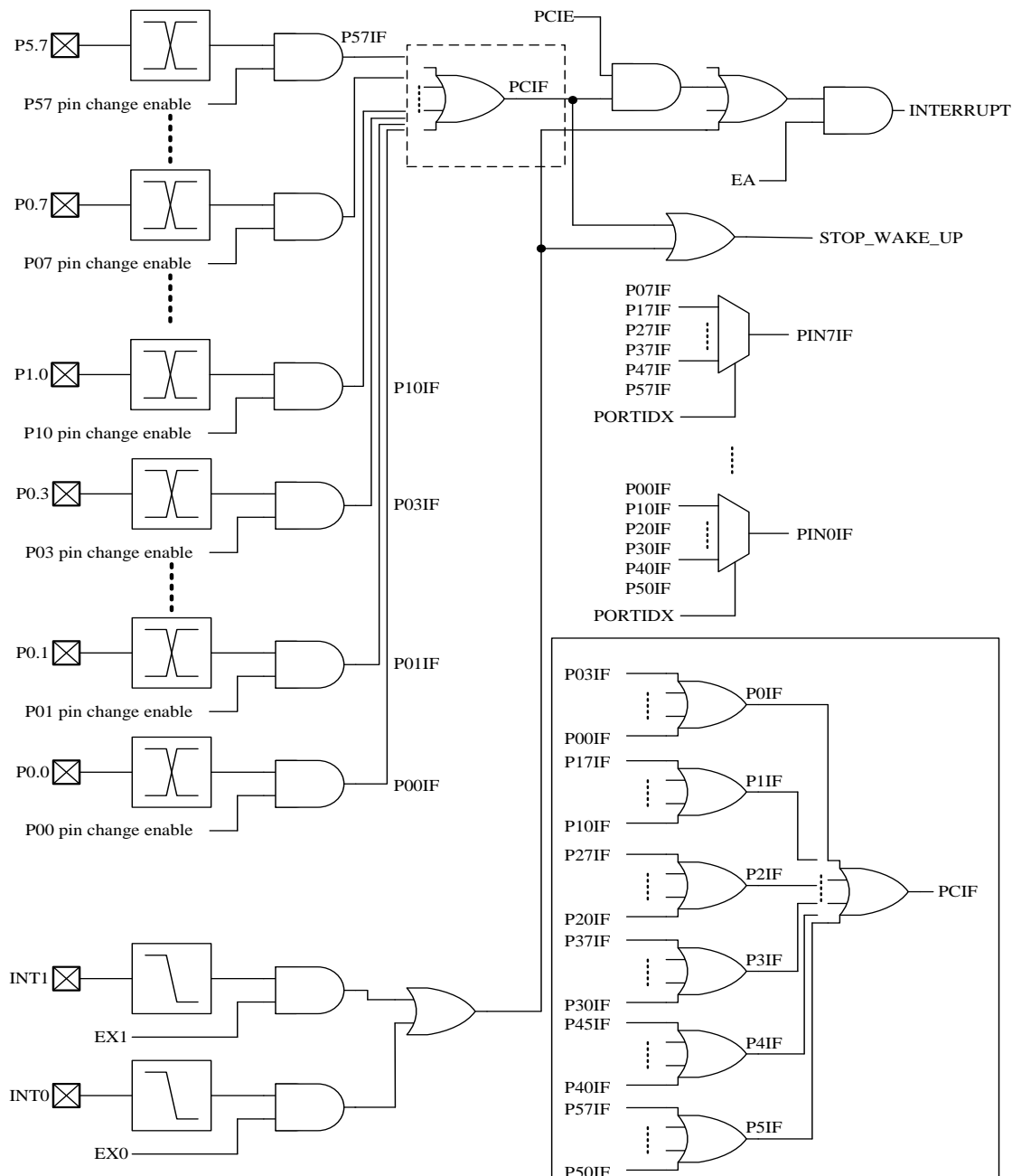
BBh.2,BAh.2 **PX2\_9LVDH, PX2\_9LVD**: 外部 INT2~INT9 引脚和 LVD 中断优先级控制。定义如上

BBh.1,BAh.1 **PPCH,PPC**:端口 0~5 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~INT1 和 Port0~Port5 引脚电平变化中断。INT0~INT1 和 Port0~Port5 的引脚变化也具有停止/暂停模式唤醒功能。INT0 和 INT1 为 8051 标准触发的下降沿或低电平。Port0~Port5 引脚更改中断由 IO 状态更改触发。引脚更改使能通过 PINMOD10 / PINMOD32 / PINMOD54 / PINMOD76 进行设置。有关详细信息，请参见第 7 章。PINMODE 和引脚更改使能设置。



引脚中断和唤醒

**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 Stop/Halt 模式。(INTn=0 and Exn=1, n=0~1)

**注:** 如果 PCIF=1, 则芯片无法进入停止/暂停模式。用户应在进入停止/暂停模式之前清除 PCIF。

SFR 88h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TCON</b>	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 H/W 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 H/W 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 85h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTPORT</b>	-	-	P5IF	P4IF	P3IF	P2IF	P1IF	P0IF
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	-	0	0	0	0	0	0

- 96h.5 **P5IF:** P5.7~P5.0 引脚电平变化旗标, 写 0 清除 P5.7~P5.0 引脚电平变化旗标
- 96h.4 **P4IF:** P4.7~P4.0 引脚电平变化旗标, 写 0 清除 P4.7~P4.0 引脚电平变化旗标
- 96h.3 **P3IF:** P3.7~P3.0 引脚电平变化旗标, 写 0 清除 P3.7~P3.0 引脚电平变化旗标
- 96h.2 **P2IF:** P2.7~P2.0 引脚电平变化旗标, 写 0 清除 P2.7~P2.0 引脚电平变化旗标
- 96h.1 **P1IF:** P1.7~P1.0 引脚电平变化旗标, 写 0 清除 P1.7~P1.0 引脚电平变化旗标
- 96h.0 **P0IF:** P0.7~P0.0 引脚电平变化旗标, 写 0 清除 P0.7~P0.0 引脚电平变化旗标

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTFLG</b>	LVDIF	-	TKIFA	ADIF	-	-	PCIF	TF3
R/W	R	-	R/W	R/W	-	-	R/W	R/W
Reset	-	-	0	0	-	-	0	0

- 95h.1 **PCIF:** Port0~Port5 引脚电平变化旗标  
 当检测到 Port0~Port5 引脚电平变化且其中断允许位被置 1 时, 由硬件置 1。  
 S / W 可以写 0 来清除所有引脚更改中断标志 (Port0~Port5), 也可以清除 PIN0IF~PIN7F  
 和 P0IF~P5IF。

SFR 96h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTPIN</b>	PIN7IF	PIN6IF	PIN5IF	PIN4IF	PIN3IF	PIN2IF	PIN1IF	PIN0IF
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

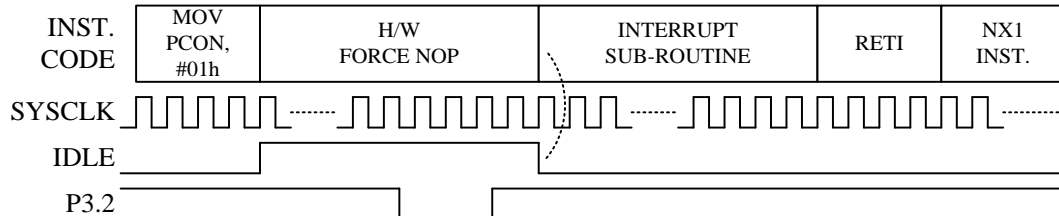
- 96h.7 **PIN7IF**: Px.7 引脚电平变化旗标, 写 0 清除 Px.7 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义
- 96h.6 **PIN6IF**: Px.6 引脚电平变化旗标, 写 0 清除 Px.6 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义
- 96h.5 **PIN5IF**: Px.5 引脚电平变化旗标, 写 0 清除 Px.5 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义
- 96h.4 **PIN4IF**: Px.4 引脚电平变化旗标, 写 0 清除 Px.4 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义
- 96h.3 **PIN3IF**: Px.3 引脚电平变化旗标, 写 0 清除 Px.3 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义
- 96h.2 **PIN2IF**: Px.2 引脚电平变化旗标, 写 0 清除 Px.2 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义
- 96h.1 **PIN1IF**: Px.1 引脚电平变化旗标, 写 0 清除 Px.1 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义
- 96h.0 **PIN0IF**: Px.0 引脚电平变化旗标, 写 0 清除 Px.1 引脚电平变化旗标  
端口号 (x) 由 PORTIDX 定义

SFR A8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>IE</b>	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 引脚中断和停止/暂停模式唤醒使能控制  
0:禁用 INT1 引脚中断和停止/暂停模式唤醒  
1:允许 INT1 引脚中断和停止/暂停模式唤醒,不管 EA 为 0 或 1,都可从停止/暂停模式下唤醒 CPU。
- A8h.0 **EX0**:INT0 引脚中断和停止/暂停模式唤醒使能控制  
0:禁用 INT0 引脚中断和停止/暂停模式唤醒  
1:允许 INT0 引脚中断和停止/暂停模式唤醒,不管 EA 为 0 或 1,都可从停止/暂停模式下唤醒 CPU。

#### 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,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 进入停止/暂停模式。

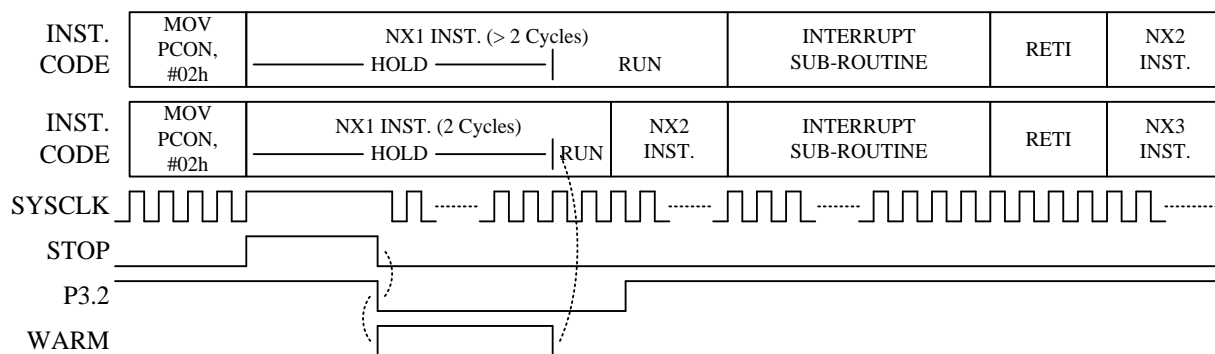
87h.0 IDL:空闲位,如果 1 进入空闲模式。

#### 6.5 停止/暂停模式唤醒和中断

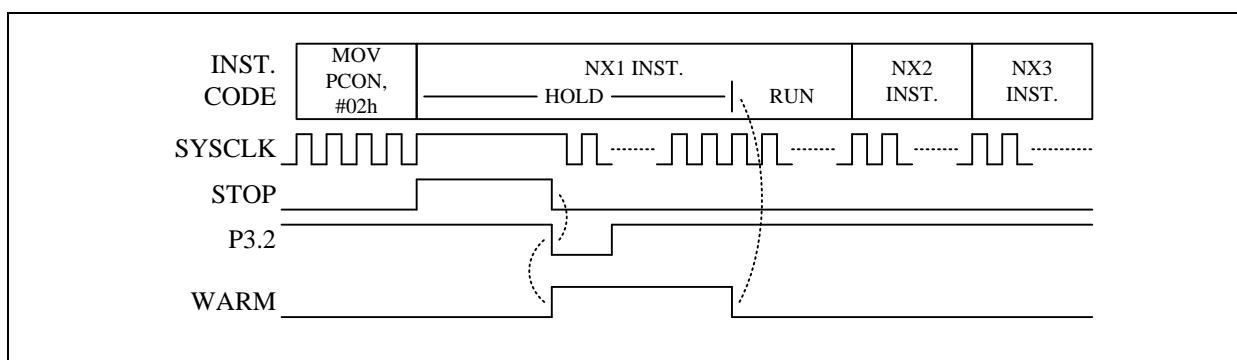
停止/暂停模式唤醒很简单,只要把各个引脚的中断使能位(如:EX0)设置,该引脚唤醒功能启用。或设置 PINMOD10 / PINMOD32 / PINMOD54 / PINMOD76 可启用 Port0~Port5 停止/暂停模式唤醒功能。一旦停止/暂停被唤醒,“PD(PCON.1)设置后的第一条指令”立即在中断服务之前被执行。中断进入需要 EA=1 和该引脚触发状态停留足够长,以被系统时钟采样到。此功能可让 CPU 停止/暂停模式唤醒后,进入或不进入中断子程序。

**注:** 如果 INT<sub>n</sub> 引脚是低电平且该唤醒功能启用,则芯片无法进入 Stop/Halt 模式。(INT<sub>n</sub>=0 and Ex<sub>n</sub>=1, n=0~1)

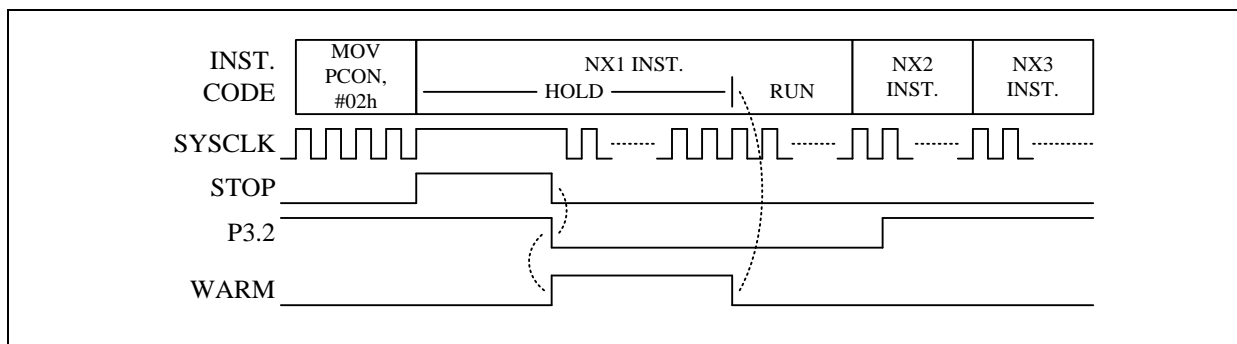
**注:** 如果 PCIF=1, 则芯片无法进入停止/暂停模式。用户应在进入停止/暂停模式之前清除 PCIF。



EA=EX0=1, P3.2(INT0)预热后被采样,停止/暂停模式唤醒和中断



EA=EX0=1, 脉冲太窄,停止/暂停模式唤醒,但没有中断



EX0=1, EA=0, P3.2 (INT0) 停止/暂停模式唤醒,但没有中断

## 7. I/O 端口

该芯片总共有 42 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）。

在进入中断程序时，除了传统已熟知应执行 PUSH, POP 的 SFR A 或 PSW 外，有一些用来索引的 SFR 也应该加入 PUSH POP 的行列，如 PORTIDX，以避免中断前后写读这些 SFR 可能造成不一致的问题。

### 7.1 端口 0~端口 5

IO 引脚可以按以下不同方式定义。

P INMOD76 PINMOD54 PINMOD32 PINMOD10					引脚状态	中断	唤醒
MODE00	0	0	0	0	开漏输出，上拉	-	-
MODE01	0	0	0	1	开漏输出(默认)	-	-
MODE02	0	0	1	0	CMOS 输出	-	-
MODE03	0	0	1	1	ADC/TK 通道, XI/XO	-	-
MODE10	0	1	0	0	开漏输出，上拉	-	-
MODE11	0	1	0	1	开漏输出(默认)	-	-
MODE12	0	1	1	0	CMOS 输出	-	-
MODE13	0	1	1	1	LCD / LED 输出	-	-
MODE20	1	0	0	0	开漏输出，上拉	Y	Y
MODE21	1	0	0	1	开漏输出(默认)	Y	Y
MODE22	1	0	1	0	CMOS 输出	-	-
MODE23	1	0	1	1	PWMO, TxO 输出	-	-
MODE30	1	1	0	0	开漏输出，上拉	Y	Y
MODE31	1	1	0	1	开漏输出(默认)	Y	Y
MODE32	1	1	1	0	CMOS 输出	-	-
MODE33	1	1	1	1	保留	-	-

**表 7.1 Port0~Port5 I / O 引脚菜单**

PINMOD76 / PINMOD54 / PINMOD32 / PINMOD10 需要 PORTIDX 来索引相应的 IO 端口。

例如：

如果 PORTIDX = 0，则 PINMOD10 设置为 P0.1 和 P0.0，高 4 位设置为 P0.1，低 4 位设置为 P0.0

如果 PORTIDX = 1，则 PINMOD10 设置为 P1.1 和 P1.0，高 4 位设置为 P1.1，低 4 位设置为 P1.0

...

如果 PORTIDX = 5，则 PINMOD10 设置为 P5.1 和 P5.0，高 4 位设置为 P5.1，低 4 位设置为 P5.0

如果 PORTIDX = 0，则 PINMOD32 设置为 P0.3 和 P0.2，高 4 位设置为 P0.3，低 4 位设置为 P0.2

...

如果 PORTIDX = 5，则 PINMOD76 设置为 P5.7 和 P5.6，高 4 位设置为 P5.7，低 4 位设置为 P5.6



模式	端口引脚功能	Px.n SFR 数据	引脚状态	电阻上拉	数位输入
MODEx0	开漏输出	0	驱动低	N	N
		1	上拉	Y	Y
MODEx1	开漏输出	0	驱动低	N	N
		1	高阻抗	N	Y
MODEx2	CMOS 推挽输出	0	驱动低	N	N
		1	驱动高	N	N
MODEx3	替代功能, 如 ADC/TK/LCD/LED/PWM/TxO	X (无关)	-	N	N

端口引脚菜单

如果一个端口 1~端口 5 引脚用于施密特触发输入,S/W 必须设置 I/O 引脚到 MODEx0 或 MODEx1, 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 5 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 MODEx3, 可以激活大多数功能。Port1 / Port3 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 MODEx0 或 MODEx1, 并将 P1.n / P3.n SFR 保持为 1。

引脚名称	唤醒 中断	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P0.3	Y	AD03	TKB19			PWM33		SDA
P0.2	Y	AD02	TKB18			PWM32		SCL
P0.1	Y	AD01	TKB17			PWM31		
P0.0	Y	AD00	TKB16			PWM30		

Port0 多重菜单

引脚名称	唤醒 中断	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P1.7	Y	AD17	TKA14	LCOM7/LED7	COM7/SEG19	PWM1		
P1.6	Y	AD16	TKA15	LCOM6/LED6	COM6/SEG18	PWM0		
P1.5	Y	AD15	TKA16	LCOM5/LED5	COM5/SEG17	PWM1		
P1.4	Y	AD14	TKA17	LCOM4/LED4	COM4/SEG16	PWM0		
P1.3	Y	AD13	TKA18	LCOM3/LED3	COM3			
P1.2	Y	AD12	TKA19	LCOM2/LED2	COM2			
P1.1	Y	AD11	TKA20	LCOM1/LED1	COM1		TXD2	
P1.0	Y	AD10	TKB20	LCOM0/LED0	COM0		RXD2	

Port1 多重菜单

引脚名称	唤醒中断	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P2.7	Y	AD27	TKA10		SEG3	PWM1		
P2.6	Y	AD26	TKA11		SEG2	PWM0		
P2.5	Y	AD25	TAK12		SEG1			
P2.4	Y	AD24	TKA13		SEG0			
P2.3	Y	AD23	TKB02		SEG11			
P2.2	Y	AD22	TKB03		SEG10			
P2.1	Y	AD21	TKB04		SEG9			
P2.0	Y	AD20	TKB00		SEG8			

Port2 多重菜单

引脚名称	唤醒中断	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P3.7	Y	AD37	TKA06		SEG7		TXD1A	
P3.6	Y	AD36	TKA07		SEG6		RXD1A	
P3.5	Y	AD35	TKA08		SEG5		TXD0B	SDA
P3.4	Y	AD34	TKA09		SEG4		RXD0B	SCL
P3.3	Y	AD33	TKA01		SEG15	PWM35		
P3.2	Y	AD32	TKA00		SEG14	PWM34		
P3.1	Y	AD31	TKB05		SEG13		TXD0A	SDA
P3.0	Y	AD30	TKB01		SEG12		RXD0A	SCL

Port3 多重菜单

引脚名称	唤醒中断	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P4.5	Y	AD39	TKA03				TXD0C	
P4.4	Y	AD38	TKA02				RXD0C	
P4.3	Y	AD29	TKB07					NSS
P4.2	Y	AD28	TKB06					MISO
P4.1	Y	AD41	TKA05					MOSI
P4.0	Y	AD40	TKA04					SCK

Port4 多重菜单

引脚名称	唤醒中断	ADC	TK	LED	LCD	PWM	UART	I2C SPI
P5.7	Y	AD19	TKB15	LSEG7		PWM2N		
P5.6	Y	AD18	TKB14	LSEG6		PWM2P		
P5.5	Y	AD09	TKB13	LSEG5			TXD1B	
P5.4	Y	AD08	TKB12	LSEG4			RXD1B	
P5.3	Y	AD07	TKB11	LSEG3		PWM2N		
P5.2	Y	AD06	TKB10	LSEG2		PWM2P		
P5.1	Y	AD05	TKB09	LSEG1		PWM2N		
P5.0	Y	AD04	TKB08	LSEG0		PWM2P		

Port5 多重菜单

下面列出了 Port0~Port5 引脚的替代功能所需的 SFR 设置。

替代功能	PINMODxx	Px.n SFR 数据	引脚状态	设其他需要的 SFR 置
T0, T1, T2, T2EX, INT0, INT1	<b>00x0</b>	1	输入上拉	
	<b>00x1</b>	1	输入	
RXD0x RXD1x RXD2	<b>xxx0</b>	1	UART RX (输入上拉)	PINMODE
	<b>xxx1</b>	1	UART RX (Input)	
TXD0x TXD1x TXD2	<b>xx10</b>	1	UART TX 输出(CMOS 推挽)	
XI, XO	<b>0011</b>	1	晶振	CLKCON
SPI Master Mode MISO	<b>xx01</b>	1	SPI 数据输入	SPCON
SPI Master Mode SCK, MOSI	<b>xx10</b>	X	SPI 时钟/数据输出(CMOS 推挽)	
SPI Slave Mode MISO	<b>xx10</b>	X	SPI 数据输出 (CMOS 推挽)	
SPI Slave Mode SCK, MOSI	<b>xx01</b>	1	SPI 时钟/数据输入	
NSS	<b>xx01</b>	1	SPI 芯片选择	
I <sup>2</sup> C Master SCL	<b>0x00</b>	X	I <sup>2</sup> C 时钟输出 (开漏输出, 上拉)	PINMODE
	<b>xx10</b>	X	I <sup>2</sup> C 时钟输出 (CMOS 推挽)	
I <sup>2</sup> C Slave SCL	<b>xx01</b>	1	I <sup>2</sup> C 时钟输出 (高阻抗)	
I <sup>2</sup> C Master/Slaver SDA	<b>xx00</b>	1	I <sup>2</sup> C 数据 (推挽)	
AD00~AD41			ADC 通道	ADCHS
TKA00~TKA20 TKB00~TKB20	<b>0011</b>	X	Touch Key 通道	TKCHSA TKCHSB ATKCHA0/1/2 ATKCHB0/1/2
LCOM0~ LCOM7 LSEG0~ LSEG7 LED0~ LED7 COM0~COM7 SEG0~SEG19	<b>0111</b>	X	LED 输出(矩阵模式)	LXDICON LXDICON2
			LED 输出(点矩阵模式)	
			LCD 输出	
T00, T20	<b>1011</b>	X	Clock 输出(CMOS 推挽)	PWMEN
PWMx			PWM 输出(CMOS 推挽)	

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。

SFR 80h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>P0</b>	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	1	1	1	1	1	1	1	1

80h.7~0 **P0:**端口0 data

SFR 90h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>P1</b>	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
<b>P2</b>	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	1	1	1	1	1	1	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
<b>P3</b>	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 E8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>P4</b>	P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0
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

E8h.7~0 **P4:** 端口4 数据

SFR C0h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>P5</b>	P5.7	P5.6	P5.5	P5.4	P5.3	P5.2	P5.1	P5.0
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

C0h.7~0 **P5:** 端口5 数据

SFR A2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD10</b>	PINMOD1				PINMOD0			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A2h.7~4 **PINMOD1**: Px.1 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

A2h.3~0 **PINMOD0**: Px.0 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

SFR A3h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD32</b>	PINMOD3				PINMOD2			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A3h.7~4 **PINMOD3**: Px.3 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

A3h.3~0 **PINMOD2**: Px.2 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

SFR A4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD54</b>	PINMOD5				PINMOD4			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A4h.7~4 **PINMOD5**: Px.5 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

A4h.3~0 **PINMOD4**: Px.4 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

SFR A5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMOD76</b>	PINMOD7				PINMOD6			
R/W	R/W				R/W			
Reset	0	0	0	1	0	0	0	1

A5h.7~4 **PINMOD7**: Px.7 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

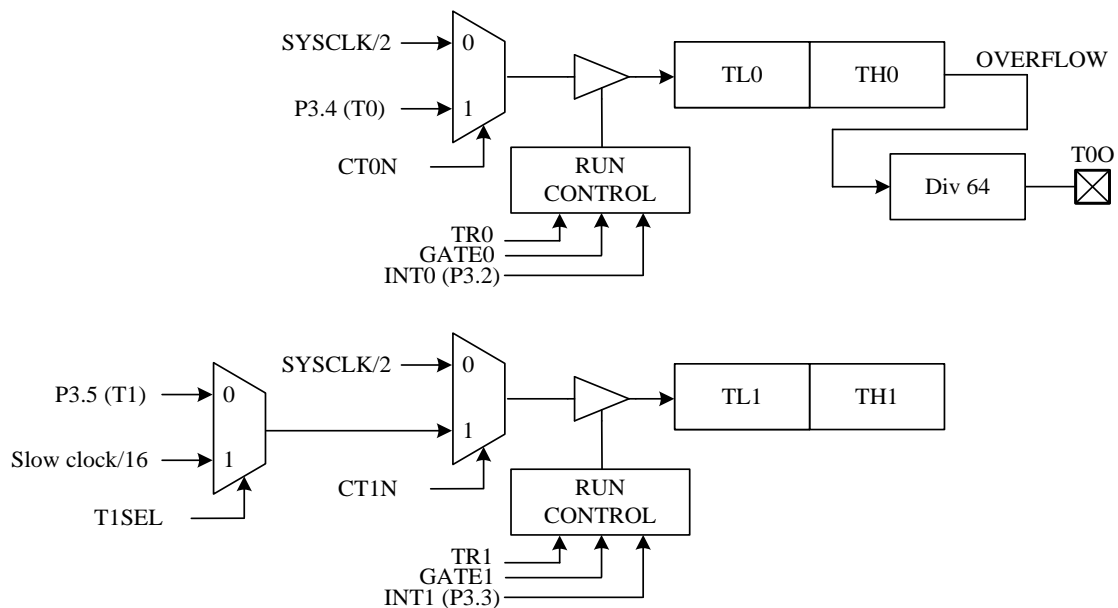
A5h.3~0 **PINMOD6**: Px.6 引脚控制, 端口索引 (x) 由 PORTIDX 定义  
0000~1111: 见表 7.1

## 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, T00 引脚输出“Timer0 溢出除以 64”的信号, 而 T20 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
<b>TCON</b>	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 H/W 设置。  
 当 CPU 转向进入中断服务程序时由 H/W 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 H/W 设置。  
 当 CPU 转向进入中断服务程序时由 H/W 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
<b>TMOD</b>	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 INTO 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
<b>TL0</b>	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
<b>TL1</b>	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
<b>TH0</b>	TH0							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

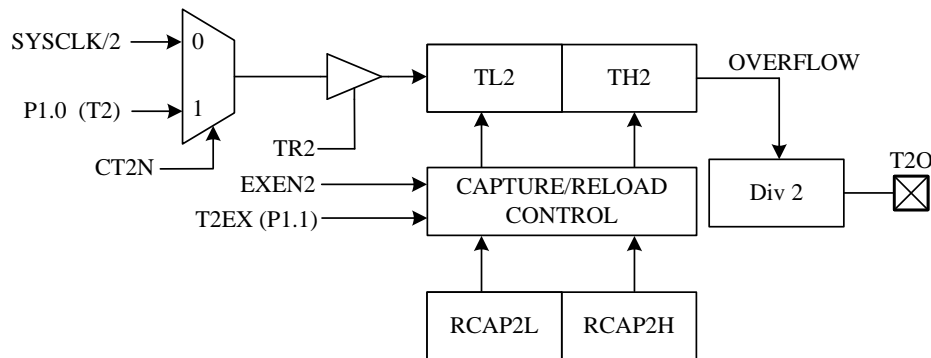
8Ch.7~0 **TH0**:Timer0 数据的高字节

SFR 8Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TH1</b>	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

## 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
<b>T2CON</b>	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 H/W 设置,除非 RCLK=1 或 TCLK=1。此位必须由 S/W 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 S/W 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
<b>RCP2L</b>	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
<b>RCP2H</b>	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
<b>TL2</b>	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
<b>TH2</b>	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
<b>AUX1</b>	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	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 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 65536,16384,4096 或 1024 取决于 TM3PSC 位。Timer3 的时钟源为慢时钟(SRC 或 SXT)或 FRC/512。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
<b>OPTION</b>	TM3CKS		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.6 **TM3CKS:** Timer3时钟源选择

00: 慢时钟 (SXT/SRC)

01: FRC/512

10: 慢时钟 (SXT/SRC) /2

11: FRC/1024

94h.1~0 **TM3PSC:**Timer3 中断率控制选择

00:中断率是 65536 个 Timer3 时钟源周期

01:中断率是 16384 个 Timer3 时钟源周期

10:中断率是 4096 个 Timer3 时钟源周期

11:中断率是 1024 个 Timer3 时钟源周期

SFR 95h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTFLG</b>	LVDIF	–	TKIFA	ADIF	–	–	PCIF	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 H/W 设置。

当程序执行中断服务程序时被自动清除。

S/W 也可以写 FEh 到 INTFLG 清除该标志。(注 2)

**注 2:**S/W 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
<b>AUX1</b>	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	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.6 **CLRTM3:**设置以清除 Timer3,H/W 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

该设备可以为蜂鸣器生成各种频率的波形引脚输出(CMOS 或开漏)。T0O 和 T2O 波形除以 Timer0 / Timer2 溢出信号。 T0O 波形是 Timer0 溢出除以 64, T2O 波形是 Timer2 溢出除以 2。用户可以通过定时器自动重装速度来控制其频率。将 P3.4 或 P1.0 的 MODE 设置为 1011b, 以输出 T0O 和 T2O。有关更多详细信息,请参见表 7.1。

## 9. UARTs

该芯片具有三个 UART，分别为 UART0，UART1 和 UART2。

UART0 是标准的 8051 全双工 UART，UART0 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。

UART1 使用 SCON1 和 SBUF1 的 SFR。SCON1 是控制寄存器，SBUF1 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。UART1 支持 UART 的大部分功能，但不支持 Mode0 和 Mode2。

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 Mode0 和 Mode2。

**UART0 波特率设置：**当 SFR UART0BRS=0

UART0 波特率设置同标准 8051 如下。

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

**UART0 波特率设置：**当 SFR UART0BRS =1

- Mode 0: 波特率= $F_{SYSCLK}/2$
- Mode 1: 波特率= $F_{sys}/32/UART0BRP$
- Mode 2: 波特率= $(SMOD + 1) \times F_{SYSCLK}/64$
- Mode 3: 波特率= $F_{sys}/32/UART0BRP$

**UART1 波特率设置：**

- Mode 0, 2: 无效的
- Mode 1, 3: 波特率= $F_{sys}/32/UART1BRP$

**UART2 波特率设置：**

- Mode 0, 2: 无效的
- Mode 1, 3: 波特率= $F_{sys}/32/UART2BRP$

SFR 87h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PCON</b>	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:UART0 双波特率控制位**  
 0:禁止 UART0 双波特率控制位  
 1:允许 UART0 双波特率控制位

SFR 98h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SCON</b>	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 UART0,波特率可变  
 10:模式 2:9 位 UART0,波特率= $F_{SYSCLK}/32$  或/64  
 11:模式 3:9 位 UART0,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:UART0 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 H/W 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 S/W 清零。

98h.0 **RI:接收中断标志**

由 H/W 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 S/W 清零。

SFR 99h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SBUF</b>	SBUF							
R/W	R/W							
Reset	–	–	–	–	–	–	–	–

99h.7~0 **SBUF:UART0 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
<b>SCON2</b>	SM2S	–	–	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 **SM2S**: 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 Mode0 / Mode2)**
- 8Eh.4 **REN2**: UART2 接收使能  
 0: 禁止接收  
 1: 允许接收
- 8Eh.3 **TB82**: UART2 传输位 8, 即在模式 3 中要传输的第 9 位
- 8Eh.2 **RB82**: UART2 接收位 8, 包含在模式 3 中接收到的第 9 位
- 8Eh.1 **TI2**: UART2 发送中断标志  
 在模式 1 和 3 中, 在停止位的开头通过 H/W 进行设置。必须通过 S/W 进行清除。
- 8Eh.0 **RI2**: UART2 接收中断标志  
 在模式 1 和 3 中, 在停止位的采样点由 H/W 设置。必须由 S/W 清除。

SFR 8Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SBUF2</b>	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。

SFR 9Ah	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SCON1</b>	SM1S	–	–	REN1	TB81	RB81	TI1	RI1
R/W	R/W	–	–	R/W	R/W	R/W	R/W	R/W
Reset	0	–	–	0	0	0	0	0

- 9Ah.7 **SM1S**: UART1 串行端口模式选择位  
 0: Mode1: 8 位 UART1, 波特率可变  
 1: Mode3: 9 位 UART1, 波特率可变  
**(UART1 不支持 Mode0 / Mode2)**
- 9Ah.4 **REN1**: UART1 接收使能  
 0: 禁止接收  
 1: 允许接收
- 9Ah.3 **TB81**: UART1 传输位 8, 即在模式 3 中要传输的第 9 位
- 9Ah.2 **RB81**: UART1 接收位 8, 包含在模式 3 中接收到的第 9 位
- 9Ah.1 **TI1**: UART1 发送中断标志  
 在模式 1 和 3 中, 在停止位的开头通过 H/W 进行设置。必须通过 S/W 进行清除。
- 9Ah.0 **RI1**: UART1 接收中断标志  
 在模式 1 和 3 中, 在停止位的采样点由 H/W 设置。必须由 S/W 清除。

SFR 9Bh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SBUF1</b>	SBUF1							
R/W	R/W							
Reset	-	-	-	-	-	-	-	-

9Bh.7~0 **SBUF1**: 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 A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE1</b>	PWMIE	I2CE	ES2	SPIE	ADTKIE	LVDIE	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.5 **ES2**: Serial Port (UART1/UART2) 中断使能  
 0: 禁用 Serial Port (UART1/UART2) 中断  
 1: 允许 Serial Port (UART1/UART2) 中断

SFR D5h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>UART2CON</b>	-	UART2BRP						
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0

D5h.6~0 **UART2BRP**: UART2 波特率预分频器

SFR DDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>UART1CON</b>	-	UART1BRP						
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	-	0	0	0	0	0	0	0

DDh.6~0 **UART1BRP**: UART1 波特率预分频器

SFR DEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>UART0CON</b>	UART0BRS	UART0BRP						
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

DEh.7 **UART0BRS**: UART0 波特率源选择

DEh.6~0 **UART0BRP**: UART0 波特率预分频器

## 10. PWMs

该芯片具有九个独立的 16 位 PWM 模块 PWM0, PWM1, PWM2 和 PWM30~PWM35。PWM0~2 具有独立的 16 位周期。PWM30~P35 共享一组 16 位周期。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<sub>SYSCLK</sub> 作为其时钟源。

使用 SFR PINMOD<sub>x</sub> 控制 PWM 输出到 IO 并设置 PWM<sub>x</sub>EN 以启用 PWM 功能。例如, PORTIDX=1, PIMOD76=BBh, PWM<sub>x</sub>EN=1, 那么 PWM1 和 PWM0 会输出到 IO。(见第 7 节)

16 位周期和占空比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。**

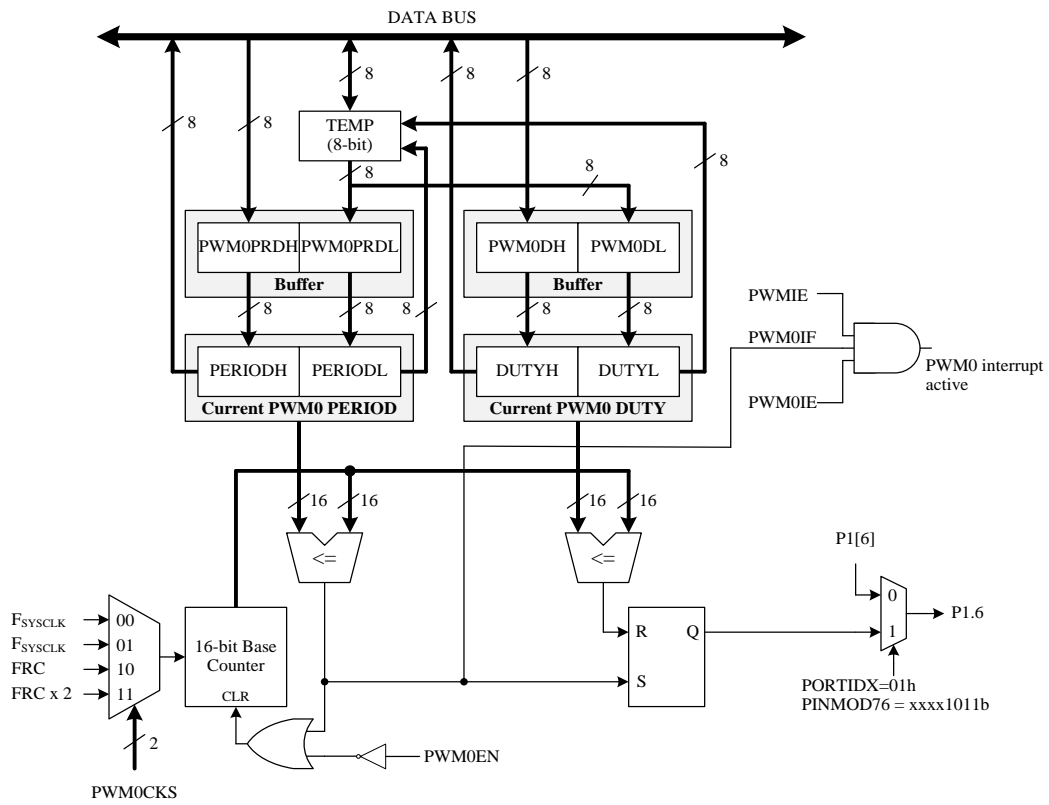
当 PWM0EN 位置 1 时, PWM0 将运行,否则 PWM0 清除并保持。PWM0 的结构如下所示。可以通过在 PWMIDX = 0 时写入 PWMDH 和 PWMDL 来更改 PWM0 占空比。每当 16 位基数计数器与 16 位 PWM0 占空比寄存器{PWM0DH, PWM0DL}匹配时, PWM0 输出信号就会复位为低电平。可以通过在 PWMIDX = 0 时将周期值写入 PWMPRDH 和 PWMPRDL 寄存器来设置 PWM0 周期。写入 PWM 占空比或周期寄存器后,新值将立即保存到其自己的缓冲区中。硬件将在当前周期结束时或清除 PWM0 时更新这些值。PWM0~3 有一个相应的中断标志,在周期结束时产生一个中断标志。

PWMDH, PWMDL, PWMPRDH 或 PWMPRDL 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,建议只在主程序内更新数据,或只在中断内更新数据,以避免可能的错误产生。

如下表所示,使用 PWMIDX 设置 PWM0~3 的 16 位周期和占空比如下:

PWMIDX (SFR 9Eh)	PWMPRDH (SFR D9h)	PWMPRDL (SFR DAh)	PWMDH (SFR D1h)	PWMDL (SFR D2h)
0xh	PWM0PRDH	PWM0PRDL	PWM0DH	PWM0DL
1xh	PWM1PRDH	PWM1PRDL	PWM1DH	PWM1DL
2xh	PWM2PRDH	PWM2PRDL	PWM2DH	PWM2DL
30h	PWM3PRDH	PWM3PRDL	PWM30DH	PWM30DL
31h			PWM31DH	PWM31DL
32h			PWM32DH	PWM32DL
33h			PWM33DH	PWM33DL
34h			PWM34DH	PWM34DL
35h			PWM35DH	PWM35DL

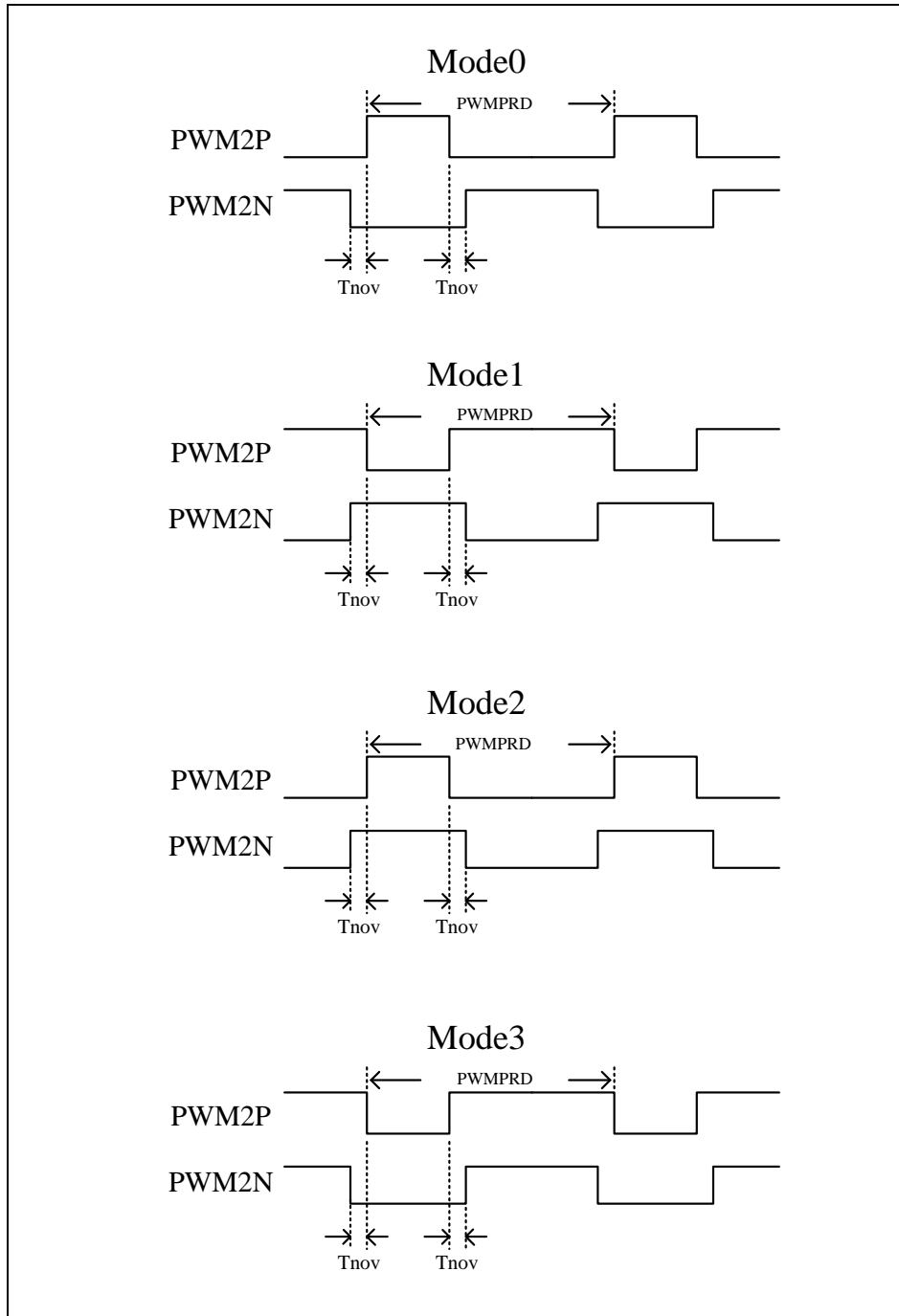
表 10.1 PWM0~2, PWM30~PWM35 周期和占空比索引表



PWM0 结构



只有 PWM2 可以通过 PWM2P 和 PWM2N 输出，有四种不同的模式。PWM 脉冲的边沿可以用 16 个不同的时间非重叠时钟间隔 ( $T_{nov}$ ) 分开。 $T_{nov}$  的宽度可以通过 PWM2DZ 在 0~15 pwm 时钟范围内选择。默认输出形式为 Mode2。四种输出模式的波形如下图所示。



**PWM2 Waveform Modes**

SFR 86h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTPWM</b>	—	—	—	—	PWM3IF	PWM2IF	PWM1IF	PWM0IF
R/W	—	—	—	—	R/W	R/W	R/W	R/W
Reset	—	—	—	—	0	0	0	0

- 86h.2 **PWM3IF:**  
 0: S / W 写入 0 清除  
 1: 周期结束由硬件设置
- 86h.2 **PWM2IF:**  
 0: S / W 写入 0 清除  
 1: 周期结束由硬件设置
- 86h.1 **PWM1IF:**  
 0: S / W 写入 0 清除  
 1: 周期结束由硬件设置
- 86h.0 **PWM0IF:**  
 0: S / W 写入 0 清除  
 1: 周期结束由硬件设置

SFR 9Dh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWM2CON</b>	PWM2OM			PWM2DZ				
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	1	0	0	0	0	0	0	0

- 9Dh.7~6 **PWM2OM:** PWM2 输出模式  
 00: 模式 0  
 01: 模式 1  
 10: 模式 2  
 11: 模式 3
- 9Dh.5~0 **PWM2DZ:** PWM2 死区控制  
 0000: 死区禁用  
 0001: 死区宽度 1\*Tpwmclk  
 0010: 死区宽度 2\*Tpwmclk  
 ...  
 1111: 死区宽度 15\*Tpwmclk

SFR 9Eh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMIDX</b>	PWMIDX							
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

- 9Eh.7~0 **PWMIDX:** PWM 周期和占空比指标。 有关更多详细信息，请参见表 10.1  
 0xh: PWM0 周期/占空比访问  
 1xh: PWM1 周期/占空比访问  
 2xh: PWM2 周期/占空比访问  
 3xh: PWM30~PWM35 周期/占空比访问  
 30h: PWM30 周期/占空比访问  
 31h: PWM31 周期/占空比访问  
 32h: PWM32 周期/占空比访问  
 33h: PWM33 周期/占空比访问  
 34h: PWM34 周期/占空比访问  
 35h: PWM35 周期/占空比访问

SFR 9Fh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMEN</b>	PWM3IE	PWM2IE	PWM1IE	PWM0IE	PWM3EN	PWM2EN	PWM1EN	PWM0EN
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

- 9Fh.7 **PWM3IE:** PWM3 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.6 **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.5 **PWM1IE:** PWM1 Interrupt Enable  
 0: 禁用  
 1: 允许(注意: PWMIE 必须同时为 1 才能产生 PWM 中断)
- 9Fh.4 **PWM0IE:** PWM0 Interrupt Enable  
 0: 禁用  
 1: 允许(注意: PWMIE 必须同时为 1 才能产生 PWM 中断)
- 9Fh.3 **PWM3EN:**  
 0: PWM3 被清除并保持      1: PWM3 正在运行
- 9Fh.2 **PWM2EN:**  
 0: PWM2 被清除并保持      1: PWM2 正在运行
- 9Fh.1 **PWM1EN:**  
 0: PWM1 被清除并保持      1: PWM1 正在运行
- 9Fh.0 **PWM0 EN:**  
 0: PWM0 被清除并保持      1: PWM0 正在运行

SFR A1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMCON</b>	PWM3CKS		PWM2CKS		PWM1CKS		PWM0CKS	
R/W	R/W	R/W	R/W		R/W		R/W	
Reset	1	0	1	0	1	0	1	0

- A1h.7~6 **PWMCKS:** PWM3 时钟源  
 00:F<sub>SYSC</sub>CLK  
 01:F<sub>SYSC</sub>CLK  
 10:FRC  
 11:FRC x 2
- A1h.5~4 **PWM0CKS:**PWM2 时钟源  
 00:F<sub>SYSC</sub>CLK  
 01:F<sub>SYSC</sub>CLK  
 10:FRC  
 11:FRC x 2
- A1h.3~2 **PWM0CKS:**PWM1 时钟源  
 00:F<sub>SYSC</sub>CLK  
 01:F<sub>SYSC</sub>CLK  
 10:FRC  
 11:FRC x 2
- A1h.1~0 **PWM0CKS:**PWM0 时钟源  
 00:F<sub>SYSC</sub>CLK  
 01:F<sub>SYSC</sub>CLK  
 10:FRC  
 11:FRC x 2

SFR A9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>INTE1</b>	PWMIE	I2CE	ES2	SPIE	ADTKIE	LVDIE	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**: PWM0~3 中断使能  
 0: 禁止 PWM0~3 中断  
 1: 使能 PWM0~3 中断

SFR D1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMDH</b>	PWMDH							
R/W	R/W							
Reset	1	0	0	0	0	0	0	0

D1h.7~0 **PWMDH**: PWM 占空比高字节, 通过 PWMIDX 索引  
 有关更多详细信息, 请参见表 10.1  
 PWMIDX = 0xh: PWM0DH 访问  
 PWMIDX = 1xh: PWM1DH 访问  
 PWMIDX = 2xh: PWM2DH 访问  
 PWMIDX = 30h: PWM30DH 访问  
 PWMIDX = 31h: PWM31DH 访问  
 PWMIDX = 32h: PWM32DH 访问  
 PWMIDX = 33h: PWM33DH 访问  
 PWMIDX = 34h: PWM34DH 访问  
 PWMIDX = 35h: PWM35DH 访问  
 Note :  
 写顺序: 先写 PWMDL, 然后再写 PWMDH  
 读顺序: 先读 PWMDH, 然后再读 PWMDL

SFR D2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMDL</b>	PWMDL							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

D2h.7~0 **PWMDL**: PWM 占空比低字节, 通过 PWMIDX 索引  
 有关更多详细信息, 请参见表 10.1  
 PWMIDX = 0xh: PWM0DL 访问  
 PWMIDX = 1xh: PWM1DL 访问  
 PWMIDX = 2xh: PWM2DL 访问  
 PWMIDX = 30h: PWM30DL 访问  
 PWMIDX = 31h: PWM31DL 访问  
 PWMIDX = 32h: PWM32DL 访问  
 PWMIDX = 33h: PWM33DL 访问  
 PWMIDX = 34h: PWM34DL 访问  
 PWMIDX = 35h: PWM35DL 访问  
 Note :  
 写顺序: 先写 PWM0DL, 然后再写 PWM0DH  
 读顺序: 先读 PWM0DH, 然后再读 PWM0DL

SFR D9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMPRDH</b>	PWMPRDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

D9h.7~0 **PWMPRDH**: PWM 周期高字节, 由 PWMIDX 索引  
 有关更多详细信息, 请参见表 10.1  
 PWMIDX = 0xh: PWM0PRDH 访问  
 PWMIDX = 1xh: PWM1PRDH 访问  
 PWMIDX = 2xh: PWM2PRDH 访问  
 PWMIDX = 3xh: PWM3PRDH 访问

Note :  
 写顺序: 先写 PWMPRDL, 然后再写 PWMPRDH  
 读顺序: 先读 PWMPRDH, 然后再读 PWMPRDL

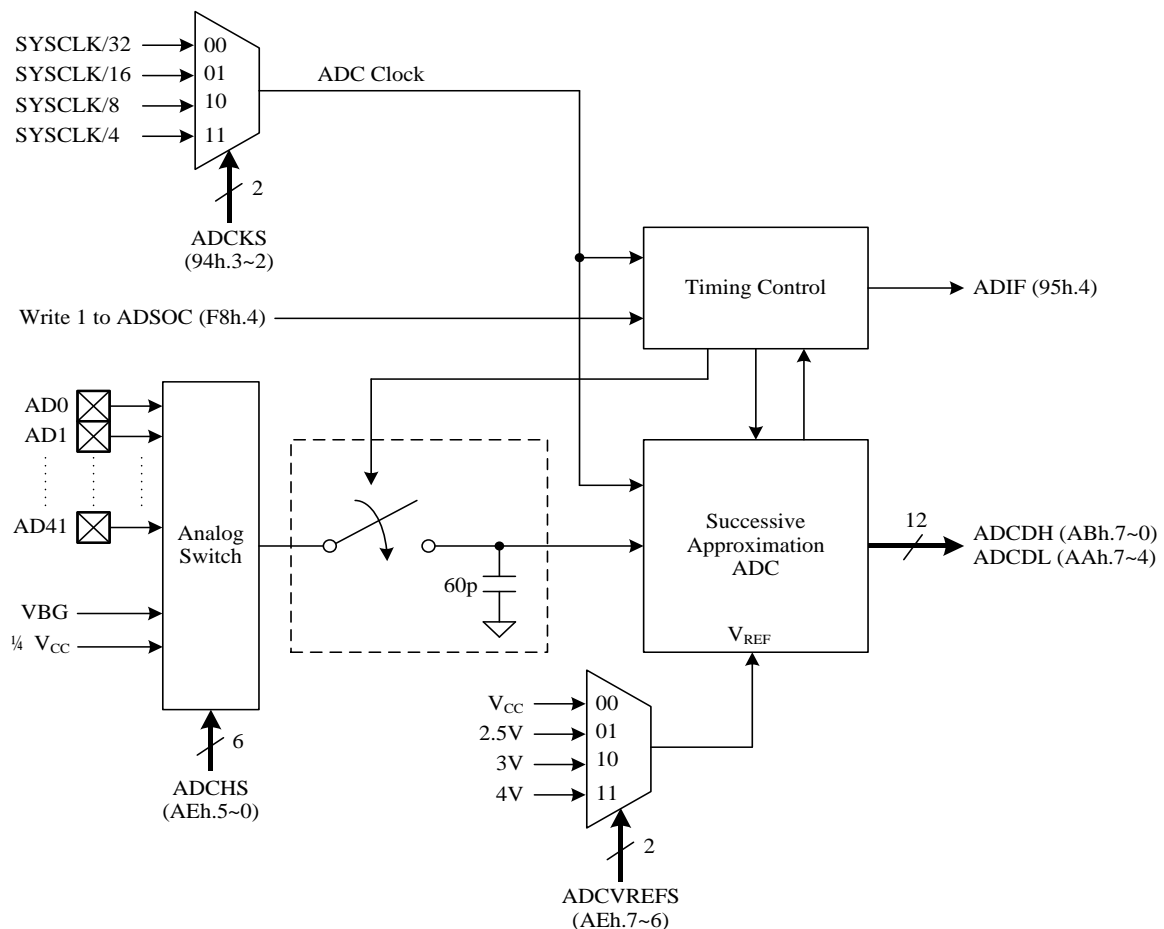
SFR DAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PWMPRDL</b>	PWMPRDL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

Dah.7~0 **PWMPRDL**: PWM 周期高字节, 由 PWMIDX 索引  
 有关更多详细信息, 请参见表 10.1  
 PWMIDX = 0xh: PWM0PRDH 访问  
 PWMIDX = 1xh: PWM1PRDH 访问  
 PWMIDX = 2xh: PWM2PRDH 访问  
 PWMIDX = 3xh: PWM3PRDH 访问

Note :  
 写顺序: 先写 PWMPRDL, 然后再写 PWMPRDH  
 读顺序: 先读 PWMPRDH, 然后再读 PWMPRDL

## 11. ADC

该芯片提供了一个 12 位 ADC 包括了 44 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 H/W 将自动清除它。转换结束后 H/W 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 信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。可以从以下四个电压中选择 ADC 的  $V_{REF}$ :  $V_{CC}$ , 2.5V, 3V 和 4V。ADCHS 選擇到 VBG 時, ADCVREFS 必需設置為  $V_{CC}$ , 否則 ADC 轉換無效。

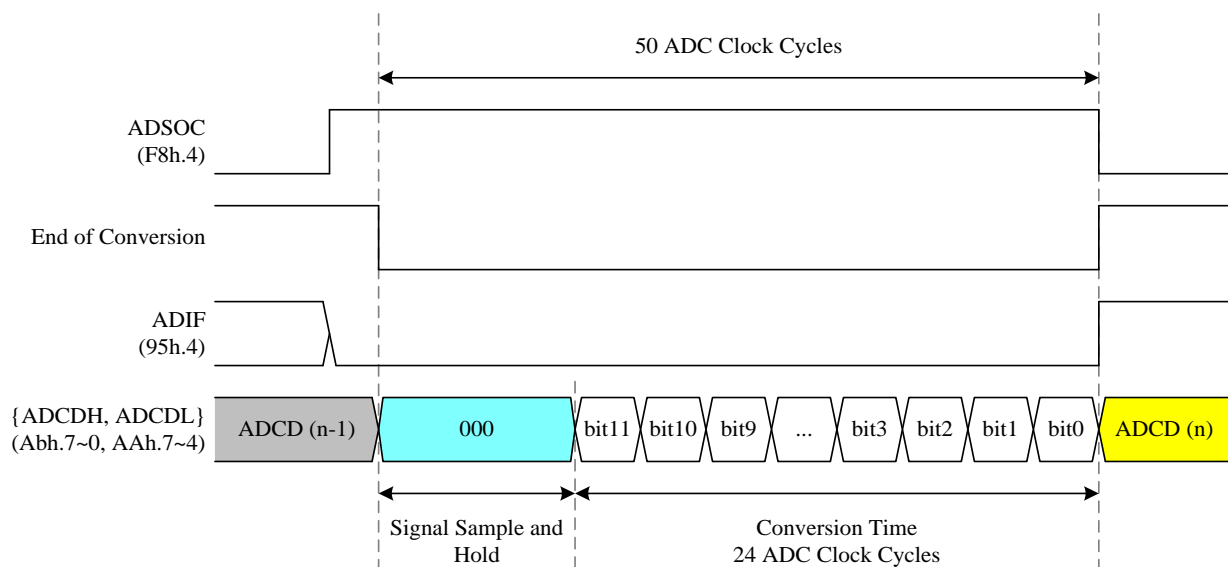


## 11.1 ADC 通道

12 位 ADC,一共有 44 个通道,指定为 AD0~AD41, VBG 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。VBG 是内部基准电压 1.22V。当 ADC 通道选择到 VBG 时, VBG 发生器将自动启用。通过设置 SFR VBGEN = 1 始终使能 VBG 发生器, 用户可以获得更稳定的 VBG 电压。ADCHS 選擇到 VBG 時, ADCVREFS 必需設置為 VCC, 否則 ADC 轉換無效。

## 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 H/W 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
<b>OPTION</b>	TM3CKS		WDTM3PSC		ADCKS		TM3PSC	
R/W	R/W	R/W	R/W		R/W		R/W	
Reset	0	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
<b>INTFLG</b>	LVDIF	-	TKIFA	ADIF	-	-	PCIF	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 H/W 设置。S/W 写入 EFh 到 INTFLG 或设置 ADSOC 位来清除该标志。

SFR A6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>PINMODE</b>	VBGEN	–	UART1PS	PSEUDOEN	I2CPS		UART0PS	
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.7 **VBGEN**: 强制 VBG 生成器启用  
 0: VBG 生成器自动启用和禁用  
 1: 强制 VBG 发生器启用包括在 IDLE 模式下, 但在停止/暂停模式下禁用

SFR AAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ADC DL</b>	ADC DL				–	–	–	–
R/W	R				–	–	–	–
Reset	–	–	–	–	–	–	–	–

Aah.7~4 **ADC DL**: ADC 数据位 3~0

SFR ABh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ADC DH</b>	ADC DH							
R/W	R							
Reset	–	–	–	–	–	–	–	–

ABh.7~0 **ADC DH**: ADC 数据位 11~4

SFR AEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>CHSEL</b>	ADC VREFS		ADC HS					
R/W	R/W		R/W	R/W	R/W	R/W	R/W	R/W
Reset	0	0	1	1	1	1	1	1

AEh.7~6 **ADC VREFS**: ADC 参考电压  
 ADC HS 选择到 VBG 时, ADC VREFS 必需设置为 VCC, 否则 ADC 转换无效  
 00: VCC  
 01: 2.5V  
 10: 3V  
 11: 4V

AEh.5~0 **ADC HS**: ADC 通道选择  
 000000: AD00  
 000001: AD01  
 ...  
 101001: AD41  
 101011: V<sub>BG</sub> (内部基准电压源)  
 101100: 1/4V<sub>CC</sub> (内部基准电压源)

SFR F8h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX1</b>	CLRWDT	CLRTM3	TKSOCA	ADSOC	–	TKSOCB	TISEL	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.4 **ADSOC**: 启动 ADC 转换  
 设置 ADSOC 位启动 ADC 转换, ADSOC 位将由 H/W 于转换结束时被清除。S/W 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 2 组 21 个通道的触摸键检测。

要使用触摸键，用户应正确设置引脚。有两种方法可以将引脚设置为 TK 通道。将 SFR PINMOD<sub>x</sub> 设置为 0011b 或经由 SFR TKPinsa / B 0~2 将引脚设置为 TK 通道。如果设置了 TKPinsa/B 0~2，则相应的引脚将被固定为 TK 通道，并且将不再受 PINMOD<sub>x</sub> 的影响。

TKPinsa	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKPinsa0</b>	TKA07	TKA06	TKA05	TKA04	TKA03	TKA02	TKA01	TKA00
<b>TKPinsa1</b>	TKA15	TKA14	TKA13	TKA12	TKA11	TKA10	TKA09	TKA08
<b>TKPinsa2</b>				TKA20	TKA19	TKA18	TKA17	TKA16

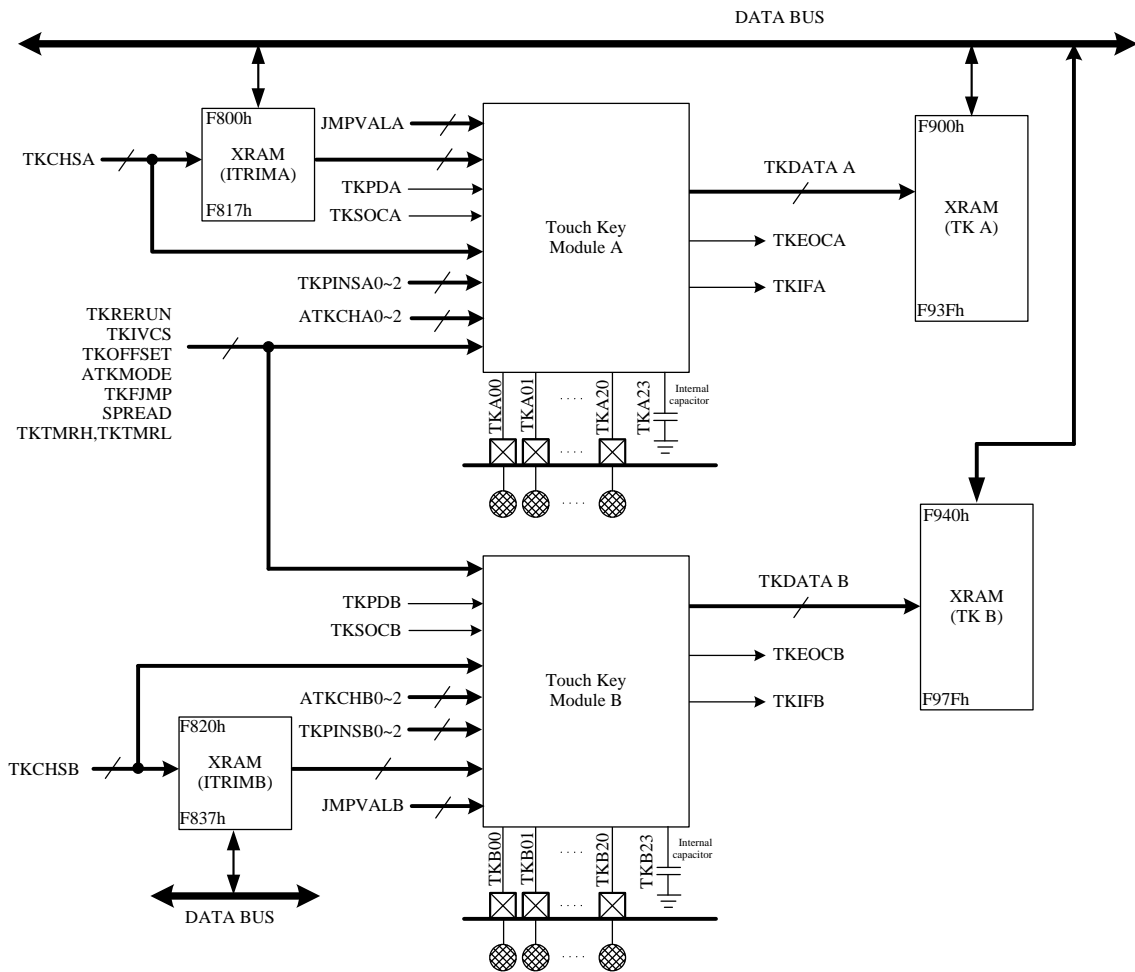
设置 TKPinsa0~2 将 IO 固定为 TKA 通道

TKPinsa	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKPinsb0</b>	TKB07	TKB06	TKB05	TKB04	TKB03	TKB02	TKB01	TKB00
<b>TKPinsb1</b>	TKB15	TKB14	TKB13	TKB12	TKB11	TKB10	TKB09	TKB08
<b>TKPinsb2</b>				TKB20	TKB19	TKB18	TKB17	TKB16

设置 TKPinsb0~2 将 IO 固定为 TKB 通道

在 TK 模式下，用户分配 TKPD = 0 来打开 TK 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<sub>SYSClk</sub> 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 TK，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按键行为。设置 TKCHS = 17h 并开始扫描可以获取内部参考电容器 (TKCAP) 的 TK 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。



**FTK 触摸按键结构**

SFR ATKCHA / B0~2 用于指定扫描 TK 通道，并且每个位都映射到 TK 引脚。TK 扫描将从低位扫描到高位。如果 ATKMODE = 0，则 TK 最多可以扫描 22 个通道，TK00~TK20 和 TKCAP（TK23），每个通道扫描一次。如果 ATKMODE = 1，则 TK 最多可以扫描 16 个信道，每个信道扫描两次。如果 ATKMODE = 2，则 TK 最多可以扫描 8 个信道，每个信道被扫描 4 次。如果 ATKMODE = 3，则 TK 最多可以扫描 4 个信道，每个信道被扫描 8 次。TKCHSA 和 TKCHSB 是用来指定 TK 开始扫描的第一个通道。

例如：

条件 ATKMODE = 0，扫描 TKA16 / TKA14 / TKA08 / TKA07 / TKA06 / TKA02

⇒ TKPinsa2=0000\_0001, TKPinsa1=0100\_0001, TKPinsa0=1100\_0100

⇒ ATKCHA2=0000\_0001, ATKCHA1=0100\_0001, ATKCHA0=1100\_0100

⇒ TKCHSA=0x02 (指定第一个扫描通道)

存储在 XRAM 中的 TK 数据的排列如下。

XRAM	
F900h	TKA00 DATAL
F901h	TKA00 DATAH
F902h	TKA01 DATAL
F903h	TKA01 DATAH
...	
F928h	TKA20 DATAL
F929h	TKA20 DATAH
...	
F92Eh	TKA23 DATAL
F92Fh	TKA23 DATAH

条件 ATKMODE = 1，扫描 TKA16 / TKA14 / TKA08 / TKA07 / TKA06 / TKA02

⇒ TKPinsa2=0000\_0001, TKPinsa1=0100\_0001, TKPinsa0=1100\_0100

⇒ ATKCHA2=0000\_0001, ATKCHA1=0100\_0001, ATKCHA0=1100\_0100

⇒ TKCHSA=0x02 (指定第一个扫描通道)

存储在 XRAM 中的 TK 数据的排列如下。

XRAM	
F900h	TKA02 1 <sup>st</sup> DATAL
F901h	TKA02 1 <sup>st</sup> DATAH
F902h	TKA02 2 <sup>nd</sup> DATAL
F903h	TKA02 2 <sup>nd</sup> DATAH
F904h	TKA06 1 <sup>st</sup> DATAL
F905h	TKA06 1 <sup>st</sup> DATAH
F906h	TKA06 2 <sup>nd</sup> DATAL
F907h	TKA06 2 <sup>nd</sup> DATAH
...	
F914h	TKA16 1 <sup>st</sup> DATAL
F915h	TKA16 1 <sup>st</sup> DATAH
F916h	TKA16 2 <sup>nd</sup> DATAL
F917h	TKA16 2 <sup>nd</sup> DATAH

TK 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 TKA16 / TKA14 / TKA08 / TKA07 / TKA06 / TKA02

- ⇒ TKPinsa2=0000\_0001, TKPinsa1=0100\_0001, TKPinsa0=1100\_0100
- ⇒ ATKCHA2=0000\_0001, ATKCHA1=0100\_0001, ATKCHA0=1100\_0100
- ⇒ TKCHSA=0x02 (指定第一个扫描通道)

存储在 XRAM 中的 TK 数据的排列如下。

XRAM	
F900h	TKA02 1 <sup>st</sup> DATAL
F901h	TKA02 1 <sup>st</sup> DATAH
F902h	TKA02 2 <sup>nd</sup> DATAL
F903h	TKA02 2 <sup>nd</sup> DATAH
F904h	TKA02 3 <sup>rd</sup> DATAL
F905h	TKA02 3 <sup>rd</sup> DATAH
F906h	TKA02 4 <sup>th</sup> DATAL
F907h	TKA02 4 <sup>th</sup> DATAH
F908h	TKA06 1 <sup>st</sup> DATAL
F909h	TKA06 1 <sup>st</sup> DATAH
F90Ah	TKA06 2 <sup>nd</sup> DATAL
F90Bh	TKA06 2 <sup>nd</sup> DATAH
F90Ch	TKA06 3 <sup>rd</sup> DATAL
F90Dh	TKA06 3 <sup>rd</sup> DATAH
F90Eh	TKA06 4 <sup>th</sup> DATAL
F90Fh	TKA06 4 <sup>th</sup> DATAH
	...
F928h	TKA16 1 <sup>st</sup> DATAL
F929h	TKA16 1 <sup>st</sup> DATAH
F92Ah	TKA16 2 <sup>nd</sup> DATAL
F92Bh	TKA16 2 <sup>nd</sup> DATAH
F92Ch	TKA16 3 <sup>rd</sup> DATAL
F92Dh	TKA16 3 <sup>rd</sup> DATAH
F92Eh	TKA16 4 <sup>th</sup> DATAL
F92Fh	TKA16 4 <sup>th</sup> DATAH

TK 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 TKA08 / TKA07 / TKA06 / TKA02

- ⇒ TKPinsa2=0000\_0000, TKPinsa1=0000\_0001, TKPinsa0=1100\_0100
- ⇒ ATKCHA2=0000\_0000, ATKCHA1=0000\_0001, ATKCHA0=1100\_0100
- ⇒ TKCHSA=0x02 (指定第一个扫描通道)

存储在 XRAM 中的 TK 数据的排列如下。

XRAM	
F900h	TKA02 1 <sup>st</sup> DATAL
F901h	TKA02 1 <sup>st</sup> DATAH
F902h	TKA02 2 <sup>nd</sup> DATAL
F903h	TKA02 2 <sup>nd</sup> DATAH
F904h	TKA02 3 <sup>rd</sup> DATAL
F905h	TKA02 3 <sup>rd</sup> DATAH
F906h	TKA02 4 <sup>th</sup> DATAL
F907h	TKA02 4 <sup>th</sup> DATAH
F908h	TKA02 5 <sup>th</sup> DATAL
F909h	TKA02 5 <sup>th</sup> DATAH
F90Ah	TKA02 6 <sup>th</sup> DATAL
F90Bh	TKA02 6 <sup>th</sup> DATAH
F90Ch	TKA02 7 <sup>th</sup> DATAL
F90Dh	TKA02 7 <sup>th</sup> DATAH
F90Eh	TKA02 8 <sup>th</sup> DATAL
F90Fh	TKA02 8 <sup>th</sup> DATAH
	...
F930h	TKA08 1 <sup>st</sup> DATAL
F931h	TKA08 1 <sup>st</sup> DATAH
F932h	TKA08 2 <sup>nd</sup> DATAL
F933h	TKA08 2 <sup>nd</sup> DATAH
F934h	TKA08 3 <sup>rd</sup> DATAL
F935h	TKA08 3 <sup>rd</sup> DATAH
F936h	TKA08 4 <sup>th</sup> DATAL
F937h	TKA08 4 <sup>th</sup> DATAH
F938h	TKA08 5 <sup>th</sup> DATAL
F939h	TKA08 5 <sup>th</sup> DATAH
F93Ah	TKA08 6 <sup>th</sup> DATAL
F93Bh	TKA08 6 <sup>th</sup> DATAH
F93Ch	TKA08 7 <sup>th</sup> DATAL
F93Dh	TKA08 7 <sup>th</sup> DATAH
F93Eh	TKA08 8 <sup>th</sup> DATAL
F93Fh	TKA08 8 <sup>th</sup> DATAH

TK 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
<b>INTFLG</b>	LVDIF	–	TKIFA	ADIF	–	–	PCIF	TF3
R/W	R	–	R/W	R/W	–	–	R/W	R/W
Reset	–	–	0	0	–	–	0	0

95h.5 **TKIFA:** 触摸键 A 中断标志

如果  $F_{SYSCLK}$  足够快，则在触摸按键转换结束时通过 H/W 设置。S/W 将 DFh 写入 INTFLG 或将 TKSOCA 设置 1 以清除该标志。

SFR 9Ch	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKCON3</b>	TKPDB	TKEOCB	TKIFB	TKXCAPB	JMPVALB			SPREAD
R/W	R/W	R	R/W	R/W	R/W			R/W
Reset	1	1	0	0	0	0	0	0

9Ch.7 **TKPDB:** 触摸键 B

0: 触摸键 B 启用

1: 触摸键 B 禁用

9C h.6 **TKEOCB:** 触摸键转换标志的末尾，在  $TKSOCB = 1$  之后 TKEOCB 可能有 3uS 的延迟，因此 F/W 必须等待足够的时间才能轮询此标志。

0: 表示正在进行转换

1: 表示转换完成

9C h.5 **TKIFB:** 触摸键 B 中断标志

如果  $F_{SYSCLK}$  足够快，则在触摸键 B 转换结束时由 H/W 进行设置。S/W TKIFB 写入 0 或将 TKSOCB 位置 1 以清除该标志。

9C h.4 **TKXCAPB:** 触摸键 B 外部电容器选择

0: 保持 0, 禁用触摸键外部电容器

1: 保留, 请勿设置为 1

9C h.3~1 **JMPVALB:** 触摸键时钟频率微调，仅在  $TKFJMP = 0$  中可用

00 = 频率最低, 111 = 频率最高

9C h.0 **SPREAD:** 内部触摸键时钟频率自动调整选项

0: 禁用

1: 启用

SFR ADh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKCON</b>	TKPDA	TKEOCA	TKRERUN	TKIVCS	TKXCAPA	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 **TKPDA:** 触摸键 A

0: 触摸键 A 启用

1: 触摸键 A 禁用

ADh.6 **TKEOCA:** 触摸键转换标志结束时, 在 TKSOCA = 1 之后 TKEOCA 可能有 3 $\mu$ s 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:** TK A / B 自动重新启动, 无需重新设置 TKSOCA / B 即可重新启动 TK A / B 转换器。

0: 自动重启禁用。 每次 TK A / B 转换都需要执行一次 TKSOCA / B

1: 自动重启启用。 TKSOCA / B 执行一次后, TK A / B 将连续转换而无需重新执行 TKSOCA / B

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 **TKXCAPA:** 触摸键 A 外部电容器选择

0: 保持 0, 禁用触摸键外部电容器

1: 保留, 请勿设置为 1

ADh.2 **TKOFFSET:** 非扫描 TK 的状态

0: 连接到 VSS

1: 连接到交流屏蔽, 连接到 VSS @ EOC

ADh.1~0 **ATKMODE:** 触摸键扫描模式

00: TKA 和 TKB 扫描方法, 每个通道扫描 1 次, 最多 22 个 TK 通道

01: TKA 和 TKB 扫描方法, 每个通道扫描 2 次, 最多 16 个 TK 通道

10: TKA 和 TKB 扫描方法, 每个通道扫描 4 次, 最多 8 个 TK 通道

11: TKA 和 TKB 扫描方法, 每个通道扫描 8 次, 最多 4 个 TK 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
<b>TKTMRL</b>	TKTMRL							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

B4h.7~0 **TKTMRL**: 触摸键 A / B 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
<b>TKCON2</b>	TKFJMP	JMPVALA			TKTMRH			
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.6~4 **JMPVALA**: 触摸键 A 时钟频率微调, 仅在 TKFJMP = 0 中可用  
000 = 频率最低, 111 = 频率最高

B5h.3~0 **TKTMRH**: 触摸键 A / B 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
<b>AUX1</b>	CLRWDT	CLRTM3	TKSOCA	ADSOC	-	TKSOCB	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.5 **TKSOCA**: 触摸键A转换开始  
设置为1即可开始触摸键A转换。如果F<sub>SYSCLK</sub>足够快且TKRERUN = 0时, 则在转换结束时, 该位将由H / W清除。S / W也可以写0清除该标志。

F8h.2 **TKSOCB**: 触摸键B转换开始  
设置为1即可开始触摸键B转换。如果F<sub>SYSCLK</sub>足够快且TKRERUN = 0时, 则在转换结束时, 该位将由H / W清除。S / W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
<b>TKCHSA</b>	–	–	–	TKCHSA				
R/W	–	–	–	R/W				
Reset	–	–	–	1	1	1	1	1

A7h.4~0 **TKCHSA:** 指定触摸按键 A 第一个扫描通道

00000: TKA00  
 00001: TKA01  
 00010: TKA02  
 00011: TKA03  
 00100: TKA04  
 00101: TKA05  
 00110: TKA06  
 00111: TKA07  
 01000: TKA08  
 01001: TKA09  
 01010: TKA10  
 01011: TKA11  
 01100: TKA12  
 01101: TKA13  
 01110: TKA14  
 01111: TKA15  
 10000: TKA16  
 10001: TKA17  
 10010: TKA18  
 10011: TKA19  
 10100: TKA20  
 10111: TKACAP 内部参考电容通道

SFR Ach	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKCHSB</b>	–	–	–	TKCHSB				
R/W	–	–	–	R/W				
Reset	–	–	–	1	1	1	1	1

Ach.4~0 **TKCHSB:** 指定触摸按键 B 第一个扫描通道

00000: TKB00  
 00001: TKB01  
 00010: TKB02  
 00011: TKB03  
 00100: TKB04  
 00101: TKB05  
 00110: TKB06  
 00111: TKB07  
 01000: TKB08  
 01001: TKB09  
 01010: TKB10  
 01011: TKB11  
 01100: TKB12  
 01101: TKB13  
 01110: TKB14  
 01111: TKB15  
 10000: TKB16  
 10001: TKB17  
 10010: TKB18  
 10011: TKB19  
 10100: TKB20  
 10111: TKBCAP 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
<b>TKPinsa0</b>	<b>TKPinsa0</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C1h.7 TKA07 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C1h.6 TKA06 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C1h.5 TKA05 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C1h.4 TKA04 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C1h.3 TKA03 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C1h.2 TKA02 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C1h.1 TKA01 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C1h.0 TKA00 引脚固定为 TK 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
<b>TKPinsa1</b>	<b>TKPinsa1</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C2h.7 TKA15 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C2h.6 TKA14 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C2h.5 TKA13 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C2h.4 TKA12 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C2h.3 TKA11 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C2h.2 TKA10 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C2h.1 TKA09 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C2h.0 TKA08 引脚固定为 TK 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
<b>TKPinsa2</b>	<b>TKPinsa2</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C3h.7~5 Reservd  
 C3h.4 TKA20 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C3h.3 TKA19 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C3h.2 TKA18 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C3h.1 TKA17 引脚固定为 TK 通道: 0: 禁用 1: 启用  
 C3h.0 TKA16 引脚固定为 TK 通道: 0: 禁用 1: 启用

SFR C4h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKPINSB0</b>	<b>TKPINSB0</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C4h.7 TKB07 引脚固定为 TK 通道：0：禁用 1：启用  
 C4h.6 TKB06 引脚固定为 TK 通道：0：禁用 1：启用  
 C4h.5 TKB05 引脚固定为 TK 通道：0：禁用 1：启用  
 C4h.4 TKB04 引脚固定为 TK 通道：0：禁用 1：启用  
 C4h.3 TKB03 引脚固定为 TK 通道：0：禁用 1：启用  
 C4h.2 TKB02 引脚固定为 TK 通道：0：禁用 1：启用  
 C4h.1 TKB01 引脚固定为 TK 通道：0：禁用 1：启用  
 C4h.0 TKB00 引脚固定为 TK 通道：0：禁用 1：启用

SFR D7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKPINSB1</b>	<b>TKPINSB1</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- D7h.7 TKB15 引脚固定为 TK 通道：0：禁用 1：启用  
 D7h.6 TKB14 引脚固定为 TK 通道：0：禁用 1：启用  
 D7h.5 TKB13 引脚固定为 TK 通道：0：禁用 1：启用  
 D7h.4 TKB12 引脚固定为 TK 通道：0：禁用 1：启用  
 D7h.3 TKB11 引脚固定为 TK 通道：0：禁用 1：启用  
 D7h.2 TKB10 引脚固定为 TK 通道：0：禁用 1：启用  
 D7h.1 TKB09 引脚固定为 TK 通道：0：禁用 1：启用  
 D7h.0 TKB08 引脚固定为 TK 通道：0：禁用 1：启用

SFR DFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>TKPINSB2</b>	<b>TKPINSB2</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- DFh.7~5 Reservd  
 DFh.4 TKB20 引脚固定为 TK 通道：0：禁用 1：启用  
 DFh.3 TKB19 引脚固定为 TK 通道：0：禁用 1：启用  
 DFh.2 TKB18 引脚固定为 TK 通道：0：禁用 1：启用  
 DFh.1 TKB17 引脚固定为 TK 通道：0：禁用 1：启用  
 DFh.0 TKB16 引脚固定为 TK 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
<b>ATKCHA0</b>	<b>ATKCHA0</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C5h.7 TKA07 扫描启用: 0: 禁用 1: 启用  
 C5h.6 TKA06 扫描启用: 0: 禁用 1: 启用  
 C5h.5 TKA05 扫描启用: 0: 禁用 1: 启用  
 C5h.4 TKA04 扫描启用: 0: 禁用 1: 启用  
 C5h.3 TKA03 扫描启用: 0: 禁用 1: 启用  
 C5h.2 TKA02 扫描启用: 0: 禁用 1: 启用  
 C5h.1 TKA01 扫描启用: 0: 禁用 1: 启用  
 C5h.0 TKA00 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
<b>ATKCHA1</b>	<b>ATKCHA1</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C6h.7 TKA15 扫描启用: 0: 禁用 1: 启用  
 C6h.6 TKA14 扫描启用: 0: 禁用 1: 启用  
 C6h.5 TKA13 扫描启用: 0: 禁用 1: 启用  
 C6h.4 TKA12 扫描启用: 0: 禁用 1: 启用  
 C6h.3 TKA11 扫描启用: 0: 禁用 1: 启用  
 C6h.2 TKA10 扫描启用: 0: 禁用 1: 启用  
 C6h.1 TKA09 扫描启用: 0: 禁用 1: 启用  
 C6h.0 TKA08 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
<b>ATKCHA2</b>	<b>ATKCHA2</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

- C7h.7 TKACAP (TKA23) 内部参考电容通道扫描使能: 0: 禁用 1: 启用  
 C7h.6~5 Reservd  
 C7h.4 TKA20 扫描启用: 0: 禁用 1: 启用  
 C7h.3 TKA19 扫描启用: 0: 禁用 1: 启用  
 C7h.2 TKA18 扫描启用: 0: 禁用 1: 启用  
 C7h.1 TKA17 扫描启用: 0: 禁用 1: 启用  
 C7h.0 TKA16 扫描启用: 0: 禁用 1: 启用

SFR B7h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ATKCHB0</b>	<b>ATKCHB0</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

B7h.7 TKB07 扫描启用: 0: 禁用 1: 启用  
 B7h.6 TKB06 扫描启用: 0: 禁用 1: 启用  
 B7h.5 TKB05 扫描启用: 0: 禁用 1: 启用  
 B7h.4 TKB04 扫描启用: 0: 禁用 1: 启用  
 B7h.3 TKB03 扫描启用: 0: 禁用 1: 启用  
 B7h.2 TKB02 扫描启用: 0: 禁用 1: 启用  
 B7h.1 TKB01 扫描启用: 0: 禁用 1: 启用  
 B7h.0 TKB00 扫描启用: 0: 禁用 1: 启用

SFR B6h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ATKCHB1</b>	<b>ATKCHB1</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

B6h.7 TKB15 扫描启用: 0: 禁用 1: 启用  
 B6h.6 TKB14 扫描启用: 0: 禁用 1: 启用  
 B6h.5 TKB13 扫描启用: 0: 禁用 1: 启用  
 B6h.4 TKB12 扫描启用: 0: 禁用 1: 启用  
 B6h.3 TKB11 扫描启用: 0: 禁用 1: 启用  
 B6h.2 TKB10 扫描启用: 0: 禁用 1: 启用  
 B6h.1 TKB09 扫描启用: 0: 禁用 1: 启用  
 B6h.0 TKB08 扫描启用: 0: 禁用 1: 启用

SFR AFh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>ATKCHB2</b>	<b>ATKCHB2</b>							
R/W	R/W							
Reset	0	0	0	0	0	0	0	0

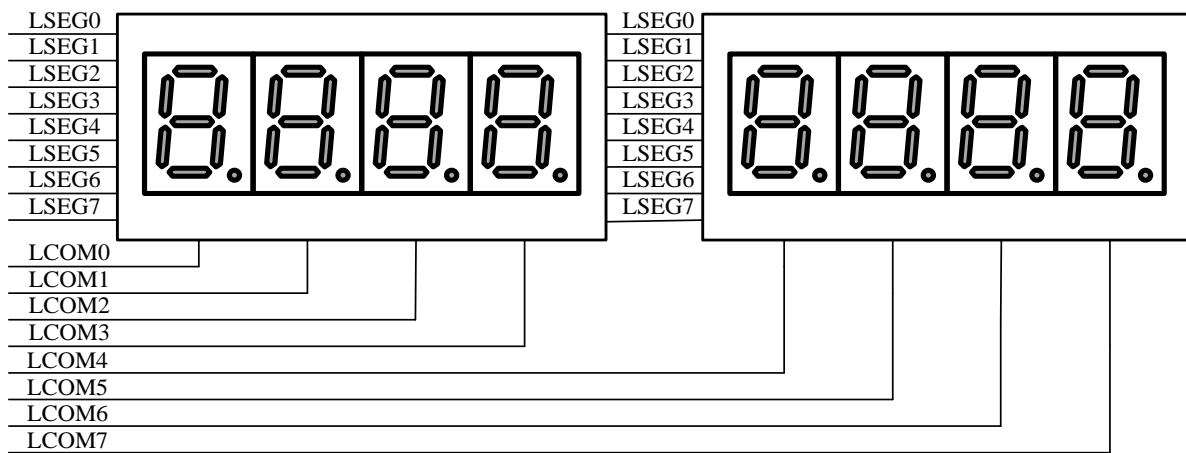
Afh.7 TKBCAP (TKB23) 内部参考电容通道扫描使能: 0: 禁用 1: 启用  
 Afh.6~5 Reservd  
 Afh.4 TKB20 扫描启用: 0: 禁用 1: 启用  
 Afh.3 TKB19 扫描启用: 0: 禁用 1: 启用  
 Afh.2 TKB18 扫描启用: 0: 禁用 1: 启用  
 Afh.1 TKB17 扫描启用: 0: 禁用 1: 启用  
 Afh.0 TKB16 扫描启用: 0: 禁用 1: 启用

### 13. LCD / LED 控制器/驱动器

该模块可以配置三种驱动模式：LED 矩阵模式，LED 点矩阵模式，LCD 驱动模式。通过寄存器配置，它仅同时支持一种操作模式。

#### 13.1 LED 矩阵模式 (MX, Matrix)

该芯片在矩阵模式下支持 LED 控制器和驱动器。如果 LEDMODE = 00b, LXDON = SELLED = 1。LED 矩阵模式将启用。它提供 8 个 Segment 引脚和 8 个 Common 引脚来驱动 64 像素的 LED 模块。COM 引脚具有较高的灌电流。可以通过 LXDBRIT 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。



XRAM 中的显示配置对应于相应地址的点亮状态（1 表示点亮，0 表示不点亮）。

XRAM Addr.	COM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FB00h	LCOM0	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB01h	LCOM1	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB02h	LCOM2	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB03h	LCOM3	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB04h	LCOM4	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB05h	LCOM5	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB06h	LCOM6	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0
FB07h	LCOM7	LSEG7	LSEG6	LSEG5	LSEG4	LSEG3	LSEG2	LSEG1	LSEG0

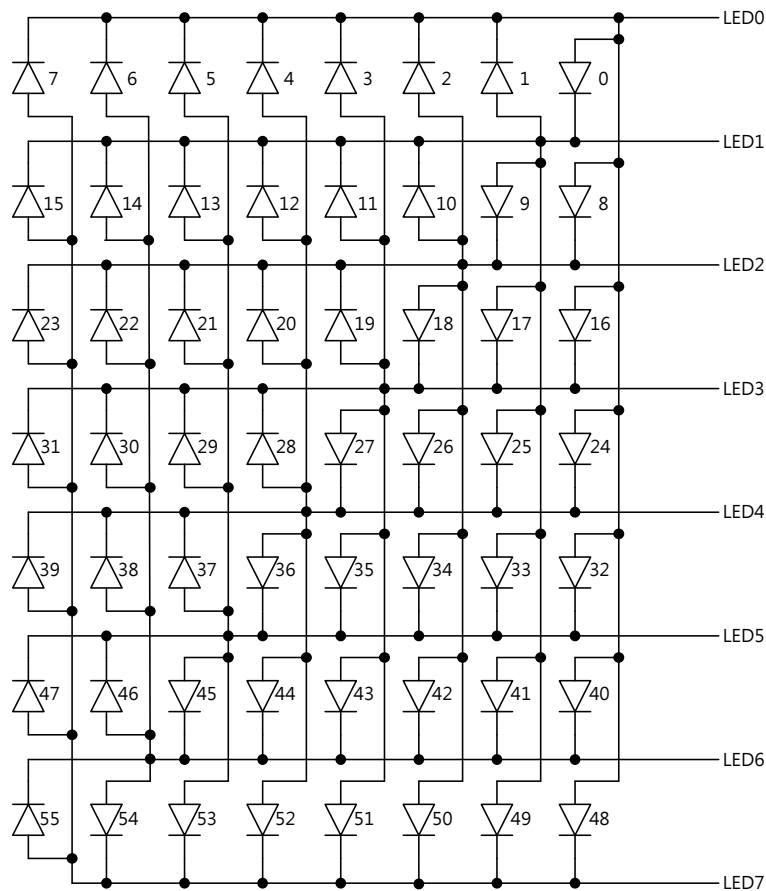
LED 矩阵驱动方式对应的显示配置表

### 13.2 LED 点矩阵模式 (DMX, Dot Matrix)

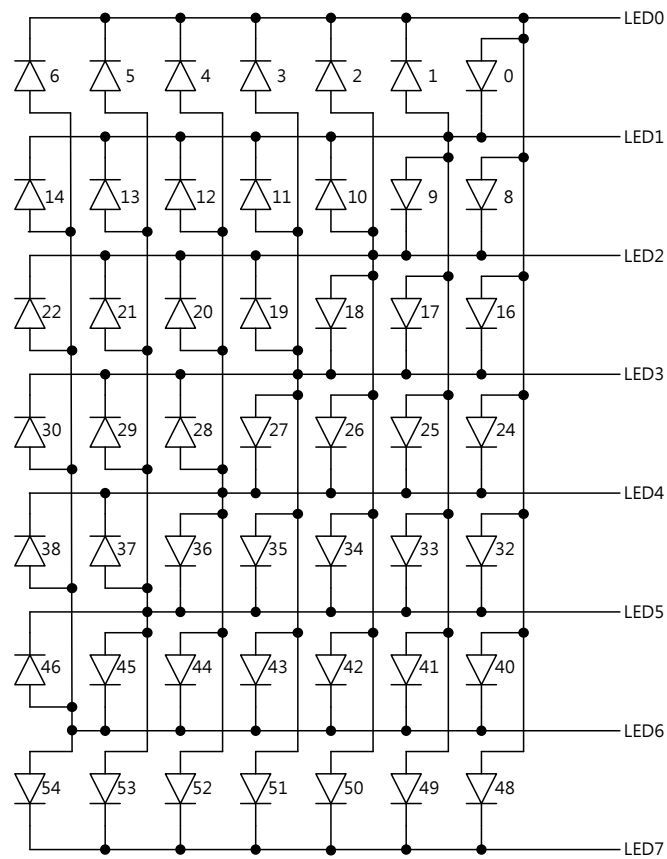
如果 LEDMODE = 10b, LXDON = SELLED = 1, LED 点矩阵模式将启用。LED 点矩阵是通用的 7 \* 8 点矩阵。对应 LED0~LED7 端口, 最多可配置 7x8 = 56 个 LED 点进行驱动, LED 的对应位置在下图的 7 \* 8 点矩阵中标记地址, XRAM 中的显示配置对应于 LED 相应地址的状态 (1 表示点亮, 0 表示不点亮)。支持多达 56 灯的 LED 驱动器。使用 LXDDUTY 选择点矩阵 4 \* 4、5 \* 5、6 \* 6、6 \* 7、7 \* 7 和 7 \* 8, 相应的 LED 地址保持不变。可以通过 LXDBRIT 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。

XRAM Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FB00h	7	6	5	4	3	2	1	0
FB01h	15	14	13	12	11	10	9	8
FB02h	23	22	21	20	19	18	17	16
FB03h	31	30	29	28	27	26	25	24
FB04h	39	38	37	36	35	34	33	32
FB05h	47	46	45	44	43	42	41	40
FB06h	55	54	53	52	51	50	49	48

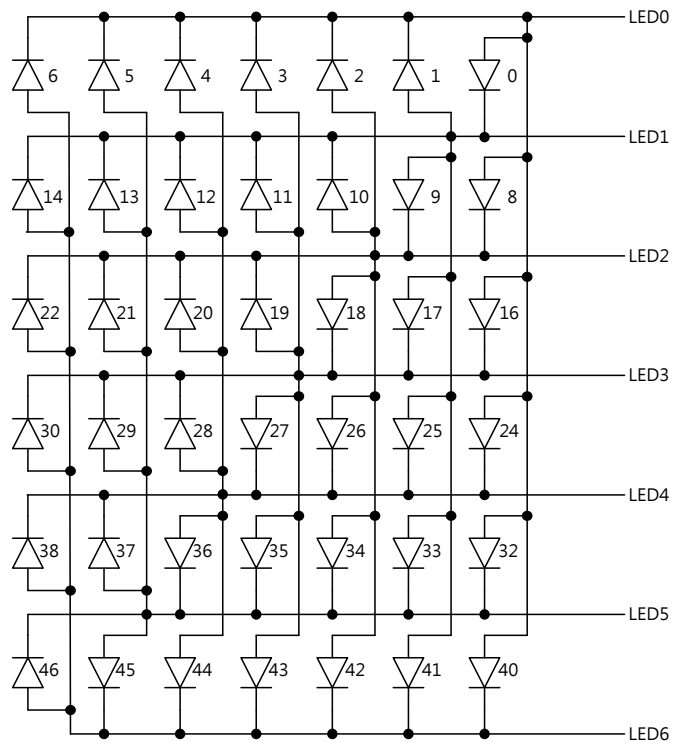
LED 点阵驱动方式对应的显示配置表



LED 7\*8 点矩阵

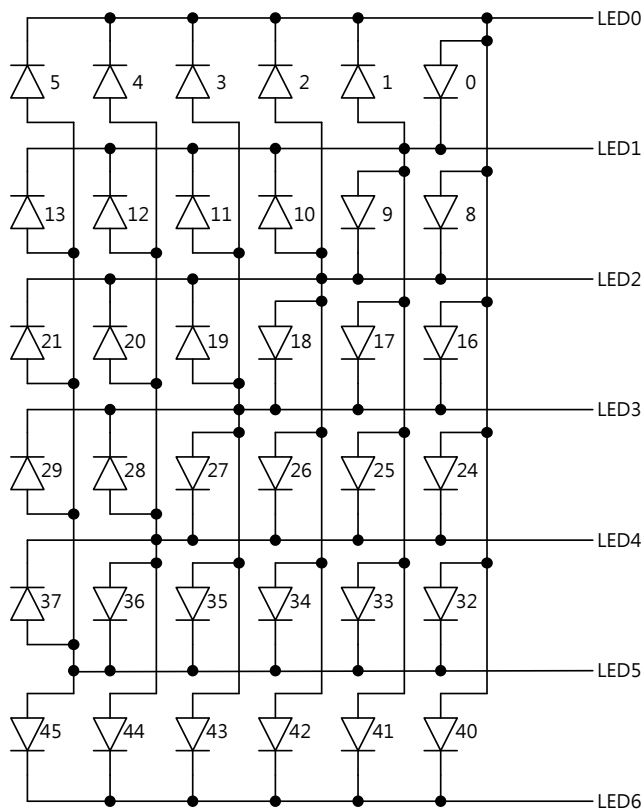


LED 7\*7 点矩阵

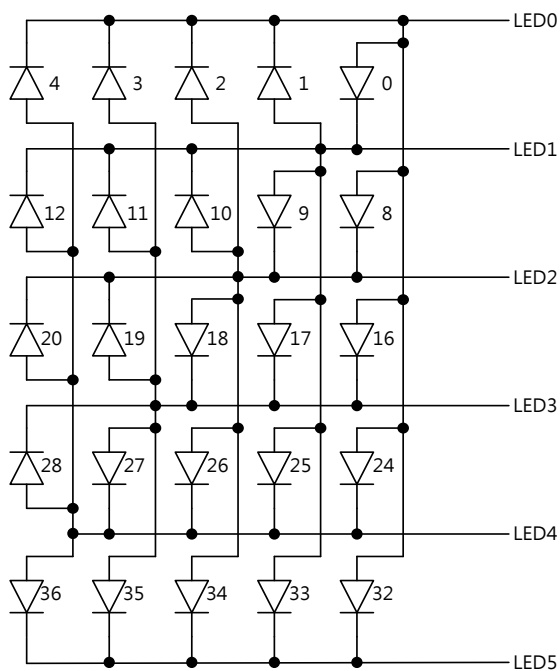


LED 6\*7 点矩阵

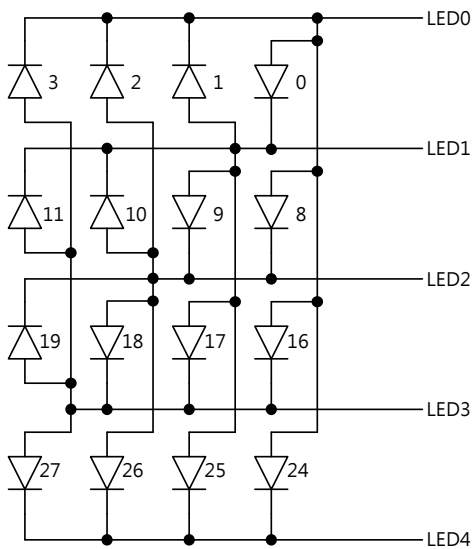




LED 6\*6 点矩阵



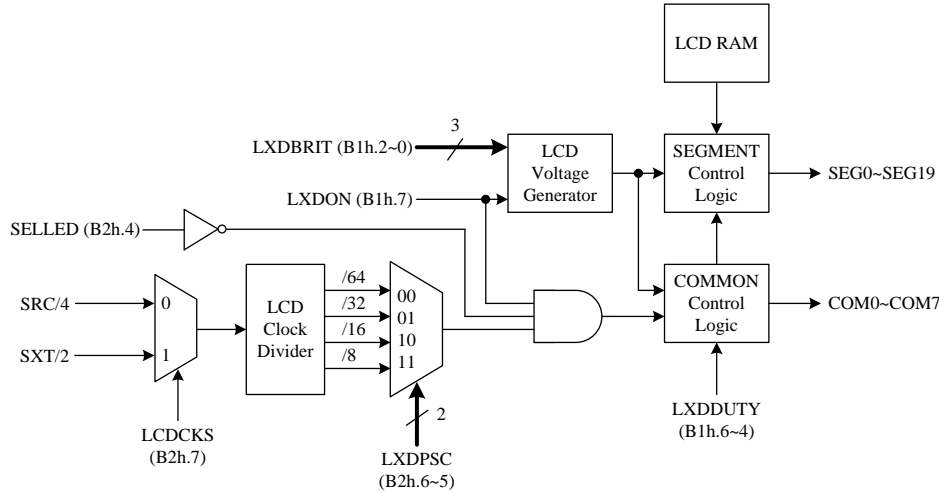
LED 5\*5 点矩阵



LED 4\*4 点矩阵

### 13.3 LCD 模式

该芯片支持 LCD 控制器和驱动器。LCD 驱动器能够选择 80 点(4 COM / 20 SEG) 或 128 点(8 COM / 16 SEG) 来驱动 LCD 面板。它能够驱动 1/3 偏压。LCD 时钟源由 SRC / 4 或 SXT / 2 产生，取决于 SFR LCDCKS。时钟速率可以通过 LXDPSC 位除以 8、16、32 和 64。如果 SRC / 4 是 LCD 时钟源，则 VCC 电压电平会影响 SRC 频率和 LCD 帧速率。LCDRAM 位于 8051 的外部数据存储空间中，地址从 FB00h 到 FB1Fh。

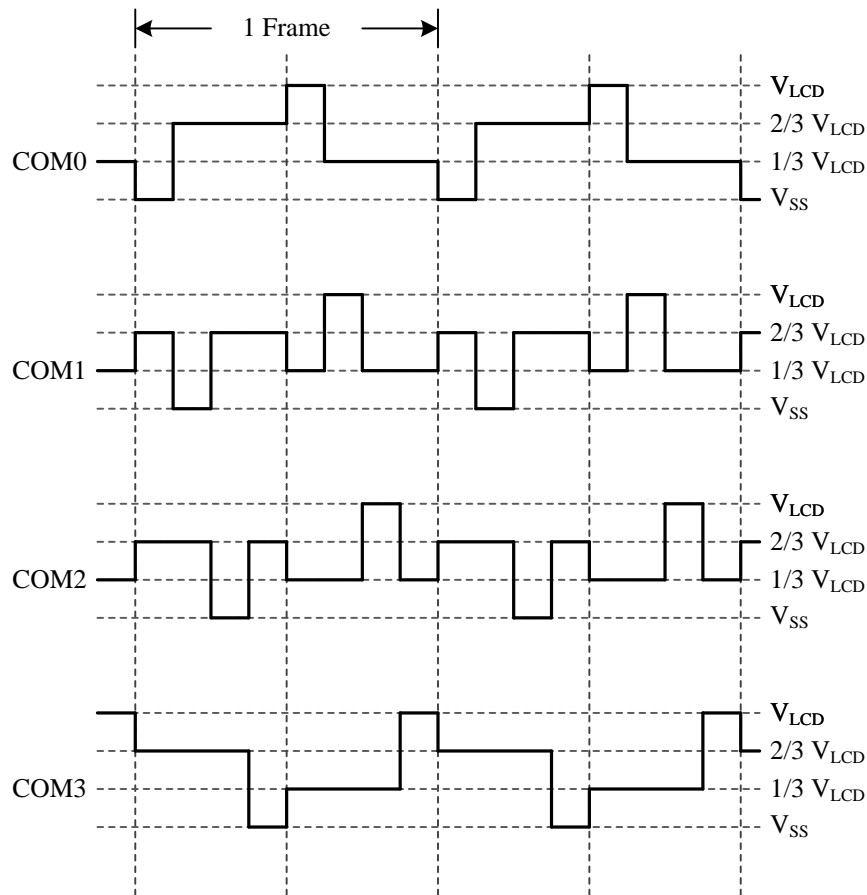


LCD COM0~7 电路

Addr.	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	COM
FB00h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM0
FB01h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM1
FB02h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM2
FB03h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM3
FB04h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM4
FB05h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM5
FB06h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM6
FB07h	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0	COM7
FB08h	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM0
FB09h	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM1
FB0Ah	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM2
FB0Bh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM3
FB0Ch	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM4
FB0Dh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM5
FB0Eh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM6
FB0Fh	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	COM7
FB10h					SEG19	SEG18	SEG17	SEG16	COM0
FB11h					SEG19	SEG18	SEG17	SEG16	COM1
FB12h					SEG19	SEG18	SEG17	SEG16	COM2
FB13h					SEG19	SEG18	SEG17	SEG16	COM3
FB14h					SEG19	SEG18	SEG17		COM4
FB15h					SEG19	SEG18			COM5
FB16h					SEG19				COM6
FB17h									COM7

LCD 对应的显示配置表

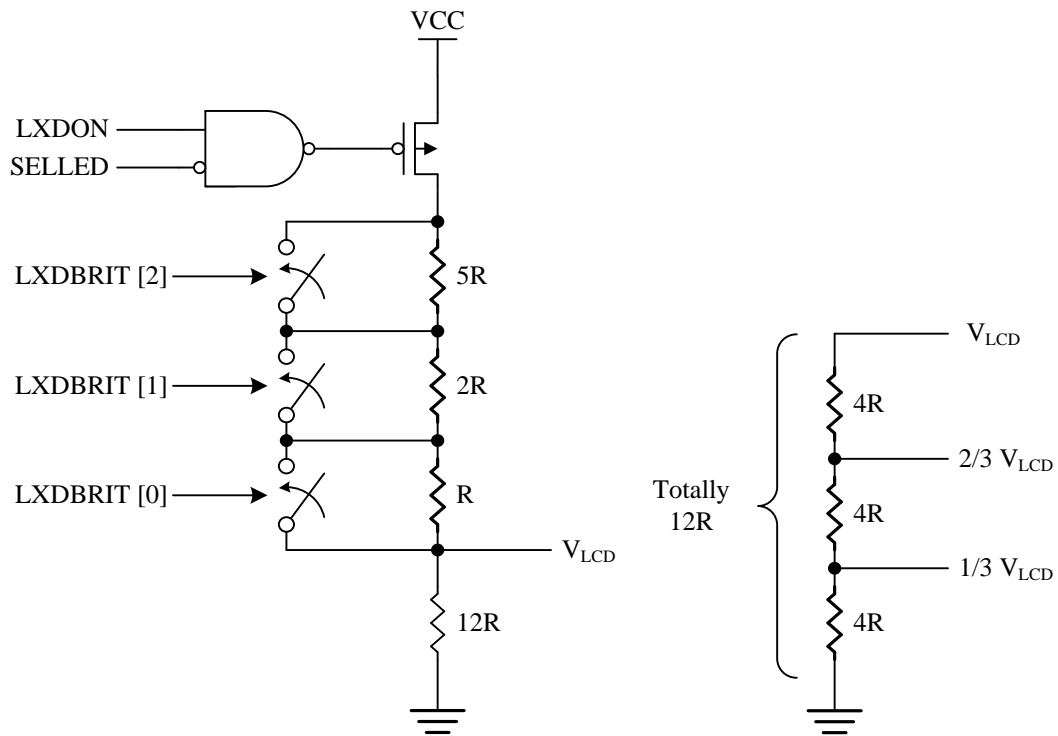
在 COM 引脚上输出的任何重复波形的频率都可以用来表示 LCD 帧速率。下图显示了一个 LCD 框架。



每个频表如下所示。

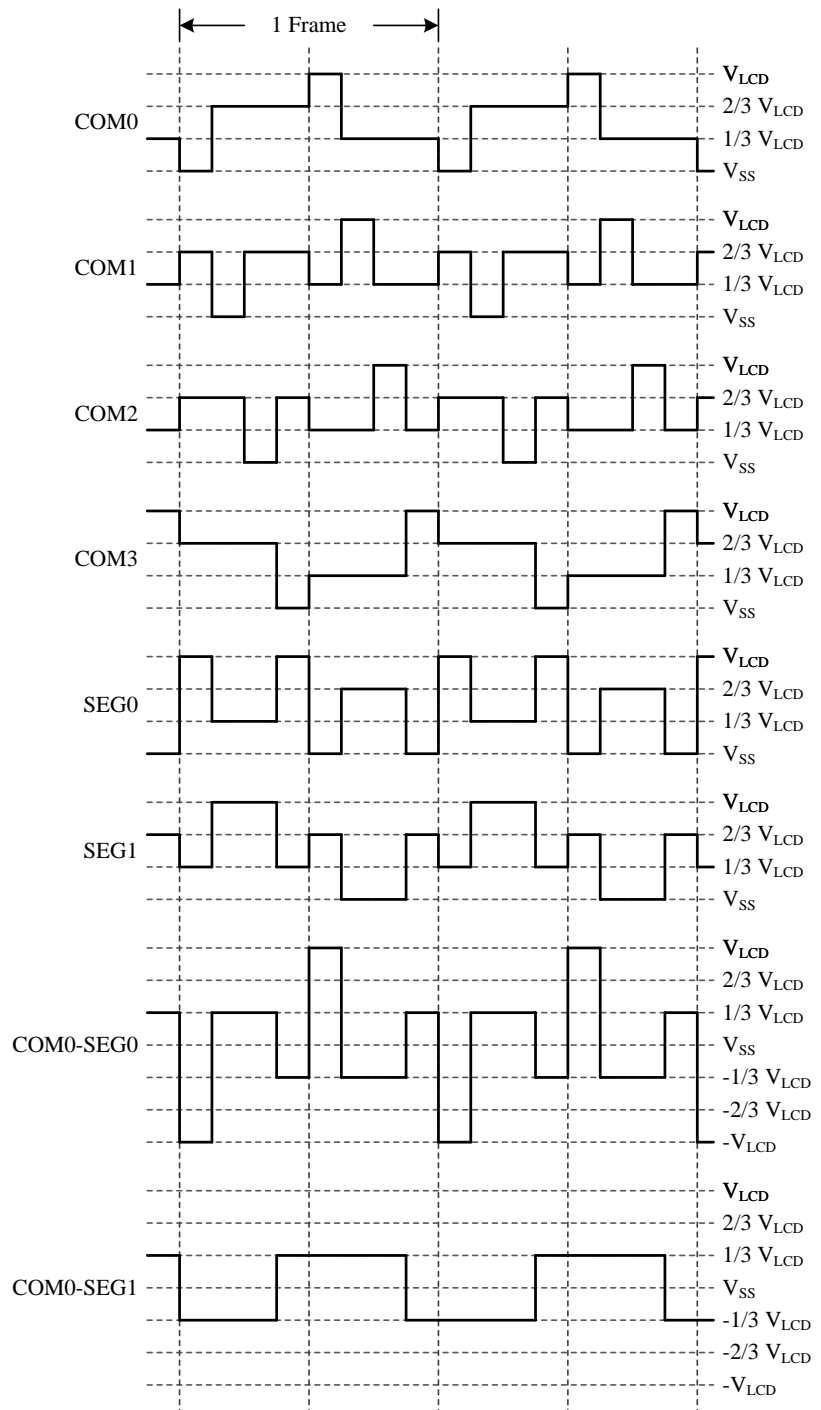
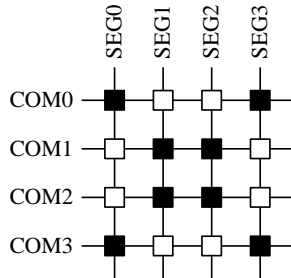
LCDCKS (B2h.7)	LXDPSC (B2h.6~5)	LCD 帧率(Hz)			
		LXDDUTY (B1h.6~4)			
		1/4 占空比	1/5 占空比	1/6 占空比	1/8 占空比
SXT/2 16384 Hz	/8	256	205	171	128
	/16	128	102	85	64
	/32	64	51	43	32
	/64	32	26	21	16
SRC/4 20000 Hz	/8	313	250	208	156
	/16	156	125	104	78
	/32	78	62	52	39
	/64	39	31	26	20

下图 LCD 电压发生器显示了由电阻组成的内部电压发生器。LXDON 和 SELLED 控制从 VCC 到地的电流。如果 LXDON = 0 或 SELLED = 1, 则 PMOS 将关闭路径, 以便所有 LCD 电压均为 0V。如果 LXDON = 1 和 SELLED = 0, 则电阻分压器将产生多个电压, 以提供 LCD 控制模块 用于生成所需的波形。LXDON 控制位将打开/短路开关以确定  $V_{LCD}$ 。下表显示了与 LXDON 对应的  $V_{LCD}$ 。由于 LXDON = 1 和 SELLED = 0 时 DC 路径始终接通, 因此分压器电路将消耗电流。



LXDON	$V_{LCD}$
000	$(12/20) \times V_{CC}$
001	$(12/19) \times V_{CC}$
010	$(12/18) \times V_{CC}$
011	$(12/17) \times V_{CC}$
100	$(12/15) \times V_{CC}$
101	$(12/14) \times V_{CC}$
110	$(12/13) \times V_{CC}$
111	$V_{CC}$

1/4 占空比, 1/3 偏置输出波形



SFR B1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LXDCON</b>	LXDON	LXDDUTY			LEDBRITM	LXDBRIT		
R/W	R/W	R/W			R/W	R/W		
Reset	0	0	0	0	0	1	1	1

B1h.7 **LXDON**: LCD / LED使能

0: LCD/LED 禁用

1: LCD / LED 使能

B1h.6~4 **LXDDUTY**: LCD / LED 占空比选择

LCD选择（如果SELLED = 0）：

000: 1/4 占空比, COM 0~3

001: 1/4 占空比, COM 0~3

010: 1/5 占空比, COM 0~4

011: 1/6 占空比, COM 0~5

100: 1/6 占空比, COM 0~5

101: 1/8 占空比, COM 0~7

110: 1/8 占空比, COM 0~7

111: 1/8 占空比, COM 0~7

LED选择：矩阵模式（如果SELLED = 1, LEDMODE = 00b）

000: 1/2 占空比, LCOM 0~1

001: 1/3 占空比, LCOM 0~2

010: 1/4 占空比, LCOM 0~3

011: 1/5 占空比, LCOM 0~4

100: 1/6 占空比, LCOM 0~5

101: 1/7 占空比, LCOM 0~6

110: 1/8 占空比, LCOM 0~7

111: 1/8 占空比, LCOM 0~7

LED选择：点矩阵模式（如果SELLED = 1, LEDMODE = 10b）

000: 4x4, LED 0~4

001: 5x5, LED 0~5

010: 6x6, LED 0~6

011: 6x7, LED 0~6

100: 7x7, LED 0~7

101: 7x8, LED 0~7

110: 保留

111: 保留

B1h.3 **LEDBRITM**: LED 亮度模式

0: 亮度均匀模式

1: 亮度增强模式

B1h.2~0 **LXDBRIT**: LCD / LED 亮度控制

000: 0 级（最暗）

...

111: 7 级（最亮）

SFR B2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>LXDCON2</b>	LCDCKS	LXDPSK		SELLED	LEDHOLD	–	LEDMODE	
R/W	R/W	R/W		R/W	R/W	–	R/W	
Reset	0	0	0	0	0	–	0	0

B2h.7 **LCDCKS**: LCD 时钟源选择 (注意, LED 时钟源固定为 FRC)

0: SRC/4

1: SXT/2

B2h.6~5 **LXDPSK**: LCD / LED 时钟预分频器选择

00: LCD / LED 时钟为 FRC 除以 64

01: LCD / LED 时钟为 FRC 除以 32

10: LCD / LED 时钟为 FRC 除以 16

11: LCD / LED 时钟为 FRC 除以 8

B2h.4 **SELLED**: LCD / LED 功能选择

0: LCD

1: LED

B2h.3 **LEDHOLD**: 保持为 0, 不能设置为 1

B2h.1~0 **LEDMODE**: LED 模式选择

00: 矩阵模式

01: 保留的

10: 点矩阵模式

11: 保留的

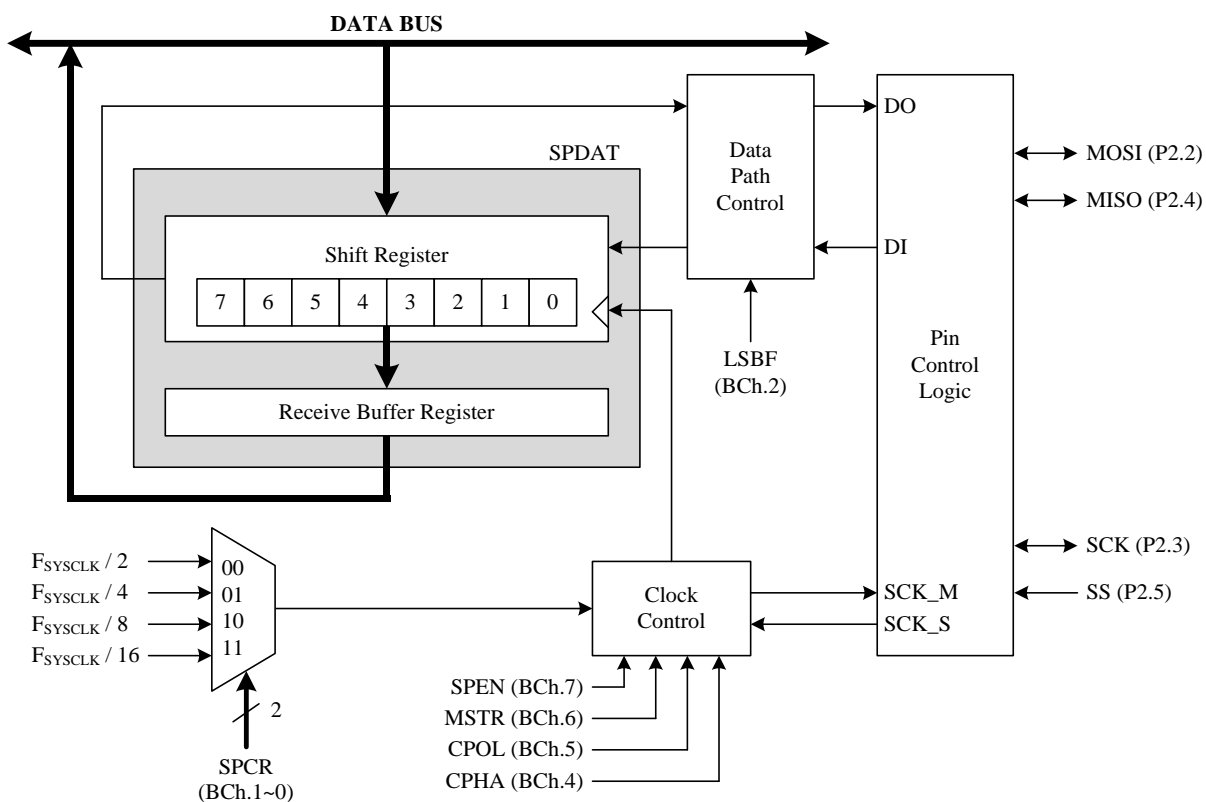


## 14. 串行外围设备接口 (SPI)

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,或闪速存储器等。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	PINMOD <sub>x</sub>	P4.n SFR 数据	引脚状态
Master Mode MISO	<b>xx01</b>	1	SPI 数据输入
Master Mode SCK, MOSI	<b>xx10</b>	X	SPI 时钟/数据输出 (CMOS 推挽)
Slave Mode MISO	<b>xx10</b>	X	SPI 数据输出 (CMOS 推挽)
Slave Mode SCK, MOSI	<b>xx01</b>	1	SPI 时钟/数据输入
SS	<b>xx01</b>	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

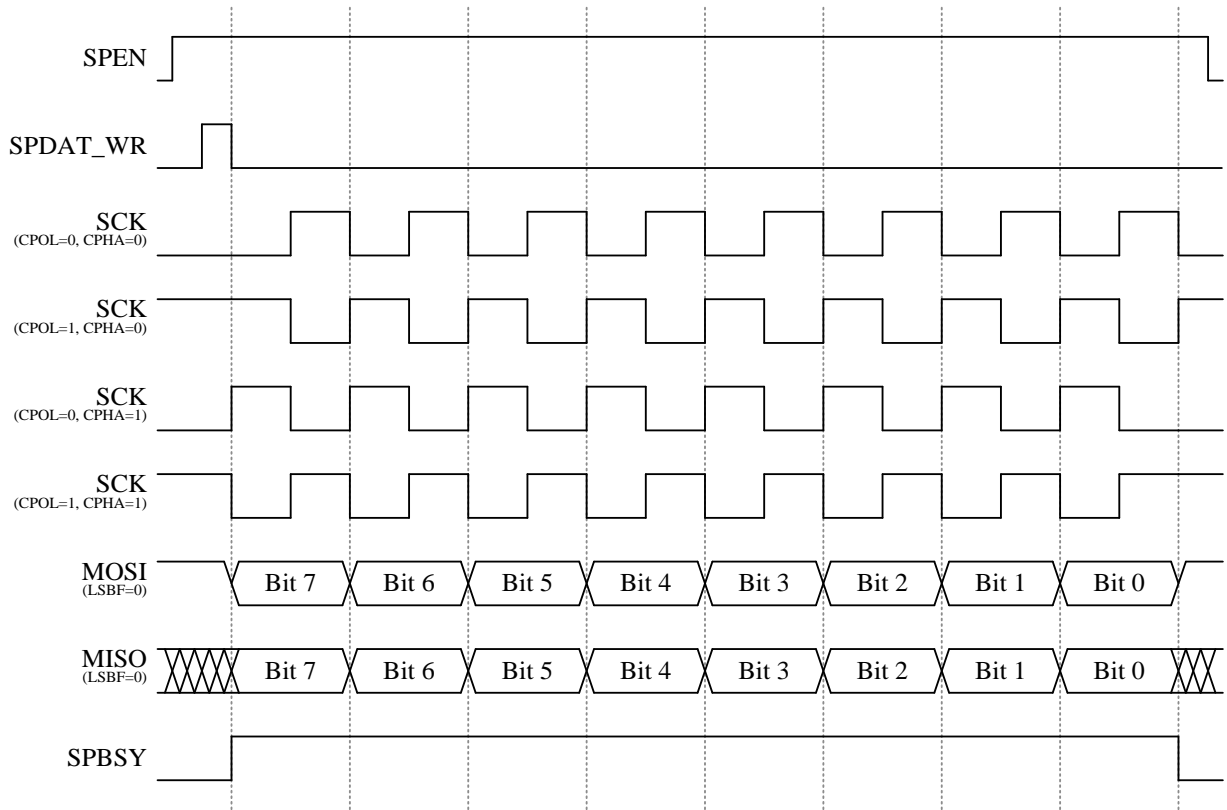
SPI 通过设置 SPCON 中的 MSTR 位以工作在主控模式。要开始传送,先将数据写到 SPDAT。如果 SPBSY 位被清除,数据将被转移到移位寄存器,并开始移出到 MOSI 线上。在同时,数据从 MISO 线移入从设备。当传输结束时在 SPSTA 中的 SPIF 位变为被设置,则接收的数据被写入到接收缓冲器,并将 SPSTA 中的 RCVBF 位设置。为了防止溢出条件,软件必须在下一个字节进入移位寄存器之前先读取 SPDAT。当数据写入 SPDAT 时 SPBSY 位将会被设置以开始传送,并在主控模式的第八个 SCK 周期结束时被清零。

### 从模式

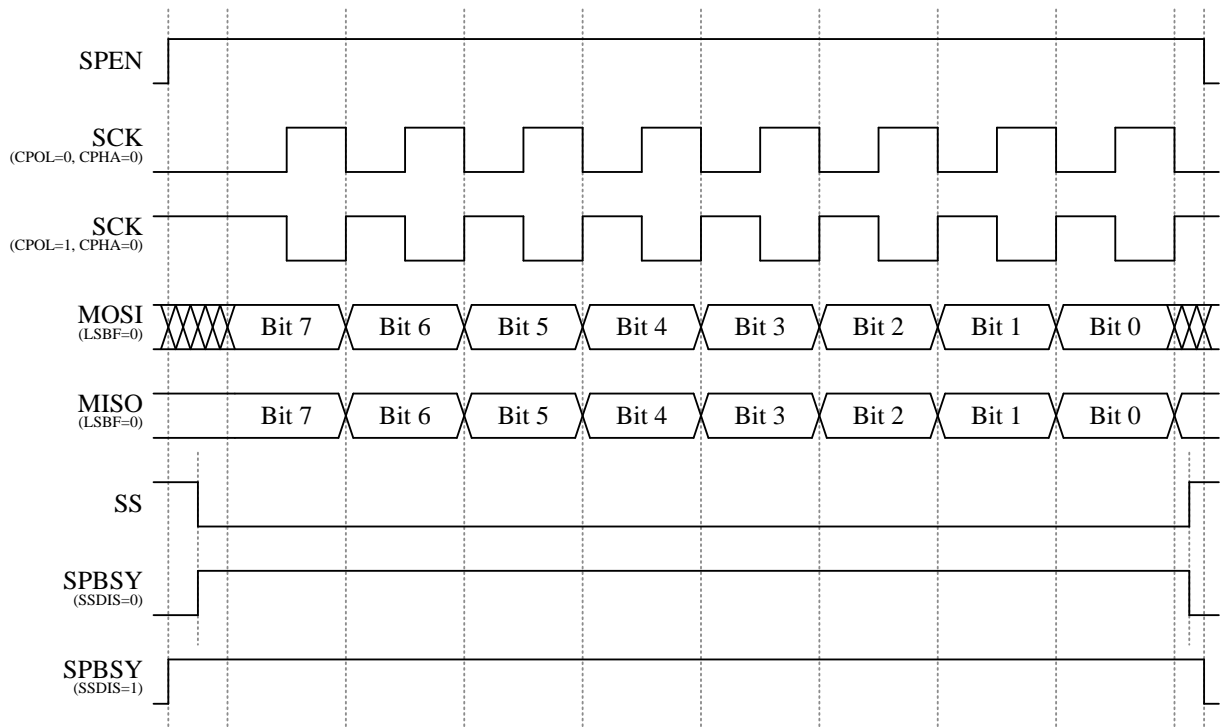
SPI 通过清除 SPCON 中的 MSTR 位成为从模式。如果 SSDIS 清零,则传输开始当 SS 变低并保持低直到数据传送结束。如果 SSDIS 设置,在 SPCON 中的 SPEN 位被设置时,传输将会开始,并不在意 SS。主设备的数据将通过 MOSI 线转移到移位寄存器,并从移位寄存器移出到 MISO 线。当一个字节进入移位寄存器时,如果 RCVBF 被清除,数据将被传输到接收缓冲器。如果 RCVBF 设置,较新的接收数据将不会传送到接收缓冲器而且 RCVOVF 位会被设置。当一个字节进入移位寄存器后,SPIF 和 RCVBF 位会被设置。为了防止溢出条件,软件必须在下一个字节进入移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  $F_{\text{SYSCLK}}/4$ 。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

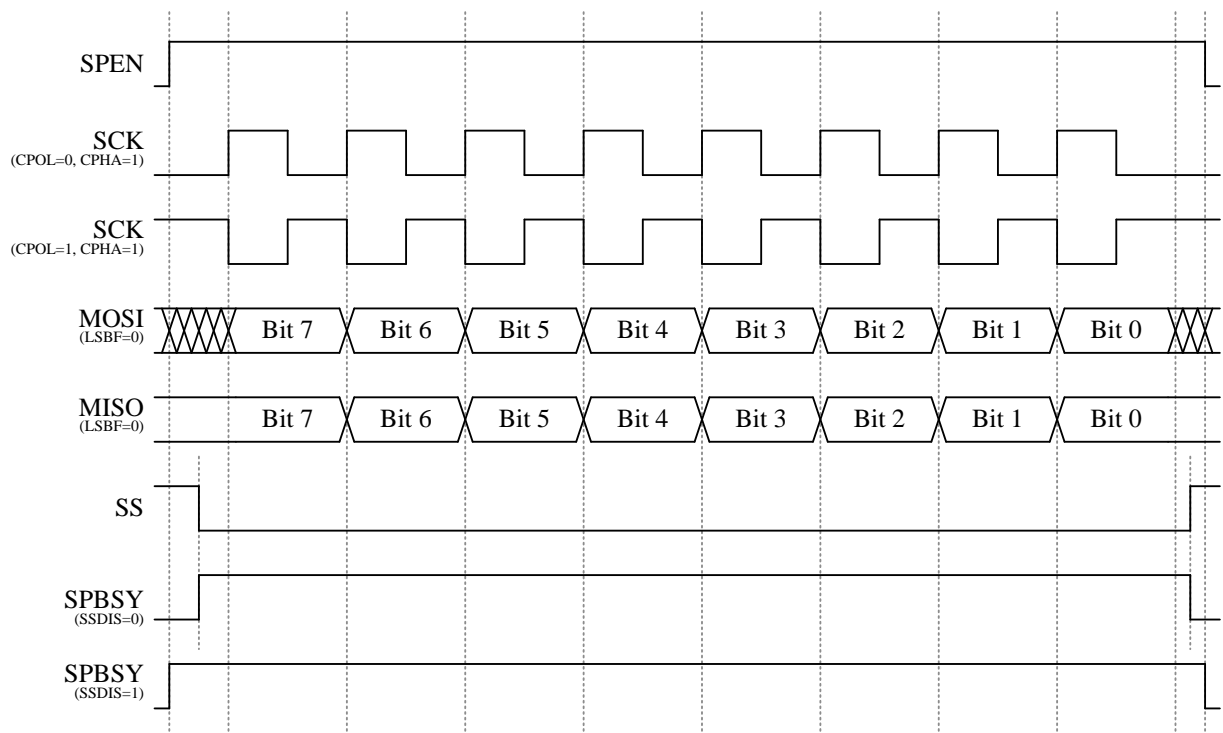
SPI 通过设置 SPCON 寄存器的 CPOL 和 CPHA 位,产生四个时钟类型。CPOL 位定义在 SPI 空闲状态的 SCK 的水平。在 CPOL 位被清零时空闲状态的 SCK 电平为低的,CPOL 位设置时是高电平。CPHA 位定义用于采样和数据移位的边沿。在 CPHA 位被清除时 SPI 在 SCK 周期的第一个边沿采样数据和在 SCK 周期的第二个边沿移动数据。CPHA 位被设置时在 SCK 期间的第二边沿的 SPI 采样数据和 SCK 周期的第一个边沿移位数据。下图显示了在主从模式的详细时间。SPEN 位被设置前主设备和从设备必须被配置为使用相同的时钟类型。SPCR 控制主模式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



从模式时序(CPHA = 1)

在主和从模式中,SPIF 位是在数据传输结束时由 H/W 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。S/W 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 H/W 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,MODF 位会被设置,如果 SPI 中断使能,中断就会产生。当 MODF 位被设置,SPCON 中的 SPEN 和 MSTR 将被 H/W 清零。写 0 到 MODF 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
<b>SPCON</b>	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<sub>SYSCLK</sub>/2 01:F<sub>SYSCLK</sub>/4 10:F<sub>SYSCLK</sub>/8 11:F<sub>SYSCLK</sub>/16

SFR BDh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SPSTA</b>	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	-	-

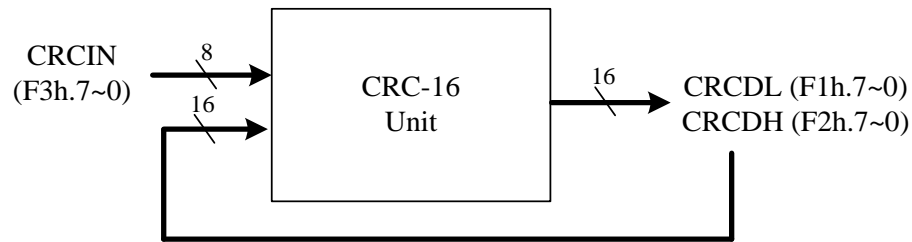
- BDh.7 **SPIF**:SPI 中断标志  
这是由 H/W 在数据传输结束时设定。中断产生时由 H/W 清零。写 0 到该位将清除该标志。
- BDh.6 **WCOL**:写入冲突中断标志  
由 H/W 设置,如果 SPBSY = 1 时写入数据到 SPDAT。写 0 到该位或 SPBSY = 0 时重写 SPDAT 数据将清除该标志。
- BDh.5 **MODF**:模式故障中断标志  
在主模式时,当 SSDIS = 0,SS 引脚被拉低时通过 H/W 设置。写 0 到该位将清除该标志。当该位被设置,SPCON 中的 SPEN 和 MSTR 将被 H/W 清零。
- BDh.4 **RCVOVF**:接收缓冲区溢出标志  
在数据传输结束和 RCVBF = 1 时将通过 H/W 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 H/W 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 H/W 设置。

SFR BEh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SPDAT</b>	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。

## 15. 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
<b>CRCDL</b>	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
<b>CRCDH</b>	CRCDH							
R/W	R/W							
Reset	1	1	1	1	1	1	1	1

F2h.7~0 **CRCDH**: 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
<b>CRCIN</b>	CRCIN							
W	W							
Reset	-	-	-	-	-	-	-	-

F3h.7~0 **CRCIN**: CRC 输入数据寄存器

## 16. 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			

<b>SFR CEh</b>	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXA2</b>	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

<b>SFR CFh</b>	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXA3</b>	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

<b>SFR E6h</b>	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXA</b>	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

<b>SFR E7h</b>	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>EXB</b>	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

<b>SFR F7h</b>	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>AUX2</b>	WDTE		PWRS AV	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 MULDVII16 = 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

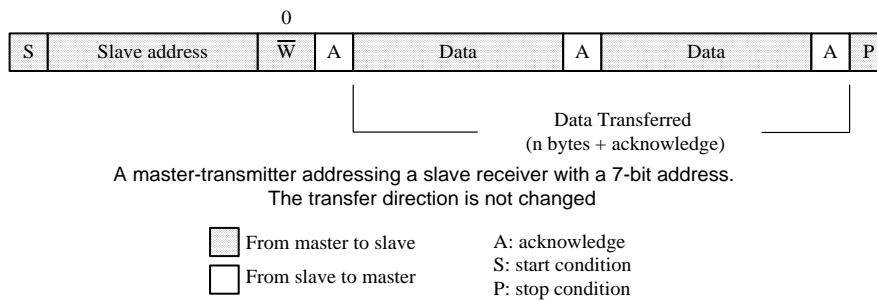


## 17. 主 I<sup>2</sup>C 接口

### 主 I<sup>2</sup>C 接口传输模式:

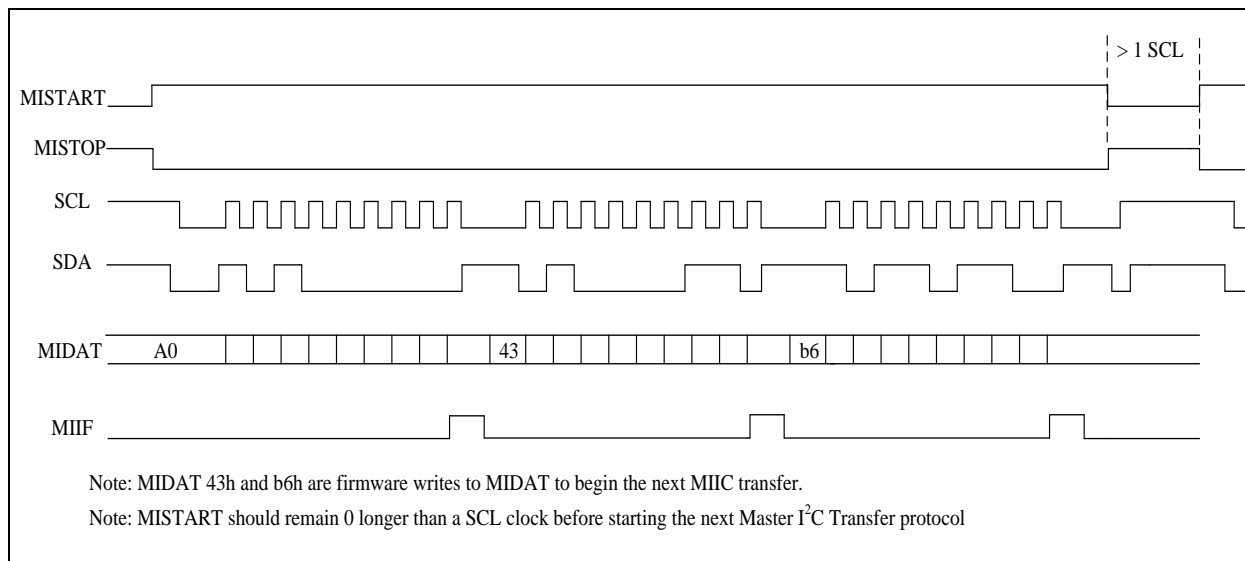
一开始先时将从机地址和方向位写入 MIDAT 并设置 MISTART。在 MISTART 设置之后, 将发送 7 位从地址和一位方向位到从机。等待直到 MIIF 转换为 1 时, 代表地址和方向位传输完成, 用户应清除 MIIF 并写入数据到 MIDAT 以开始第一次数据传输。当 MIIF 转换为 1 时, 代表数据传输到从机完成。用户可以再次写入数据到 MIDAT 以将开始下一次数据传输到从机。设置 MISTOP 以完成传送模式。

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 I<sup>2</sup>C 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 I<sup>2</sup>C 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 I<sup>2</sup>C 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 I<sup>2</sup>C 传输



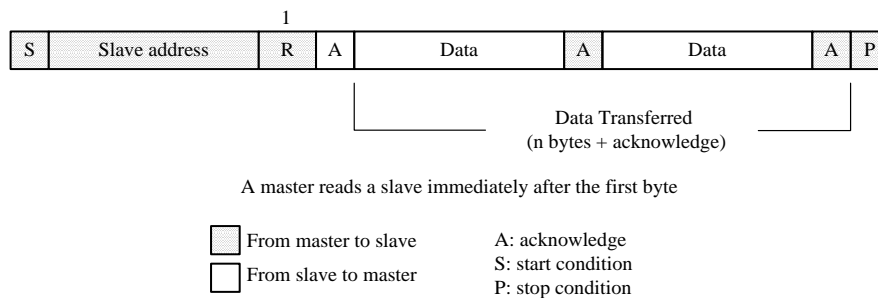
主发送时序

**注:** 在开始下一个主 I<sup>2</sup>C 协议之前, MISTART 应该保持 0 比 SCL 周期更长。

**I<sup>2</sup>C主接口接收模式:**

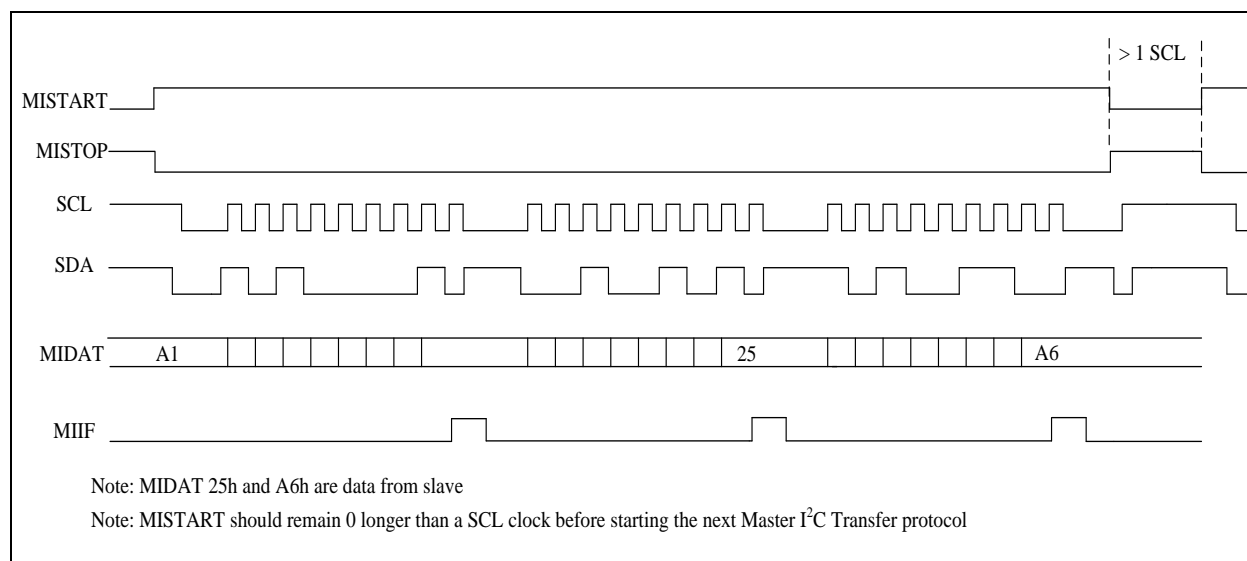
一开始先将从机地址和方向位写入MIDAT并设置MISTART。在MISTART设置之后，将发送7位从地址和一位方向位到从机。当MIIF转换为1时，代表地址和方向位传输完成。用户应清除MIIF并读取MIDAT以开始第一次接收数据（此时尚未完成接收数据,应丢弃读入的MIDAT）。当MIIF转换为1时，代表对从机接收的数据已完成。用户可以读取MIDAT以得到接收数据，硬件会同时开始下一次接收。设置MISTOP以完成接收模式。

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I<sup>2</sup>C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 I<sup>2</sup>C 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 I<sup>2</sup>C 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 I<sup>2</sup>C 传输



主接收时序

I <sup>2</sup> C 功能引脚	PINMOD <sub>x</sub>	Px.n SFR data	引脚状态
I <sup>2</sup> C Master SCL	<b>0000</b>	X	时钟输出（开漏输出）
	<b>0010</b>	X	时钟输出（CMOS 推挽）
I <sup>2</sup> C Master/Slaver SDA	<b>0000</b>	1	数据（上拉）

**主 I<sup>2</sup>C 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
<b>INTE1</b>	PWMIE	I2CE	ES2	SPIE	ADTKIE	LVDIE	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: I<sup>2</sup>C 中断使能**

 0: 禁用 I<sup>2</sup>C 中断      1: 使能 I<sup>2</sup>C 中断

SFR E1h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>MICON</b>	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 I<sup>2</sup>C 使能**

0: 禁用      1: 使能

**E1h.6 MIACKO: 当主 I<sup>2</sup>C 接收数据时, 向 I<sup>2</sup>C 总线发送 ACK**

0: ACK 到从机      1: NACK 到从机

**E1h.5 MIIF: 主 I<sup>2</sup>C 中断标志**

 当主 I<sup>2</sup>C 发送或接收一个字节完成时, 由 H/W 设置。向该位写入"0"将清除该标志

**E1h.4 MIACKI: 当主 I<sup>2</sup>C 传输时, ACK 来自 I<sup>2</sup>C 总线 (只读)**

0: 收到 ACK      1: 收到 NACK

**E1h.3 MISTART: 主 I<sup>2</sup>C 起动脉**

 1: 启动 I<sup>2</sup>C 总线传输

**E1h.2 MISTOP: 主 I<sup>2</sup>C 停止位**

 1: 发送停止信号以停止 I<sup>2</sup>C 总线

**E1h.1~0 MICR: 主 I<sup>2</sup>C 时钟频率选择**

 00: F<sub>sys</sub>/4      (ex. If F<sub>sys</sub>=16MHz, I<sup>2</sup>C clock is 4M Hz)

 01: F<sub>sys</sub>/16      (ex. If F<sub>sys</sub>=16MHz, I<sup>2</sup>C clock is 1M Hz)

 10: F<sub>sys</sub>/64      (ex. If F<sub>sys</sub>=16MHz, I<sup>2</sup>C clock is 250K Hz)

 11: F<sub>sys</sub>/256      (ex. If F<sub>sys</sub>=16MHz, I<sup>2</sup>C clock is 62.5K Hz)

SFR E2h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>MIDAT</b>	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 I<sup>2</sup>C 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 I<sup>2</sup>C 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 I<sup>2</sup>C 总线的接收

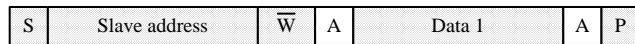
SFR EAh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SICON</b>	MIIE	TXDIE	RCD2IE	RCD1IE	-	TXDF	RCD2F	RCD1F
R/W	R/W	R/W	R/W	R/W	-	R/W	R/W	R/W
Reset	0	0	0	0	-	1	0	0

**EAh.7 MIIE: I<sup>2</sup>C 主中断使能**

0: 禁用      1: 使能

### 18. 从机 I<sup>2</sup>C 接口

该芯片提供如下的从机 I<sup>2</sup>C 接口接收协议。从机 I<sup>2</sup>C 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。

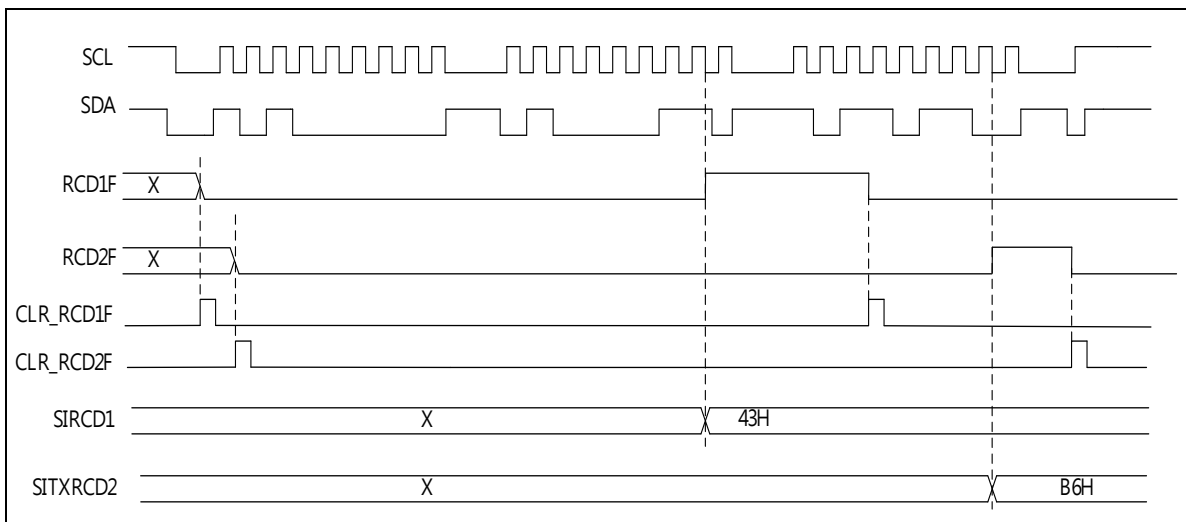


Slave I<sup>2</sup>C Receive Byte protocol



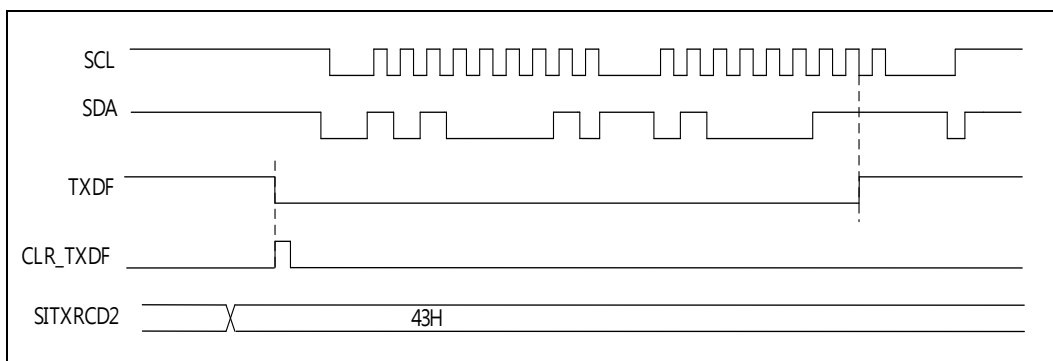
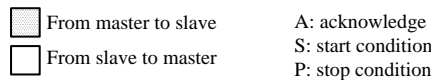
Slave I<sup>2</sup>C Receive Two Byte protocol

- From master to slave
- From slave to master
- A: acknowledge
- S: start condition
- P: stop condition



从机接收时序

该芯片提供以下从机设备 I<sup>2</sup>C 接口传输协议。从 I<sup>2</sup>C 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 I<sup>2</sup>C Transmit protocol


从机发送时序

I <sup>2</sup> C 功能引脚	PINMOD <sub>x</sub>	P2.n SFR data	引脚状态
I <sup>2</sup> C 从 SCL	<b>xx01</b>	1	时钟输入(高阻抗)
I <sup>2</sup> C 主/从 SDA	<b>xx00</b>	1	数据(上拉)

 从 I<sup>2</sup>C 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
<b>INTE1</b>	PWMIE	I2CE	ES2	SPIE	ADTKIE	LVDIE	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: I<sup>2</sup>C 中断使能**

0: 禁用 I<sup>2</sup>C 中断      1: 使能 I<sup>2</sup>C 中断

SFR E9h	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SIADR</b>	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 I<sup>2</sup>C 地址**

E9h.0 **SIEN: 从机 I<sup>2</sup>C 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
<b>SICON</b>	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 I<sup>2</sup>C 发送完成中断使能  
0: 禁用  
1: 使能
- EAh.5 **RCD2IE**: 从机 I<sup>2</sup>C DATA2 (SITXRCD2) 接收完成中断使能  
0: 禁用  
1: 使能
- EAh.4 **RCD1IE**: 从机 I<sup>2</sup>C DATA1 (SIRCD1) 接收完成中断使能  
0: 禁用  
1: 使能
- EAh.2 **TXDF**: 从机 I<sup>2</sup>C 传输完成中断标志  
从机 I<sup>2</sup>C 传输完成时由硬件设置, 写 0 清除
- EAh.1 **RCD2F**: 从机 I<sup>2</sup>C DATA2 (SITXRCD2) 接收完成中断标志  
从机 I<sup>2</sup>C DATA2 (SITXRCD2) 接收完成后由硬件置位, 写 0 清除
- EAh.0 **RCD1F**: 从机 I<sup>2</sup>C DATA1 (SIRCD1) 接收完成中断标志  
从机 I<sup>2</sup>C 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
<b>SIRCD1</b>	SIRCD1							
R/W	R	R	R	R	R	R	R	R
Reset	–	–	–	–	–	–	–	–

- EBh.7~0 **SIRCD1**: 从机 I<sup>2</sup>C 数据接收寄存器 1 (DATA1)

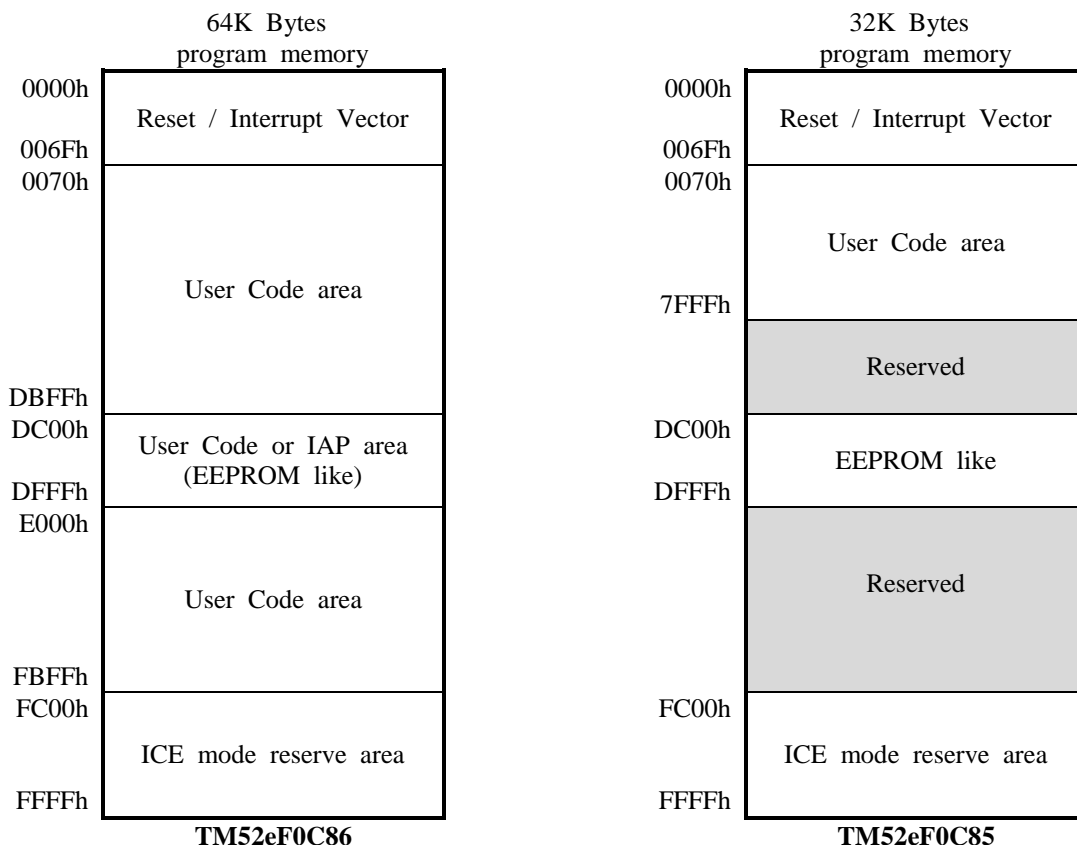
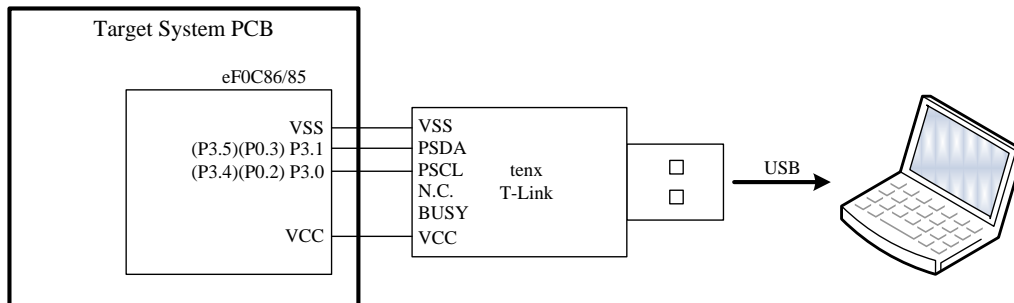
SFR ECh	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
<b>SITXRCD2</b>	SITXRCD2							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Reset	–	–	–	–	–	–	–	–

- ECh.7~0 **SITXRCD2**: 从机 I<sup>2</sup>C 发送和接收数据寄存器  
Read: 从机 I<sup>2</sup>C 数据接收寄存器 2 (DATA2)  
Write: 从机 I<sup>2</sup>C 数据传输寄存器 (TXD)

### 19. 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/P3.1, P02/P30 或 P34/P35 引脚连接到 tenx 专用 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, P02/P30 或 P34/P35 引脚必须在输入模式下工作(PINMODx=xx00b, or xx01b)。
3. Tenx EV 模块占用了程序存储器的寻址空间 FC00h~FFFFh 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 P02/P03 或 P34/P35。  
(P02/P03 或 P34/P35 只能支持 ICE 功能，不支持烧录器)
6. SFR PWRS AV (F7h.5) 在使用 T-Link 模块时会被清除。



**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	<b>P0</b>	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
81h	0000-0111	<b>SP</b>	SP							
82h	0000-0000	<b>DPL</b>	DPL							
83h	0000-0000	<b>DPH</b>	DPH							
85h	xx00-0000	<b>INTPORT</b>	-	-	P5IF	P4IF	P3IF	P2IF	P1IF	P0IF
86h	xxxx-0000	<b>INTPWM</b>	-	-	-	-	PWM3IF	PWM2IF	PWM1IF	PWM0IF
87h	0xxx-0000	<b>PCON</b>	SMOD	-	-	-	GF1	GF0	PD	IDL
88h	0000-0000	<b>TCON</b>	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
89h	0000-0000	<b>TMOD</b>	GATE1	CT1N	TMOD1		GATE0	CTON	TMOD0	
8Ah	0000-0000	<b>TL0</b>	TL0							
8Bh	0000-0000	<b>TL1</b>	TL1							
8Ch	0000-0000	<b>TH0</b>	TH0							
8Dh	0000-0000	<b>TH1</b>	TH1							
8Eh	0100-0000	<b>SCON2</b>	SM2S	-	-	REN2	TB82	RB82	TI2	RI2
8Fh	xxxx-xxxx	<b>SBUF2</b>	SBUF2							
90h	1111-1111	<b>P1</b>	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
91h	xxxx-x000	<b>PORTIDX</b>	-	-	-	-	-	PORTIDX		
94h	0000-0000	<b>OPTION</b>	TM3CKS		WDTPSC		ADCKS		TM3PSC	
95h	xx00-x000	<b>INTFLG</b>	LVDIF	-	TKIFA	ADIF	-	-	PCIF	TF3
96h	0000-0000	<b>INTPIN</b>	PIN7IF	PIN6IF	PIN5IF	PIN4IF	PIN3IF	PIN2IF	PIN1IF	PIN0IF
97h	xxxx-xx00	<b>SWCMD</b>	IAPEN / SWRST / WDTO							
98h	0000-0000	<b>SCON</b>	SM0	SM1	SM2	REN	TB8	RB8	TI	RI
99h	xxxx-xxxx	<b>SBUF</b>	SBUF							
9Ah	0100-0000	<b>SCON1</b>	SM1S	-	-	REN1	TB81	RB81	TI1	RI1
9Bh	xxxx-xxxx	<b>SBUF1</b>	SBUF1							
9Ch	1100-xxxx	<b>TKCON3</b>	TKPDB	TKEOCB	TKIFB	TKXCAPB	JMPVALB			SPREAD
9Dh	1000-0000	<b>PWM2CON</b>	PWM2OM			PWM2DZ				
9Eh	0000-0000	<b>PWMIDX</b>	PWMIDX							
9Fh	0000-0000	<b>PWMEN</b>	PWM3IE	PWM2IE	PWM1IE	PWM0IE	PWM3EN	PWM2EN	PWM1EN	PWM0EN
A0h	1111-1111	<b>P2</b>	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
A1h	1010-1010	<b>PWMCON</b>	PWM3CKS		PWM2CKS		PWM1CKS		PWM0CKS	
A2h	0001-0001	<b>PINMOD10</b>	PINMOD1				PINMOD0			
A3h	0001-0001	<b>PINMOD32</b>	PINMOD3				PINMOD2			
A4h	0001-0001	<b>PINMOD54</b>	PINMOD5				PINMOD4			
A5h	0001-0001	<b>PINMOD76</b>	PINMOD7				PINMOD6			
A6h	0000-0000	<b>PINMODE</b>	VBGEN	-	UART1PS	PSEUDOEN	I2CPS		UART0PS	
A7h	xxx1-1111	<b>TKCHSA</b>	-	-	-	TKCHSA				
A8h	0x00-0000	<b>IE</b>	EA	-	ET2	ES	ET1	EX1	ET0	EX0
A9h	xx00-0000	<b>INTE1</b>	PWMIE	I2CE	ES2	SPIE	ADTKIE	LVDIE	PCIE	TM3IE
AAh	xxxx-xxxx	<b>ADC DL</b>	ADC DL							
ABh	xxxx-xxxx	<b>ADC DH</b>	ADC DH							
ACH	xxx1-1111	<b>TKCHSB</b>	-	-	-	TKCHSB				
ADh	1100-0000	<b>TKCON</b>	TKPDA	TKEOCA	TKRERUN	TKIVCS	TKXCAPA	TKOFFSET	ATKMODE	
Aeh	0011-1111	<b>CHSEL</b>	ADC VREFS			ADC HS				
AFh	0000-0000	<b>ATKCHB2</b>	ATKCHB2							
B0h	1111-1111	<b>P3</b>	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
B1h	0000-0111	<b>LXDCON</b>	LXDON	LXDDUTY			LEDBRITM	LXD BRIT		
B2h	0000-0000	<b>LXDCON2</b>	LCDCKS	LXD PSC		SELLED	LEDHOLD	LEDMODE		
B4h	1111-1111	<b>TKTMRL</b>	TKTMRL							
B5h	0000-0000	<b>TKCON2</b>	TKFJMP	JMPVALA			TKTMRH			
B6h	0000-0000	<b>ATKCHB1</b>	ATKCHB1							
B7h	0000-0000	<b>ATKCHB0</b>	ATKCHB0							
B8h	xx00-0000	<b>IP</b>	-	-	PT2	PS	PT1	PX1	PT0	PX0
B9h	xx00-0000	<b>IPH</b>	-	-	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
Bah	xx00-0000	<b>IP1</b>	PPWM	PI2C	PS2	PSPI	PADTKI	PLVD	PPC	PT3
BBh	xx00-0000	<b>IP1H</b>	PPWMH	PI2CH	PS2H	PSPIH	PADTKIH	PLVDH	PPCH	PT3H



Adr	RST	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BCh	0000-0000	<b>SPCON</b>	SPEN	MSTR	CPOL	CPHA	SSDIS	LSBF	SPCR	
BDh	0000-0xxx	<b>SPSTA</b>	SPIF	WCOL	MODF	RCVOVF	RCVBF	SPBSY	-	-
BEh	0000-0000	<b>SPDAT</b>	SPDAT							
BFh	0x00-0000	<b>LVDCON</b>	LVDM	LVDO	LVDDBS	LVDPD	LVDS			
C0h	1111-1111	<b>P5</b>	P5							
C1h	0000-0000	<b>TKPinsa0</b>	TKPinsa0							
C2h	0000-0000	<b>TKPinsa1</b>	TKPinsa1							
C3h	0000-0000	<b>TKPinsa2</b>	TKPinsa2							
C4h	0000-0000	<b>TKPinsb0</b>	TKPinsb0							
C5h	0000-0000	<b>ATKCHA0</b>	ATKCHA0							
C6h	0000-0000	<b>ATKCHA1</b>	ATKCHA1							
C7h	0000-0000	<b>ATKCHA2</b>	ATKCHA2							
C8h	0000-0000	<b>T2CON</b>	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	CT2N	CPRL2N
C9h	00xx-xxxx	<b>IAPWE</b>	IAPWE / IAPTO							
CAh	0000-0000	<b>RCP2L</b>	RCP2L							
CBh	0000-0000	<b>RCP2H</b>	RCP2H							
CCh	0000-0000	<b>TL2</b>	TL2							
CDh	0000-0000	<b>TH2</b>	TH2							
CEh	0000-0000	<b>EXA2</b>	EXA2							
CFh	0000-0000	<b>EXA3</b>	EXA3							
D0h	0000-0000	<b>PSW</b>	CY	AC	F0	RS1	RS0	OV	F1	P
D1h	0000-0000	<b>PWMDH</b>	PWMDH							
D2h	0000-0000	<b>PWMDL</b>	PWMDL							
D5h	x000-0000	<b>UART2CON</b>	-	UART2BRP						
D6h	00x0-0011	<b>LVRCON</b>	SXTGAIN		-	LVRPD	-	LVRS		
D7h	0000-0000	<b>TKPinsb1</b>	TKPinsb1							
D8h	00x0-0011	<b>CLKCON</b>	SCKTYPE	FCKTYPE	STPSCK	STPPCK	STPFCK	SELFCK	CLKPSC	
D9h	1111-1111	<b>PWMPRDH</b>	PWMPRDH							
DAh	1111-1111	<b>PWMPRDL</b>	PWMPRDL							
DDh	x000-0000	<b>UARTICON</b>	-	UART1BRP						
DEh	0000-0000	<b>UART0CON</b>	UART0BRS	UART0BRP						
DFh	0000-0000	<b>TKPinsb2</b>	TKPinsb2							
E0h	0000-0000	<b>ACC</b>	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
E1h	000x-0100	<b>MICON</b>	MIEN	MIACKO	MIIF	MIACKI	MISTART	MISTOP	MICR	
E2h	0000-0000	<b>MIDAT</b>	MIDAT							
E5h	0000-0000	<b>EFTCON</b>	EFT2CS	EFT1CS	EFT1S		EFTSLOW	EFTWCPU	EFTWOUT	CKHLDE
E6h	0000-0000	<b>EXA</b>	EXA							
E7h	0000-0000	<b>EXB</b>	EXB							
E8h	1111-1111	<b>P4</b>	P4							
E9h	0110-1000	<b>SIADR</b>	SA							SIEN
EAh	0000-x100	<b>SICON</b>	MIIE	TXDIE	RCD2IE	RCD1IE	-	TXDF	RCD2F	RCD1F
EBh	xxxx-xxxx	<b>SIRCD1</b>	SIRCD1							
ECh	xxxx-xxxx	<b>SITXRCD2</b>	SITXRCD2							
EEh	xxxx-x1xx	<b>BOOTV</b>	-	-	-	-	-	RSTV	BOOTVR	
EFh	xxx0-0000	<b>PWRCON</b>	-	-	-	AVPULL	WARMTIME	ENVPULL	PWRIDLE	PWRSLOW
F0h	0000-0000	<b>B</b>	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
F1h	1111-1111	<b>CRCDL</b>	CRCDL							
F2h	1111-1111	<b>CRCDH</b>	CRCDH							
F3h	0000-0000	<b>CRCIN</b>	CRCIN							
F5h	xxxx-xxxx	<b>CFGBG</b>	-	-	-	BGTRIM				
F6h	xxxx-xxxx	<b>CFGWL</b>	-	FRCF						
F7h	0000-1110	<b>AUX2</b>	WDTE		PWRSV	VBGOUT	DIV32	IAPTE		MULDIV16
F8h	0000-0000	<b>AUX1</b>	CLRWDT	CLRTM3	TKSOCA	ADSOC	-	TKSOCB	T1SEL	DPSEL

Flash Address	NAME	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INFO2 0200h	<b>CFGWH</b>	PROTN	XRSTEN	-	-	-	-	BOOTV	

## 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	<b>P0</b>	7~0	P0	R/W	00h	Port0 data
81h	<b>SP</b>	7~0	SP	R/W	07h	Stack Point
82h	<b>DPL</b>	7~0	DPL	R/W	00h	Data Point low byte
83h	<b>DPH</b>	7~0	DPH	R/W	00h	Data Point high byte
85h	<b>INTPORT</b>	5	P5IF	R/W	0h	P5.7~P5.0 pin change interrupt flag, Write 0 to clear P5.7~P5.0 pin change interrupt flag
		4	P4IF	R/W	0h	P4.7~P4.0 pin change interrupt flag, Write 0 to clear P4.7~P4.0 pin change interrupt flag
		3	P3IF	R/W	0h	P3.7~P3.0 pin change interrupt flag, Write 0 to clear P3.7~P3.0 pin change interrupt flag
		2	P2IF	R/W	0h	P2.7~P2.0 pin change interrupt flag, Write 0 to clear P2.7~P2.0 pin change interrupt flag
		1	P1IF	R/W	0h	P1.7~P1.0 pin change interrupt flag, Write 0 to clear P1.7~P1.0 pin change interrupt flag
		0	P0IF	R/W	0h	P0.7~P0.0 pin change interrupt flag, Write 0 to clear P0.7~P0.0 pin change interrupt flag
86h	<b>INTPWM</b>	3	PWM3IF	R/W	0h	PWM3 Interrupt Flag. 1: interrupt asserted, write 0 to clear int flag
		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	<b>PCON</b>	7	SMOD	R/W	0	Set 1 to enable UART0 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 STOP/Halt mode
		0	IDL	R/W	0	Idle control bit, set 1 to enter IDLE mode
88h	<b>TCON</b>	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	<b>TMOD</b>	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	<b>TL0</b>	7~0	TL0	R/W	00h	Timer0 data low byte
8Bh	<b>TL1</b>	7~0	TL1	R/W	00h	Timer1 data low byte
8Ch	<b>TH0</b>	7~0	TH0	R/W	00h	Timer0 data high byte
8Dh	<b>TH1</b>	7~0	TH1	R/W	00h	Timer1 data high byte
8Eh	<b>SCON2</b>	7	SM2S	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	<b>SBUF2</b>	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	<b>P1</b>	7~0	P1	R/W	FFh	Port1 data
91h	<b>PORTIDX</b>	2~0	PORTIDX	R/W	00h	PORT index of INTPIN, PINMOD10, PINMOD32, PINMOD54, PINMOD76

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
94h	OPTION	6	TM3CKS	R/W	0	Timer3 Clock Source Select. 00: SLOW clock (SXT/SRC) 01: FRC/512 10: SLOW clock (SXT/SRC) /2 11: FRC/1024
		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 <sub>SYSCLK</sub> /32 01: F <sub>SYSCLK</sub> /16 10: F <sub>SYSCLK</sub> /8 11: F <sub>SYSCLK</sub> /4
		1~0	TM3PSC	R/W	00	Timer3 Interrupt rate 00: Timer3 Interrupt rate is 65536 Timer3 Clock Source cycle 01: Timer3 Interrupt rate is 16384 Timer3 Clock Source cycle 10: Timer3 Interrupt rate is 4096 Timer3 Clock Source cycle 11: Timer3 Interrupt rate is 1024 Timer3 Clock Source cycle
95h	INTFLG	7	LVDIF	R	-	Low Voltage Detect flag Set by H/W when a low voltage occurs.
		5	TKIFA	R/W	0	Touch Key A 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 TKSOCA 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	PCIF	R/W	0	Port0~Port5 Pin change interrupt flag Set by H/W when Port0~Port5 pin state change is detected and its interrupt enable bit is set. S/W can write 0 to clear all pin interrupt flags (Port0~Port5), it will also clear PIN0IF~PIN7F and P0IF~P5IF.
		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.

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
96h	INTPIN	7	PIN7IF	R/W	0	Px.7 pin change interrupt flag, Write 0 to clear Px.7 pin change interrupt flag port number (x) define by PORTIDX
		6	PIN6IF	R/W	0	Px.6 pin change interrupt flag, Write 0 to clear Px.6 pin change interrupt flag port number (x) define by PORTIDX
		5	PIN5IF	R/W	0	Px.5 pin change interrupt flag, Write 0 to clear Px.5 pin change interrupt flag port number (x) define by PORTIDX
		4	PIN4IF	R/W	0	Px.4 pin change interrupt flag, Write 0 to clear Px.4 pin change interrupt flag port number (x) define by PORTIDX
		3	PIN3IF	R/W	0	Px.3 pin change interrupt flag, Write 0 to clear Px.3 pin change interrupt flag port number (x) define by PORTIDX
		2	PIN2IF	R/W	0	Px.2 pin change interrupt flag, Write 0 to clear Px.2 pin change interrupt flag port number (x) define by PORTIDX
		1	PIN1IF	R/W	0	Px.1 pin change interrupt flag, Write 0 to clear Px.1 pin change interrupt flag port number (x) define by PORTIDX
		0	PIN0IF	R/W	0	Px.0 pin change interrupt flag, Write 0 to clear Px.0 pin change interrupt flag port number (x) define by PORTIDX
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.
98h	SCON	7	SM0	R/W	0	UART0 Serial port mode select bit 0, 1 (SM0, SM1) = 00: Mode0: 8 bit shift register, Baud Rate= $F_{SYSCLK}/2$ 01: Mode1: 8 bit UART0, Baud Rate is variable 10: Mode2: 9 bit UART0, Baud Rate= $F_{SYSCLK}/32$ or $/64$ 11: Mode3: 9 bit UART0, 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 UART0 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	-	UART0 transmit and receive data. Transmit data is written to this location and receive data is read from this location, but the paths are independent.

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
9Ah	SCON1	7	SM1S	R/W	0	UART1 Serial port mode select bit 0: Mode1: 8 bit UART1, Baud Rate is variable 1: Mode3: 9 bit UART1, Baud Rate is variable
		4	REN1	R/W	0	UART1 reception enable 0: Disable reception 1: Enable reception
		3	TB81	R/W	0	Transmit Bit 8, the ninth bit to be transmitted in Mode3
		2	RB81	R/W	0	Receive Bit 8, contains the ninth bit that was received in Mode3
		1	TI1	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	RI1	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.
9Bh	SBUF1	7~0	SBUF1	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.
9Ch	TKCON3	7	TKPDB	R/W	1	Touch Key B Power Down 0: Touch Key B enable; 1: Touch Key B disable
		6	TKEOCB	R	1	Touch Key B end of conversion flag 0: Indicates conversion is in progress 1: Indicates conversion is finished
		5	TKIFB	R/W	0	Touch Key B Interrupt Flag Set by H/W at the end of TK conversion if SYSCLK is fast enough. S/W clear TKIFB or sets the TKSOCB bit to clear this flag.
		4	TKXCAPB	R/W	0	Touch Key B external capacitor select 0: Keep 0, disable Touch Key B external capacitor 1: reserved (Do not set to 1)
		3~1	JMPVALB	R/W	0	Touch Key Clock frequency fine tune , only available in TKFJMP=0 000=frequency slowest, 111=frequency fastest
		0	SPREAD	R/W	0	TK spread spectrum 0: Disable 1: Enable
9Dh	PWM2CON	7~6	PWM2OM	R/W	10	output mode 00: mode 0 01: mode 1 10: mode 2 11: mode 3
		5~0	PWM2DZ	R/W	0	PWM2 Dead zone Control 0000: dead zone disabled 0001: Dead zone width 1*Tpwmclk 0010: Dead zone width 2*Tpwmclk ... 1111: Dead zone width 15*Tpwmclk
9Eh	PWMIDX	7~0	PWMIDX	R/W	FFh	PWM period and duty index. See table 10.1 for more detail 0xh: PWM0 Period/Duty access 1xh: PWM1 Period/Duty access 2xh: PWM2 Period/Duty access 3xh: PWM30~PWM35 Period/Duty access 30h: PWM30 Period/Duty access 31h: PWM31 Period/Duty access 32h: PWM32 Period/Duty access 33h: PWM33 Period/Duty access 34h: PWM34 Period/Duty access 35h: PWM35 Period/Duty access

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
9Fh	PWMEN	7	PWM3IE	R/W	0	PWM3 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		6	PWM2IE	R/W	0	PWM2 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		5	PWM1IE	R/W	0	PWM1 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		4	PWM0IE	R/W	0	PWM0 Interrupt Enable 0: disable 1: enable (note: PWMIE must be 1 at the same time to generate PWM interrupt)
		3	PWM3EN	R/W	0	PWM3 enable 0: PWM3 is cleared and held 1: PWM3 is running
		2	PWM2EN	R/W	0	PWM2 enable 0: PWM2 is cleared and held 1: PWM2 is running
		1	PWM1EN	R/W	0	PWM1 enable 0: PWM1 is cleared and held 1: PWM1 is running
		0	PWM0EN	R/W	0	PWM0 enable 0: PWM0 is cleared and held 1: PWM0 is running
A0h	P2	7~0	P2	R/W	FF	P2.7~P2.0 data
A1h	PWMCON	7~6	PWM3CKS	R/W	10	PWM3 clock source 00: F <sub>SYSCLK</sub> 01: F <sub>SYSCLK</sub> 10: FRC 11: FRC x 2
		5~4	PWM2CKS	R/W	10	PWM2 clock source 00: F <sub>SYSCLK</sub> 01: F <sub>SYSCLK</sub> 10: FRC 11: FRC x 2
		3~2	PWM1CKS	R/W	10	PWM1 clock source 00: F <sub>SYSCLK</sub> 01: F <sub>SYSCLK</sub> 10: FRC 11: FRC x 2
		1~0	PWM0CKS	R/W	10	PWM0 clock source 00: F <sub>SYSCLK</sub> 01: F <sub>SYSCLK</sub> 10: FRC 11: FRC x 2
A2h	PINMOD10	7~4	PINMOD1	R/W	0001	Px.1 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD0	R/W	0001	Px.0 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A3h	PINMOD32	7~4	PINMOD3	R/W	0001	Px.3 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD2	R/W	0001	Px.2 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
A4h	PINMOD54	7~4	PINMOD5	R/W	0001	Px.5 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1
		3~0	PINMOD4	R/W	0001	Px.4 pin control, port index (x) is defined by PORTIDX 0000~1111: see table 7.1

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
A5h	PINMOD76	7~4	PINMOD7	R/W	0001	Px.7 pin control, port index (x) is defined by PORTIDX 0000~1111; see table 7.1
		3~0	PINMOD6	R/W	0001	Px.6 pin control, port index (x) is defined by PORTIDX 0000~1111; see table 7.1
A6h	PINMOD	7	VBGEN	R/W	0	Force VBG generator enable 0: VBG generator is automatically enable and disable 1: Force VBG generator enable included in IDLE mode but disabled in Stop/Halt mode
		5	UART1PS	R/W	0	UART1 Pin Select 0: RXD1/TXD1 = P3.6/P3.7 1: RXD1/TXD1 = P5.4/P5.5
		4	PSEDOEN	R/W	0	P30~P32 pseudo open-drain 0: disable 1: enable
		3~2	I2CPS	R/W	0	I2C Pin Select 00: SCL/SDA = P3.4/P3.5 01: SCL/SDA = P3.0/P3.1 1x: SCL/SDA = P0.2/P0.3
		1~0	UART0PS	R/W	0	UART0 Pin Select 00: RXD0/TXD0 = P3.0/P3.1 01: RXD0/TXD0 = P3.4/P3.5 10: RXD0/TXD0 = P4.4/P4.5 11: Reservd
A7h	TKCHSA	4~0	TKCHS	R/W	1Fh	Specify the first Touch Key A scan channel 0000: TKA00 00001: TKA01 00010: TKA02 00011: TKA03 00100: TKA04 00101: TKA05 00110: TKA06 00111: TKA07 01000: TKA08 01001: TKA09 01010: TKA10 01011: TKA11 01100: TKA12 01101: TKA13 01110: TKA14 01111: TKA15 10000: TKA16 10001: TKA17 10010: TKA18 10011: TKA19 10100: TKA20 10111: TKACAP: internal reference capacitor channel
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 (UART0) 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 & Stop/Halt 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 & Stop/Halt mode wake up capability



Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
A9h	INTE1	7	PWMIE	R/W	0	Set 1 to enable PWM0~PWM2 interrupt
		6	I2CE	R/W	0	Set 1 to enable I <sup>2</sup> C (master/slave) interrupt
		5	ES2	R/W	0	Set 1 to enable Serial Port (UART1/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	LVDIE	R/W	0	Set 1 to enable LVD interrupt.
		1	PCIE	R/W	0	Port0~Port5 pin change interrupt enable. This bit does not affect Stop/Halt mode wake up capability. 0: Disable Port0~Port5 pin change interrupt 1: Enable Port0~Port5 pin change interrupt
		0	TM3IE	R/W	0	Set 1 to enable Timer3 Interrupt and Halt mode wake up
AAh	ADC <sub>DL</sub>	7~4	ADC <sub>DL</sub>	R	-	ADC data bit 3~0
ABh	ADC <sub>DH</sub>	7~0	ADC <sub>DH</sub>	R	-	ADC data bit 11~4
ACh	TKCHSB	4~0	TKCHS	R/W	1Fh	Specify the first Touch Key B scan channel 00000: TKB00 00001: TKB01 00010: TKB02 00011: TKB03 00100: TKB04 00101: TKB05 00110: TKB06 00111: TKB07 01000: TKB08 01001: TKB09 01010: TKB10 01011: TKB11 01100: TKB12 01101: TKB13 01110: TKB14 01111: TKB15 10000: TKB16 10001: TKB17 10010: TKB18 10011: TKB19 10100: TKB20 10111: TKBCAP: internal reference capacitor channel

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
ADh	TKCON	7	TKPDA	R/W	1	Touch Key A Power Down 0: Touch Key A enable; 1: Touch Key A disable
		6	TKEOCA	R	1	Touch Key A end of conversion flag 0: Indicates conversion is in progress 1: Indicates conversion is finished
		5	TKRERUN	R/W	0	TK A/B Auto re-start 0: Auto re-start disable. TKSOCA/B needs to be executed once for each TK A/B conversion 1: Auto re-start enable. After TKSOCA/B is executed once, TK A/B will be converted continuously without re-executing TKSOCA/B
		4	TKIVCS	R/W	0	Touch Key internal voltage control select 0: VCHG=2.8V; VINT=1.4V 1: VCHG=3.6V; VINT=1.8V
		3	TKXCAPA	R/W	0	Touch Key A external capacitor select 0: Keep 0, disable Touch Key A external capacitor 1: reserved (Do not set to 1)
		2	TKOFFSET	R/W	0	status of non-scan TK 0: connect to VSS 1: connect to AC shielding , connect to VSS@EOC
		1~0	ATKMODE	R/W	00	Touch Key Scan Mode 00: TKA and TKB scan method, each channel scan 1 time, max 22 TK channels 01: TKA and TKB scan method, each channel scan 2 times, max 16 TK channels 10: TKA and TKB scan method, each channel scan 4 times, max 8 TK channels 11: TKA and TKB scan method, each channel scan 8 times, max 4 TK channels
AEh	CHSEL	7~6	ADCVREFS	R/W	00	ADC reference voltage. When ADCHS is selected to VBG, ADCVREFS must be set to VCC, otherwise ADC conversion will be invalid 00: VCC 01: 2.5V 10: 3V 11: 4V
		5~0	ADCHS	R/W	1111	ADC channel select 000000: AD00 000001: AD01 ... 101001: AD41 101010: Reserved 101011: V <sub>BG</sub> (Internal Bandgap Reference Voltage) 101100: 1/4V <sub>CC</sub> (Internal Reference Voltage)
AFh	ATKCHB2	7	ATKCHB2	R/W	0	TKBCAP (TKB23) internal reference capacitor channel scan enable: 0: disable      1: enable
		4		R/W	0	TKB20 scan enable: 0: disable      1: enable
		3		R/W	0	TKB19 scan enable: 0: disable      1: enable
		2		R/W	0	TKB18 scan enable: 0: disable      1: enable
		1		R/W	0	TKB17 scan enable: 0: disable      1: enable
		0		R/W	0	TKB16 scan enable: 0: disable      1: enable
B0h	P3	7~0	P3	R/W	FFh	Port3 data

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B1h	LXDCON	7	LXDON	R/W	0	LCD/LED enable 0: LCD/LED disable 1: LCD/LED enable
		4~4	LXDDUTY	R/W	00	LCD/LED duty select LCD select (SELLED=0): 000: 1/4 Duty, COM 0~3 001: 1/4 Duty, COM 0~3 010: 1/5 Duty, COM 0~4 011: 1/6 Duty, COM 0~5 100: 1/6 Duty, COM 0~5 101: 1/8 Duty, COM 0~7 110: 1/8 Duty, COM 0~7 111: 1/8 Duty, COM 0~7  LED select: Matrix mode (SELLED=1, LEDMODE=00b) 000: 1/2 Duty, LCOM 0~1 001: 1/3 Duty, LCOM 0~2 010: 1/4 Duty, LCOM 0~3 011: 1/5 Duty, LCOM 0~4 100: 1/6 Duty, LCOM 0~5 101: 1/7 Duty, LCOM 0~6 110: 1/8 Duty, LCOM 0~7 111: 1/8 Duty, LCOM 0~7  LED select: Dot Matrix mode (SELLED=1, LEDMODE=10b) 000: 4x4, LED 0~4 001: 5x5, LED 0~5 010: 6x6, LED 0~6 011: 6x7, LED 0~6 100: 7x7, LED 0~7 101: 7x8, LED 0~7 110: Reserved 111: Reserved
		3	LEDBRITM	R/W	0	LED Brightness Mode 0: Uniform brightness mode 1: Brightness enhancement mode
		2~0	LXDBRIT	R/W	111	LCD/LED Brightness control 000: Level 0 (Darkest) ... 111: Level 7 (Brightest)

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
B2h	LCNCON2	7	LCDCKS	R/W	0	LCD clock source select and LED Clock Source fixed as FRC 0: SRC/4 1: SXT/2
		6~5	LXDpsc	R/W	00	LCD/LED clock prescaler select 00: LCD/LED clock is FRC divided by 64 01: LCD/LED clock is FRC divided by 32 10: LCD/LED clock is FRC divided by 16 11: LCD/LED clock is FRC divided by 8
		4	SELLED	R/W	0	LCD/LED function select 0: LCD 1: LED
		3	LEDHOLD	R/W	0	Keep at 0, cannot be set to 1
		1~0	LEDMODE	R/W	00	LED Mode select 00: Matrix scan mode 01: Reserved 10: Dot Matrix scan mode 11: Reserved
B4h	TKTMRL	7~0	TKTMRL	R/W	FFh	Touch Key A/B Scan length bit 7~0 adjustment 00: shortest, FF: longest
B5h	TKCON2	7	TKFJMP	R/W	0	Internal Touch Key clock frequency auto adjust option 0: Disable 1: Enable
		6~4	JMPVAL	R/W	000	Touch Key Clock frequency fine tune , only available in TKFJMP=0 00: frequency slowest ... 11: frequency fastest
		1~0	TKTMRH	R/W	00	Touch Key A/B Scan length 9~8 adjustment 00: shortest ... 11: longest
B6h	ATKCHB1	7	ATKCHB1	R/W	0	TKB15 scan enable: 0: disable 1: enable
		6		R/W	0	TKB14 scan enable: 0: disable 1: enable
		5		R/W	0	TKB13 scan enable: 0: disable 1: enable
		4		R/W	0	TKB12 scan enable: 0: disable 1: enable
		3		R/W	0	TKB11 scan enable: 0: disable 1: enable
		2		R/W	0	TKB10 scan enable: 0: disable 1: enable
		1		R/W	0	TKB09 scan enable: 0: disable 1: enable
		0		R/W	0	TKB08 scan enable: 0: disable 1: enable
B7h	ATKCHB0	7	ATKCHB0	R/W	0	TKB07 scan enable: 0: disable 1: enable
		6		R/W	0	TKB06 scan enable: 0: disable 1: enable
		5		R/W	0	TKB05 scan enable: 0: disable 1: enable
		4		R/W	0	TKB04 scan enable: 0: disable 1: enable
		3		R/W	0	TKB03 scan enable: 0: disable 1: enable
		2		R/W	0	TKB02 scan enable: 0: disable 1: enable
		1		R/W	0	TKB01 scan enable: 0: disable 1: enable
		0		R/W	0	TKB00 scan enable: 0: disable 1: enable

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
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	PLVD	R/W	0	External INT2~INT9 Pin Interrupt Priority Low bit
		1	PPC	R/W	0	Port0~Port5 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	PLVDH	R/W	0	External INT2~INT9 Pin Interrupt Priority High bit
		1	PPCH	R/W	0	Port0~Port5 Interrupt Priority High bit
		0	PT3H	R/W	0	Timer3 Interrupt Priority High bit
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: FSYSCLK/2 01: FSYSCLK/4 10: FSYSCLK/8 11: FSYSCLK/16

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
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.
BFh	LVDCON	7	LVDM	R/W	0	Low Voltage Detect interrupt enable 0: LVDIF =1 and LVDO =1 while VCC < LVDS 1: LVDIF =1 and LVDO =1 while VCC > LVDS
		6	LVDO	R	-	Low Voltage Detect output
		5	LVDDBS	R/W	0	Low Voltage Detect debounce select 0: Disable 1: Enable
		4	LVDPD	R/W	0	Low Voltage Detect select 0: Enable LVD 1: Disable LVD
		3~0	LVDS	R/W	0	Low Voltage Detect select 0000: Set LVD at 2.52V 0001: Set LVD at 2.62V 0010: Set LVD at 2.74V 0011: Set LVD at 2.86V 0100: Set LVD at 2.99V 0101: Set LVD at 3.1V 0110: Set LVD at 3.23V 0111: Set LVD at 3.35V 1000: Set LVD at 3.48V 1001: Set LVD at 3.6V 1010: Set LVD at 3.72V 1011: Set LVD at 3.84V 1100: Set LVD at 3.96V 1101: Set LVD at 4.08V 1110: Set LVD at 4.2V 1111: Set LVD at 4.32V
C0h	P5	7~0	P5	R/W	FFh	Port5 data

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
C1h	TKPinsa0	7	TKPinsa0	R/W	0	TKA07 Pin fix as TK channel: 0: disable 1: enable
		6			0	TKA06 Pin fix as TK channel: 0: disable 1: enable
		5			0	TKA05 Pin fix as TK channel: 0: disable 1: enable
		4			0	TKA04 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKA03 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKA02 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKA01 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKA00 Pin fix as TK channel: 0: disable 1: enable
		C2h			TKPinsa1	7
6	0		TKA14 Pin fix as TK channel: 0: disable 1: enable			
5	0		TKA13 Pin fix as TK channel: 0: disable 1: enable			
4	0		TKA12 Pin fix as TK channel: 0: disable 1: enable			
3	0		TKA11 Pin fix as TK channel: 0: disable 1: enable			
2	0		TKA10 Pin fix as TK channel: 0: disable 1: enable			
1	0		TKA09 Pin fix as TK channel: 0: disable 1: enable			
0	0		TKA08 Pin fix as TK channel: 0: disable 1: enable			
C3h	TKPinsa2		4	TKPinsa2		R/W
		3	0		TKA19 Pin fix as TK channel: 0: disable 1: enable	
		2	0		TKA18 Pin fix as TK channel: 0: disable 1: enable	
		1	0		TKA17 Pin fix as TK channel: 0: disable 1: enable	
		0	0		TKA16 Pin fix as TK channel: 0: disable 1: enable	
C4h	TKPINSB0	7	TKPINSB0	R/W	0	TKB07 Pin fix as TK channel: 0: disable 1: enable
		6			0	TKB06 Pin fix as TK channel: 0: disable 1: enable
		5			0	TKB05 Pin fix as TK channel: 0: disable 1: enable
		4			0	TKB04 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKB03 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKB02 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKB01 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKB00 Pin fix as TK channel: 0: disable 1: enable
		C5h			ATKCHA0	7
6	R/W		0	TKB06 scan enable: 0: disable 1: enable		
5	R/W		0	TKB05 scan enable: 0: disable 1: enable		
4	R/W		0	TKB04 scan enable: 0: disable 1: enable		
3	R/W		0	TKB03 scan enable: 0: disable 1: enable		
2	R/W		0	TKB02 scan enable: 0: disable 1: enable		
1	R/W		0	TKB01 scan enable: 0: disable 1: enable		
0	R/W		0	TKB00 scan enable: 0: disable 1: enable		
C6h	ATKCHA1		7	ATKCHB1		R/W
		6	R/W		0	TKB14 scan enable: 0: disable 1: enable
		5	R/W		0	TKB13 scan enable: 0: disable 1: enable
		4	R/W		0	TKB12 scan enable: 0: disable 1: enable
		3	R/W		0	TKB11 scan enable: 0: disable 1: enable
		2	R/W		0	TKB10 scan enable: 0: disable 1: enable
		1	R/W		0	TKB09 scan enable: 0: disable 1: enable
		0	R/W		0	TKB08 scan enable: 0: disable 1: enable
		C7h	ATKCHA2		7	ATKCHA2
4	R/W			0	TKB20 scan enable: 0: disable 1: enable	
3	R/W			0	TKB19 scan enable: 0: disable 1: enable	
2	R/W			0	TKB18 scan enable: 0: disable 1: enable	
1	R/W			0	TKB17 scan enable: 0: disable 1: enable	
0	R/W			0	TKB16 scan enable: 0: disable 1: enable	

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[FA00~FBFF] Write 4Ch to enable one byte IAP write to ROM[FC00~FDFF] Write BAh to enable ERASE 512 byte of ROM[FA00~FBFF] Write BCh to enable ERASE 512 byte of ROM[FC00~FDFF] 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
		6	IAPTO	R	0	IAP Time-Out flag Set by H/W when IAP Time-out occurs. Cleared by H/W when IAPWE=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



Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
D1h	PWMDH	7~0	PWM0DH	R/W	80H	PWM duty high byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0DH access PWMIDX = 1xh: PWM1DH access PWMIDX = 2xh: PWM2DH access PWMIDX = 30h: PWM30DH access PWMIDX = 31h: PWM31DH access PWMIDX = 32h: PWM32DH access PWMIDX = 33h: PWM33DH access PWMIDX = 34h: PWM34DH access PWMIDX = 35h: PWM35DH access  write sequence: PWMDL then PWMDH read sequence: PWMDH then PWMDL
D2h	PWMDL	7~0	PWM0DL	R/W	00H	PWM duty low byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0DH access PWMIDX = 1xh: PWM1DH access PWMIDX = 2xh: PWM2DH access PWMIDX = 30h: PWM30DH access PWMIDX = 31h: PWM31DH access PWMIDX = 32h: PWM32DH access PWMIDX = 33h: PWM33DH access PWMIDX = 34h: PWM34DH access PWMIDX = 35h: PWM35DH access  write sequence: PWMDL then PWMDH read sequence: PWMDH then PWMDL
D5h	UART2CON	6~0	UART2BRP	R/W	00H	UART2 baud rate pre-scaler
D6h	LVRCON	7~6	SXTGAIN	R/W	00	SXT GAIN select 00: lowest ~ 11: highest
		4	LVRPD	R/W	0	Low Voltage Reset function select 0: Disable 1: Enable
		2~0	LVRS	R/W	0000	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
D7h	TKPINSB1	7	TKPINSB1	R/W	0	TKB15 Pin fix as TK channel: 0: disable 1: enable
		6			0	TKB14 Pin fix as TK channel: 0: disable 1: enable
		5			0	TKB13 Pin fix as TK channel: 0: disable 1: enable
		4			0	TKB12 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKB11 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKB10 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKB09 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKB08 Pin fix as TK channel: 0: disable 1: enable
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
		6	FCKTYPE	R/W	0	Fast clock type. This bit can be changed only in Slow mode (SELFCK=0).

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
						0: FRC 1: FXT
		5	STPSCK	R/W	1	Set 1 to stop Slow clock after PD=1 (Halt / Stop mode control)
		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	PWMPRDH	7~0	PWMPRDH	R/W	FFH	PWM period high byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0PRDH access PWMIDX = 1xh: PWM1PRDH access PWMIDX = 2xh: PWM2PRDH access PWMIDX = 3xh: PWM3PRDH access  write sequence: PWMPRDL then PWMPRDH read sequence: PWMPRDH then PWMPRDL
DAh	PWMPRDL	7~0	PWMPRDL	R/W	FFH	PWM period low byte, index by PWMIDX See table 10.1 for more detail PWMIDX = 0xh: PWM0PRDH access PWMIDX = 1xh: PWM1PRDH access PWMIDX = 2xh: PWM2PRDH access PWMIDX = 3xh: PWM3PRDH access  write sequence: PWMPRDL then PWMPRDH read sequence: PWMPRDH then PWMPRDL
DDh	UART1CON	6~0	UART1BRP	R/W	00H	UART1 baud rate pre-scaler
DEh	UART0CON	7	UART0BRS	R/W	00H	UART0 baud rate source select
		6~0	UART0BRP	R/W	00H	UART0 baud rate pre-scaler
DFh	TKPINSB2	4	TKPINSB2	R/W	0	TKB20 Pin fix as TK channel: 0: disable 1: enable
		3			0	TKB19 Pin fix as TK channel: 0: disable 1: enable
		2			0	TKB18 Pin fix as TK channel: 0: disable 1: enable
		1			0	TKB17 Pin fix as TK channel: 0: disable 1: enable
		0			0	TKB16 Pin fix as TK channel: 0: disable 1: enable
E0h	ACC	7~0	ACC	R/W	00h	Accumulator
E1h	MICON	7	MIEN	R/W	0	Master I <sup>2</sup> C enable 0: disable 1: enable
		6	MIACKO	R/W	0	When Master I <sup>2</sup> C receive data, send acknowledge to I <sup>2</sup> C Bus 0: ACK to slave device 1: NACK to slave device
		5	MIF	R/W	0	Master I <sup>2</sup> C Interrupt flag 0: write 0 to clear it 1: Master I <sup>2</sup> C transfer one byte complete
		4	MIACKI	R	-	When Master I <sup>2</sup> C transfer, acknowledgement form I <sup>2</sup> C bus (read only) 0: ACK received 1: NACK received
		3	MISTART	R/W	0	Master I <sup>2</sup> C Start bit 1: start I <sup>2</sup> C bus transfer
		2	MISTOP	R/W	1	Master I <sup>2</sup> C Stop bit 1: send STOP signal to stop I <sup>2</sup> C bus
		1~0	MICR	R/W	00	Master I <sup>2</sup> C (SCL) clock frequency selection 00: Fsys/4 (ex. If Fsys=16MHz, I <sup>2</sup> C clock is 4M Hz) 01: Fsys/16 (ex. If Fsys=16MHz, I <sup>2</sup> C clock is 1M Hz) 10: Fsys/64 (ex. If Fsys=16MHz, I <sup>2</sup> C clock is 250K Hz)

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
						11: Fsys/256 (ex. If Fsys=16MHz, I <sup>2</sup> C clock is 62.5K Hz)
E2h	MIDAT	7~0	MIDAT	R/W	00	Master I <sup>2</sup> C data shift register (W): After Start and before Stop condition, write this register will resume transmission to I <sup>2</sup> C bus (R): After Start and before Stop condition, read this register will resume receiving from I <sup>2</sup> C bus
E5h	EFTCON	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	-	Force SYSCLK to SLOWCLK while EFT detected, Disable ONLY 0: Disable 1: Reserved
		2	EFTWCPU	R/W	0	CPU enter Wait state while EFT detected 0: Disable 1: Enable
		1	EFTWOUT	R/W	1	EFTWAIT output to pin 0: P03 = normal I/O 1: P03 = !EFTWAIT
		0	CKHLDE	R/W	00	clock hold enable 0: Disable 1: Enable
E6h	EXA	7~0	EXA	R/W	00h	Expansion accumulator
E7h	EXB	7~0	EXB	R/W	00h	Expansion B register
E8h	P4	7~0	P4	R/W	FFh	Port 4 data
E9h	SIADR	7~1	SA	R/W	64h	Slave I <sup>2</sup> C address assigned
		0	SIEN	R/W	0	Slave I <sup>2</sup> C enable 0: disable 1: enable
EAh	SICON	7	MIIE	R/W	0	I <sup>2</sup> C Master interrupt enable 0: disable 1: enable
		6	TXDIE	R/W	0	Slave I <sup>2</sup> C transmission completed interrupt enable 0: disable 1: enable
		5	RCD2IE	R/W	0	Slave I <sup>2</sup> C DATA2(SITXRCD2) reception completed interrupt enable 0: disable 1: enable
		4	RCD1IE	R/W	0	Slave I <sup>2</sup> C DATA1(SIRCD1) reception completed interrupt enable 0: disable 1: enable
		2	TXDF	R/W	1	Slave I <sup>2</sup> C transmission completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I <sup>2</sup> C transmission complete
		1	RCD2F	R/W	0	Slave I <sup>2</sup> C DATA2(SITXRCD2) reception completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I <sup>2</sup> C DATA2(SITXRCD2) reception complete enable
		0	RCD1F	R/W	0	Slave I <sup>2</sup> C DATA1(SIRCD1) reception completed interrupt flag 0: write 0 to clear it 1: Set by H/W when Slave I <sup>2</sup> C DATA1(SIRCD1) reception complete
EBh	SIRCD1	7~0	SIRCD1	R	-	Slave I <sup>2</sup> C data receive register1 (DATA1)
ECh	SITXRCD2	7~0	SITXRCD2	R/W	-	Slave I <sup>2</sup> C transmit and receive data register Read: Slave I <sup>2</sup> C data receive register2 (DATA2) Write: Slave I <sup>2</sup> C data transmission register (TXD)

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
EEF	BOOTV	2	RSTV	R/W	1	Change the reset vector (except power on reset) 0: Reset vector = 0x0000 1: Reset vector = 0xE800 or 0xE000 (Decided by BOOTV)
		1~0	BOOTVR	R	-	Power on reset vector select. Read only. Load from CFGWH.BOOTV 00: 0x0000 01: 0x0000 10: 0xE000 (BOOT area 7K bytes) 11: 0xE800 (BOOT area 5K bytes, default)
EFh	PWRCON	4	AVPULL	R/W	0	Auto turn-on VPULL when SLOW to FAST 0: disable 1: enable
		3	WARMTIME	R/W	0	Warm up time after PDOWN 0: 128 Clock 1: 64 Clock
		2	ENVPULL	R/W	0	Power control, force VPULL enable 0: disable 1: enable
		1	PWRIDLE	R/W	0	Power control, VPULL control at IDLE mode 0: VDD = LDO @ IDLE mode 1: VDD = VPULL @ IDLE mode
		0	PWRSLOW	R/W	0	Power control, VPULL control at SLOW mode 0: VDD = LDO @ SLOW mode 1: VDD = VPULL @ SLOW mode
F0h	<b>B</b>	7~0	<b>B</b>	R/W	00h	B register
F1h	<b>CRCDL</b>	7~0	CRCDL	R/W	FFh	16-bit CRC data bit 7~0
F2h	<b>CRCDH</b>	7~0	CRCDH	R/W	FFh	16-bit CRC data bit 15~8
F3h	<b>CRCIN</b>	7~0	CRCIN	W	-	CRC input data
F5h	<b>CFGGB</b>	3~0	BGTRIM	R/W	-	VBG trimming value (Chip Reserved)
F6h	<b>CFGWL</b>	6~0	FRCF	R/W	-	FRC frequency adjustment 00h: lowest frequency 7Fh: highest frequency
F7h	AUX2	7~6	WDTE	R/W	-	Watchdog Timer Reset control 0x: WDT disable 10: WDT enable in Fast/Slow mode, disable in Idle/Stop/Halt mode 11: WDT always enable
		5	PWRSV	R/W	-	Power saving mode control 0: No power saving 1: Power saving, disable LVR in 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, The additional condition VGEN=1 (AEh.1) should be set.
		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 0.8mS trigger watchdog time-out flag 10: wait 3.2mS trigger watchdog time-out flag 11: wait 6.4mS 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

Adr	SFR	Bit#	Bit Name	R/W	Rst	Description
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	TKSOCA	R/W	0	Touch Key A 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 while TKRERUN=0. 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.
		2	TKSOCB	R/W	0	Touch Key B 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 while TKRERUN=0. S/W can also write 0 to clear this flag.
		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
INFO2 0200h	CFGWH	7	PROTN	Flash Code Protect, 0=Protect
		6	XRSTEN	External Pin Reset enable, 0=enable.
		0	BOOTV	Power-on Reset Vector Selection 00: 0x0000 01: 0x0000 10: 0xE000 (BOOT area 7K bytes) 11: 0xE800 (BOOT area 5K bytes, default)

## 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

LOGICAL				
Mnemonic	Description	byte	cycle	opcode
RL A	Rotate A left	1	2	23
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
MOVX A, @Ri	Move external data (A8) to A	1	4	E2-E3
MOVX A, @DPTR	Move external data (A16) to A	1	4	E0
MOVX @Ri, A	Move A to external data (A8)	1	4	F2-F3
MOVX @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≠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( $T_A=25^{\circ}\text{C}$ ,  $V_{CC}=2.0\text{V} \sim 5.5\text{V}$ )

参数	符号	条件	最小值	典型值	最大值	单位	
工作电压	$V_{CC}$	$F_{SYSCLK}=18.432\text{ MHz}$	2.3	–	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
			$V_{CC}=5\text{V}$ , $V_{OH}=0.6V_{CC}$	20	34	–	
			$V_{CC}=3\text{V}$ , $V_{OH}=0.9V_{CC}$	2.5	5	–	
			$V_{CC}=3\text{V}$ , $V_{OH}=0.6V_{CC}$	8	14	–	
I/O 端口 灌电流	$I_{OL}$	所有输出	$V_{CC}=5\text{V}$ , $V_{OL}=0.1V_{CC}$	40	60	–	mA
			$V_{CC}=3\text{V}$ , $V_{OL}=0.1V_{CC}$	20	30	–	
		P1 @ LED mode (High sink)	$V_{CC}=5\text{V}$ , $V_{OL}=0.1V_{CC}$	100	130	–	
			$V_{CC}=3\text{V}$ , $V_{OL}=0.1V_{CC}$	40	58	–	
电源电流	$I_{DD}$	快钟 $V_{CC}=5\text{V}$	FRC=18.432 MHz	–	3.5	–	mA
		快钟 $V_{CC}=3\text{V}$	FRC=18.432 MHz	–	3.5	–	
		慢钟	$V_{CC}=5\text{V}$	–	0.21	–	
			$V_{CC}=3\text{V}$	–	0.2	–	
		空闲 PWRSAV=0	SRC, $V_{CC}=5\text{V}$	–	180	–	$\mu\text{A}$
			SRC, $V_{CC}=3\text{V}$	–	165	–	
		空闲 PWRSAV=1	$V_{CC}=5\text{V}$	–	165	–	
			$V_{CC}=3\text{V}$	–	150	–	
		停止 PWRSAV=0	$V_{CC}=5\text{V}$	–	55	–	
			$V_{CC}=3\text{V}$	–	45	–	
		停止 PWRSAV=1	$V_{CC}=5\text{V}$	–	11	–	
			$V_{CC}=3\text{V}$	–	3.8	–	
暂停 PWRSAV=0	$V_{CC}=5\text{V}$	–	58	–			
	$V_{CC}=3\text{V}$	–	47	–			
暂停 PWRSAV=1	$V_{CC}=5\text{V}$	–	13.5	–			
	$V_{CC}=3\text{V}$	–	5.5	–			
系统时钟频率	$F_{SYSCLK}$	$V_{CC} > LVR_{TH}$	$V_{CC}=2.5\text{V}$	–	–	18.432	MHz

参数	符号	条件	最小值	典型值	最大值	单位	
LVR 参考电压	$V_{LVR}$	$T_A=25^{\circ}\text{C}$	-	2.24	-	V	
			-	2.48	-		
			-	2.72	-		
			-	2.96	-		
			-	3.20	-		
			-	3.44	-		
			-	3.68	-		
LVR 滞后电压	$V_{HYST}$	$T_A=25^{\circ}\text{C}$	-	$\pm 0.1$	-	V	
LVD 参考电压	$V_{LVD}$	$T_A=25^{\circ}\text{C}$	-	2.52	-	V	
			-	2.62	-		
			-	2.74	-		
			-	2.86	-		
			-	2.99	-		
			-	3.1	-		
			-	3.23	-		
			-	3.35	-		
			-	3.48	-		
			-	3.6	-		
			-	3.72	-		
			-	3.84	-		
			-	3.96	-		
			-	4.08	-		
-	4.2	-					
-	4.32	-					
低电压 检测时间	$t_{LVR}$	$T_A=25^{\circ}\text{C}$	100	-	-	$\mu\text{s}$	
上拉电阻	$R_P$	$V_{IN}=0\text{V}$	$V_{CC}=5\text{V}$	-	34	-	K $\Omega$
			$V_{CC}=3\text{V}$	-	58		

**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	$-20^{\circ}\text{C} \sim 50^{\circ}\text{C}, V_{CC}=2.5\text{V} \sim 5.0\text{V}$	-1%	18.432	+1%	MHz
	$-40^{\circ}\text{C} \sim 70^{\circ}\text{C}, V_{CC}=2.5\text{V} \sim 5.0\text{V}$	-1.5%	18.432	+1.5%	
	$-40^{\circ}\text{C} \sim 85^{\circ}\text{C}, V_{CC}=2.5 \sim 5.0\text{V}$	-2%	18.432	+2%	
	$-40^{\circ}\text{C} \sim 105^{\circ}\text{C}, V_{CC}=2.5 \sim 5.0\text{V}$	-3%	18.432	+3%	

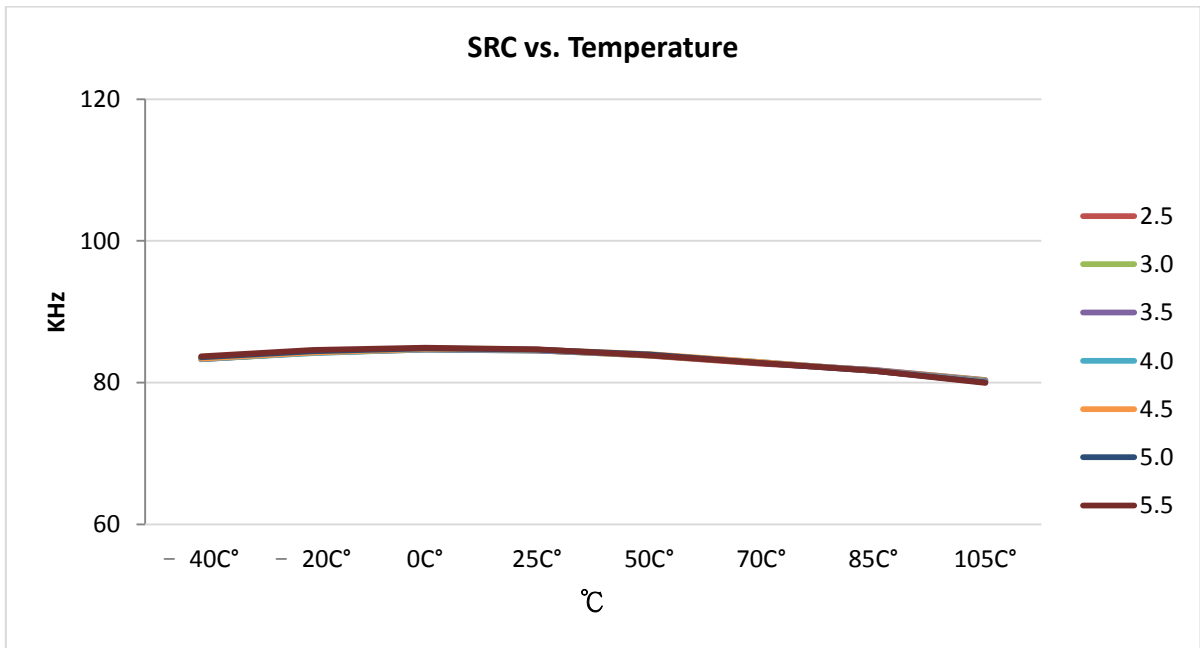
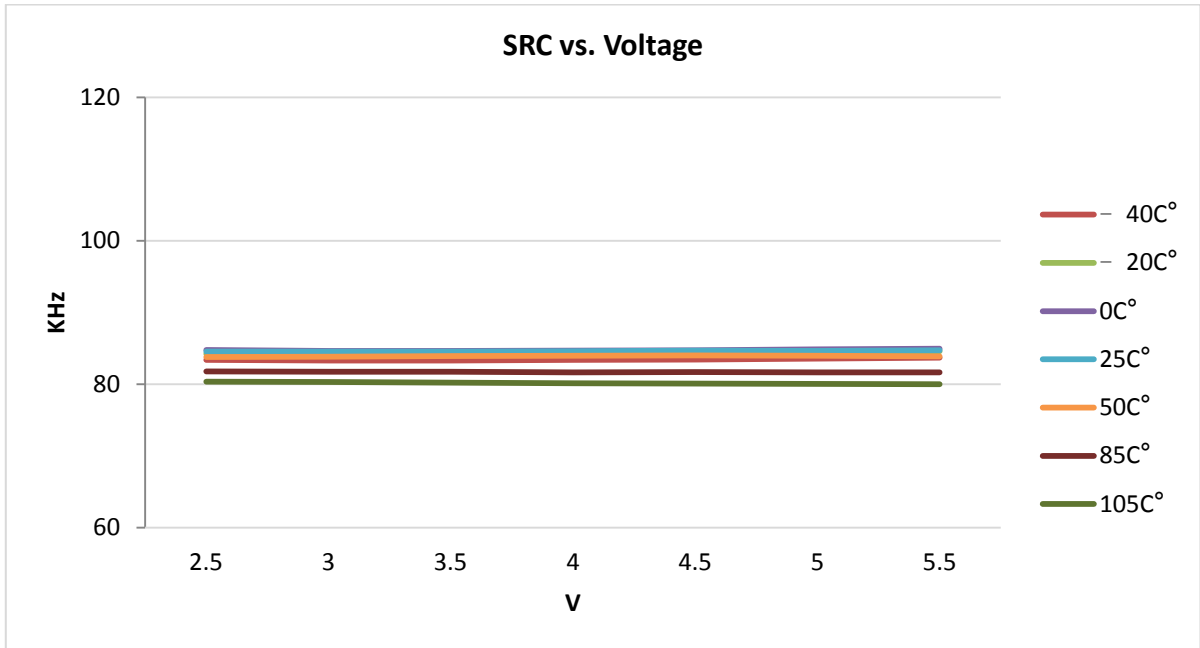
**4. 复位时序特性**( $T_A = -40^{\circ}\text{C} \sim +105^{\circ}\text{C}$ )

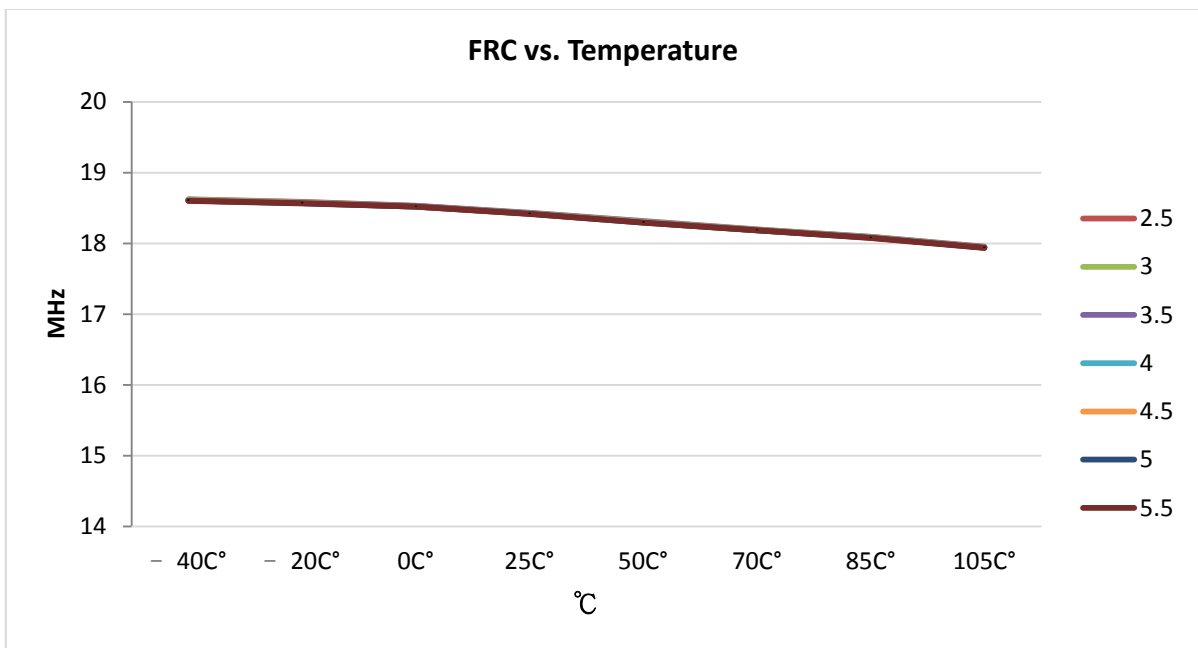
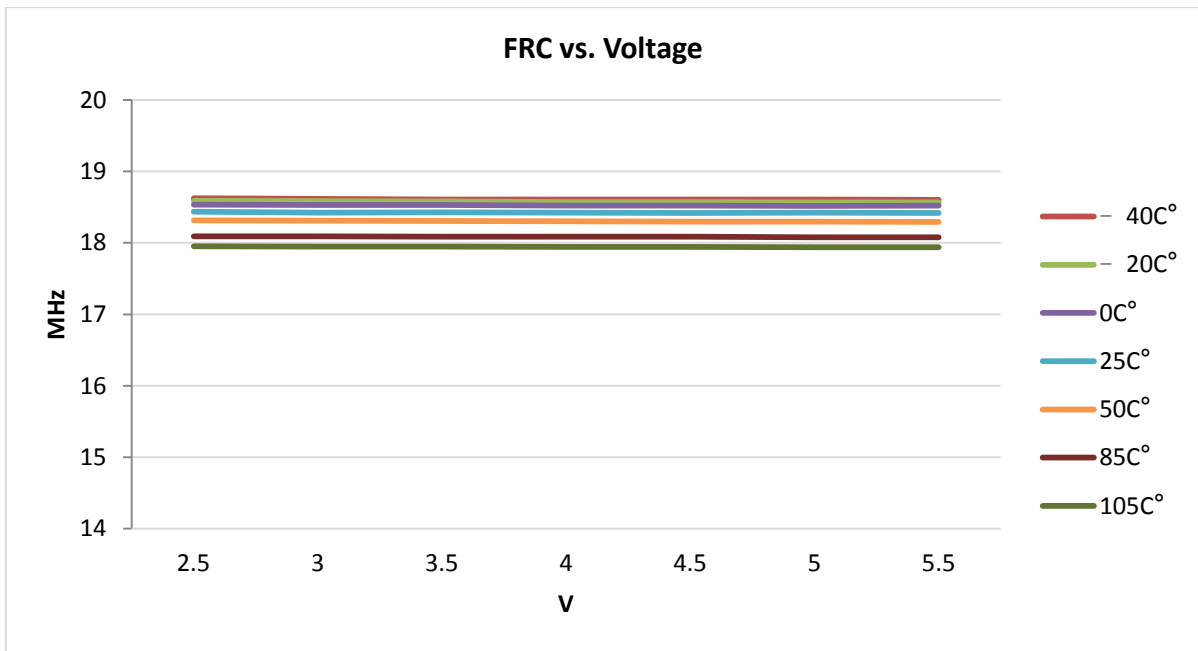
参数	条件	最小值	典型值	最大值	单位
RESET 输入低电平宽度	$V_{CC}=5.0\text{V} \pm 10\%$	30	-	-	$\mu\text{s}$
WDT 唤醒时间	$V_{CC}=5.0\text{V}, \text{WDTPSC}=11$	-	52	-	ms
	$V_{CC}=3.0\text{V}, \text{WDTPSC}=11$	-	57	-	

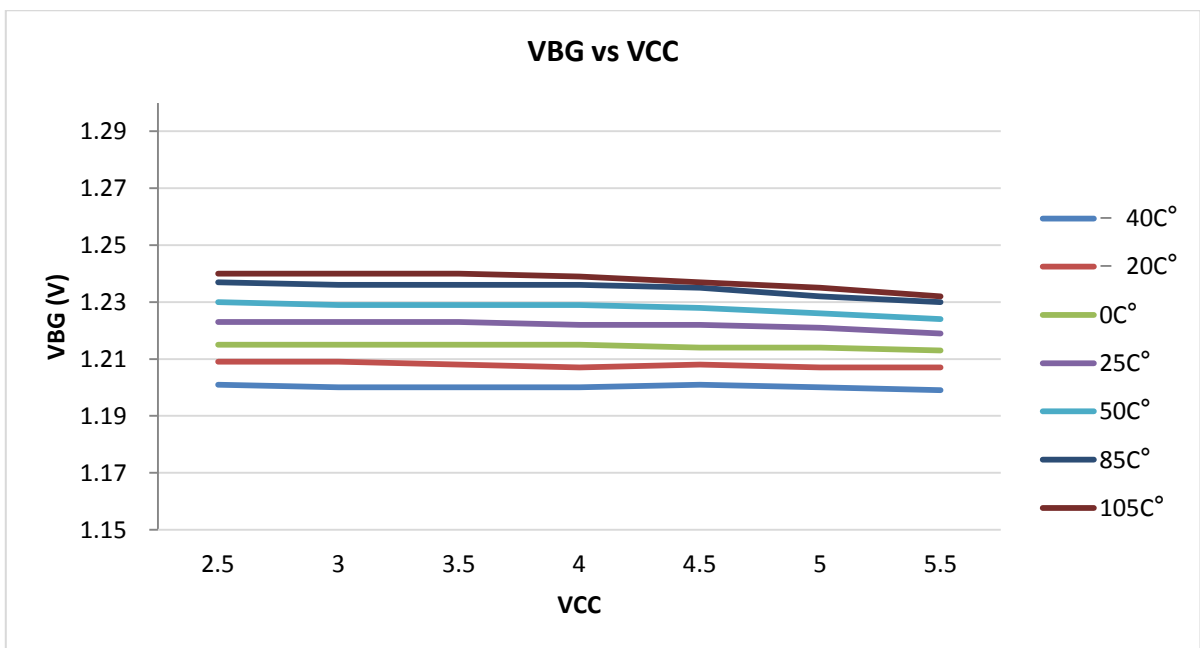
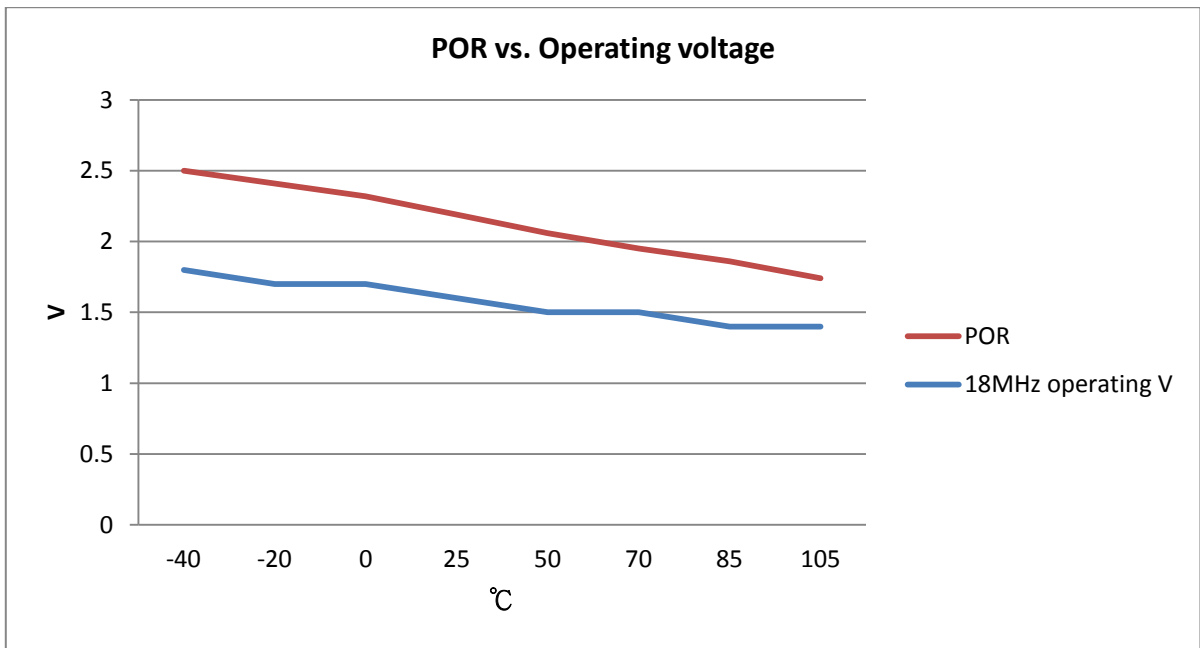
**5. ADC 电气特性**( $T_A = 25^{\circ}\text{C}, V_{CC} = 3.0\text{V} \sim 5.5\text{V}, V_{SS} = 0\text{V}$ )

参数	条件	最小值	典型值	最大值	单位	
总绝对误差	$V_{CC}=5.12\text{V}, V_{SS}=0\text{V}$	-	$\pm 2.5$	$\pm 4$	LSB	
积分非线性误差		-	$\pm 3.2$	$\pm 5$		
最大输入时钟( $f_{\text{ADC}}$ )	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 VBG (ADCHS=1011b)	-	-	0.5		
转换时间	$F_{\text{ADC}} = 1\text{MHz}$	-	50	-	$\mu\text{s}$	
带隙基准电压 ( $V_{\text{BG}}$ )	-	$V_{CC}=2.5\text{V} \sim 5.5\text{V}$ $-20^{\circ}\text{C} \sim 85^{\circ}\text{C}$	-1.5%	1.22	+1.5%	V
		$V_{CC}=2.5\text{V} \sim 5.5\text{V}$ $-40^{\circ}\text{C} \sim 105^{\circ}\text{C}$	-2%	1.22	+2%	
ADC 参考电压 ( $V_{\text{ADC}}$ )	ADCVREFS=1	$V_{CC}=2.5\text{V} \sim 5\text{V}$ $-20^{\circ}\text{C} \sim 85^{\circ}\text{C}$	-1.5%	2.5	+1.5%	
		$V_{CC}=2.5\text{V} \sim 5\text{V}$ $-40^{\circ}\text{C} \sim 105^{\circ}\text{C}$	-2%	2.5	+2%	
VCC / 4 基准电压 ( $V_{1/4}$ )	-	$V_{CC}=5\text{V}, 25^{\circ}\text{C}$	-0.8%	1.26	+0.8%	
		$V_{CC}=3.6\text{V}, 25^{\circ}\text{C}$	-0.8%	0.907	+0.8%	
输入电压	-	$V_{SS}$	-	$V_{CC}$		

### 3. 特性曲线图







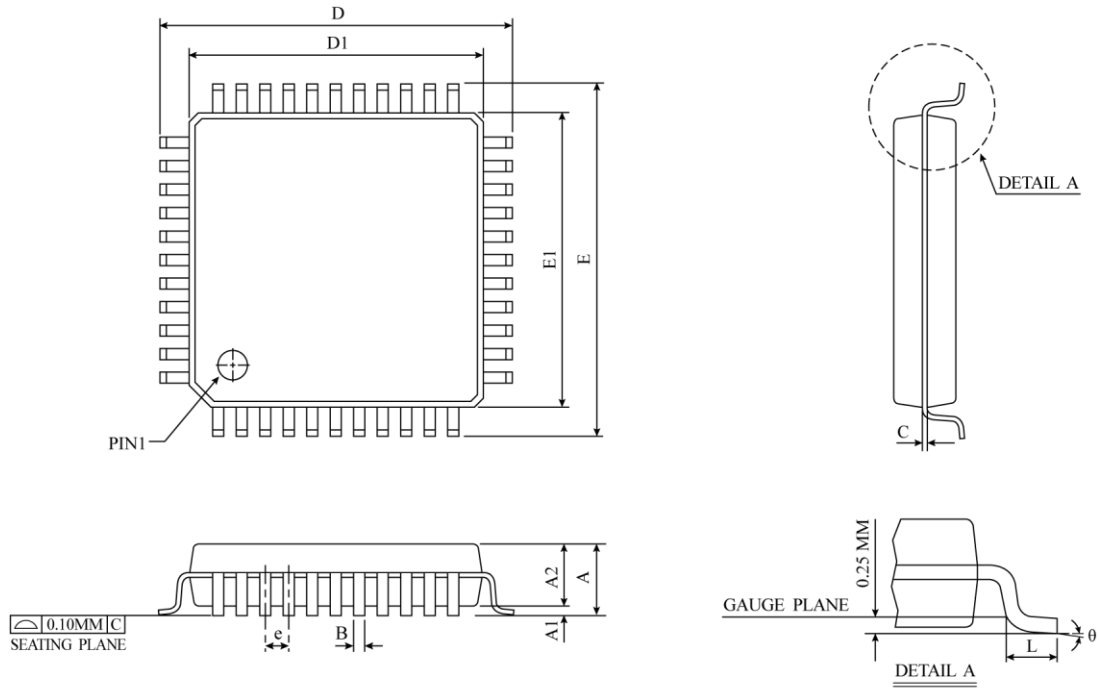
## 封装说明

请注意，此處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
TM52eF0C86-MTP	Wafer/Dice blank chip
TM52eF0C85-MTP	
TM52eF0C86-COD	Wafer/Dice with code
TM52eF0C85-COD	
TM52eF0C86-MTP-74	LQFP 44-pin (10*10*1.4mm)
TM52eF0C85-MTP-74	
TM52eF0C86-MTP-23	SOP 28-pin (300 mil)
TM52eF0C85-MTP-23	

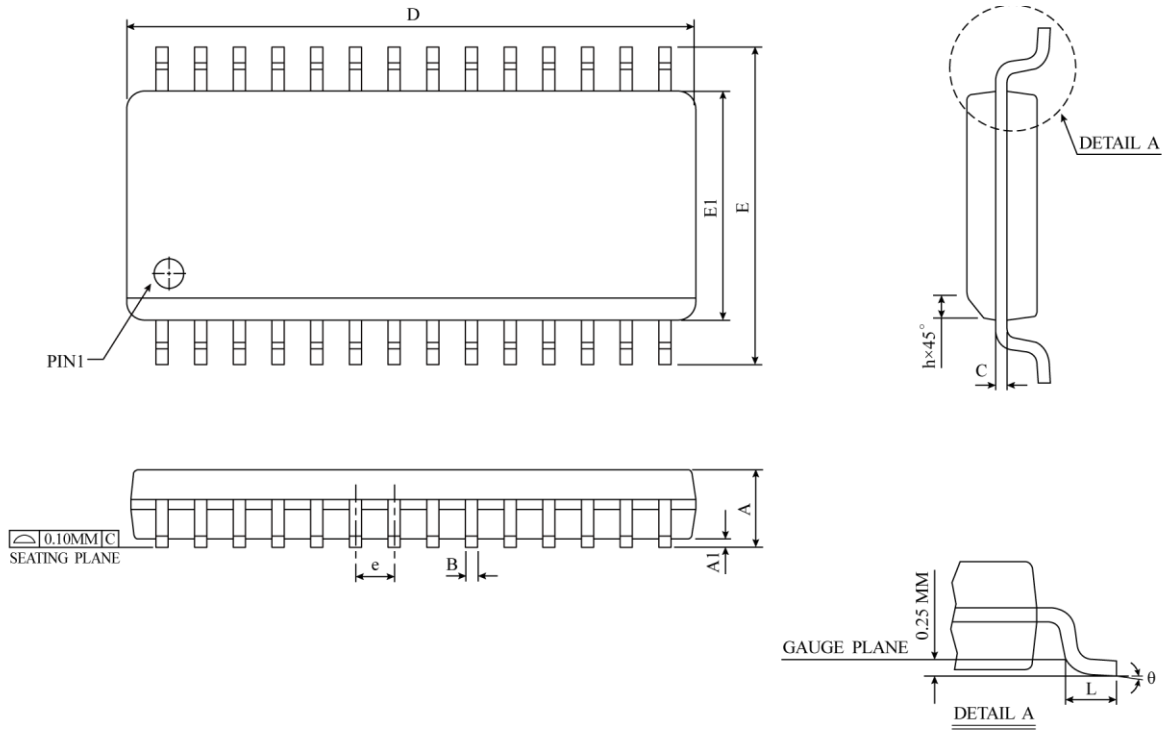


**LQFP-44 ( 10×10mm ) Package Dimension**


SYMBOL	DIMENSION IN MM			DIMENSION IN INCH		
	MIN	NOM	MAX	MIN	NOM	MAX
A	-	-	1.60	-	-	0.063
A1	0.05	0.10	0.15	0.002	0.004	0.006
A2	1.35	1.40	1.45	0.053	0.055	0.057
B	0.30	0.35	0.40	0.012	0.012	0.016
C	0.09	0.13	0.16	0.004	0.006	0.008
D	12.00 BSC			0.472 BSC		
D1	10.00 BSC			0.394 BSC		
E	12.00 BSC			0.472 BSC		
E1	10.00 BSC			0.394 BSC		
e	0.80 BSC			0.031 BSC		
L	0.45	0.60	0.75	0.018	0.024	0.030
θ	0°	3.5°	7°	0°	3.5°	7°
JEDEC	MS-026 (BCB)					

▲ \* NOTES : DIMENSION " D1 " AND " E1 " DO NOT INCLUDE MOLD PROTRUSIONS. ALLOWABLE PROTRUSIONS IS 0.25 mm PER SIDE.  
 " D1 " AND " E1 " ARE MAXIMUM PLASTIC BODY SIZE DIMENSIONS INCLUDING MOLD MISMATCH.

SOP-28 ( 300mil ) Package Dimension



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.