



苏州华芯微电子股份有限公司
SUZHOU HUAXIN MICRO-ELECTRONICS CO.,LTD.

www.h-sun.com

苏州华芯微电子股份有限公司



HuaXin Micro-electronics Co., Ltd

HS1306

参考手册



目 录

1. 概述.....	1
2. 特征.....	1
3. 引脚分配（三种方式,32 脚固定，其他封装可以随要求改变）.....	2
4. 系统框图.....	4
5. 器件描述.....	4
5.1 存储器结构.....	4
5.1.1 程序存储器.....	4
5.1.2 配置存储器.....	5
5.1.3 数据存储器.....	5
5.2 操作寄存器.....	8
5.2.1 H/L（间接寻址寄存器）.....	8
5.2.2 Y/Z（间接寻址寄存器）.....	8
5.2.3 R（间接寻址寄存器）.....	9
5.2.4 PCL（程序计数器）与堆栈.....	10
5.2.5 OPTION（功能寄存器）.....	11
5.2.6 STATUS（状态寄存器）.....	11
5.2.7 PORT（端口寄存器）.....	12
5.2.8 TMR0、TMR1（定时/计数器寄存器）.....	14
5.2.8.1 TnMD 寄存器.....	14
5.2.8.2 TnRL 寄存器.....	15
5.2.8.3 TC 时钟频率输出（蜂鸣器输出）.....	16
5.2.8.4 PWM.....	16
5.2.9 INTCON（中断控制器）.....	20
5.2.10 SIM（SIM 控制器）.....	21
5.2.10.1 SPI 模式.....	22
5.2.10.2 I2C 模式.....	24
5.2.11 触摸控制模块.....	26
5.2.11.1 TKOP0（Touch 功能 0 寄存器）.....	26
5.2.11.2 TKOP1（Touch 功能 1 寄存器）.....	27
5.2.11.3 TKOP2（Touch 功能 2 寄存器）.....	28
5.2.11.4 TKTM0.....	29
5.2.11.5 TKM0EN.....	29
5.2.11.6 TKM1EN.....	29
5.2.11.6 TKM2EN.....	30
5.2.11.7 TKMCT.....	30
5.2.12 OPTION2.....	31
5.2.13 RBANK.....	31
5.2.14 STKP(堆栈控制器).....	31
5.2.15 堆栈操作.....	32
5.2.16 OSCM(堆栈控制器).....	33



5.3 看门狗定时器.....	34
5.4 省电模式.....	34
5.5 从 SLEEP 状态唤醒.....	34
5.6 中断.....	35
5.7 复位.....	35
5.8 CODE 操作寄存器.....	35
6. 电气参数.....	37
6.1 极限参数.....	37
6.2 DC 特性.....	37
6.3 AC 特性.....	38
6.4 WDT 的溢出时间 (1:1)	38
7. 封装及尺寸.....	39
7.1 封装图.....	39
7.2 尺寸.....	39
8. 修正记录.....	40



HS1306

产品说明书

Ver 1.10A

1. 概述

HS1306 是采用低耗高速 CMOS 工艺制造的 8 位单片机，它内部包含一个 6K*16-bit 的一次性可编程只读电存储器(OTP-ROM)。有 16 位选项位可满足用户要求。

2. 特征

- ◆ 宽工作电压范围：2.2V~5.5V
- ◆ 工作温度范围：0°C~70°C（实际可工作-10~85°C）
- ◆ 工作频率范围：
 - 内置 RC（4MHZ/8MHz 可选）±2%
 - 外置双端振荡器，2MHz~12MHz
- ◆ 系统操作模式：
 - 普通模式：高速时钟工作。
 - 睡眠模式：高、低速时钟同时停止工作，此模式下可设置作M-KEY键盘扫描，功耗<1uA。
 - 低速模式：低速时钟工作，高速时钟停止工作，此模式下可设置作T-KEY键盘扫描，功耗<1uA。
- ◆ 6K*16 位片内 ROM
- ◆ 一个安全位（代码寄存器中）保护程序不被读出
- ◆ 384*8bits 片内寄存器组（bank0, bank1）
- ◆ 4 组双向 I/O 端口
- ◆ 8 级用于子程序嵌套的堆栈
- ◆ 2 个 8 位定时/计数器，可用以 PWM 输出
- ◆ 6 个中断源
- ◆ 1 个 16 位定时/计数器
- ◆ 每个指令周期为两个时钟周期，大部分指令为单周期指令
- ◆ SPI 协议接口，I2C 协议接口（SPI 和 I2C 共用寄存器，所以同时只能用其一）
- ◆ 24 个可检测电容端口，可与检测手指触摸电容。每个电容端口可单独控制
- ◆ 内置红外发射放大驱动电路
- ◆ 内部上电复位电路(POR)
- ◆ 内置低电压检测（LVD），用于欠压复位（BOR）
- ◆ 内置电压检测（CVD，Check Voltage Device 2.4V/3.2V），用于电源电压检测



3. 引脚分配（三种方式,32脚固定，其他封装可以随要求改变）

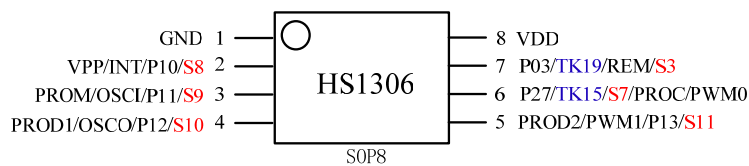
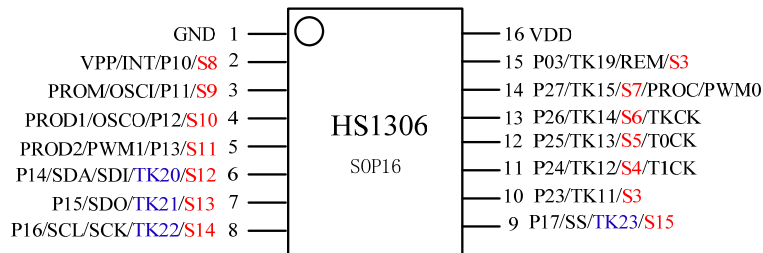
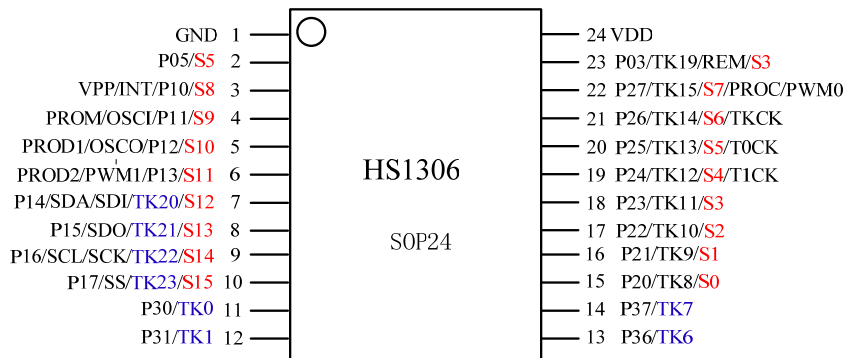
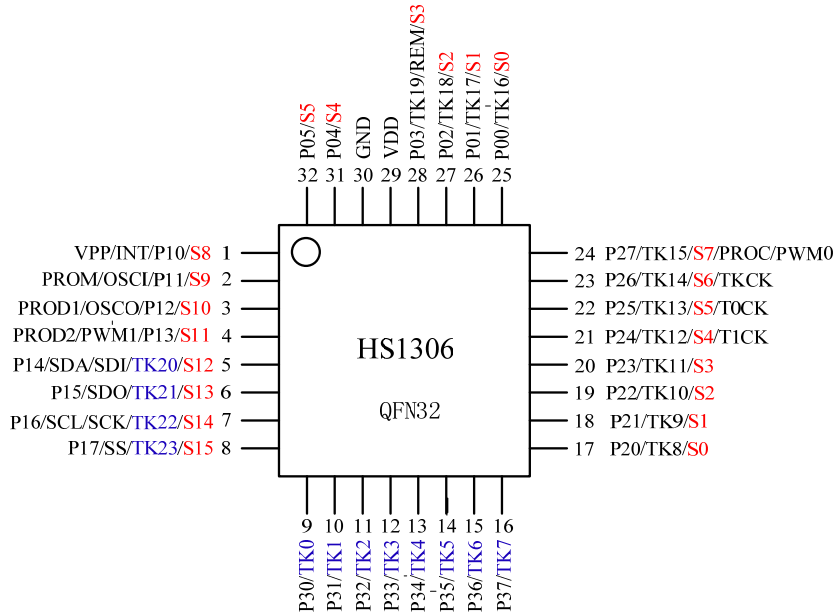




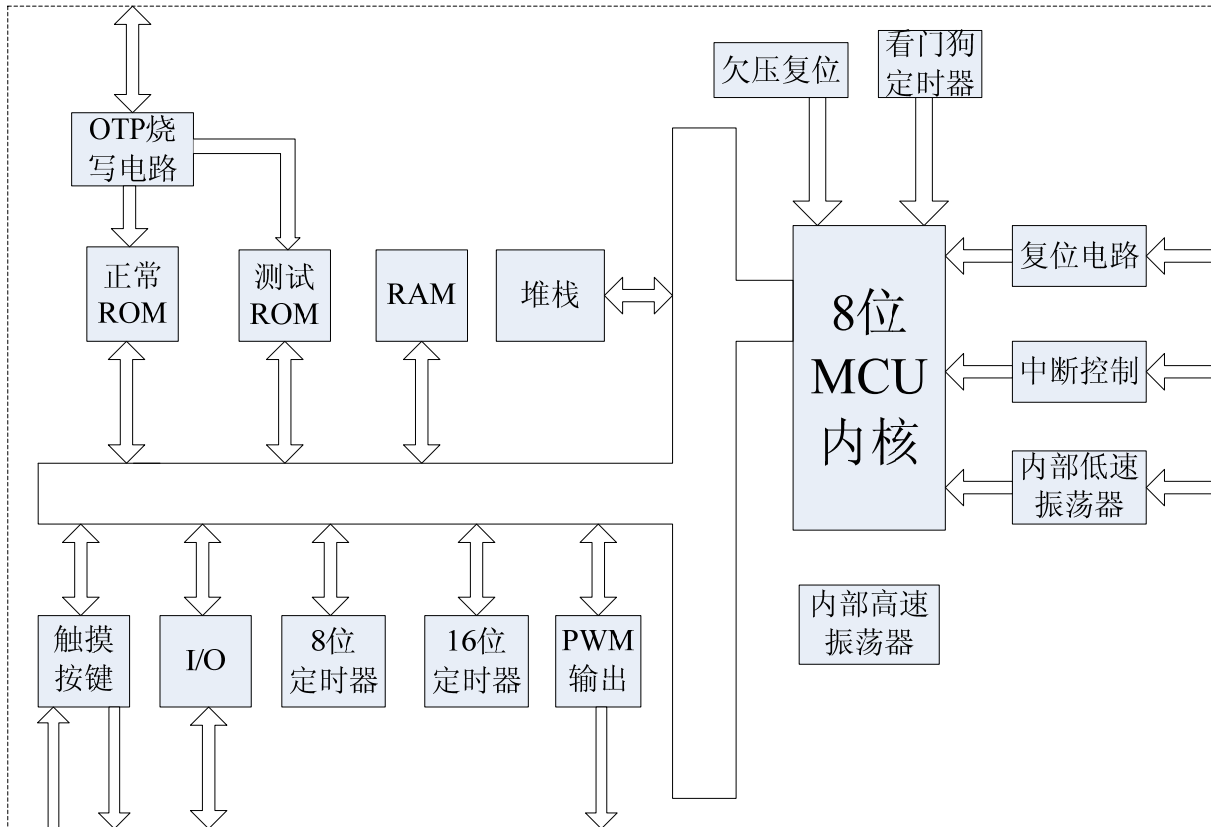
表 3-1 封装管脚说明

管脚名称	输入/输出	说 明
VDD	I	电源正端（+）输入端；
GND	I	电源负端（-）输入端；
P00~P37	I/O	普通端口，有四种状态，分别是输入，输出，转 I/O，带上拉输入
T1CK,T0CK	I	TMR1, TMR0 计数功能输入脚
S0~S13	I/O	T-KEY 扫描脚（可通过寄存器设置睡眠时的动态扫描脚）：应用于不同方案的键盘电路；
TK0~TK23	I/O	触摸输入。设置该功能时，端口无 I/O 特性，端口状态呈三角波振荡波形。
VPP	A	OTP 烧写高压脚
PWM1,PWM2	O	PWM 波形输出脚，驱动与 I/O 一致。
REM	O	红外驱动，也可以进行红外输入学习。
SDI,SCK SDO,SS	I/O	SPI 协议定义脚位
INT	I	中断输入脚。
PROC	I	OTP 烧写时时钟输入端
PROM	I	OTP 烧写时，模式选择端。
PROD1, PROD2	I/O	OTP 烧写时，数据输入输出端。2 个可加快烧写效率。
TKCK	I	TKTMR 计数器输入端
OSCI	I	振荡器输入端。
OSCO	O	振荡器输出端。振荡器可以 455K~8MHz.
SDA,SCL	I/O	I2C 协议定义脚位

注：I=input,O=output, I/O=input/output, A=Analog input



4. 系统框图



5. 器件描述

5.1 存储器结构

CORE 的存储器是由程序存储器和数据存储器构成。

5.1.1 程序存储器

CORE 有 13 位程序计数器，能够寻址 $6K \times 16$ 个地址空间。

复位时，程序计数器的地址为 0000h，为第一个地址。

响应中断后，PC 地址转到 0008H 执行。

CORE 支持 CALL/GOTO 执行全地址跳转。

8 级 13 位 STACK，可以实现 8 级程序嵌套。

如图 5-1 所示 ROM 结构表。



图 5-1

5.1.2 配置存储器

配置寄存器在上电初始化时寻址，和 ROM 同一个地址空间。配置寄存器可以对内置 Rc 振荡器精度、器件工作方式等进行选择。



5.1.3 数据存储

数据存储器分为Bank0 和Bank1 两个区域。通过寄存器“RBANK”决定访问哪个Bank。如RBANK=0，直接访问Bank0，而当RBANK=1，则直接访问Bank1。如果在一个Bank 区域但需要访问另一个Bank 区域，就必须设置寄存器“RBANK”。Bank0的后半部分为特殊寄存器地址，主要一些特殊功能的硬件配置使用。

00H~7FH	通用寄存器 RAM	BANK0
80H~FFH	特殊寄存器	
100H~1FFH	通用寄存器 RAM	BANK1

地址	寄存器	地址	寄存器	地址	寄存器	地址	寄存器
80H	H			A0H	P0WP	A8H	TKTM0H
81H	L			A1H	P1WP	A9H	TKMCTH
82H	R			A2H	P2WP	AAH	TKMCTL
83H	Z						
84H	Y	9CH	TKM1En	A4H	TKOP1		
		9DH	TKM2En	A5H	SPIC1		
86H	STATUS			A6H	SPIBUF		
87H	RBANK			A7H	TKM0En		

地址	寄存器	地址	寄存器	地址	寄存器	地址	寄存器
		B8H	TKTM0L	C0H	P0M0	C8H	INTFLAG
				C1H	P1M0	C9H	INTCON
				C2H	P2M0	CAH	OSCM
				C3H	P3M0		
B4H	TKOP0					CCH	WDTR
B5H	SPIC0					CDH	OPTION2
B6H	TKOP2					CEH	PCL
						CFH	PCH



地址	寄存器	地址	寄存器	地址	寄存器	地址	寄存器
D0H	P0	D8H	T0MD	E0H	P0M1		
D1H	P1	D9H	T0C	E1H	P1M1		
D2H	P2	DAH	T0RL	E2H	P2M1		
D3H	P3	DBH	OPTION	E3H	P3M1		
		DCH	T1MD				
		DDH	T1C				
		DEH	T1RL	E6H	@HL		
		DFH	STKP	E7H	@YZ		



5.2 操作寄存器

5.2.1 H/L（间接寻址寄存器）

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
80H(R/W)	H	1、通用工作寄存器； 2、RAM 数据寻址指针@HL。							
81H(R/W)	L								
复位状态	BOR	x	x	x	x	x	x	x	x
	WDT	u	u	u	u	u	u	u	u

例：用 H、L 作为数据指针，访问bank0中020H处的内容。

```
B0MOV H, #00H
B0MOV L, #20H
B0MOV A, @HL
```

例：对bank 0 中的数据进行清零处理。

```
CLR H           ; H = 0, 指向bank 0。
B0MOV L, #7FH  ; L = 7FH。
CLR_HL_BUF:
CLR @HL        ; @HL 清零。
DECMS L        ; L - 1, 如果 L = 0, 程序结束。
JMP CLR_HL_BUF
CLR @HL
END_CLR:
...
...
```

5.2.2 Y/Z（间接寻址寄存器）

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
83H(R/W)	Z	1、通用工作寄存器； 2、RAM 数据寻址指针@HL。 3、配合指令 MOVC 对 ROM 数据进行查表。							
84H(R/W)	Y								
复位状态	BOR	x	x	x	x	x	x	x	x
	WDT	u	u	u	u	u	u	u	u

例：用 Y、Z 作为数据指针，访问bank0 中020H 处的内容。

```
B0MOV Y, #00H
B0MOV Z, #20H
B0MOV A, @YZ
```

例：对bank 0 中的数据进行清零处理。

```
CLR Y           ; Y = 0, 指向bank 0。
B0MOV Z, #7FH  ; Z = 7FH。
```



CLR_YZ_BUF:

```

CLR @YZ                ;@YZ 清零。
DECMS Z                ;Z - 1, 如果 Z = 0, 程序结束。
JMP CLR_YZ_BUF
CLR @YZ
END_CLR:
...
...

```

5.2.3 R (间接寻址寄存器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
82H(R/W)	R	1、作为工作寄存器使用; 2、存储执行查表指令后的高字节数据。(执行MOVC指令,指定ROM单元的高字节数据会被存入R寄存器而低字节数据则存入ACC。)							
复位状态	BOR	x	x	x	x	x	x	x	x
	WDT	u	u	u	u	u	u	u	u

例：查找 ROM 地址为“TABLE1”的值。

```

B0MOV Y, #TABLE1$M    ; 设置 TABLE1 地址高字节。
B0MOV Z, #TABLE1$L    ; 设置 TABLE1 地址低字节。
MOVC                   ; 查表, R = 00H, ACC = 35H。
                       ; 查找下一地址。

INCMS Z
JMP @F                 ; Z 没有溢出。
INCMS Y                 ; Z 溢出 (FFH -> 00), -> Y=Y+1
NOP                    ;

@@:    MOVC             ; 查表, R = 51H, ACC = 05H。
... ;

TABLE1:
DW 0035H                ; 定义数据表 (16 位) 数据。
DW 5105H
DW 2012H
...

```



5.2.4 PCL（程序计数器）与堆栈

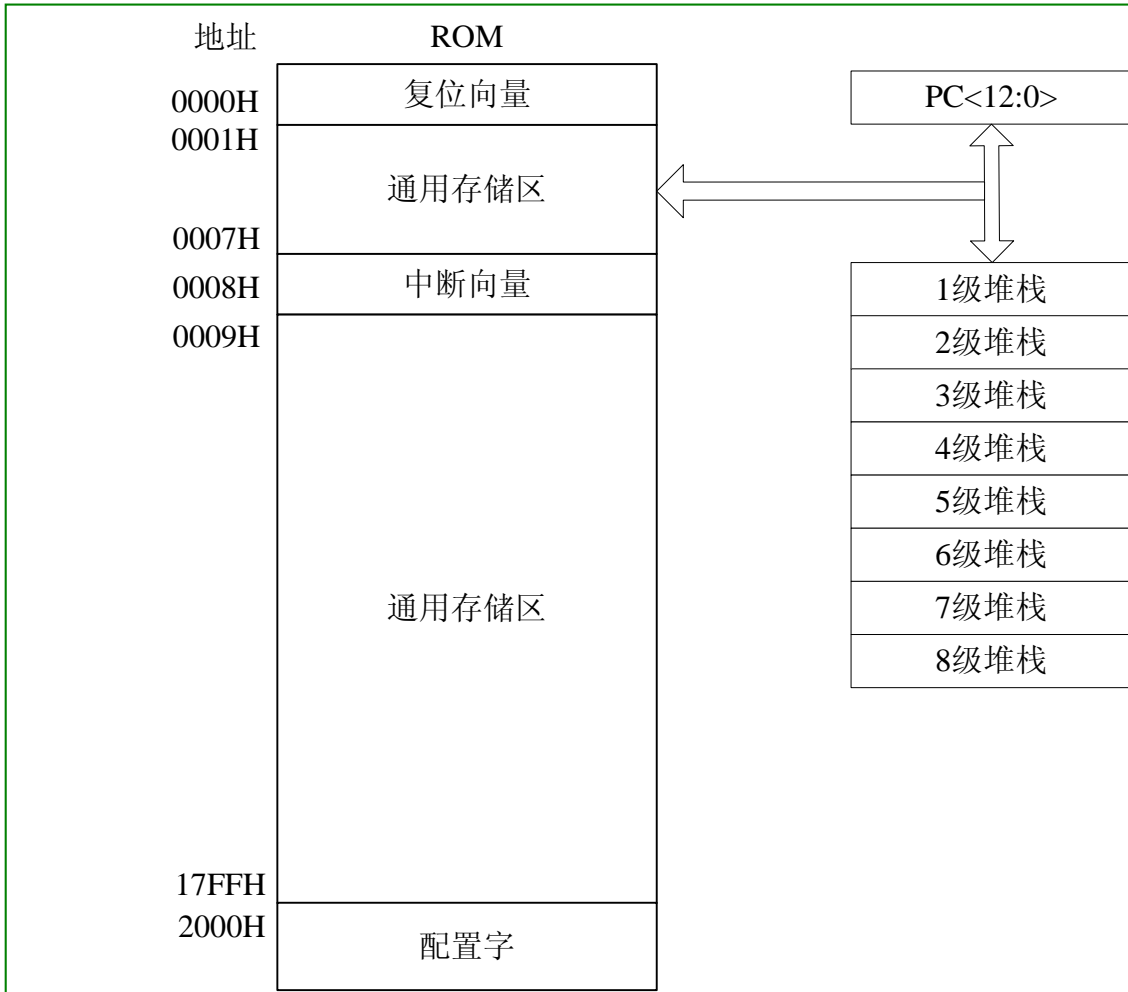


图 5-3 程序计数器结构

	Bit12	Bit11	Bit10	Bit9	Bit8	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PC	PC12	PC11	PC10	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
复位	0	0	0	0	0	0	0	0	0	0	0	0	0
	PCH						PCL						

PCH, PCL 都是可读可写的。

- ◆ 产生 6K*16bits 片内 OTP ROM 地址以获取对应程序指令代码。
- ◆ 复位后 PCL 的所有位均清零
- ◆ “GOTO” 指令直接装载 PC 的 13 位。因此，“GOTO” 指令跳转范围 4K 程序区间。
- ◆ “CALL” 指令加载 PC 的 13 位，然后 PC+1 进入堆栈。
- ◆ “RET” (“RETLW K”, “RETI”, “RETI”) 指令将栈顶数据装入 PC
- ◆ “ADDWF PCL,1” 允许 “A” 的值加到当前 PC，PC 的高 5 位将自然进位
- ◆ “MOVWF PCL” 允许将寄存器 “A” 的值装入 PC 的低 8 位，同时 PC 的高 5 位保持不变。
- ◆ “SUBWF PCL,1” 允许 “A” 的值被减到当前 PC，但 PC 的高 5 位保持不变！
- ◆ 改变 PCL 内容的指令需要 2 个指令周期，除此之外，所有的指令均只需 1 个指令周期



5.2.5 OPTION (功能寄存器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
DBH(R/W)	OPTION	WDTE	REM_en	P3WU	P1SN	P20SN	WDTS2	WDTS1	WDTS0
RESET 状态		0	0	0	0	0	0	0	0

注 x: 未定; q: 随不同操作改变; u 不变。

WDTS2	WDTS1	WDTS0	WDT 的分频比
0	0	0	1:1 (大约 24ms, 3v)
0	0	1	1:2
0	1	0	1:4
0	1	1	1:8
1	0	0	1:16
1	0	1	1:32
1	1	0	1:64
1	1	1	1:128

REM_en: 内置三极管控制位。

=0, 无效

=1, 有效。有效后, 对应 I/O 口功能失效, 只是 NMOS 管开漏输出。

P3WU: PORT3 口唤醒控制位

=1. 端口变化可唤醒。(低电平唤醒)

=0. 不能。

P1SN: PORT1 端口扫描使能位

=1. 当端口处于输入输出模式 (P1M1:P1M0=10) 时, 相应端口会出现扫描信号。

=0. 无论端口处于什么状态, 都不会出现扫描信号。

P20SN: PORT2,PORT0 端口扫描使能位

=1. 当端口处于输入输出模式 (P2M1:P2M0=10、P0M1:P0M0=10) 时, 相应端口会出现扫描信号。

=0. 无论端口处于什么状态, 都不会出现扫描信号。

WDTE: 看门狗使能位

=1, 关闭 WDT

=0, 使能 WDT (默认是打开的)。

5.2.6 STATUS (状态寄存器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
86H(R/W)	STATUS	/TO	/PD	LVDF28	LVDF24		C	DC	Z
复位状态	BOR	1	1	0	0	0	x	x	x
	WDT	0	0	0	0	0	u	u	u

寄存器包含 ALU 的算术状态, RESET 状态。

如果 STATUS 寄存器作为目的操作数, 将要影响 Z、DC 和 C。这 3 位不能写操作。/TO 和/PD



不能写。因此对 STATUS 进行操作可能会和你想要得结果不同。例如，CLRR STATUS 指令将会清高 3 位且 Z=1。结果将会是 STATUS=uu00 0uu1（这里 u=unchange）。

C: 进位/借位（加减法指令）

=1, 有进位/无借位

=0, 无进位/有借位

DC: 半进位/半借位（加减法指令）

=1, 有低 4 位进位/无低 4 位借位

=0, 无低 4 位进位/有低 4 位借位

Z: 零标志位

=1, 逻辑操作结果为 0

=0, 逻辑操作结果不为 0

LVDF24: 电压检查位

=1, 电源电压小于 2.4v

=0, 电源电压大于 2.4v。

LVDF28: 电压检查位

=1, 电源电压小于 2.8v

=0, 电源电压大于 2.8v。

/PD: 掉电标志位

=1, 系统上电或执行 CLRWDT 指令后

=0, 执行 SLEEP 指令后

/TO: 时间溢出标志位

=1, 系统上电或执行 CLRWDT 或执行 SLEEP 指令后

=0, 看门狗溢出后

5.2.7 PORT（端口寄存器）

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
D0h(R/W)	PORT0	-			P0 端口寄存器				
C0h(R/W)	P0M0	-			P0 模式寄存器 0				
E0h(R/W)	P0M1	-			P0 模式寄存器 1				
B0h(R/W)	P0WP	-			P0 唤醒寄存器				
D1h(R/W)	P1	P1 端口寄存器							
C1h(R/W)	P1M0	P1 模式寄存器 0							
E1h(R/W)	P1M1	P1 模式寄存器 1							
B1h(R/W)	P1WP	P1 唤醒寄存器							
D2h(R/W)	P2	P2 端口寄存器							
C2h(R/W)	P2M0	P2 模式寄存器 0							
E2h(R/W)	P2M1	P2 模式寄存器 1							



B2h(R/W)	P2WP	P2 唤醒寄存器							
D3h(R/W)	P3	P3 端口寄存器							
C3h(R/W)	P3M0	P3 模式寄存器 0							
E3h(R/W)	P3M1	P3 模式寄存器 1							
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

PnM1:PnM0	PORTn (n=A,B,C,D,E) 端口功能	
1 0	输出, 强 0 或强 1	
1 1	准输入输出端口。实际为输出口, 只是输出弱 1 强 0。如果此时 PDSN、PCSN 为 1, 则相应端口会输出扫描信号。	
0 1	输入且有上拉电阻。此电阻与准输入输出口输出 1 时一致。电阻约 20K 左右	
0 0	纯输入态。	

P1M1:P1M0	P1[0]端口功能	
0 0	悬空	
1 0 0 1	准输入输出端口。实际为输出口, 输出强 0, 微弱 1, 可以作为输入, 但作为输入时, 内部端口寄存器应置 1。	
1 1	扫描管脚, 但必须 PDSN 有效才行。	

REM_en	P0M1:P0M0	P0[3]端口功能
0	1 0	输出, 强 0 或强 1
0	1 1	准输入输出端口。实际为输出口, 只是输出弱 1 强 0
0	0 1	输入且有上拉电阻。此电阻与准输入输出口输出 1 时一致。电阻约 20K 左右
0	0 0	纯输入态。
1	x x	<p>内置大 NMOS 管开漏, 相当于内置三极管, 控制逻辑于 I/O 一致 当处于此模式时, 对应的 P0M1<3>可以控制是否省电解电容。</p> <p>P0M1<3></p> <p>0 省电解, 输出脉冲边沿交换。</p> <p>1 不省电解, 输出脉冲边沿较陡, 电源需加电容才能工作。</p>



PnWP	功能描述
n=0,1,2 三个寄存器	设置端口变化可唤醒，下降沿有效。

5.2.8 TMR0、TMR1（定时/计数器寄存器）

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0D9h(R/W)	T0C	定时/计数器 0 的 8 位数据							
0D8h(R/W)	T0MD	定时/计数器 0 模式控制位							
0DAh(R/W)	T0RL	定时/计数器 0 自动装载位							
0DDh(R/W)	T1C	定时/计数器 1 的 8 位数据							
0DCh(R/W)	T1MD	定时/计数器 1 模式控制位							
0DEh(R/W)	T1RL	定时/计数器 1 自动装载位							
TnC\TnRL	BOR	x	x	x	x	x	x	x	x
复位状态	WDT	u	u	u	u	u	u	u	u

概述（以下如无特殊说明，都是同时指 TMR0 和 TMR1）

定时/计数器TnC（n=0，1）具有双时钟源，可根据实际需要选择内部时钟或外部时钟作为计时标准。其中，内部时钟来自Fcpu，外部时钟TnCK(n=0,1)（下降沿触发）输入。寄存器TnMD 控制时钟源的选择。当TnC 从0FFH 溢出到00H 时，TnC 在继续计数的同时产生一个溢出信号，TnIF=1，触发TnC 中断请求。

TC0 的主要作用如下：

- ◆ 8 位可编程定时器：根据选定的时钟频率在特定时间产生中断信号；
- ◆ 外部事件计数：对外部事件计数；
- ◆ 蜂鸣器输出；
- ◆ PWM 输出。

5.2.8.1 TnMD 寄存器

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0D8h(R/W)	T0MD	TR	TPS2	TPS1	TPS0	TnCKS	ALOAD	TCOUT	PWMEN
0DCh(R/W)	T1MD								
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

PWMEN: PWM 输出控制。

0 = 禁止 PWM 输出；

1 = 使能 PWM 输出，PWM 输出占空比由 TC0OUT 和 ALOAD0 控制。

TCOUT: TC 超时输出信号控制。仅当 PWMEN = 0 时有效。

0 = 禁止，P13 或者 P27 作为输入/输出口；



1 = 使能, P13 或者 P27 输出 TCOUT 信号。可用作驱动 Buzzer。

ALOAD: 自动装载控制。仅当 PWMEN = 0 时有效。

0 = 禁止 TC 自动装载;

1 = 使能 TC 自动装载。

TCCKS: TC 时钟信号控制位。

0 = 内部时钟 (Fcpu 或 Fosc);

1 = 外部时钟, T0CK 或 T1CK 输入。

TPS[2:0]: TC 分频选择位。

111 = fcpu/256;

110 = fcpu/128;

... ;

001 = fcpu/4;

000 = fcpu/2。

TR: TC 启动控制位。

0 = 禁止 TC 定时器;

1 = 开启 TC 定时器。

5.2.8.2 TnRL 寄存器

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0DBh(R/W)	T0RL	自动装载寄存器							
0DEh(R/W)	T1RL								

T0和T1的自动装载功能由TnMD 的ALOAD位控制。当TC溢出时, TnRL 的值自动装入TC 中。这样, 用户在使用的过程中就不需要在中断中重新赋值。TC为双重缓存器结构。若程序对TnRL 进行了修改, 那么修改后的TnRL值先被暂存在TnRL的第一个缓存器中, 直到当前TC溢出后, 才被真正存入TnRL缓存器中, 从而避免TC中断时间出错以及PWM 和蜂鸣器误动作。

注: 在PWM 模式下, 系统自动开启自动装载功能。位寄存器ALOAD 用于控制溢出范围。

TnRL初始值计算公式如下:

$$\text{TnRL初始值} = N - (\text{TC 中断间隔时间} * \text{输入时钟})$$

上式中, N 为TC 的最大计数范围。TC 的溢出时间有如下六种可能情况:

TnCKS	PWMEN	ALOAD	TCOUT	N	TC0R 计数范围	TC0R 二进制计数范围
0	0	X	X	256	00H~0FFH	00000000b~11111111b
	1	0	0	256	00H~0FFH	00000000b~11111111b
	1	0	1	64	00H~3FH	xx000000b~xx111111b
	1	1	0	32	00H~1FH	xxx00000b~xxx11111b
	1	1	1	16	00H~0FH	xxxX0000b~xxxX1111b
1	-	-	-	256	00H~0FFH	00000000b~11111111b

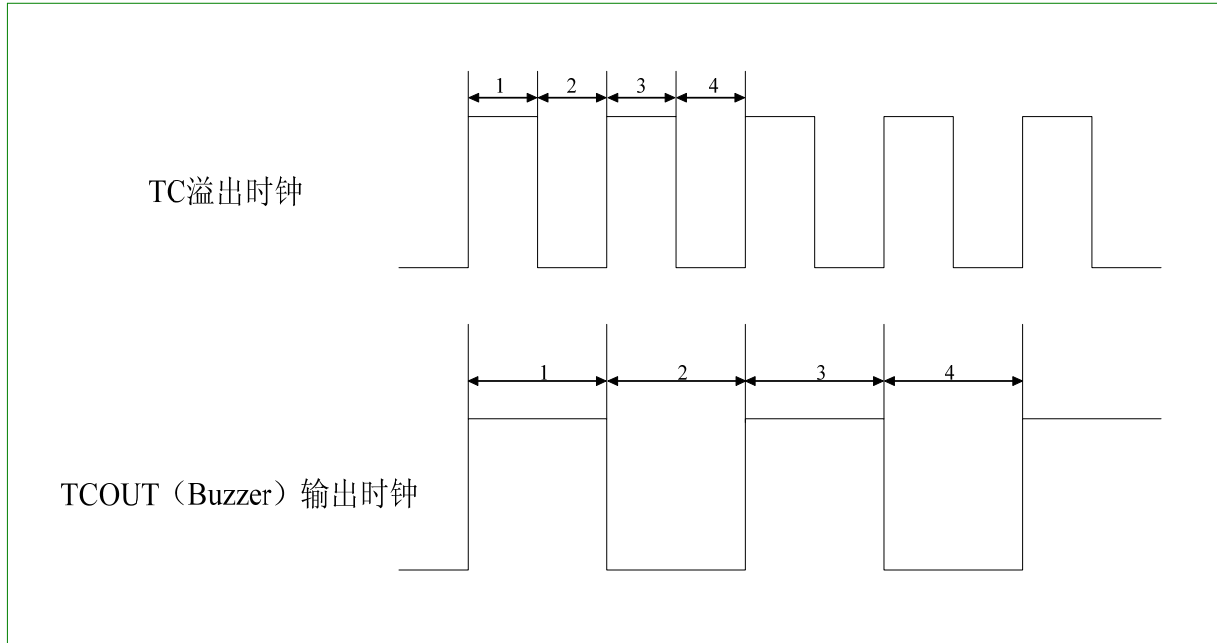
例: TC 中断间隔时间设置为10ms, 时钟源选Fcpu (TnCKS = 0), 无PWM 输出 (PWMEN = 0), 高速时钟为4MHz, Fcpu = Fosc/4, TPS = 010 (Fcpu/64)。

$$\begin{aligned} \text{TnRL 初始值} &= N - (\text{TC 中断间隔时间} * \text{输入时钟源}) \\ &= 256 - (10\text{ms} * 4\text{MHz} / 4 / 64) \\ &= 256 - (10^{-2} * 4 * 10^6 / 4 / 64) = 100 = 64\text{H} \end{aligned}$$



5.2.8.3 TC 时钟频率输出（蜂鸣器输出）

对TC 时钟频率进行适当设置可得到特定频率的蜂鸣器输出(TCOUT)，并通过PWM引脚 输出。单片机内部设置TC 的溢出频率经过2 分频后作为TCOUT 的频率，即TC 每溢出2 次TCOUT 输出一个完整的脉冲，此时，I/O 功能自动被禁止。TCOUT 输出波形如下：



若时钟选择4MHz，系统时钟源采用外部时钟 $F_{osc}/4$ ，程序中设置 $TPS = 110$ ， $TC = TCOR = 131$ ，则TC 的溢出频率为2KHz，TCOUT 的输出频率为1KHz。

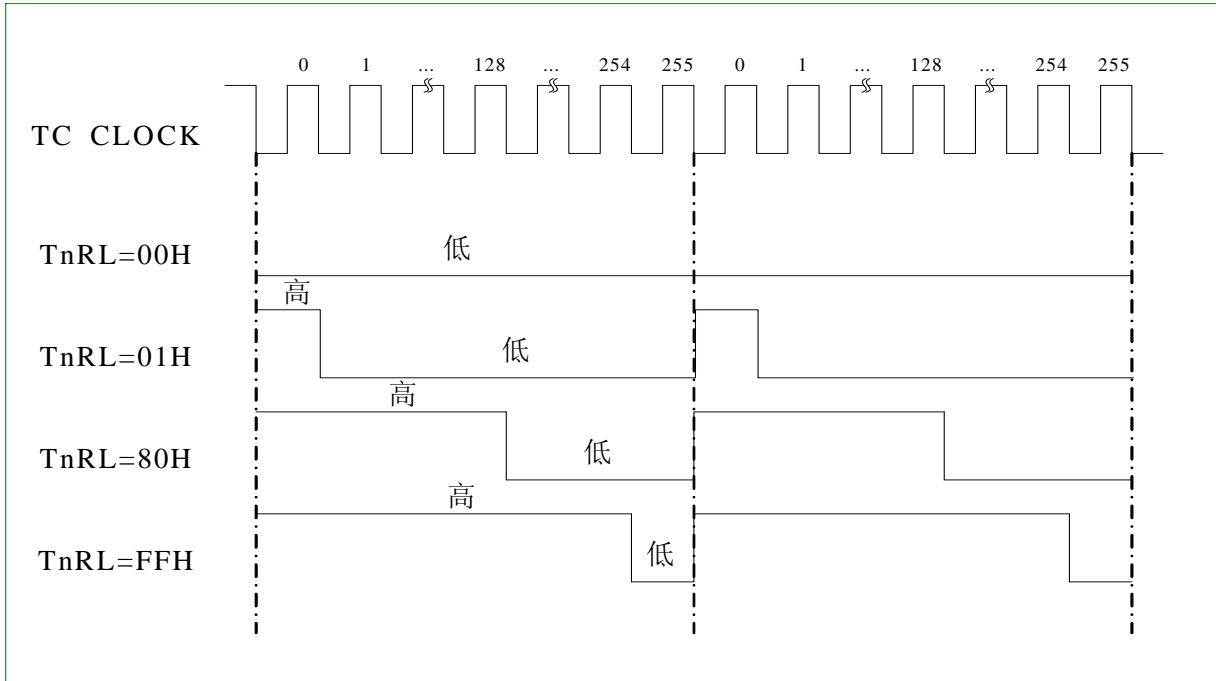
5.2.8.4 PWM

PWM信号输出到PWMOUT,TCOUT和ALOAD 标志位控制PWM 输出的阶数256、64、32和16)。8 位计数器TC 计数过程中不断与TNRL 相比较，当TC计数到两者相等时，PWM输出低电平，当TC再次从零开始计数时，PWM 被强制输出高电平。PWM输出占空比 = $TNRL / \text{计数量程}$ （计数量程 = 256、64、32 或16）。参考寄存器保持输入00H 可使PWM 的输出长时间维持在低电平，通过修改TNRL 可改变PWM 输出占空比。

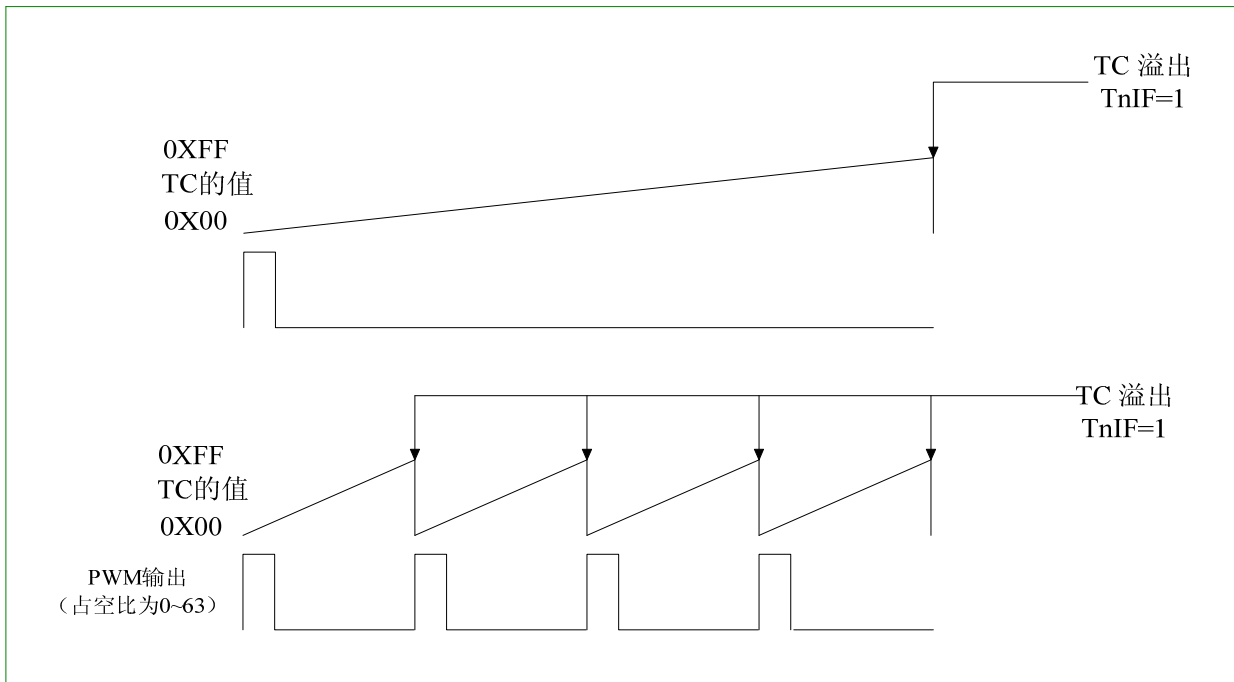
注：TMR0和TMR1 为双重缓存器结构，调整TNRL 的值可以改变PWM 的输出占空比。用户可随时改变TNRL 的值，但是只有在TC(TIMER COUNT) 溢出后，这一修改值才真正被写入TnRL 中。

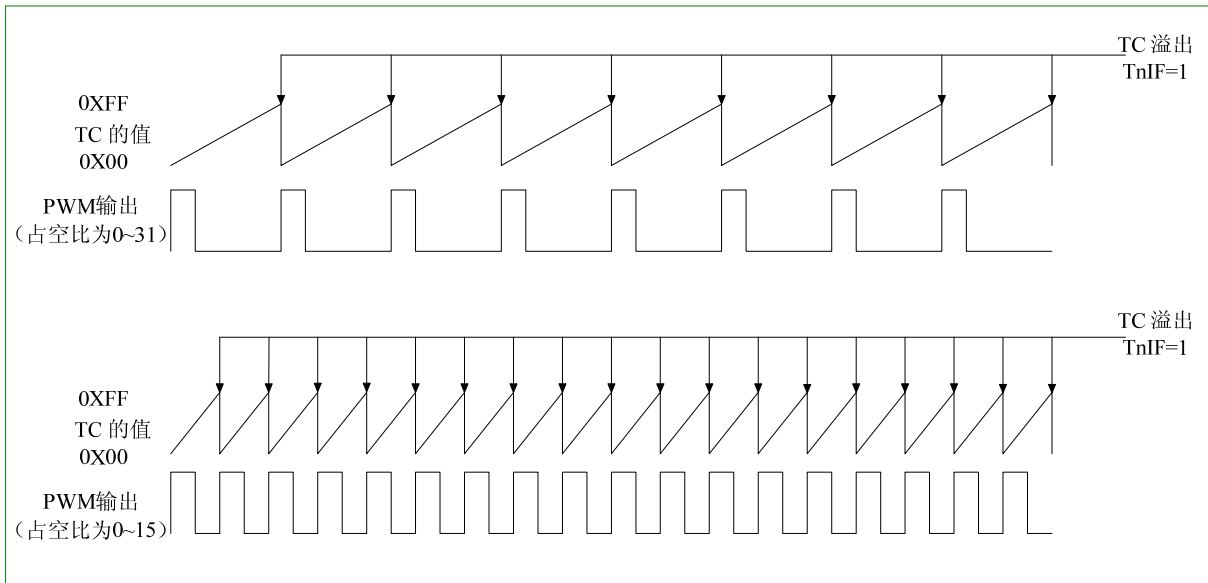
ALOAD	TCOUT	PWM 占空比范围	TC 有效值	TnRL 有效值	MAX. PWM 输出频率 (Fcpu=4MHZ)	注释
0	0	0/256~255/256	00H~FFH	00H~FFH	7.8125K	每计数256次溢出
0	1	0/64~63/64	00H~3FH	00H~3FH	31.25K	每计数64次溢出
1	0	0/32~31/32	00H~1FH	00H~1FH	62.5K	每计数32次溢出
1	1	0/16~15/16	00H~0FH	00H~0FH	125K	每计数16次溢出

PWM 输出占空比随 TnRL 的变化而变化：0/256~255/256。

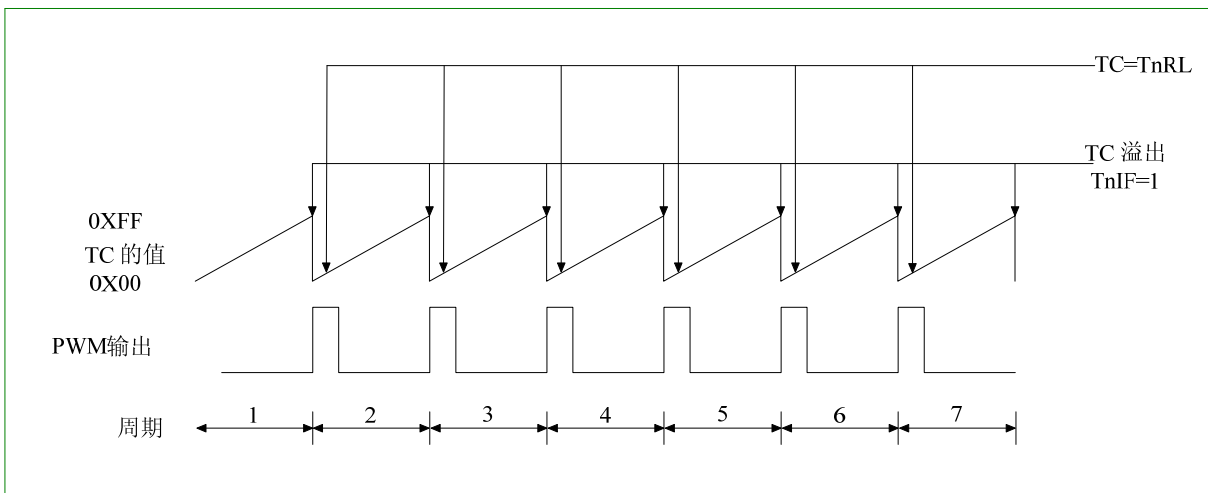


在 PWM 模式下，TnIF 的频率与 PWM 的占空比有关，具体情况如下图所示：

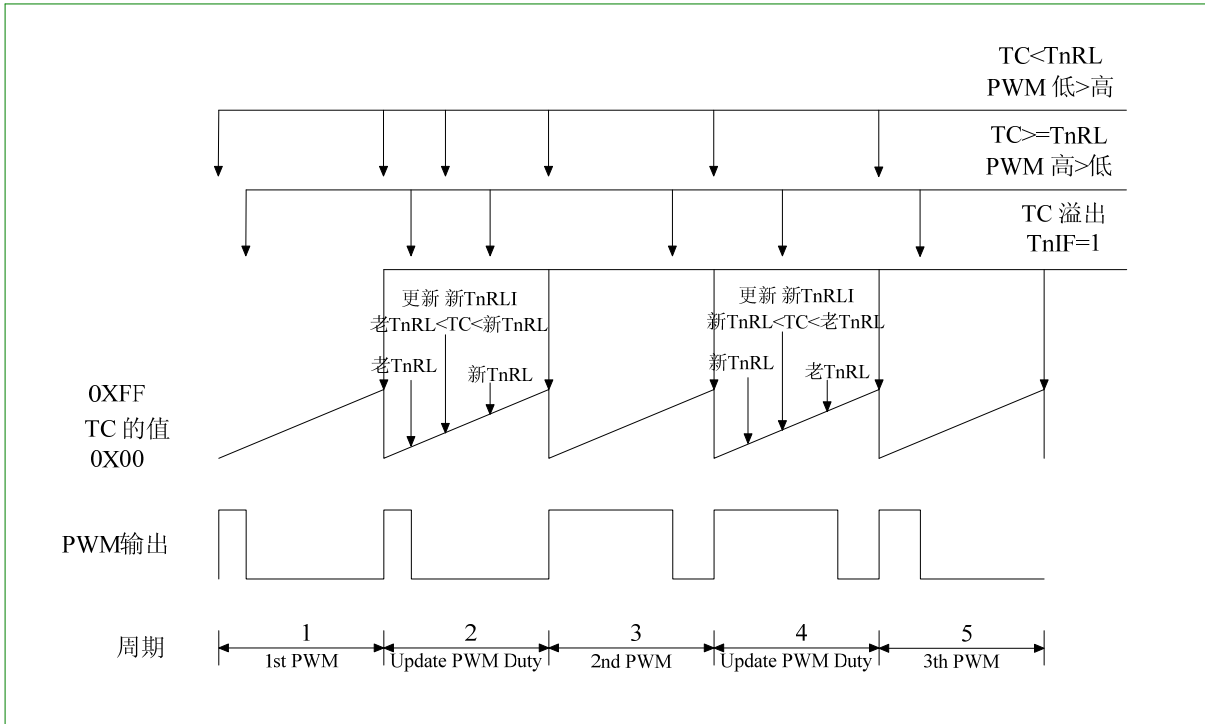




在PWM 模式下，系统随时比较TC 和TnRL 的异同。若 $TC < TnRL$ ，PWM 输出高电平，反之则输出低电平。当TC 发生改变的时候，PWM 将在下一周期改变输出占空比。如果TnRL 保持恒定，那么PWM 输出波形也保持稳定。



上图所示是 TnRL 恒定时的波形。每当TC 溢出时，PWM 都输出高电平， $TC \geq TnRL$ 时，PWM 即输出低电平。下面所示是TnRL 发生变化时对应的波形图：



在period 2 和period 4 中，显示新的占空比（TnRL），但PWM 在period 2 和period 4 的占空比要在下一个period才会改变。这样，可以避免PWM 不随设定改变或在同一个周期内改变两次，从而避免系统发生不可预知的误动作。



5.2.9 INTCON (中断控制器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0C9h(R/W)	INTCON			TKTMOIE	SIMIE	T1IE	TOIE	INTE	RPIE
0C8h(R/W)	INTFLAG			TKTMOIF	SIMIF	T1IF	TOIF	INTF	RPIF
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

Bit7,6: 未用

TKTMOIF: 触摸模块中 TKMR0 的溢出标志位

=0, 无效。

=1, 有效。

TKTMOIE: 触摸模块中 TKMR0 的中断使能位

=0, 无效。

=1, 有效。

SIMIE: SPI 或 I2C 协议传送结束 中断使能

=0, SPI 或 I2C 传送结束 中断无效。

=1, SPI 或 I2C 传送结束 中断有效。

SIMIF: SPI 或 I2C 协议传送结束 标志

=0, SPI 或 I2C 正在传送。

=1, SPI 或 I2C 传送结束 , 可唤醒。

T1IE:TIMER1 溢出中断使能

=0, Timer1 溢出中断无效。

=1, Timer1 溢出中断有效。

T1IF: TIMER1 溢出

=0, TIMER1 未溢出。

=1, TIMER1 溢出。

TOIE:TIMER0 溢出中断使能

=0, timer0 溢出中断无效。

=1, timer0 溢出中断有效。

TOIF: TIMER0 溢出

=0, TIMER0 未溢出。

=1, TIMER0 溢出。

INTE: INT 脚输入沿变化中断使能位

=0, INT 输入变化中断无效。

=1, INT 输入变化中断有效。

INTF: INT 沿边化

=0, 无效。

=1, 有效, 可唤醒。

RPIE: P0、P1、P2、P3 输入变化 (由高到低) 中断使能

=0, 无效。

=1, 有效。

RPIF: P0、P1、P2、P3 输入变化 (由高到低)



=0, 未变化。

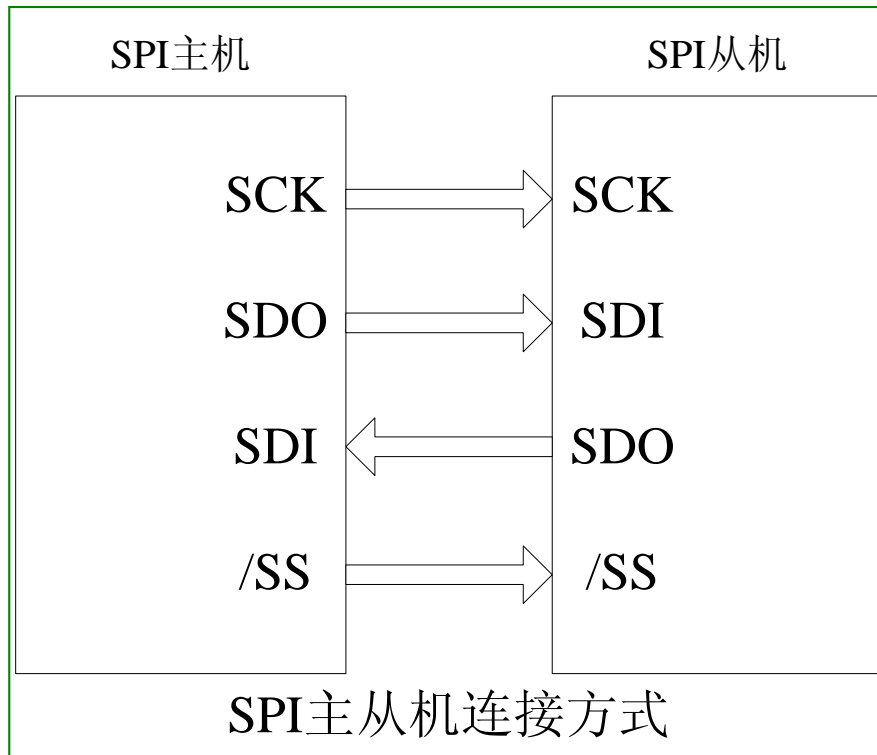
=1, 有变化, 可唤醒。

TCM0、TCM1、TCM2 和 TKTM0 用以触摸模块的计数。本触摸原理使用 RC 振荡器来识别 C 的变化。当 C 变化时, RC 振荡器的频率也变化, 在固定的时间内, 通过 TCM0、TCM1 和 TCM2 计数模块对 RC 进行计数, 通过得到的数据就可以计算出频率, 进而算出 C 的变化量。

5.2.10 SIM (SIM 控制器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0A6h(R/W)	SIMBUF	SIM 数据寄存器, 存储发送或接收的数据							
0B5h(R/W)	SIMC0				SIMM2	SIMM1	SIMM0	SIMEN	
0A5h(R/W)	SIMC (SPI)			CKEG	CKP	MLS	SSEN	WCOL	
0A5h(R/W)	SIMC (I2C)		HBB		HTX	TXAK		WCOL	
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

SPI 接口是一个全双工串行数据传输器。SPI 接口的四线为: SDI、SDO、SCK 和/SS。SDI 和 SDO 是数据的输入和输出线。SCK 是串行时钟线, /SS 是从机的选择线。SPI 的接口引脚与普通 I/O 口。通过设定 SIMC0、SIMC1 寄存器的对应位, 来使能 SPI 接口。连接到 SPI 接口的单片机以主/从模式进行通信, 且主机完成所有的数据传输初始化, 并控制时钟信号。由于单片机只有一个/SS 引脚, 所以只能拥有一个从机设备。可通过软件控制/SS 引脚使能, 设置 SSEN 位为“1”使能/SS 引脚功能, 设置 SSEN 位为“0”/SS 引脚将作为普通输入/输出引脚。





SPI功能具有以下特点:

- ◆ 全双工同步数据传输
- ◆ 主从模式
- ◆ 最低有效位先传或最高有效位先传的数据传输模式
- ◆ 传输完成标志位
- ◆ 时钟源上升沿或下降沿有效

SIMBUF:

SIM的缓冲器,用以存放接收和发送的数据。在单片机尚未将数据写入到SPI总线中时,要传输的数据应先存在SIMBUF寄存器中。SPI总线接收到数据之后,单片机就可以从SIMBUF寄存器中读取。所有通过SPI总线传输或接收的数据都必须通过SIMBUF寄存器实现。

SIMC0: SIM控制器0

SIMM2~SIPM0: SIM工作模式控制位

- 000: SPI主机模式; SPI时钟为 $F_{sys}/4$
- 001: SPI主机模式; SPI时钟为 $F_{sys}/16$
- 010: SPI主机模式; SPI时钟为 $F_{sys}/64$
- 011: SPI主机模式; SPI时钟为 $TMR0/2$
- 100: 未使用模式
- 101: SPI从机模式
- 110: I2C从机模式
- 111: 未使用模式

这几位用于设置SPI功能的工作模式,用于选择SPI的主从模式和SPI的主机时钟频率。SPI时钟源可来自于系统时钟也可以选择来自TMR0。若选择的是作为SPI从机,则其时钟源从外部主机而得。

Bit 0 SIMEN: SIM控制位。

- 0: 无效
- 1: 有效

此位为SIM接口的开/关控制位。选择SPI模式时,此位为“0”时,SPI接口除能,SDI、SDO、SCK和SS脚用于输入/输出功能,SPI工作电流减小到最小值。此位为“1”时,SPI接口使能。配置选项中首先将SPI接口使能才能使此位有效。若SPI经由SIMM2~SIMM0位设置为工作在SPI接口,当SIMEN位由低到高转变时,SPI控制寄存器中的设置不会发生变化,其首先应在应用程序中初始化。

选择I2C模式时,用以控制I2C模块的使能。其相关寄存器控制I2C模式。

5.2.10.1 SPI模式

SIMC1: SIM控制器1

Bit 7~6 未定义位 用户可通过软件程序对这两位进行读写。

CKEG: SPI的SCK有效时钟边沿类型位

CKP=0

- 0: SCK为高电平且在SCK上升沿抓取数据
- 1: SCK为高电平且在SCK下降沿抓取数据

CKP=1

- 0: SCK为低电平且在SCK下降沿抓取数据
- 1: SCK为低电平且在SCK上升沿抓取数据



CKEG和CKP位用于设置SPI总线上时钟信号输入和输出方式。在执行数据传输前，这两位必须被设置，否则将产生错误的时钟边沿信号。CKP位决定时钟线的基本状态，若时钟无效且此位为高，则SCK为低电平，若时钟无效且此位为低，则SCK为高电平。CKEG位决定有效时钟边沿类型，取决于CKP的状态。

CKP: 时钟线的基础状态位

0: 当时钟无效时，SCK引脚为高电平

1: 当时钟无效时，SCK引脚为低电平

此位决定了时钟线的基础状态，当时钟无效时，若此位为高，SCK为低电平，若此位为低，SCK为高电平。

MLS: SPI数据移位命令位

0: LSB

1: MSB

数据移位选择位，用于选择数据传输时高位优先传输还是低位优先传输。此位设置为高时高位优先传输，为低时低位优先传输。

SSEN: SPI /SS引脚控制位

0: 除能

1: 使能

SSEN位用于/SS引脚的使能/除能控制。此位为低时，/SS除能并处于浮空状态。此位为高时，/SS作为选择脚。

注:在主机模式下，SSEN=0，则/SS脚做I/O口使用。SSEN=1，则/SS脚在SPIEN=1时，变低。

在从机模式下，SSEN=0，则SPI模块不受控与/SS脚，不论/SS脚为高还是为低都可以接收数据。此时/SS端可做I/O口使用。在SSEN=1时，SPI模块必须在/SS脚为低时才会接收数据。

WCOL: SPI写冲突标志位 WCOL标志位用于监测数据冲突的发生

0: 无冲突

1: 数据在传输时又有数据被写入SIMBUF寄存器。此位可被软件清零。

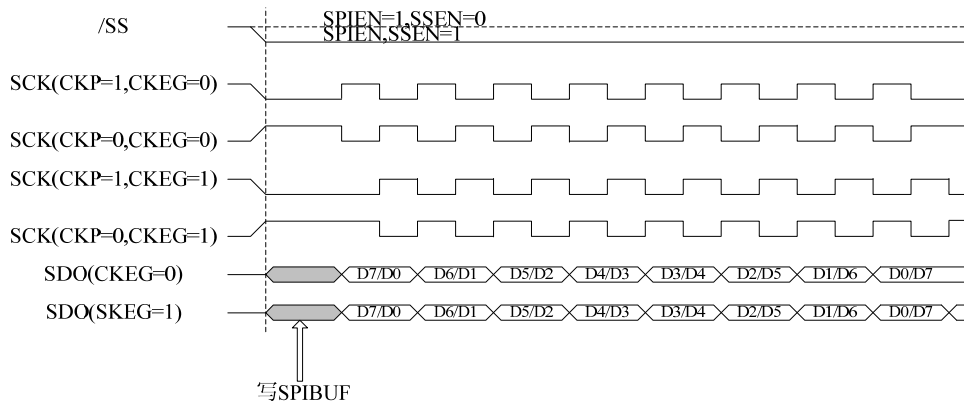
SIMIF: SPI 或 I2C 协议传送结束 标志（其地址在 INTCON 和 INTFLAG 中）

0: SPI 或 I2C 正在传送

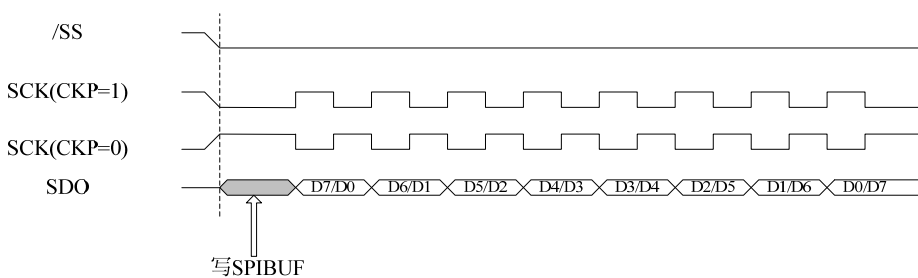
1: SPI 或 I2C 传送结束，可唤醒

SPI通信

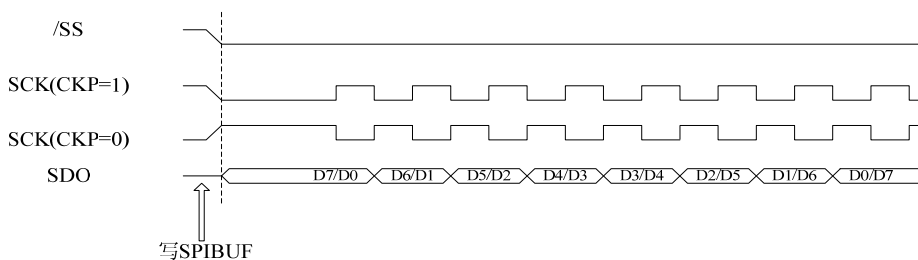
将SPIEN设置为高，使能SPI功能之后，单片机处于主机模式，当数据写入到寄存器SIMBUF的同时传输/接收开始进行。数据传输完成时，TRF位将自动被置位但清除只能通过应用程序完成。单片机处于从机模式时，收到主机发来的信号之后，会传输SIMBUF中的数据，而且在SDI引脚上的数据也会被移位到SIMBUF寄存器中。主机应在输出时钟信号之前先输出一个/SS信号以使能从机，从机的数据传输功能也应在与/SS信号相关的适当时候准备就绪，这由CKP和CKEG位决定。所附时序图表明了CKP和CKEG位各种设置情况下从机数据与/SS信号的关系。



SPI主机模式时序



SPI从机模式时序 (CKEG=0)



SPI从机模式时序 (CKEG=1)

注：使用SPI前，应先定义CKEG,CKP，然后才使能SPIEN信号。

5.2.10.2 I2C 模式

I²C 串行接口是一个双线的接口，有一条串行数据线 SDA 和一条串行时钟线 SCL。由于可能有多个设备在同一条总线上相互连接，所以这些设备的输出都是开漏型输出。因此应在这些输出口上都应加上拉电阻。应注意的是：I²C 总线上的每个设备都没有选择线，但分别与唯一的地址一一对应，用于 I²C 通信。

如果有两个设备通过双向的I²C总线进行通信，那么就存在一个主机和一个从机。主机和从机都可以用于传输和接收数据，但只有主机才可以控制总线动作。那些处于从机模式的设备，要在I²C 总线上传输数据只有两种方式，一是从机发送模式，二是从机接收模式。

SIMC1寄存器



0A5h(R/W)	SIMC (I2C)		HBB		HTX	TXAK		WCOL	
-----------	------------	--	-----	--	-----	------	--	------	--

Bit 6 **HBB**: I²C总线忙标志位，只读

0: I²C总线闲

1: I²C总线忙

当检测到START信号时I²C忙，此位变为高电平。当检测到STOP信号时I²C总线停止，该位变为低电平。

Bit 4 **HTX**: 从机处于发送或接收模式标志位。SCK由主机产生。起始、停止都是由主机控制。

0: 从机处于接收模式

1: 从机处于发送模式

Bit 3 **TXAK**: I²C总线发送确认标志位

0: 从机发送确认标志

1: 从机没有发送确认标志

单片机接收8位数据之后会将该位在第九个时钟传到总线上。如果单片机想要接收更多的数据，则应在接收数据之前将此位设置为“0”。

Bit 1 **WCOL**: I2C写冲突标志位 WCOL标志位用于监测数据冲突的发生

0: 无冲突

1: 数据在传输时又有数据被写入SIMBUF寄存器。此位可被软件清零。

其他未用bit读时为0，如果SPI寄存器使用，而在I2C未用时，可做为普通寄存器使用。

注：

- ◆ I2C的传输完成标志位和SPI用的是同一个。
- ◆ I2C的发送数据时，SIMBUF只有重新写入数据才会更新。即传送时不会改变SIMBUF的值。
- ◆ 在SPI模式时，数据传送结束后，SIMBUF总是反映的是SDI接收到的数据。
- ◆ 数据传送结束后，如果不去除SIM_TRF信号，则下次传送接收时，SIMBUF不会更新。正确的做法是：当一个数据结束后，把数据读入缓存，然后清除SIM_TRF，等待下一次接收！
- ◆ 按照通用I2C操作规则，当主机读取本I2C模块数值时，主机应该在第九个时钟进行响应。即如果第九个时钟进行响应，设置为0，则从机默认主机会继续读取数据，从机不会释放数据线；反之，如果第九个时钟设置为1，则从机认为主机接收数据完毕，从机就会释放数据线，主机就可以发送停止信号。如果此时主机不发送停止信号，从机也不会继续传送数据。
- ◆ I2C模式时，SCL和SDA都有弱上拉，SPI中的其他端口用作I/O口；SPI模式从机使用时，SCK有弱上拉，SDI不论何种模式一直有弱上拉。



5.2.11 触摸控制模块

触控脚与P1、P2、P3的I/O引脚共用。通过寄存器的位来选择此功能。当选择触摸功能时，I/O口功能消失，如果是SPI功能，则SPI功能也消失，即触摸功能优先，其次是SPI或I2C。按键被分成三个部分，即M0~M2四个模块。每个模块具有自己的控制逻辑电路和设置寄存器。寄存器的名称和它对应的模块编号相关联。

	D7	D6	D5	D4	D3	D2	D1	D0
TKOP0	TKEN	三个采集模块的使能			HTK_2	选择通道，8选1，同时只有一路有效，作用于三个采集模块		
	TK总控制	M2使能	M1使能	M0使能				
TKOP1	D7	D6	D5	D4				
	TKCTM1	TKCTM0	改动振荡频率，实际上改动电流，值越小，电流越大，频率越快。用于寄生电容不同，频率偏的情况					
TKOP2	D7	D6	D5	D4	D3	D2	D1	D0
	TKTM0TR	TKMCT2 TR	TKMCT1 TR	TKMCT0 TR	TKTM0C S	TKMCT2C LR	TKMCT1 CLR	TKMCT0C LR
	TM0启动	TC2启动	TC1启动	TC0启动	时钟选择	触摸计数器清0位，0清0，1计数。		
当 MnCT 当做计数器进行计数时，只有 TM0tr 有效时，才能够开始计数，这样能保证三者同时计数，一致性可以得到保证。当定时器时无此限制。计数时只能对触摸模块进行计数。								
	D7	D6	D5	D4	D3	D2	D1	D0
TKTM0L	公共定时/计数器的低8位							
TKTM0H	公共定时/计数器的高8位							
TKM0En	I/O口及触摸脚选择。0：普通端口；1：触摸脚，I/O无效							
TKM1En	I/O口及触摸脚选择。0：普通端口；1：触摸脚，I/O无效							
TKM2En	I/O口及触摸脚选择。0：普通端口；1：触摸脚，I/O无效							
TKMC/TL	触摸模块的定时器计数器低8位							
TKMC/TH	触摸模块的定时器计数器高8位，3个计数器共用地址							
	“-”的标注未作特殊用途，可用作通用寄存器，正常读写。							

5.2.11.1 TKOP0 (Touch 功能 0 寄存器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
B4h(R/W)	TKOP0	TKEN	TKM2EN	TKM1EN	TKM0EN	HTK_2	CHS2	CHS1	CHS0
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

TKEN: 触摸模块的总控制位。

=0, 所有触摸模块无效。

=1, 控制模块有效。需要方法 TKM2EN、TKM1EN、TKM0EN 有效。



CHS2~CHS0: 触摸管脚选择位

CHS2	CHS1	CHS0	功能
0	0	0	TK0, TK8, TK16 有效, 如果触摸选择, 则 I/O 无效
0	0	1	TK1, TK9, TK17 有效, 如果触摸选择, 则 I/O 无效
0	1	0	TK2, TK10, TK18 有效, 如果触摸选择, 则 I/O 无效
0	1	1	TK3, TK11, TK19 有效, 如果触摸选择, 则 I/O 无效
1	0	0	TK4, TK12, TK20 有效, 如果触摸选择, 则 I/O 无效
1	0	1	TK5, TK13, TK21 有效, 如果触摸选择, 则 I/O 无效
1	1	0	TK6, TK14, TK22 有效, 如果触摸选择, 则 I/O 无效
1	1	1	TK7, TK15, TK23 有效, 如果触摸选择, 则 I/O 无效

端口的无 I/O 特性是指端口输出无效, 读时为 1。

HTK_2: 三个触摸控制器的频率分频控制。

=0, 充电电流不变;

=1, 充电电流减小 1 倍。

TKM2EN: 触摸模块 2 的控制位。

=0, 触摸模块 2 无效。

=1, 控制模块 2 有效。需要 TKEN 同时有效。

TKM1EN: 触摸模块 1 的控制位。

=0, 触摸模块 1 无效。

=1, 控制模块 1 有效。需要 TKEN 同时有效。

TKM0EN: 触摸模块 0 的控制位。

=0, 触摸模块 0 无效。

=1, 控制模块 0 有效。需要 TKEN 同时有效。

注: 只有触摸控制位有效后, 其他寄存器控制位才有意义!

5.2.11.2 TKOP1 (Touch 功能 1 寄存器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0A4h(R/W)	TKOP1	TKCTM1	TKCTM0	改动振荡频率, 实际上改动电流, 值越小, 电流越大, 频率越快。用于寄生电容不同, 频率偏的情况					
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

TKCTM1:TKCTM0:

用以选择触摸计数器读取数据通道。当采集完成后, 要把触摸计数器的值读出处理。本芯片有三路计数器, 三路计数器是占用同一个地址的, 读时会根据 TKCTM1~TKCTM0 的值来确认是那个通道。

TKCTM1	TKCTM0	通道
0	0	TKM0CT (TK0~TK7)
0	1	TKM1CT (TK8~TK15)
1	x	TKM2CT (TK16~TK23)

D5~D0:



改动振荡频率，实际上改动电流，值越小，电流越大，频率越快。用于寄生电容不同，频率偏的情况。用以调整触摸端口初始寄生电容不同的情况。

5.2.11.3 TKOP2 (Touch 功能 2 寄存器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0B6h(R/W)	TKOP2	TKTM0TR	TKMC T2TR	TKMC T1TR	TKMC T0TR	TKTM0CS	TKMC T2CLR	TKMC T1CLR	TKMC T0CLR
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

此寄存器控制触摸模块的核心单元定时器、计数器的开启。触摸模块有一个公共定时器和三个定时/计数器。

TKTM0TR: 公共定时器 0 启动位。

=0, 停止。

=1, 启动。

TKMCT2TR: 触摸模块 2 的定时/计数器的启动位。

=0, 停止。

=1, 启动。

TKMCT1TR: 触摸模块 1 的定时/计数器的启动位。

=0, 停止。

=1, 启动。

TKMCT0TR: 触摸模块 0 的定时/计数器的启动位。

=0, 停止。

=1, 启动。

注：当 TKMCT2-TKMCT0 进行计数时，只有 TKTM0tr 有效时，才能够开始计数，这样能保证三者同时计数，一致性可以得到保证。

TKTM0CS: 公共定时器 0 信号选择位。

=0, Fcpu。一条指令周期的时间。

=1, PD2 口的输入波形计数。

TKMCT2CLR: 触摸模块 2 的定时/计数器的清 0 位。

=0, 清 0。

=1, 允许触摸模块 2 的 RC 的振荡频率信号输入。

TKMCT1CLR: 触摸模块 1 的定时/计数器的清 0 位。

=0, 清 0。

=1, 允许触摸模块 1 的 RC 的振荡频率信号输入。

TKMCT0CLR: 触摸模块 0 的定时/计数器的清 0 位。

=0, 清 0。

=1, 允许触摸模块 0 的 RC 的振荡频率信号输入。

注：一次计数完成后，下次重新计数时，应先把 3 个触摸计数器清 0，然后才重置定时器；如下所示：

//设置定时时间

BCLR TKOP2.0 ;计数器 1 清 0

BCLR TKOP2.1 ;计数器 2 清 0

BCLR TKOP2.2 ;计数器 3 清 0



BSET TKOP2.0

BSET TKOP2.1

BSET TKOP2.2

//通道切换

MOV A,#0F1H

MOV TKTM0H,A

MOV A,#5FH

MOV TKTM0L,A ;定时 1ms

BCLR INTFLAG.5 ;清除定时器标志

BSET TKOP2.7 ;启动计数

5.2.11.4 TKTM0

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0B8h(R/W)	TKTM0L	TKTM0的低8位							
0A8h(R/W)	TKTM0H	TKTM0的高8位							
复位状态	BOR	x	x	x	x	x	x	x	x
	WDT	u	u	u	u	u	u	u	u

5.2.11.5 TKM0EN

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0A7h(R/W)	TKM0KEN	I/O 口及触摸脚选择。0: 普通端口; 1: 触摸脚, I/O 无效							
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

5.2.11.6 TKM1EN

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
09Ch(R/W)	TKM1KEN	I/O 口及触摸脚选择。0: 普通端口; 1: 触摸脚, I/O 无效							
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

5.2.11.6 TKM2EN

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
09Dh(R/W)	TKM2KEN	I/O 口及触摸脚选择。0: 普通端口; 1: 触摸脚, I/O 无效							
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0



5.2.11.7 TKMCT

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0AAh(R/W)	TKMCTL	TKMCTL 的低8位							
0A9h(R/W)	TKMCTH	TKMCTH 的高8位							
复位状态	BOR	x	x	x	x	x	x	x	x
	WDT	u	u	u	u	u	u	u	u

触控按键操作:

手指接近或接触到触控面板时, 面板的电容量会增大, 电容量的变化会轻微改变内部感应振荡器的频率, 通过测量频率的变化可以感知触控动作。参考时钟通过内部可编程分频器能够产生一个固定的时间周期。在这个时间周期内, 通过对感应振荡器产生的时钟周期计数, 可确定触控按键的动作。

每个触控按键模块包含8个与I/O引脚共用的触控按键。通过寄存器可设置相应引脚功能。每个触控按键模块共用中断向量和中断标志。

在参考时钟固定的时间间隔内, 感应振荡器产生的时钟周期数是可以测量的。这个周期数可以用于判断触控动作是否发生。在最后一个时间间隔后, 会产生一个触控按键中断信号。

触控按键中断:

每个触控按键模块包含8个触控按键, 只有一个是16-bit C/F计数器中断, TKTMn计数器不会产生溢出信号, 也没有中断信号。需要注意的是, 包含在多功能中断中的16-bit C/F计数器中断标志位不会自动复位, 必须通过应用程序将其复位。

触控模块的操作顺序:

- 1、启动三个模块的使能;
- 2、启动总控制, 使触摸模块工作起来;
- 3、设置所用的端口;
- 4、选择通道;
- 5、调整初始的频率, 以实际情况而定, 设置到 1M 左右;
- 6、三个 TKMCTn 计数器清 0, 并启动;
- 7、设置 TKTM0 定时器, 并启动; 此时 TKMCTn 才能计数;
- 8、TKTM0 溢出后, 通过 TKCTM1~0 分别读取三个 TKMnCT 的值, 并另存; 重新设置 TKTM0, TKMCTn 计数器, 并清除响应标志位;
- 9、重新选择通道;
- 10、循环执行 8、9 步, 把所有通道的值都读完;
- 11、分析所得数据, 进行相关手指动作。



5.2.12 OPTION2

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0Cdh(R/W)	OPTION2				INTedge1	INTedge0			
复位状态	BOR	0	0	0	0	0	0	0	0
	WDT	0	0	0	0	0	0	0	0

INTedge[1:0]: 中断触发边沿选择位

00 =保留

01 = 上升沿

10 = 下降沿

11 = 上升/下降双向边沿

5.2.13 RBANK

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
87H(R/W)	RBANK								RBANKS0
复位状态	BOR	-	-	-	-	-	-	-	0
	WDT	-	-	-	-	-	-	-	0

RBANKS0: 数据存储器选择。

=0, bank0

=1, bank1

5.2.14 STKP(堆栈控制器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0DFh(R/W)	STKP	GIE					STKPB0	STKPB0	STKPB0
复位状态	BOR	0	0	0	0	0	1	1	1
	WDT	0	0	0	0	0	1	1	1

GIE: 总中断使能

=0, 所有中断无效。对于在中断后从 SLEEP 模式唤醒, 则程序会从 SLEEP 后的指令执行。

=1, 所有打开的中断有效。从中断后唤醒, 则程序会从 008H 开始执行。

堆栈指针STKP 是一个3 位寄存器, 存放被访问的堆栈单元地址, 13 位数据存储器STKnH 和 STKnL 用于暂存堆栈数据。以上寄存器都位于bank 0。

使用入栈指令PUSH 和出栈指令POP 可对堆栈缓存器进行操作。堆栈操作遵循后进先出(LIFO)的原则, 入栈时堆栈指针STKP 的值减1, 出栈时STKP 的值加1, 这样, STKP 总是指向堆栈缓存器顶层单元。

系统进入中断或执行CALL指令之前, 程序计数器PC的值被存入堆栈缓存器中进行入栈保护。



5.2.15 堆栈操作

执行程序调用指令CALL 和响应中断服务时，堆栈指针STKP 的值减1，指针指向下一个堆栈缓存器。同时，对程序计数器PC 的内容进行入栈保存。

堆栈层数	STKP 寄存器			堆栈缓存器		注释
	STKPB2	STKPB1	STKPB0	高字节	低字节	
0	1	1	1	Free	Free	——
1	1	1	0	STK0H	STK0L	——
2	1	0	1	STK1H	STK1L	——
3	1	0	0	STK2H	STK2L	——
4	0	1	1	STK3H	STK3L	——
5	0	1	0	STK4H	STK4L	——
6	0	0	1	STK5H	STK5L	——
7	0	0	0	STK6H	STK6L	——
8	1	1	1	STK7H	STK7L	——
>8	1	1	0	——	——	堆栈溢出，出错

对应每个入栈操作，都有一个出栈操作来恢复程序计数器PC的值。RETI指令用于中断服务程序中，RET用于子程序调用。出栈时，STKP加1并指向下一个空闲堆栈缓存器。堆栈恢复操作如下表所示：

堆栈层数	STKP 寄存器			堆栈缓存器		注释
	STKPB2	STKPB1	STKPB0	高字节	低字节	
8	1	1	1	STK7H	STK7L	——
7	0	0	0	STK6H	STK6L	——
6	0	0	1	STK5H	STK5L	——
5	0	1	0	STK4H	STK4L	——
4	0	1	1	STK3H	STK3L	——
3	1	0	0	STK2H	STK2L	——
2	1	0	1	STK1H	STK1L	——
1	1	1	0	STK0H	STK0L	——
0	1	1	1	Free	Free	——



5.2.16 OSCM(堆栈控制器)

地址	名称	D7	D6	D5	D4	D3	D2	D1	D0
0CAh(R/W)	OSCM				CPUM1	CPUM0			
复位状态	BOR	-	-	-	0	0	-	-	-
	WDT	-	-	-	0	0	-	-	-

Bit[4:3] CPUM[1:0]: CPU 工作模式控制位。

00 = 普通模式;

01 = 睡眠模式;

1X= 系统保留。



5.3 看门狗定时器

看门狗定时器是由内部慢速 RC 振荡器独立提供的，不需要外部时钟源。在 SLEEP 模式中，如果看门狗使能打开，则看门狗定时器继续计数。

如果 WDTE=1，则 WDT 不工作。

WDT 的基本溢出时间有 2 个管脚扫描周期。一般一个扫描周期为 12ms，因此 WDT 的周期是 24ms（VDD=3V 时）。扫描周期随电压的升高而减小。WDT 溢出后，系统复位。复位完成后无需等待时间，系统就能重新正常工作。

写 WDTR 为 5Ah 可以清除 WDT，防止看门狗溢出，导致系统复位。Sleep 指令也可复位 WDT，系统从睡眠中唤醒后也可复位 WDT。

WDT 也配置了预分频器，可以 1~128 倍可选，最大可定时 3.1s。预分频器通过 PS2~PS0 配置。WDT 溢出后，预分频器会复位。同时，能够是 WDT 复位的信号也可是预分频器复位。

WDT 和管脚扫描的计数器是共用的。在非 sleep 状态，如果 WDT 未使能，则管脚不会出现扫描信号。Sleep 状态下，在 WDT 使能或者管脚需要扫描时，扫描信号源才会打开。

5.4 省电模式

省电模式有 SLEEP 指令进入。

当 SLEEP 指令执行后，/PD 位(STATUS<3>)被清除，/TO 位被设置，看门狗定时器被清 0 后继续运行（如果 WDTE=1），振荡器关闭，系统时钟停止。

如果是 T 型模式，管脚扫描继续进行。如果 WDT 使能，则 WDT 继续运行，直至溢出。

所有端口保持 SLEEP 执行前的状态。

省电模式有两种方式：

睡眠模式：高、低速时钟同时停止工作，此模式下可设置作 M-KEY 键盘扫描，功耗 < 0.3uA。

低速模式：低速时钟工作，高速时钟停止工作，此模式下可设置作 T-KEY 键盘扫描，功耗 < 1uA。

5.5 从 SLEEP 状态唤醒

系统可以从下列事件中从 SLEEP 状态唤醒：

- ◆ BOR 掉电复位唤醒
- ◆ 端口变化(由高到低)RPIF 唤醒
- ◆ INT 脚变化 INTF 唤醒
- ◆ WDT 溢出唤醒
- ◆ SPI 传输完成唤醒（只在从机模式下）

BOR 掉电复位会使系统复位，从而唤醒系统。

如果系统被中断事件唤醒，则相应的中断位被设置。不论 GIE 为何种状态，系统都会被唤醒。如果 GIE=1，则唤醒后程序从中断服务程序处（008h）开始执行，如果 GIE=0，则程序从 SLEEP 后的下一个 PC 处开始执行。

唤醒时间：外置 XT：1024 个 OSC 振荡周期；

内置 RC，4 个 OSC 振荡周期。

注意：睡眠之前先把各个标志位清除，才能正常唤醒。



5.6 中断

CORE 有三个中断源：

- ◆ 端口变化 RPIF 中断
- ◆ TMR0 溢出中断
- ◆ TMR1 溢出中断
- ◆ INTF 中断
- ◆ SPI 中断
- ◆ 触摸模块中 1 个定时器的溢出中断

INTCON 是中断控制寄存器。

全局中断使能位 GIE，如果设定，则可以允许所有能够中断部件的能够产生中断。个体中断可以通过 INTCON 寄存器相应的位来设置，与 GIE 的状态没有关系。

如果一个中断事件发生且中断允许位全部重置后，GIE 将被硬件清除，使中断使能无效，即在执行 reti 之前，不会重新中断。执行中断后，下一条指令将在 008H 处开始执行。

在重新设置 GIE 是中断重新有效前，中断标志位须软件清除。

所有中断的中断入口均在地址 008H，各个中断都没有单独的中断向量。

5.7 复位

系统复位有下列方式：

- ◆ 上电复位 (POR)
- ◆ 掉电复位 (BOR)
- ◆ 看门狗溢出复位

一些寄存器不会被任何一种复位影响。他们的状态在上电复位和其他未变得复位下是未知的。其他寄存器都有一个复位状态。如果不是特别指出，复位初态均为 0。

POR or BOR 复位时间：24ms+n*Tosc（外置振荡：n=1024，内置 RC：n=4）

看门狗溢出：n*Tosc（外置振荡：n=1024，内置 RC：n=4）

5.8 CODE 操作寄存器

HS505x 有一个代码选择寄存器，它不属于常规程序存储器。在常规执行程序期间，它不可被访问。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
LVD32	LVD24	-	OSC	ENWDT	CP	-	IRC

Bit15(LVD32)：3.2V 电压检测使能位。SLEEP 状态时该电压检测电路无效。

1：有效，当电源电压小于 3.2v 时，其对应的标志位为 1。

0：无效。

Bit14(LVD24)：2.4V 电压检测使能位。SLEEP 状态时该电压检测电路无效。

1：有效，当电源电压小于 2.4v 时，其对应的标志位为 1。

0：无效。

Bit13：该位和 LVD24 联合使用。



1: 当 2.4V 检测有效时, 如果电源电压低于 2.4v, 则会引起芯片复位;

0: 无论 2.4v 检测是否有效, 都不会在电源电压低于 2.4v 时使芯片复位。

注: 该位是 LVD24 检测的一个延伸, 如果该位为 1, 则可以在 LVD24 检测有效时, 同时可以使芯片复位, 且原 LVD24 检测功能在 sleep 时是无效的, 但该位一旦为 1, 则 LVD24 检测无论是否在 sleep 状态都会一直有效 (前提是 LVD24 先有效)。

所有电压检测的误差在 -20~60°C 有最大 ±0.3v 的偏差。

Bit12(OSC): 内外置振荡器选择。

0: 内置。

1: 外置。

Bit11(ENWDT): 看门狗定时器使能位

0: 使能。

1: 禁止。

Bit10 (CP): 代码加密选择位

1: 加密。

0: 不加密。

Bit9

Bit8: 内置振荡器频率选择。

0: 4MHz。

1: 8MHz。

Bit7~Bit0: 内部 RC 的修调位, C7~C0 只能被置为 1。



6. 电气参数

6.1 极限参数

参数	符号	参数范围	单位
电源电压	Vcc	-0.3~4.5	V
输入电压	Vi	-0.3~Vcc+0.3	V
输出电压	Vo	-0.3~Vcc+0.3	V
最大功耗 (Vcc=3V,空载)	Pa	10	mW
工作温度	Topr	-10~+70	°C
贮存温度	Tstg	-40~+125	°C

6.2 DC 特性

参数	符号	测试条件	最小值	典型值	最大值	单位
电源电压	Vcc		2.0	3	5.5	V
电源电流	Icc	Sleep 执行, Vcc=3V		1	2	μA
REM 灌电流	IoL	Vcc=3V, (内置三极管) VOL=0.2VDD,	280	300	350	mA
I/O 端口做输出时	IoL	VOL=0.2VDD, VDD=3V		16.5		mA
		VOL=0.2VDD, VDD=5V		36		
	Ioh	VDD=3V VOH=0.8VDD		6		
		VDD=5V VOH=0.8VDD		14.3		
VPP (高电平较弱)	IoL	VOL=0.2VDD, VDD=3V		9		mA
	Ioh	VDD=3V VOH=0.8VDD,			0.1	uA
准 I/O 端口做输出时	IoL	与 I/O 一致				mA
	Ioh	VDD=3V VOH=0.8VDD,	43	64	89	uA
	R	折合电阻	35	20	12	K Ω
	Ioh	VDD=3V VOH=0	83	130	190	uA
I/O 口输入电平	Vil				0.3vdd	v
	Vih		0.7vdd			v

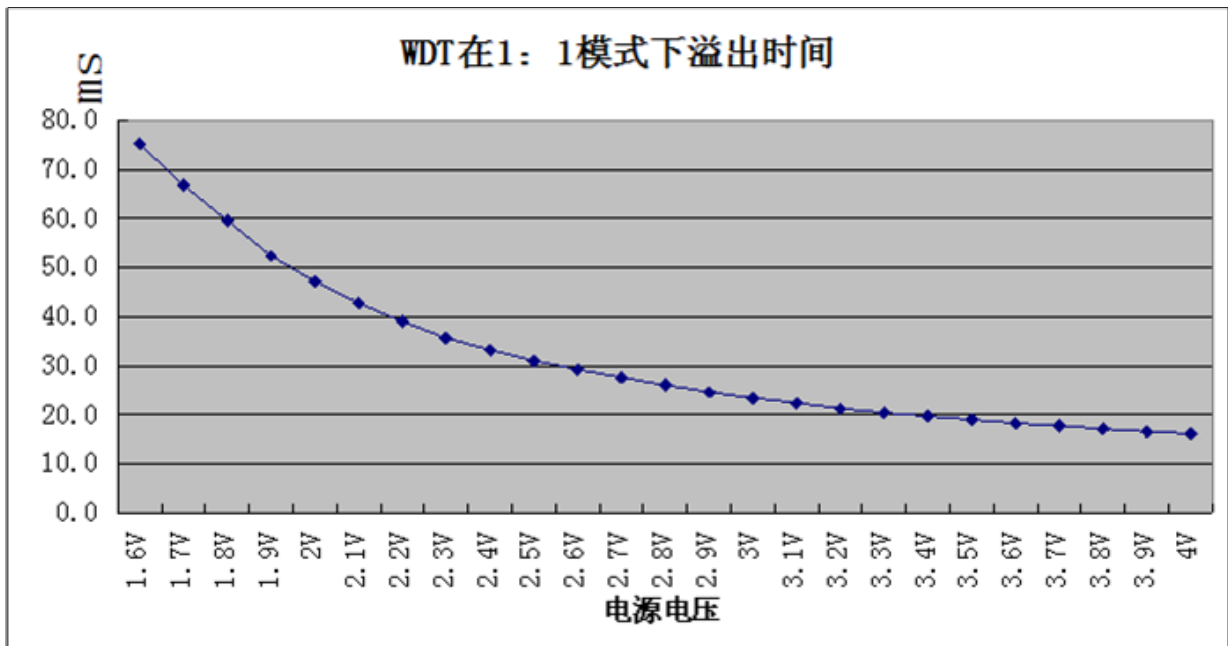


6.3 AC 特性

参 数	符号	测试条件	最小值	典型值	最大值	单位
工作频率	F	2~5.5			8M	Hz
上电复位时间	OST	内置振荡时		$T_p+1024osc$		ms
		外置振荡时		T_p+4osc		
看门狗复位时间	Twdt			$x*T_p$		ms
端口由低到高 上升时间	Tr	VDD>2.3V			10	ms
端口由高到低 下降时间	Tf	VDD>2.3V			100	us

注：Tp 为静态下 2 倍管脚扫描周期。

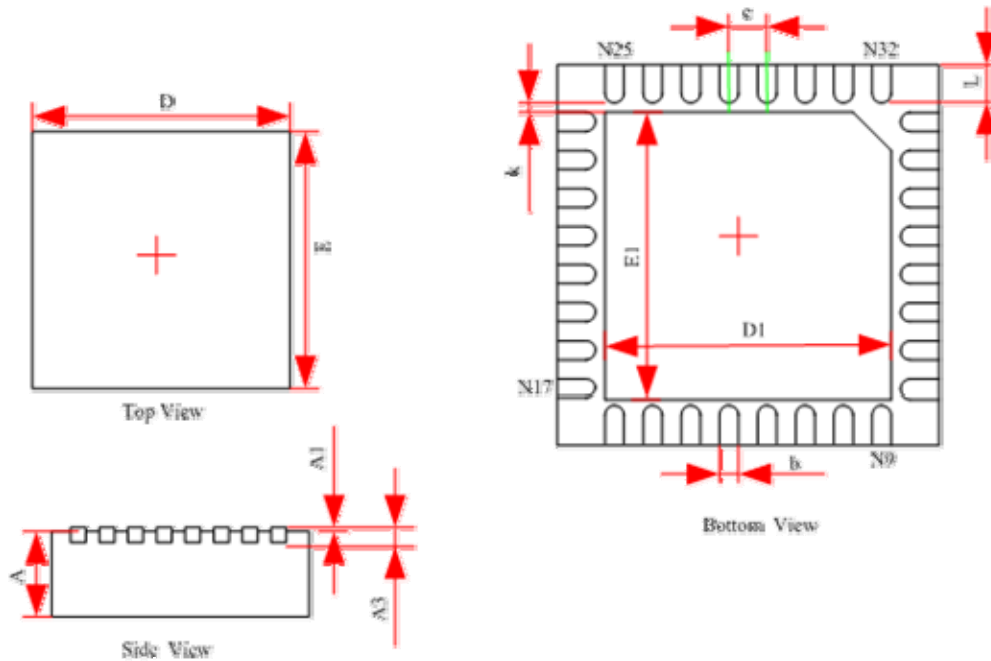
6.4 WDT 的溢出时间 (1:1)





7. 封装及尺寸

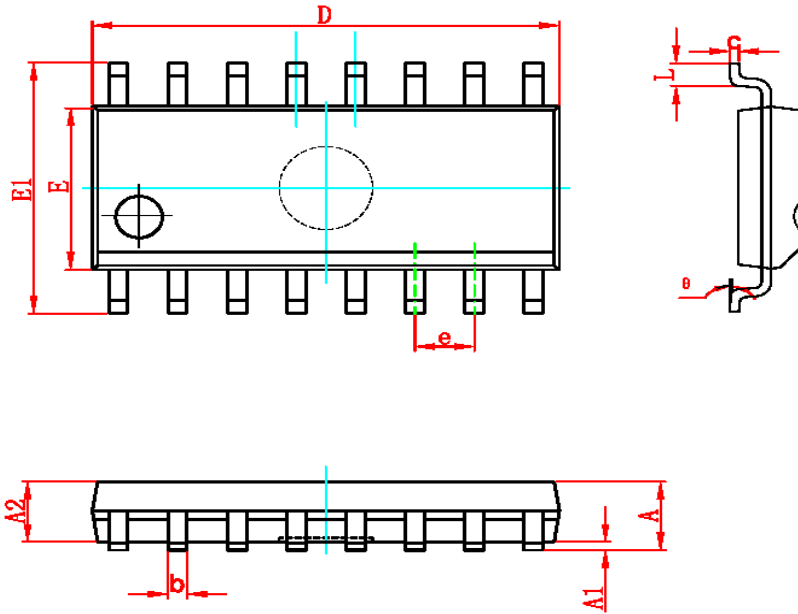
7.1 封装图及尺寸 (QFN32)



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	0.700/0.800	0.800/0.900	0.028/0.031	0.031/0.035
A1	0.000	0.050	0.000	0.002
A3	0.203REF		0.008 REF	
D	4.924	5.076	0.194	0.200
E	4.924	5.076	0.194	0.200
D1	3.300	3.500	0.130	0.138
E1	3.300	3.500	0.130	0.138
k	0.200MIN		0.008MIN	
b	0.180	0.300	0.007	0.012
e	0.500TYP		0.020TYP	
L	0.324	0.476	0.013	0.019



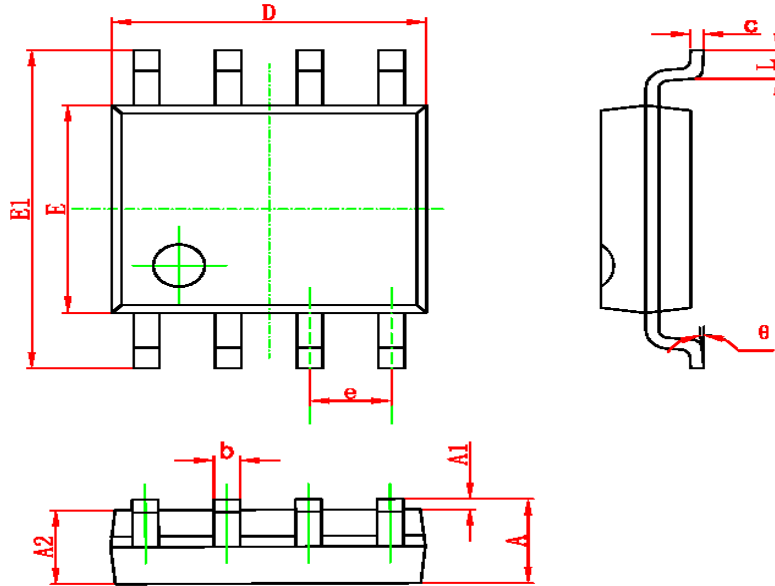
7.2 封装图及尺寸 (SOP16)



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	1.350	1.750	0.053	0.069
A1	0.100	0.250	0.004	0.010
A2	1.350	1.550	0.053	0.061
b	0.330	0.510	0.013	0.020
c	0.170	0.250	0.007	0.010
D	9.800	10.200	0.386	0.402
E	3.800	4.000	0.150	0.157
E1	5.800	6.200	0.228	0.244
e	1.270 (BSC)		0.050 (BSC)	
L	0.400	1.270	0.016	0.050
θ	0°	8°	0°	8°



7.2 封装图及尺寸 (SOP8)



Symbol	Dimensions In Millimeters		Dimensions In Inches	
	Min	Max	Min	Max
A	1.350	1.750	0.053	0.069
A1	0.100	0.250	0.004	0.010
A2	1.350	1.550	0.053	0.061
b	0.330	0.510	0.013	0.020
C	0.170	0.250	0.006	0.010
D	4.700	5.100	0.185	0.200
E	3.800	4.000	0.150	0.157
E1	5.800	6.200	0.228	0.244
E	1.270(BSC)		0.050(BSC)	
L	0.400	1.270	0.016	0.050
H	0.250(TYP)		0.01(TYP)	



8. 修正记录

版本	时间	内容	作者
V1.00A	2016.3.30	初版	HLJ
V1.10A	2017.7.3	第二版（封装添加）	CYN

Information furnished is believed to be accurate and reliable. However, HuaXin Micro-electronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. The actual part delivered may not completely agree with the description written here and it is user's responsibility to make wise judgment on the performance. HuaXin Micro-electronics assumes no responsibility for the mismatch occurred. No license is granted by implication or otherwise under any patent or patent rights of HuaXin Micro-electronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. HuaXin Micro-electronics products are not authorized for use as critical components in life support devices or systems without express written approval of HuaXin Micro-electronics.