



### 集成8路触摸按键，12位PWM和12位ADC的增强型8051微控制器

#### 1. 特性

- 基于8051指令流水线结构的8位单片机
- Flash ROM: 64K字节 (烧写次数1万次)
- RAM: 内部256字节, 外部1280字节, LED RAM 28字节, 触摸按键RAM 16字节
- 类EEPROM: 2048字节 (烧写次数10万次)
- 工作电压:
  - $f_{OSC} = 2\text{MHz} - 12\text{MHz}$ ,  $V_{DD} = 2.7\text{V} - 5.5\text{V}$
- 振荡器 (代码选项):
  - 晶体谐振器: 32.768kHz
  - 晶体谐振器: 2MHz - 12MHz
  - 陶瓷谐振器: 2MHz - 12MHz
  - 内部高频RC振荡器: 16.6MHz ( $T_A = +25^\circ\text{C}$ ,  $\pm 1\%$ )  
全温度范围 ( $T_A = -40^\circ\text{C} + 85^\circ\text{C}$ ,  $\pm 2\%$ )
  - 内部低频RC振荡器: 128KHz ( $\pm 10\%$ )
- 28pin: 26个CMOS双向I/O管脚  
20pin: 18个CMOS双向I/O管脚
- I/O内建上拉电阻 (典型值30K)
- 7个大电流驱动口 (7路可以驱动共阴/共阳LED)
- 5个16位定时器/计数器T0, T1, T2, T3, T4
- 2个12位PWM定时器
- 中断源:
  - 定时器0, 1, 2, 3, 4
  - 外部中断4: 4输入
  - 外部中断3: 3输入
  - 外部中断2
  - ADC, EUART, TWI, SCM, LPD
  - PWM1, 2, TOUCH, CRC
- 集成8路触摸按键功能
- 增强型EUART
- TWI (主从模式)
- 13通道12位模数转换器 (ADC), 内建常温输出1.25V ( $\pm 1\%$ ), 全温度范围 ( $T_A = -40^\circ\text{C} + 85^\circ\text{C}$ ,  $\pm 2.5\%$ ), 基准电压作为ADC通道输入
- LED驱动器 (7COM驱动共阴/共阳LED)
  - 7 X 13段
- 内建低电压检测功能 (LPD)
  - LPD电压2.8V - 4.2V可设置
- 内建的低电压复位功能 (LVR) (代码选项)
  - LVR电压1: 3.7V
  - LVR电压2: 3.1V
  - LVR电压3: 2.8V
- CPU机器周期:
  - 1个振荡周期
- 看门狗定时器 (WDT)
- 预热计数器
- 循环冗余校验 (CRC)
- 支持省电运行模式:
  - 空闲模式
  - 掉电模式
- Flash型ROM
- 封装:
  - SOP28/20

#### 2. 概述

SH79F6412是一种高速高效率8051可兼容单片机。在同样振荡频率下,较之传统的8051芯片它有着运行更快速的优越特性。

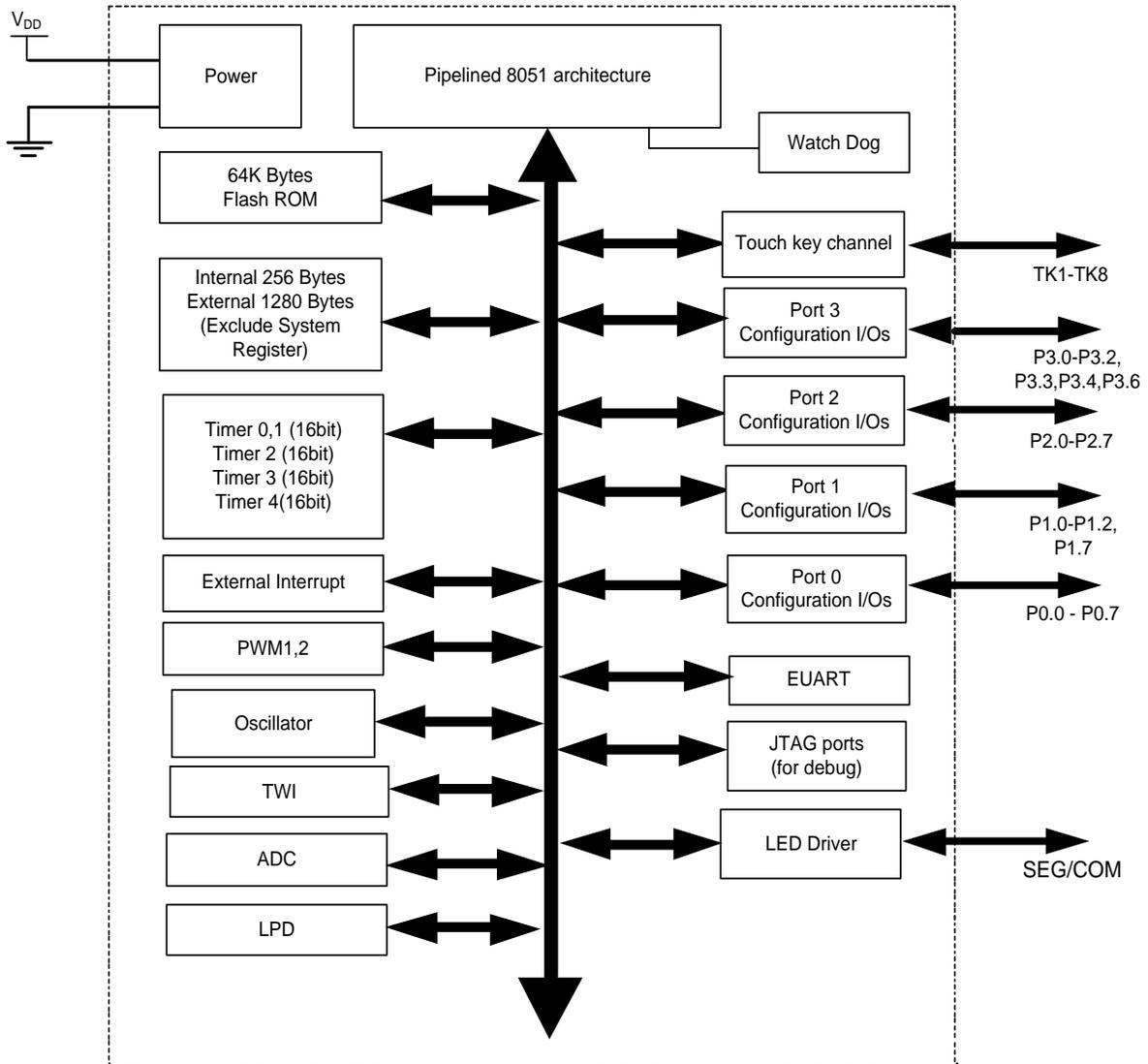
SH79F6412保留了标准8051芯片的大部分特性。这些特性包括内置256字节RAM, 1个UART和外置中断INT2, INT3和INT4, 一个TWI中断, 一个TK中断。此外, SH79F6412还集成了外部1280字节RAM, 可兼容8052芯片的16位定时器 (Timer2)。该单片机还包括适合于程序和数据的64K字节Flash块。

SH79F6412不仅集成了如EUART/TWI等标准通讯模块, 此外还集成了LED驱动器, 具有内建比较功能的ADC, PWM定时器等模块。

为了达到高可靠性和低功耗, SH79F6412内建看门狗定时器, 低电压复位功能及系统时钟监控功能。此外SH79F6412还提供了2种低功耗省电模式。



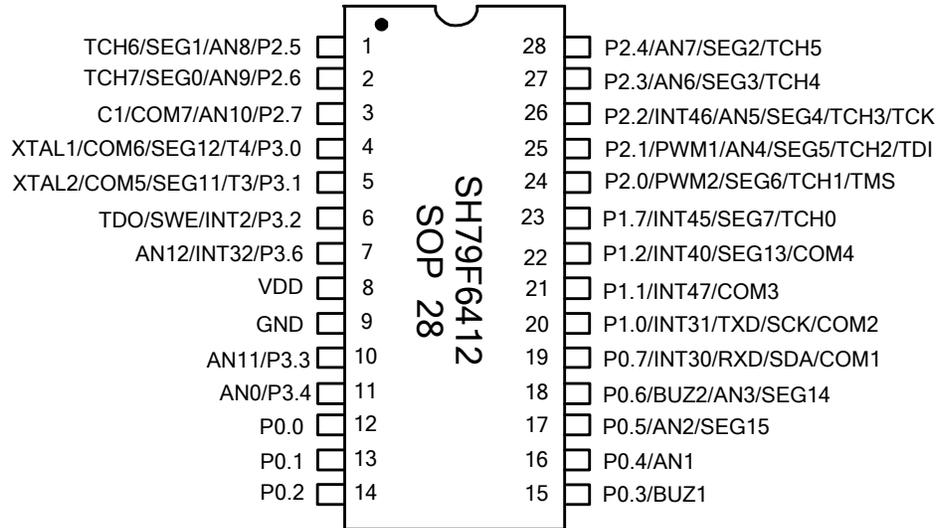
3. 方框图



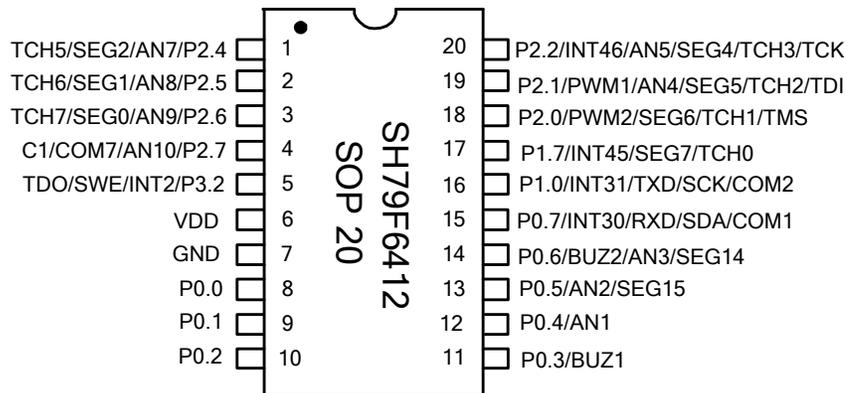


#### 4. 引脚配置

##### 28Pin SOP封装引脚图



##### 20Pin SOP封装引脚图



引脚配置图

**注意:**

- (1) 未使用的I/O需要设置输出（保持低电平）或者输入上拉（保持高电平），避免引脚浮动导致的漏电及干扰。  
SOP28封装的P1.3-P1.6, P3.5, P4.0-P4.3需要设置。（PxPCRx, x=0-4, y=0-7）（Px.y, x=0-4, y=0-7）  
SOP20封装的P1.1-P1.6, P2.3, P3.0-P3.1, P3.3-P3.5, P4.0-P4.3需要设置。（PxPCRx, x=0-4, y=0-7）（Px.y, x=0-4, y=0-7）
- (2) 命名中，写在最外侧的功能具有最高优先级，最内侧的功能具有最低优先级（参见配置图）。当一个被高优先级的功能占用时，即使低优先级功能被允许，也不能作为低优先级功能。只有当软件禁止高优先级功能，才能被释放作为低优先级端口使用。



Table 4.1 引脚功能

引脚编号 (SOP28)	引脚编号 (SOP20)	引脚命名	默认功能
1	2	TCH6/SEG1/AN8/P2.5	P2.5
2	3	TCH7/SEG0/AN9/P2.6	P2.6
3	4	C1/COM7/AN10/P2.7	P2.7
4	-	XTAL1/COM5/SEG12/T4/P3.0	P3.0
5	-	XTAL2/COM6/SEG11/T3/P3.1	P3.1
6	5	SWE/INT2/P3.2	P3.2
7	-	AN11/INT32/P3.6	P3.6
8	6	V <sub>DD</sub>	---
9	7	GND	---
10	-	AN11/P3.3	P3.3
11	-	AN0/P3.4	P3.4
12	8	P0.0	P0.0
13	9	P0.1	P0.1
14	10	P0.2	P0.2
15	11	BUZ1/P0.3	P0.3
16	12	AN1/P0.4	P0.4
17	13	SEG15/AN2/P0.5	P0.5
18	14	SEG14/AN3/BUZ2/P0.6	P0.6
19	15	COM1/SDA/RXD/INT30/P0.7	P0.7
20	16	COM2/SCK/TXD/INT31/P1.0	P1.0
21	-	COM3/INT47/P1.1	P1.1
22	-	COM4/SEG13/INT40/P1.2	P1.2
23	17	TCH0/SEG7/INT45/P1.7	P1.7
24	18	TMS/TCH1/SEG6/PWM2/P2.0	P2.0
25	19	TDI/TCH2/SEG5/AN4/PWM1/P2.1	P2.1
26	20	TCK/TCH3/SEG4/AN5/INT46/P2.2	P2.2
27	-	TCH4/SEG3/AN6/P2.3	P2.3
28	1	TCH5/SEG2/AN7/P2.4	P2.4

**注意:**

未使用的I/O需要设置输出（保持低电平）或者输入上拉（保持高电平），避免引脚浮动导致的漏电及干扰。

SOP28封装的P1.3-P1.6, P3.5, P4.0-P4.3需要设置。（PxPCRY, x=0-4, y=0-7）（Px.y, x=0-4, y=0-7）

SOP20封装的P1.1-P1.6, P2.3, P3.0-P3.1, P3.3-P3.5, P4.0-P4.3需要设置。（PxPCRY, x=0-4, y=0-7）（Px.y, x=0-4, y=0-7）



## 5. 引脚描述

引脚编号	类型	说明
<b>I/O端口</b>		
P0.0 - P0.7	I/O	8位双向I/O端口
P1.0 - P1.2, P1.7	I/O	4位双向I/O端口
P2.0 - P2.7	I/O	8位双向I/O端口
P3.0 - P3.2, P3.3, P3.4, P3.6	I/O	6位双向I/O端口
<b>定时器</b>		
T3	I/O	定时器3外部输入
T4	I/O	定时器4外部输入/比较输出
<b>PWM控制器</b>		
PWM1	O	12位PWM0定时器输出引脚
PWM2	O	12位PWM1定时器输出引脚
<b>EUART</b>		
RXD	I	EUART数据输入引脚
TXD	O	EUART数据输出引脚
<b>ADC</b>		
AN0 - AN12	I	ADC输入通道
<b>LED驱动器</b>		
LED_C1 - LED_C7	O	LED显示COM信号输出引脚
LED_S0 - S7 LED_S11 - S15	O	LED显示Segment信号输出引脚
<b>IIC</b>		
SDA	I/O	TWI数据输入/输出引脚
SCK	I/O	TWI时钟引脚
<b>TK</b>		
TCH0 - 7	I	触摸按键输入引脚
<b>蜂鸣器</b>		
BUZ1, 2	O	蜂鸣器输出



续上表

中断&复位&时钟&电源		
INT2	I	外部中断2
INT30	I	外部中断30
INT31	I	外部中断31
INT32	I	外部中断32
INT40, INT45 - 47	I	外部中断40, 45 - 47
XTAL1	I	谐振器输入
XTAL2	O	谐振器输出
GND	P	接地
VDD	P	电源 (2.7 - 5.5V)
单线仿真		
SWE (P3.2)	I/O	单线仿真
TDO (P3.2)	O	调试接口: 测试数据输出
TCK (P2.2)	I	调试接口: 测试模式选择
TDI (P2.1)	I	调试接口: 测试数据输入
TMS (P2.0)	I	调试接口: 测试时钟输入
<b>注意:</b> 当P3.2, P2.2 - P2.0作为调试接口时, P3.2, P2.2 - P2.0功能被禁止。 当P3.2为单线仿真时, 其他功能无效。		



## 6. SFR映像

SH79F6412内置256字节的直接寻址寄存器，包括通用数据存储器 and 特殊功能寄存器（SFR），SH79F6412的SFR有以下几种：

<b>CPU内核寄存器：</b>	ACC, B, PSW, SP, DPL, DPH
<b>CPU内核增强寄存器：</b>	AUXC, DPL1, DPH1, INSCON, XPAGE
<b>电源时钟控制寄存器：</b>	PCON, SUSLO
<b>Flash寄存器：</b>	IB_OFFSET, IB_DATA, IB_CON1, IB_CON2, IB_CON3, IB_CON4, IB_CON5, FLASHCON
<b>数据页面控制寄存器：</b>	XPAGE
<b>看门狗定时器寄存器：</b>	RSTSTAT
<b>系统时钟控制寄存器：</b>	CLKCON
<b>中断寄存器：</b>	IEN0, IEN1, IENC, IENC1, IENC2, IPH0, IPL0, IPH1, IPL1, EXF0, EXF1, EXF3, EXCON0, EXCON1, EXCON2
<b>I/O口寄存器：</b>	P0, P1, P2, P3, P4, P0CR, P1CR, P2CR, P3CR, P4CR, P0PCR, P1PCR, P2PCR, P3PCR, P4PCR, P1OS
<b>定时器寄存器：</b>	TCON, TMOD, TL0, TH0, TL1, TH1, TCON1, T2CON, T2MOD, TH2, TL2, RCAP2L, RCAP2H, T3CON, TH3, TL3, T4CON, TH4, TL4, SWTHL
<b>EUART寄存器：</b>	SCON, SBUF, SADEN, SADDR, PCON, RxCON
<b>ADC寄存器：</b>	ADCON, ADT, ADC1H, ADC2H, ADDL, ADDH
<b>LED寄存器：</b>	DISPCON, SEG01, SEG02, DISPCLK, LEDCOM, DISCOM, LIGHTCOM, SHARECON
<b>BUZZER寄存器：</b>	BUZCON, BUZD, BUZP
<b>TWI寄存器：</b>	TWIDAT, TWIADR, TWISTA, TWICON
<b>PWM寄存器：</b>	PWMEN, PWMEN1, PWML0, PWM1C, PWM2C, PWM1PL, PWM1PH, PWM1DL, PWM1DH, PWM2C, PWM2PL, PWM2PH, PWM2DL, PWM2DH
<b>LPD寄存器：</b>	LPDCON
<b>TK寄存器：</b>	TKCON1, TKF0, TKU1, TKDIV01, TKDIV02, TKDIV03, TKDIV04, TKVREF, TKST, TKRANDOM, TKCOUNT, TKW, P2SS, TKWAIT
<b>CRC寄存器：</b>	CRCCON, CRCDL, CRCDH
<b>ISP寄存器：</b>	ISPCON, ISPLO



## SH79F6412

**Table 6.1** C51核SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ACC	E0H	累加器	00000000	ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0
B	F0H	B寄存器	00000000	B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0
AUXC	F1H	C寄存器	00000000	C.7	C.6	C.5	C.4	C.3	C.2	C.1	C.0
PSW	D0H	程序状态字	00000000	CY	AC	F0	RS1	RS0	OV	F1	P
SP	81H	堆栈指针	00000111	SP.7	SP.6	SP.5	SP.4	SP.3	SP.2	SP.1	SP.0
DPL	82H	数据指针低位字节	00000000	DPL0.7	DPL0.6	DPL0.5	DPL0.4	DPL0.3	DPL0.2	DPL0.1	DPL0.0
DPH	83H	数据指针高位字节	00000000	DPH0.7	DPH0.6	DPH0.5	DPH0.4	DPH0.3	DPH0.2	DPH0.1	DPH0.0
DPL1	84H	数据指针1低位字节	00000000	DPL1.7	DPL1.6	DPL1.5	DPL1.4	DPL1.3	DPL1.2	DPL1.1	DPL1.0
DPH1	85H	数据指针1高位字节	00000000	DPH1.7	DPH1.6	DPH1.5	DPH1.4	DPH1.3	DPH1.2	DPH1.1	DPH1.0
INSCON	86H	数据指针选择	-0--00-0	-	BKS0	-	-	DIV	MUL	-	DPS

**Table 6.2** 电源时钟控制SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PCON	87H	电源控制	00--0000	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL
SUSLO	8EH	电源控制保护字	00000000	SUSLO.7	SUSLO.6	SUSLO.5	SUSLO.4	SUSLO.3	SUSLO.2	SUSLO.1	SUSLO.0



## SH79F6412

**Table 6.3** Flash控制SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_OFF SET	FBH Bank0	可编程flash低位字节偏移	00000000	IB_OFF SET.7	IB_OFF SET.6	IB_OFF SET.5	IB_OFF SET.4	IB_OFF SET.3	IB_OFF SET.2	IB_OFF SET.1	IB_OFF SET.0
IB_DATA	FCH Bank0	可编程flash数据寄存器	00000000	IB_DATA.7	IB_DATA.6	IB_DATA.5	IB_DATA.4	IB_DATA.3	IB_DATA.2	IB_DATA.1	IB_DATA.0
IB_CON1	F2H Bank0	flash控制寄存器1	00000000	IB_CON1.7	IB_CON1.6	IB_CON1.5	IB_CON1.4	IB_CON1.3	IB_CON1.2	IB_CON1.1	IB_CON1.0
IB_CON2	F3H Bank0	flash控制寄存器2	----0000	-	-	-	-	IB_CON2.3	IB_CON2.2	IB_CON2.1	IB_CON2.0
IB_CON3	F4H Bank0	flash控制寄存器3	----0000	-	-	-	-	IB_CON3.3	IB_CON3.2	IB_CON3.1	IB_CON3.0
IB_CON4	F5H Bank0	flash控制寄存器4	----0000	-	-	-	-	IB_CON4.3	IB_CON4.2	IB_CON4.1	IB_CON4.0
IB_CON5	F6H Bank0	flash控制寄存器5	----0000	-	-	-	-	IB_CON5.3	IB_CON5.2	IB_CON5.1	IB_CON5.0
XPAGE	F7H Bank0	编程用地址选择寄存器	00000000	XPAGE.7	XPAGE.6	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
FLASHCON	A7H Bank0	flash控制寄存器	-----0	-	-	-	-	-	-	-	FAC

**Table 6.4** 在系统编程控制SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ISPLO	AAH Bank0	ISP辅助寄存器	00000000	ISPLO.7	ISPLO.6	ISPLO.5	ISPLO.4	ISPLO.3	ISPLO.2	ISPLO.1	ISPLO.0
ISPCON	ABH Bank0	ISP控制寄存器	00000000	ISPCON.7	ISPCON.6	ISPCON.5	ISPCON.4	ISPCON.3	ISPCON.2	ISPCON.1	ISPCON.0

**Table 6.5** WDT SFR

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
RSTSTAT	B1H Bank0	看门狗定时器控制寄存器	0000-000*	WDOF	SWRF	PORF	LVRF	-	WDT.2	WDT.1	WDT.0

**注意:** \*表示不同情况的复位决定RSTSTAT寄存器中的复位值, 详见WDT章节



## SH79F6412

**Table 6.6** 时钟控制SFR

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
CLKCON	B2H Bank0	系统时钟选择	111000--	32k_SPDUP	CLKS1	CLKS0	SCMIF	HFON	FS	-	-

**Table 6.7** 中断SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IEN0	A8H Bank0	中断允许控制0	00000000	EA	EADC	ET2	ES	ET1	EPWM2	ET0	TKIE
IEN1	A9H Bank0	中断允许控制1	00000000	ESCM	EX3	EPWM1	ET3	ETWI	EX4	EX2	ET4
IENC	BAH Bank0	中断通道允许控制	000----0	EXS47	EXS46	EXS45	-	-	-	-	EXS40
IENC1	BBH Bank0	中断通道允许控制1	--000-00	-	-	ECRC	EPWM1	EPWM2	-	ESCM1	ELPD
IENC2	BCH Bank0	中断通道允许控制2	-----000	-	-	-	-	-	EXS32	EXS31	EXS30
IPH0	B4H Bank0	中断优先权控制高位0	-0000000	-	PADH	PT2H	PESH	PX3H	PW2H	PT0H	PTKH
IPL0	B8H Bank0	中断优先权控制低位0	-0000000	-	PADL	PT2L	PESL	PX3L	PW2L	PTKL	PT0L
IPH1	B5H Bank0	中断优先权控制高位1	00000000	PSCH	PT1H	PW1H	PT3H	PTWH	PX4H	PX2H	PT4H
IPL1	B9H Bank0	中断优先权控制低位1	00000000	PSCL	PT1L	PW1L	PT3L	PTWL	PX4L	PX2L	PT4L
EXF0	E8H Bank0	外部中断寄存器0	00000000	IT4.1	IT4.0	IT3.1	IT3.0	IT2.1	IT2.0	IE3	IE2
EXF1	D8H Bank0	外部中断寄存器1	000----0	IF47	IF46	IF45	-	-	-	-	IF40
EXF3	D1H Bank0	外部中断寄存器2	-----000	-	-	-	-	-	IF32	IF31	IF30
EXCON	A4H Bank0	外部中断采样次数控制	00000000	EXCON	I1PS1	I1PS0	I1SN1	I1SN0	I0PS1	I0PS0	I0SN1



## SH79F6412

Table 6.8 端口SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P0	80H Bank0	8位端口0	00000000	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
P1	90H Bank0	8位端口1	00000000	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
P2	A0H Bank0	8位端口2	00000000	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
P3	B0H Bank0	3位端口3	-0000000	-	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
P4	90H Bank1	3位端口3	----0000	-	-	-	-	P4.3	P4.2	P4.1	P4.0
P0CR	E1H Bank0	端口0输入/输出方向控制	00000000	P0CR.7	P0CR.6	P0CR.5	P0CR.4	P1CR.3	P1CR.2	P1CR.1	P1CR.0
P1CR	E2H Bank0	端口1输入/输出方向控制	00000000	P1CR.7	P1CR.6	P1CR.5	P1CR.4	P1CR.3	P1CR.2	P1CR.1	P1CR.0
P2CR	E3H Bank0	端口2输入/输出方向控制	00000000	P2CR.7	P2CR.6	P2CR.5	P2CR.4	P2CR.3	P2CR.2	P2CR.1	P2CR.0
P3CR	E4H Bank0	端口3输入/输出方向控制	-0000000	-	P3CR.6	P3CR.5	P3CR.4	P3CR.3	P3CR.2	P3CR.1	P3CR.0
P4CR	91H Bank1	端口4输入/输出方向控制	----0000	-	-	-	-	P4CR.3	P4CR.2	P4CR.1	P4CR.0
P0PCR	E9H Bank0	端口0内部上拉允许	00000000	P0PCR.7	P0PCR.6	P0PCR.5	P0PCR.4	P0PCR.3	P0PCR.2	P0PCR.1	P0PCR.0
P1PCR	EAH Bank0	端口1内部上拉允许	00000000	P1PCR.7	P1PCR.6	P1PCR.5	P1PCR.4	P1PCR.3	P1PCR.2	P1PCR.1	P1PCR.0
P2PCR	EBH Bank0	端口2内部上拉允许	00000000	P2PCR.7	P2PCR.6	P2PCR.5	P2PCR.4	P2PCR.3	P2PCR.2	P2PCR.1	P2PCR.0
P3PCR	ECH Bank0	端口3内部上拉允许	-0000000	-	P3PCR.6	P3PCR.5	P3PCR.4	P3PCR.3	P3PCR.2	P3PCR.1	P3PCR.0
P4PCR	92H Bank1	端口4内部上拉允许	----0000	-	-	-	-	P4PCR.3	P4PCR.2	P4PCR.1	P4PCR.0
P1OS	EFH Bank0	输出模式选择	----0000	-	-	-	-	P1OSCR.1	P1OSCR.0	P1OS.1	P1OS.0
P2SS	9DH Bank0	触摸按键模式选择	00000000	P2SS.7	P2SS.6	P2SS.5	P2SS.4	P2SS.3	P2SS.2	P2SS.1	P2SS.0

**注意:**

未使用的I/O需要设置输出（保持低电平）或者输入上拉（保持高电平），避免引脚浮动导致的漏电及干扰。

SOP28封装的P1.3-P1.6, P3.5, P4.0-P4.3需要设置。(PxPCRY, x=0-4, y=0-7) (Px.y, x=0-4, y=0-7)

SOP20封装的P1.1-P1.6, P2.3, P3.0-P3.1, P3.3-P3.5, P4.0-P4.3需要设置。(PxPCRY, x=0-4, y=0-7) (Px.y, x=0-4, y=0-7)



## SH79F6412

Table 6.9 定时器SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TCON	88H Bank0	定时器/计数器控制寄存器	0000----	TF1	TR1	TF0	TR0	-	-	-	-
TMOD	89H Bank0	定时器/计数器0, 1模式寄存器	--00--00	-	-	M11	M10	-	-	M01	M00
TL0	8AH Bank0	定时器/计数器0低位字节	00000000	TL0.7	TL0.6	TL0.5	TL0.4	TL0.3	TL0.2	TL0.1	TL0.0
TH0	8BH Bank0	定时器/计数器0高位字节	00000000	TH0.7	TH0.6	TH0.5	TH0.4	TH0.3	TH0.2	TH0.1	TH0.0
TL1	8CH Bank0	定时器/计数器1低位字节	00000000	TL1.7	TL1.6	TL1.5	TL1.4	TL1.3	TL1.2	TL1.1	TL1.0
TH1	8DH Bank0	定时器/计数器1高位字节	00000000	TH1.7	TH1.6	TH1.5	TH1.4	TH1.3	TH1.2	TH1.1	TH1.0
TCON1	8FH Bank0	定时器/计数器0, 1控制寄存器	-00-00--	-	TCLK_S1	TCLK_S0	-	TCLKP1	TCLKP0	-	-
T2CON	C8H Bank0	定时器/计数器2控制寄存器	0----0--	TF2	-	-	-	-	TR2	-	-
T2MOD	C9H Bank0	定时器/计数器2模式寄存器	0-----	TCLKP2	-	-	-	-	-	-	-
RCAP2L	CAH Bank0	定时器/计数器2重载/截获低位字节	00000000	RCAP2L.7	RCAP2L.6	RCAP2L.5	RCAP2L.4	RCAP2L.3	RCAP2L.2	RCAP2L.1	RCAP2L.0
RCAP2H	CBH Bank0	定时器/计数器2重载/截获高位字节	00000000	RCAP2H.7	RCAP2H.6	RCAP2H.5	RCAP2H.4	RCAP2H.3	RCAP2H.2	RCAP2H.1	RCAP2H.0
TL2	CCH Bank0	定时器/计数器2低位字节	00000000	TL2.7	TL2.6	TL2.5	TL2.4	TL2.3	TL2.2	TL2.1	TL2.0
TH2	CDH Bank0	定时器/计数器2高位字节	00000000	TH2.7	TH2.6	TH2.5	TH2.4	TH2.3	TH2.2	TH2.1	TH2.0
T3CON	C0H Bank1	定时器/计数器3控制寄存器	0-00-000	TF3	-	T3PS.1	T3PS.0	-	TR3	T3CLKS.1	T3CLKS.0
TL3	C2H Bank1	定时器/计数器3低位字节	00000000	TL3.7	TL3.6	TL3.5	TL3.4	TL3.3	TL3.2	TL3.1	TL3.0
TH3	C3H Bank1	定时器/计数器3高位字节	00000000	TH3.7	TH3.6	TH3.5	TH3.4	TH3.3	TH3.2	TH3.1	TH3.0
T4CON	C8H Bank1	定时器/计数器4控制寄存器	00000000	TF4	TC4	T4PS1	T4PS0	T4M1	T4M0	TR4	T4CLKS
TL4	CCH Bank1	定时器/计数器4低位字节	00000000	TL4.7	TL4.6	TL4.5	TL4.4	TL4.3	TL4.2	TL4.1	TL4.0
TH4	CDH Bank1	定时器/计数器4高位字节	00000000	TH4.7	TH4.6	TH4.5	TH4.4	TH4.3	TH4.2	TH4.1	TH4.0



## SH79F6412

**Table 6.10** EUART SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
SCON	98H Bank0	串行控制	00000000	SM0/FE	SM1/RXOV	SM2/TXCOL	REN	TB8	RB8	TI	RI
SBUF	99H Bank0	串行数据缓冲器	00000000	SBUF.7	SBUF.6	SBUF.5	SBUF.4	SBUF.3	SBUF.2	SBUF.1	SBUF.0
SADEN	9BH Bank0	从属地址掩码	00000000	SADEN.7	SADEN.6	SADEN.5	SADEN.4	SADEN.3	SADEN.2	SADEN.1	SADEN.0
SADDR	9AH Bank0	从属地址	00000000	SADDR.7	SADDR.6	SADDR.5	SADDR.4	SADDR.3	SADDR.2	SADDR.1	SADDR.0
SBRTH	C7H Bank0	波特率发生寄存器	00000000	SBRTEN	SBRT.14	SBRT.13	SBRT.12	SBRT.11	SBRT.10	SBRT.9	SBRT.8
SBRTL	BFH Bank0	波特率发生寄存器	00000000	SBRT.7	SBRT.6	SBRT.5	SBRT.4	SBRT.3	SBRT.2	SBRT.1	SBRT.0
SFINE	BEH Bank0	波特率微调寄存器	----0000	-	-	-	-	SFINE.3	SFINE.2	SFINE.1	SFINE.0
PCON	87H Bank0	电源和串行控制	00--0000	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL

**Table 6.11** BUZZER SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
BUZCON	BDH Bank0	蜂鸣器输出控制	00--00--	BUZEN	BUSLE	-	-	BCA1	BCA0	-	-
BUZP	AFH Bank0	蜂鸣器周期控制	00000000	BZP.7	BZP.6	BZP.5	BZP.4	BZP.3	BZP.2	BZP.1	BZP.0



## SH79F6412

**Table 6.12** ADC SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ADCON	93H Bank0	ADC控制寄存器	00000000	ADON	ADCIF	EC	SCH3	SCH2	SCH1	SCH0	GO/DONE
ADT	94H Bank0	ADC时间配置	000-0000	TADC2	TADC1	TADC0	-	TS3	TS2	TS1	TS0
ADC1H	95H Bank0	ADC通道配置1	00000000	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
ADC2H	91H Bank0	ADC通道配置2	0---0000	ADCV	-	-	-	CH11	CH10	CH9	CH8
ADDL	96H Bank0	ADC数据低位字节	----0000	-	-	-	-	A3	A2	A1	A0
ADDH	97H Bank0	ADC数据高位字节	00000000	A11	A10	A9	A8	A7	A6	A5	A4

**Table 6.13** CRC SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
CRCCON	ACH Bank0	CRC控制寄存器	00-00000	CRC_GO	CRCIF	-	CRCADDR4	CRCADDR3	CRCADDR2	CRCADDR1	DAT.0
CRCDL	ADH Bank0	CRC结果低位寄存器	00000000	CRCD.7	CRCD.6	CRCD.5	CRCD.4	CRCD.3	CRCD.2	CRCD.1	CRCD.0
CRCDH	AEH Bank0	CRC结果高位寄存器	00000000	CRCD.15	CRCD.14	CRCD.13	CRCD.12	CRCD.11	CRCD.10	CRCD.9	CRCD.8



## SH79F6412

**Table 6.14 TK SFRs**

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKCON1	A1H Bank0	TK功能控制寄存器	0-000000	TKCON	-	TKGO/DONE	SHARE	MODE	OVDD	FSW1	FSW0
TKF0	C1H Bank0	TK中断标志位寄存器	-00000--	-	IFERR	IFGO	IFAVE	IFCOUNT	IFTKOV	-	-
TKU1	C2H Bank0	TK通道选择寄存器	00000000	TK8	TK7	TK6	TK5	TK4	TK3	TK2	TK1
TKDIV01	C3H Bank0	TK放大系数寄存器	00000000	DIV7	DIV6	DIV5	DIV4	DIV3	DIV2	DIV1	DIV0
TKDIV02	C4H Bank0	TK放大系数寄存器	00000000	DIV15	DIV14	DIV13	DIV12	DIV11	DIV10	DIV9	DIV8
TKDIV03	C5H Bank0	TK放大系数寄存器	00000000	DIV23	DIV22	DIV21	DIV20	DIV19	DIV18	DIV17	DIV16
TKDIV04	C6H Bank0	TK放大系数寄存器	----0000	-	-	-	-	DIV27	DIV26	DIV25	DIV24
TKVREF	CEH Bank0	基准电压源选择寄存器	00000000	VREF1	VREF0	CMPD1	CMPD0	VTK1	VTK0	TUNE1	TUNE0
TKST	CFH Bank0	触摸按键频率选择寄存器	-0000000	-	ST.6	ST.5	ST.4	ST.3	ST.2	ST.1	ST.0
TKRANDOM	B6H Bank0	触摸按键频率选择寄存器	0000--00	TKRADON	TKOFFSET	TKVDD	TKOUT	-	-	RANDOM1	RANDOM0
TKW	B7H Bank0	TK通道错误显示寄存器	-----000	-	-	-	-	-	TW.2	TW.1	TW.0
TKCOUNT	E6H Bank0	触摸按键宽度选择寄存器	00000000	COUNT0.7	COUNT0.6	COUNT0.5	COUNT0.4	COUNT0.3	COUNT0.2	COUNT0.1	COUNT0.0
TKWAIT	A2H Bank0	触摸按键等待宽度选择寄存器	00000000	TKWAIT.7	TKWAIT.6	TKWAIT.5	TKWAIT.4	TKWAIT.3	TKWAIT.2	TKWAIT.1	TKWAIT.0

**Table 6.15 LPD SFR**

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
LPDCON	B3H Bank0	低电压检测控制寄存器	00---000	LPDEN	LPDF	-	-	-	LPDOP	LPDS1	LPDS0



## SH79F6412

**Table 6.16 LED SFRs**

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
DISPCON	D9H Bank0	LED功能选择寄存器	-0-000--	-	LEDON	-	MODSW	LEDMODE0	LEDMODE1	-	-
SEG01	DAH Bank0	SEG功能选择寄存器	00000000	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
SEG02	DBH Bank0	SEG功能选择寄存器	-0000---	-	SEG14	SEG13	SEG12	SEG11	-	-	-
DISPCLK	DCH Bank0	LED时钟频率选择寄存器	00000000	DCK0.7	DCK0.6	DCK0.5	DCK0.4	DCK0.3	DCK0.2	DCK0.1	DCK0.0
LEDCOM	DDH Bank0	COM功能选择寄存器	-0000000	-	COM7	COM6	COM5	COM4	COM3	COM2	COM1
DISCOM	DEH Bank0	LED COM扫描宽度寄存器	00000000	DCOM.7	DCOM.6	DCOM.5	DCOM.4	DCOM.3	DCOM.2	DCOM.1	DCOM.0
LIGHTCOM	DFH Bank0	LED COM亮度选择寄存器	-----000	-	-	-	-	-	CC3	CC2	CC1
SHARECON	D7H Bank0	SHARE控制寄存器	-----000	-	-	-	-	-	SHARE.2	SHARE.1	SHARE.0

**Table 6.17 TWI SFRs**

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TWICON	F8H Bank0	TWI设置寄存器	00000000	TOUT	ENTWI	STA	STO	TWINT	AA	TFREE	EFREE
TWISTA	F9H Bank0	TWI状态寄存器	11111000	TWISTA.7	TWISTA.6	TWISTA.5	TWISTA.4	TWISTA.3	CR.1	CR.0	ETOT
TWIADR	FAH Bank0	TWI数据地址寄存器	00000000	TWA.6	TWA.5	TWA.4	TWA.3	TWA.2	TWA.1	TWA.0	GC
TWIDAT	FDH Bank0	TWI数据输入/输出寄存器	00000000	TWIDAT.7	TWIDAT.6	TWIDAT.5	TWIDAT.4	TWIDAT.3	TWIDAT.2	TWIDAT.1	TWIDAT.0



## SH79F6412

**Table 6.18** PWM SFRs

符号	地址	名称	POR/WDT/LVR /PIN复位值	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWM1CON	E8H Bank1	PWM1功能选择寄存器	0000-000	PWM1EN	PWM1S	PWM1CK1	PWM1CK0	-	PWM1IE	PWM1IF	PWM1SS
PWM2CON	E9H Bank1	PWM2功能选择寄存器	0000-000	PWM2EN	PWM2S	PWM2CK1	PWM2CK0	-	PWM2IE	PWM2IF	PWM2SS
PWM1PH	EAH Bank1	12位PWM1周期控制高位	----0000	-	-	-	-	PWM1P.11	PWM1P.10	PWM1P.9	PWM1P.8
PWM1PL	EBH Bank1	12位PWM1周期控制低位	00000000	PWM1P.7	PWM1P.6	PWM1P.5	PWM1P.4	PWM1P.3	PWM1P.2	PWM1P.1	PWM1P.0
PWM2PH	ECH Bank1	12位PWM2周期控制高位	----0000	-	-	-	-	PWM2P.11	PWM2P.10	PWM2P.9	PWM2P.8
PWM2PL	EDH Bank1	12位PWM2周期控制低位	00000000	PWM2P.7	PWM2P.6	PWM2P.5	PWM2P.4	PWM2P.3	PWM2P.2	PWM2P.1	PWM2P.0
PWM1DH	E4H Bank1	12位PWM1占空比控制高位	----0000	-	-	-	-	PWM1D.11	PWM1D.10	PWM1D.9	PWM1D.8
PWM1DL	E5H Bank1	12位PWM1占空比控制低位	00000000	PWM1D.7	PWM1D.6	PWM1D.5	PWM1D.4	PWM1D.3	PWM1D.2	PWM1D.1	PWM1D.0
PWM2DH	E6H Bank1	12位PWM2占空比控制高位	----0000	-	-	-	-	PWM2D.11	PWM2D.10	PWM2D.9	PWM2D.8
PWM2DL	E7H Bank1	12位PWM2占空比控制低位	00000000	PWM2D.7	PWM2D.6	PWM2D.5	PWM2D.4	PWM2D.3	PWM2D.2	PWM2D.1	PWM2D.0

注意: -:保留位。



SFR映像图

Bank0

	可位寻址	不可位寻址							
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8H	TWICON	TWISTA	TWIADR	IB_OFFSET	IB_DATA	TWIDAT			FFH
F0H	B	AUXC	IB_CON1	IB_CON2	IB_CON3	IB_CON4	IB_CON5	XPAGE	F7H
E8H	EXF0	P0PCR	P1PCR	P2PCR	P3PCR			P1OS	EFH
E0H	ACC	P0CR	P1CR	P2CR	P3CR		TKCOUNT		E7H
D8H	EXF1	DISPCON	SEG01	SEG02	DISPCLK	LEDCOM	DISCOM	LIGHTCOM	DFH
D0H	PSW	EXF3						SHARECON	D7H
C8H	T2CON	T2MOD	RCAP2L	RCAP2H	TL2	TH2	TKVREF	TKST	CFH
C0H	HLVCON	TKF0	TKU1	TKDIV01	TKDIV02	TKDIV03	TKDIV04	SBRTH	C7H
B8H	IPL0	IPL1	IENC	IENC1	IENC2	BUZCON	SFINE	SBRTL	BFH
B0H	P3	RSTSTAT	CLKCON	LPDCON	IPH0	IPH1	TKRANDOM	TKW	B7H
A8H	IEN0	IEN1	ISPLO	ISPCON	CRCCON	CRCDL	CRCDH	BUZP	AFH
A0H	P2	TKCON1	TKWAIT		EXCON			FLASHCON	A7H
98H	SCON	SBUF	SADDR	SADEN		P2SS			9FH
90H	P1	ADC2H		ADCON	ADT	ADC1H	ADDL	ADDH	97H
88H	TCON	TMOD	TL0	TH0	TL1	TH1	SUSLO	TCON1	8FH
80H	P0	SP	DPL	DPH	DPL1	DPH1	INSCON	PCON	87H
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	

Bank1

	可位寻址	不可位寻址							
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8H									FFH
F0H	B	AUXC						XPAGE	F7H
E8H	PWM1CON	PWM2CON	PWM1PH	PWM1PL	PWM2PH	PWM2PL			EFH
E0H	ACC				PWM1DH	PWM1DL	PWM2DH	PWM2DL	E7H
D8H									DFH
D0H	PSW								D7H
C8H	T4CON				TL4	TH4			CFH
C0H	T3CON		TL3	TH3					C7H
B8H	IPL0	IPL1							BFH
B0H					IPH0	IPH1			B7H
A8H	IEN0	IEN1							AFH
A0H									A7H
98H									9FH
90H	P4	P4CR	P4PCR						97H
88H							SUSLO		8FH
80H		SP	DPL	DPH	DPL1	DPH1	INSCON	PCON	87H
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	

注意：未使用的SFR地址禁止读写。



## 7. 标准功能

### 7.1 CPU

#### 7.1.1 CPU内核特殊功能寄存器

##### 特性

- CPU内核寄存器: ACC, B, PSW, SP, DPL, DPH

##### 累加器

累加器ACC是一个常用的专用寄存器，指令系统中采用A作为累加器的助记符。

##### B寄存器

在乘除法指令中，会用到B寄存器。在其它指令中，B寄存器可作为暂存器来使用。

##### 栈指针 (SP)

栈指针SP是一个8位专用寄存器，在执行PUSH、各种子程序调用、中断响应等指令时，SP先加1，再将数据压栈；执行POP、RET、RETI等指令时，数据退出堆栈后SP再减1。堆栈栈顶可以是片上内部RAM (00H-FFH) 的任意地址，系统复位后，SP初始化为07H，使得堆栈事实上由08H地址开始。

##### 程序状态字 (PSW) 寄存器

程序状态字 (PSW) 寄存器包含了程序状态信息。

##### 数据指针 (DPTR)

数据指针DPTR是一个16位专用寄存器，其高位字节寄存器用DPH表示，低位字节寄存器用DPL表示。它们既可以作为一个16位寄存器DPTR来处理，也可以作为2个独立的8位寄存器DPH和DPL来处理。

Table 7.1 PSW寄存器

D0H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PSW	CY	AC	F0	RS1	RS0	OV	F1	P
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	CY	<b>进位标志位</b> 0: 算术或逻辑运算中，没有进位或借位发生 1: 算术或逻辑运算中，有进位或借位发生
6	AC	<b>辅助进位标志位</b> 0: 算数逻辑运算中，没有辅助进位或借位发生 1: 算数逻辑运算中，有辅助进位或借位发生
5	F0	<b>F0标志位</b> 用户自定义标志位
4-3	RS[1:0]	<b>R0-R7寄存器页选择位</b> 00: 页0 (映射到00H-07H) 01: 页1 (映射到08H-0FH) 10: 页2 (映射到10H-17H) 11: 页3 (映射到18H-1FH)
2	OV	<b>溢出标志位</b> 0: 没有溢出发生 1: 有溢出发生
1	F1	<b>F1标志位</b> 用户自定义标志位
0	P	<b>奇偶校验位</b> 0: 累加器A中值为1的位数为偶数 1: 累加器A中值为1的位数为奇数



### 7.1.2 CPU增强内核特殊功能寄存器

- 扩展的'MUL'和'DIV'指令：16位\*8位，16位/8位
- 双数据指针
- CPU增强内核寄存器：AUXC, DPL1, DPH1, INSCON

SH79F6412扩展了'MUL'和'DIV'的指令，使用一个新寄存器-AUXC寄存器保存运算数据的高8位，以实现16位运算。在16位乘法指令中，会用到AUXC寄存器。在其它指令中，AUXC寄存器可作为暂存器来使用。

CPU在复位后进入标准模式，'MUL'和'DIV'的指令操作和标准8051指令操作一致。当INSCON寄存器的相应位置1后，'MUL'和'DIV'指令的16位操作功能被打开。

	操作		结果		
			A	B	AUXC
MUL	INSCON.2 = 0; 8位模式	(A)*(B)	低位字节	高位字节	---
	INSCON.2 = 1; 16位模式	(AUXC A)*(B)	低位字节	中位字节	高位字节
DIV	INSCON.3 = 0; 8位模式	(A)/(B)	商低位字节	余数	---
	INSCON.3 = 1; 16位模式	(AUXC A)/(B)	商低位字节	余数	商高位字节

#### 双数据指针

使用双数据指针能加速数据存储移动。标准数据指针被命名为DPTR而新型数据指针命名为DPTR1。

数据指针DPTR1与DPTR类似，是一个16位专用寄存器，其高位字节寄存器用DPH1表示，低位字节寄存器用DPL1表示。它们既可以作为一个16位寄存器DPTR1来处理，也可以作为2个独立的8位寄存器DPH1和DPL1来处理。

通过对INSCON寄存器中的DPS位置1或清0选择两个数据指针中的一个。所有读取或操作DPTR的相关指令将会选择最近一次选择的数据指针。

### 7.1.3 寄存器

Table 7.2 数据指针选择寄存器

86H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
INSCON	-	BKS0	-	-	DIV	MUL	-	DPS
读/写	-	读/写	-	-	读/写	读/写	-	读/写
复位值 (POR/WDT/LVR/PIN)	-	0	-	-	0	0	-	0

位编号	位符号	说明
6	BKS0	特殊功能寄存器页选择位 0: 选择特殊功能寄存器页0 1: 选择特殊功能寄存器页1
3	DIV	16位/8位除法选择位 0: 8位除法 1: 16位除法
2	MUL	16位/8位乘法选择位 0: 8位乘法 1: 16位乘法
0	DPS	数据指针选择位 0: 数据指针 1: 数据指针1



## 7.2 随机数据存储器 (RAM)

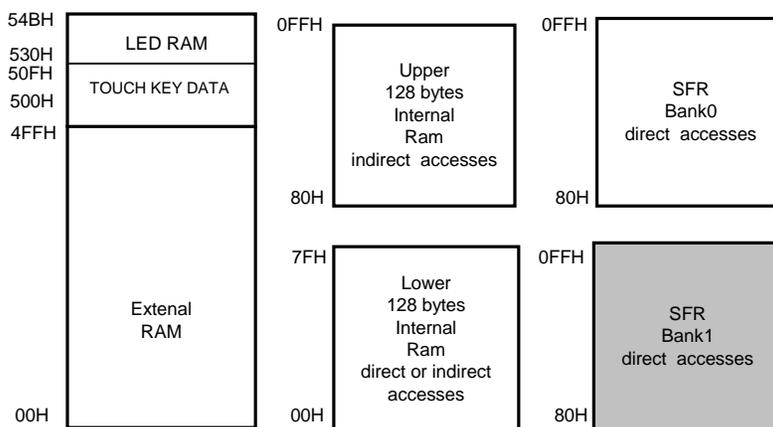
### 7.2.1 特性

SH79F6412为数据存储提供了内部RAM和外部RAM。下列为存储器空间分配:

- 低位128字节的RAM (地址从00H到7FH) 可直接或间接寻址。
- 高位128字节的RAM (地址从80H到FFH) 只能间接寻址。
- 特殊功能寄存器 (SFR, 地址从80H到FFH) 只能直接寻址。
- 外部RAM可通过MOVX指令间接访问。

高位128字节的RAM占用的地址空间和SFR相同,但在物理上与SFR的空间是分离的。当一个指令访问高于地址7FH的内部位置时,CPU可以根据访问的指令类型来区分是访问高位128字节数据RAM还是访问SFR。

SH79F6412在外部数据空间额外提供了1280字节RAM,支持高级语言。SH79F6412还配置了28字节的LED RAM和16字节的触摸按钮RAM。



内部和外部RAM配置

SH79F6412支持传统的访问外部RAM方法。使用MOVX A, @Ri或MOVX@Ri, A; 来访问外部低256字节RAM; 使用MOVX A, @DPTR或MOVX@DPTR, A来访问外部1280字节RAM。

用户也能用XPAGE寄存器来访问外部RAM, 仅用MOVXA, @Ri或MOVX@Ri, A指令即可。用户能用XPAGE来表示高于256字节的RAM地址。

在Flash SSP模式下, XPAGE也能用作分段选择器 (详见SSP章节)。

### 7.2.2 寄存器

Table 7.3 数据存储页寄存器 (XPAGE)

F7H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
XPAGE	-	XPAGE.6	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
读/写	-	读/写						
复位值 (POR/WDT/LVR/PIN)	-	0	0	0	0	0	0	0

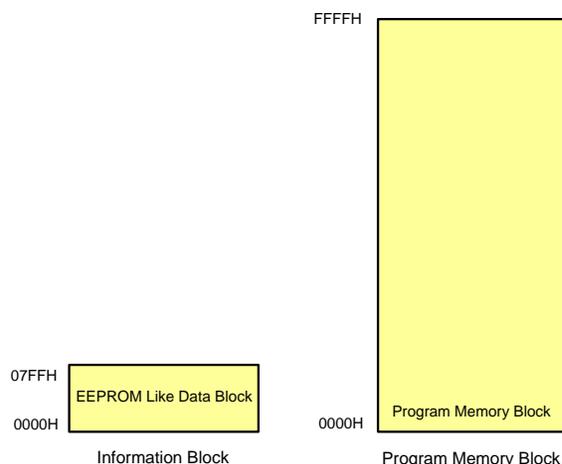
位编号	位符号	说明
6-0	XPAGE[6:0]	RAM页选择控制位



## 7.3 Flash程序存储器

### 7.3.1 特性

- Flash存储器包括64 X 1KByte，总共64KB
- 集成类EEPROM存储器8 X 256B，总共2KB
- 在工作电压范围内都能进行编程和擦除操作
- 支持4种代码保护模式
- 在线编程（ICP）操作支持写入、读取和擦除操作
- 支持整体/扇区擦除和编程
- 编程/擦除次数：程序区：至少10,000次  
类EEPROM区：至少100,000次
- 数据保存年限：至少10年
- 低功耗



SH79F6412为存储程序代码内置64K可编程Flash程序存储区（Program Memory Block），支持在线编程（ICP）模式和扇区自编程（SSP）模式对Flash存储器操作。每个扇区1024字节。

SH79F6412还内置2048字节的类EEPROM存储区用于存放用户数据。每个扇区256字节，总共8个扇区。

Flash操作定义：

在线编程（ICP）模式：通过Flash编程器对Flash存储器进行擦、读、写操作。

扇区自编程（SSP）模式：用户程序代码在Flash代码区中运行，对Flash存储器（包括Flash代码区和类EEPROM区）进行擦、读、写操作，但无法擦除代码自身所在的扇区。

**Flash存储器支持以下操作：**

#### (1) 代码保护控制模式编程

SH79F6412的代码保护功能为用户代码提供了高性能的安全措施。共提供4种保护模式。

代码保护模式0：对烧写器加密，允许/禁止任何编程器的写入/读取操作（不包括整体擦除），以4K（4个扇区）为单位，可以分开保护。

代码保护模式1：对MOVC指令加密，允许/禁止在其它扇区中通过MOVC指令进行读取操作，或通过SSP模式进行擦除/写入操作，以4K（4个扇区）为单位，可以分开保护。

代码保护模式2：SSP功能允许/禁止控制，选中后，芯片对code区域的SSP操作（擦除或者写入，不包括读取）是禁止的，但是不会禁止芯片对类EEPROM的操作。

代码保护模式3：客户密码保护，可由客户自设密码，密码由6字节组成。如果将此功能开启，表示在烧写器或仿真器工具对芯片做任何操作（读出，写入，擦除或者仿真）之前先输入这个密码，如果这个密码正确，则芯片允许烧写器或仿真器工具进行相应的操作，反之则报错，无法执行相应操作。

用户必须使用下列方式才能完成代码保护控制模式的设定：

1. Flash编程器在ICP模式设置相应的保护位，以进入所需的保护模式。
2. SSP模式不支持代码保护控制模式编程。



## (2) 整体擦除

无论代码保护控制模式的状态如何，整体擦除操作都将会擦除所有程序，代码选项，代码保护位，但是不会擦除类EEPROM存储区。

用户必须使用下列方式才能完成整体擦除：

1. Flash编程器在ICP模式发出整体擦除指令，进行整体擦除。
2. SSP模式不支持整体擦除。

## (3) 扇区擦除

扇区擦除操作将会擦除所选扇区中内容。用户程序（SSP）和Flash编程器都能执行该操作。

若需用户程序执行该操作，必须禁止所选扇区的代码保护控制模式1和保护模式2。

若需编程器执行该操作，必须禁止所选扇区的代码保护控制模式0，如果保护模式3使能，必须输入正确密码。

用户必须使用下列2种方式之一才能完成扇区擦除：

1. Flash编程器在ICP模式发出扇区擦除指令，进行扇区擦除。
2. 通过SSP功能发出扇区擦除指令，进行扇区擦除（详见在扇区自编程章节）。

## (4) 类EEPROM存储区擦除

类EEPROM存储区擦除操作将会擦除类EEPROM存储区中的内容。用户程序（SSP）和Flash编程器都能执行该操作。

用户必须使用下列2种方式之一才能完成类EEPROM存储区擦除：

1. Flash编程器在ICP模式发出类EEPROM存储区擦除指令，进行类EEPROM存储区擦除。
2. 通过SSP功能发出类EEPROM存储区擦除指令，进行类EEPROM存储区擦除（详见在扇区自编程章节）。

## (5) 写/读代码

读/写代码操作可以将代码从Flash存储器中读出或写入。用户程序（SSP）和编程器都能执行该操作。

若需用户程序执行读代码操作，必须禁止所选扇区的代码保护控制模式1。但不管保护位如何设置，用户程序都能读程序自身所在扇区（1K为单位）。

若需用户程序执行写代码操作，必须禁止所选扇区的代码保护控制模式1和代码保护模式2。

**注意：**若只使能扇区的代码保护控制模式1，用户程序不能写其它扇区，但能写程序自身所在扇区（1K为单位），若需编程器执行该操作，必须禁止所选扇区的代码保护控制模式0。

用户必须使用下列2种方式之一才能完成写/读代码：

1. Flash编程器在ICP模式发出写/读代码指令，进行写/读代码。
2. 通过SSP功能发出写代码指令，进行写代码操作；通过MOVC指令执行读代码操作。

## (6) 写/读类EEPROM存储区

读/写类EEPROM存储区操作可以将数据从类EEPROM存储区中读出或写入。用户程序（SSP）和Flash编程器都能执行该操作。

用户必须使用下列2种方式之一才能完成写/读类EEPROM存储区：

1. Flash编程器在ICP模式发出写/读类EEPROM存储区指令，进行写/读类EEPROM存储区。
2. 通过SSP功能发出写类EEPROM存储区指令，进行写类EEPROM操作；通过MOVC指令执行读类EEPROM操作。

### Flash存储器操作汇总

操作	ICP	SSP
代码保护	支持	不支持
扇区擦除	支持（无安全位）	支持（无安全位）
整体擦除	支持	不支持
类EEPROM存储区擦除	支持	支持
写/读代码	支持（无安全位）	支持（无安全位）
读/写类EEPROM存储区	支持	支持



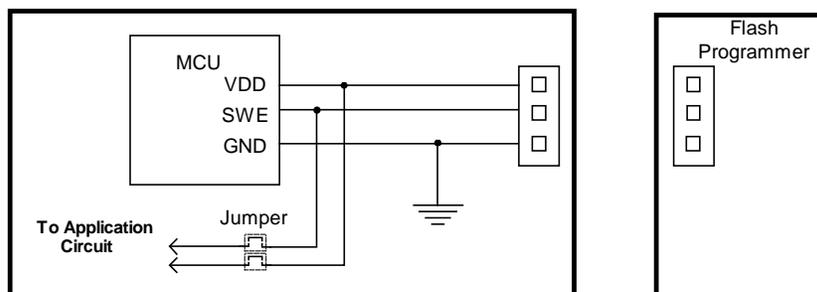
### 7.3.2 ICP模式下的Flash操作

#### 单线仿真模式：

ICP模式为通过Flash编程器对MCU进行编程，可以在MCU焊在用户板上以后编程。ICP模式下，用户系统必须关机后Flash编程器才能通过ICP编程接口刷新Flash存储器。ICP编程接口包括3个引脚（V<sub>DD</sub>，GND，SWE）。

编程器使用1个JTAG引脚（SWE）进入编程模式。只有将特定波形输入1个引脚后，CPU才能进入编程模式。如需详细说明请参考Flash编程器用户指南。

在ICP模式中，通过3线接口编程器能完成所有Flash操作。因为编程信号非常敏感，所以使用编程器编程时用户需要先用3个跳线将芯片的编程引脚（V<sub>DD</sub>，GND，SWE）从应用电路中分离出来，如下图所示。



当采用ICP模式进行操作时，建议按照如下步骤进行操作：

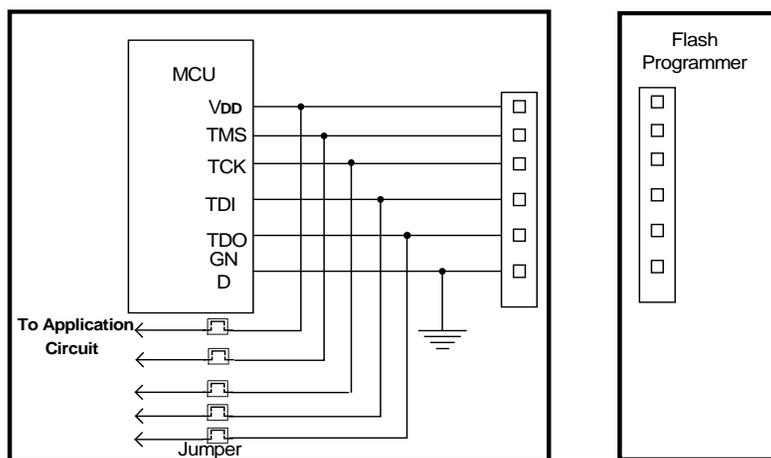
- (1) 在开始编程前断开跳线（jumper），从应用电路中分离编程引脚；
- (2) 将芯片编程引脚连接至Flash编程器编程接口，开始编程；
- (3) 编程结束后断开Flash编程器接口，连接跳线恢复应用电路。

#### 四线仿真模式：

ICP模式为通过Flash编程器对MCU进行编程，可以在MCU焊在用户板上以后编程。ICP模式下，用户系统必须关机后Flash编程器才能通过ICP编程接口刷新Flash存储器。ICP编程接口包括6个引脚（V<sub>DD</sub>，GND，TCK，TDI，TMS，TDO）。

编程器使用4个JTAG引脚（TDO，TDI，TCK，TMS）进入编程模式。只有将特定波形输入4个引脚后，CPU才能进入编程模式。如需详细说明请参考Flash编程器用户指南。

在ICP模式中，通过6线接口编程器能完成所有Flash操作。因为编程信号非常敏感，所以使用编程器编程时用户需要先用6个跳线将芯片的编程引脚（V<sub>DD</sub>，GND，TCK，TDI，TMS，TDO）从应用电路中分离出来，如下图所示。



当采用ICP模式进行操作时，建议按照如下步骤进行操作：

- (1) 在开始编程前断开跳线（jumper），从应用电路中分离编程引脚；
- (2) 将芯片编程引脚连接至编程器编程接口，开始编程；
- (3) 编程结束后断开编程器接口，连接跳线恢复应用电路。



## 7.4 SSP功能

## 7.4.1 寄存器

## (1) 擦除/编程用扇区选择和编程用地址偏移量寄存器

此寄存器用来选择待擦除或者待编程扇区的区号，配合IB\_OFFSET寄存器来表示待编程字节在扇区内的地址偏移量。

对于程序存储区，一个扇区为1024字节，寄存器定义如下：

Table 7.4 擦除/编程用扇区选择和地址偏移寄存器

F7H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
XPAGE	XPAGE.7	XPAGE.6	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-2	XPAGE[7:2]	被擦除/编程的存储单元扇区号（也即存储单元的最高4位地址），0000代表扇区0，依此类推
1-0	XPAGE[1:0]	被擦除/编程的存储单元次高2位地址

Table 7.5 编程用地址偏移寄存器

FBH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_OFFSET	IB_OFF SET.7	IB_OFF SET.6	IB_OFF SET.5	IB_OFF SET.4	IB_OFF SET.3	IB_OFF SET.2	IB_OFF SET.1	IB_OFF SET.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	IB_OFFSET[7:0]	被编程的存储单元低8位地址

XPAGE[1:0]和IB\_OFFSET[7:0]共10位，可以表示1个程序存储扇区内全部1024个字节的偏移量。

类EEPROM存储区，一个扇区为256字节，共8个扇区，寄存器定义如下：

Table 7.6 擦除/编程用扇区选择寄存器

F7H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
XPAGE	XPAGE.7	XPAGE.6	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-3	XPAGE[7:3]	在擦除/编程扇区时无意义
2-0	XPAGE[2:0]	被擦除/编程的扇区选择位 000: 扇区0 001: 扇区1 ... 111: 扇区7

类EEPROM块区的访问可通过指令“MOVC A, @A+DPTR”或“MOVC A, @A+PC”实现。

注意：需要将FLASHCON寄存器中的FAC位置1。



Table 7.7 编程用地址偏移寄存器

FBH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_OFFSET	IB_OFF SET.7	IB_OFF SET.6	IB_OFF SET.5	IB_OFF SET.4	IB_OFF SET.3	IB_OFF SET.2	IB_OFF SET.1	IB_OFF SET.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7-0	IB_OFFSET[7:0]	被擦除/编程的块单元地址						

IB\_OFFSET[7:0]共8位，可以表示1个块区内全部256个字节的偏移量。

(2) 编程用数据寄存器

Table 7.8 编程用数据寄存器

FCH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_DATA	IB_DATA.7	IB_DATA.6	IB_DATA.5	IB_DATA.4	IB_DATA.3	IB_DATA.2	IB_DATA.1	IB_DATA.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7-0	IB_DATA[7:0]	待编程数据						

(3) 操作模式选择寄存器

Table 7.9 SSP操作模式选择寄存器

F2H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_CON1	IB_CON1.7	IB_CON1.6	IB_CON1.5	IB_CON1.4	IB_CON1.3	IB_CON1.2	IB_CON1.1	IB_CON1.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7-0	IB_CON1[7:0]	<b>SSP操作选择位</b> 0xE6: 扇区擦除 0x6E: 存储单元编程						

(4) SSP流程控制寄存器

Table 7.10 SSP流程控制寄存器1

F3H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_CON2	-	-	-	-	IB_CON2.3	IB_CON2.2	IB_CON2.1	IB_CON2.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0
位编号	位符号	说明						
3-0	IB_CON2[3:0]	必须为05H，否则Flash编程将会终止						



Table 7.11 SSP流程控制寄存器2

F4H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_CON3	-	-	-	-	IB_CON3.3	IB_CON3.2	IB_CON3.1	IB_CON3.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

位编号	位符号	说明
3-0	IB_CON3[3:0]	必须为0AH, 否则Flash编程将会终止

Table 7.12 SSP流程控制寄存器3

F5H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_CON4	-	-	-	-	IB_CON4.3	IB_CON4.2	IB_CON4.1	IB_CON4.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

位编号	位符号	说明
3-0	IB_CON4[3:0]	必须为09H, 否则Flash编程将会终止

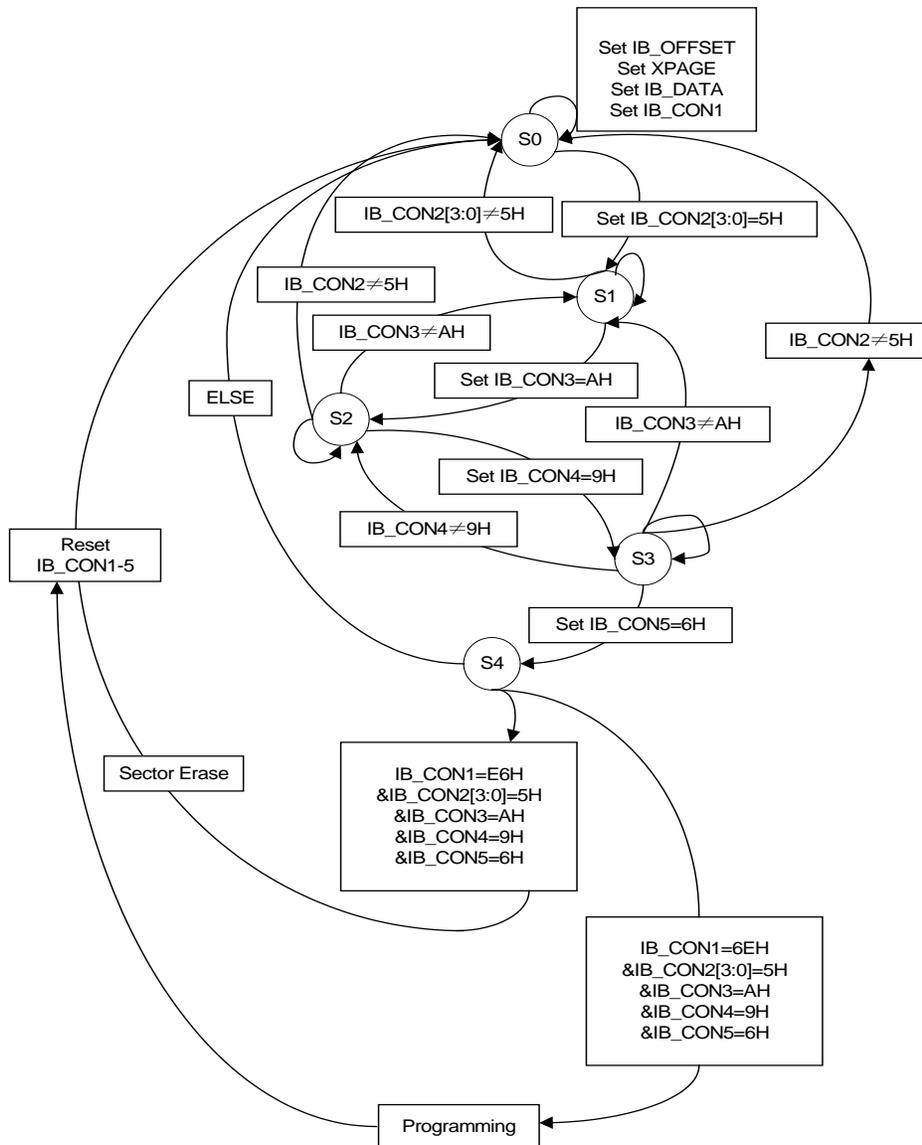
Table 7.13 SSP流程控制寄存器4

F6H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IB_CON5	-	-	-	-	IB_CON5.3	IB_CON5.2	IB_CON5.1	IB_CON5.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

位编号	位符号	说明
3-0	IB_CON5[3:0]	必须为06H, 否则Flash编程将会终止



7.4.2 Flash控制流程图





### 7.4.3 SSP编程注意事项

为确保顺利完成SSP编程，用户软件必须按以下步骤设置：

**(1) 对主程序区烧写：**（注：需关闭代码保护模式1和模式2）

1. 关闭中断；
2. 按相应的待编程扇区号设置XPAGE，IB\_OFFSET；
3. 按编程需要，设置IB\_DATA；
4. 按照顺序设置IB\_CON1 - 5；
5. 添加4个NOP指令；
6. 开始编程，CPU将进入IDLE模式；编程完成后自动退出IDLE模式；
7. 如需继续写入数据，跳转至第2步；
8. XPAGE寄存器清0，恢复中断设置。

**(2) 对主程序区扇区擦除：**（注：需关闭代码保护模式1和模式2）

1. 关闭中断；
2. 按相应的扇区设置XPAGE；
3. 按照顺序设置IB\_CON1 - 5；
4. 添加4个NOP指令；
5. 开始擦除，CPU将进入IDLE模式；擦除完成后自动退出IDLE模式；
6. 更多扇区擦除操作跳转至第2步；
7. 清除XPAGE，恢复中断设置。

**(3) 读取：**

使用“MOVC A, @A+DPTR”或“MOVC A, @A+PC”。

**(4) 对类EEPROM区域擦除烧写动作：**（注：此功能不受代码保护模式控制）

对于类EEPROM的操作类似于主程序区Flash的操作，即类似上述(1)/(2)/(3)部分的描述。区别在于：

1. 在对类EEPROM进行擦除、写或读之前，应首先将FLASHCON寄存器的最低位FAC位置1。
2. 类EEPROM的扇区为256字节，而不是1024字节。

**注意：**当不需对类EEPROM操作时，必须将FAC位清0

### 7.4.4 可读识别码

SH79F6412每颗芯片出厂后都固化有一个40位的可读识别码，它的值为0 - 0xffffffff的随机值，它是无法擦除的，可以由程序或编程工具读出。

读识别码时，首先，设FAC位为1，然后给DPTR赋值“0127BH - 127FH”，将A清0，再使用“MOVC A, @A+DPTR”来读取。

**注意：**读完识别码后必须将FAC位清0，否则会影响用户程序读代码区的指令执行。

FLASHCON寄存器的描述如下：

Table 7.14 访问控制寄存器

A7H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
FLASHCON	-	-	-	-	-	-	-	FAC
读/写	-	-	-	-	-	-	-	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	-	-	0

位编号	位符号	说明
0	FAC	访问控制 0: MOVC指令或者SSP功能访问主程序区域 1: MOVC指令或者SSP功能访问类EEPROM区域



## 7.5 系统时钟和振荡器

### 7.5.1 特性

- 支持4种振荡器类型：晶体谐振器，陶瓷谐振器和内部16.6M/128K RC振荡器，32.768kHz晶体谐振器
- 2个振荡器引脚（XTAL1，XTAL2）从4种振荡器类型中产生1种或者2种时钟
- 内建16.6MHz RC振荡器常温（±1%）RC振荡器，全温度范围（±2%）
- 内建系统时钟分频器

### 7.5.2 时钟定义

SH79F6412几个内部时钟定义如下：

**OSCCLK:** 从4个可选振荡器类型中（晶体谐振器和陶瓷谐振器以及内部16.6M RC/128K振荡器）选中的那个振荡器的时钟。 $f_{osc}$ 定义为OSCCLK的频率。 $t_{osc}$ 定义为OSCCLK的周期。

**OSC1CLK:** 从3个可选振荡器类型中（从XTALX输入的晶体谐振器和陶瓷谐振器以及内部16.6M RC振荡器）选中的那个振荡器的时钟。 $f_{osc1}$ 定义为OSC1CLK的频率。 $t_{osc1}$ 定义为OSC1CLK的周期。

**注意:** 当代码选项OP\_OSC不是0011, 0110, 1010时（128K振荡器与32.68K晶体振荡器没有被选中，详见代码选项章节），OSC1CLK不存在。

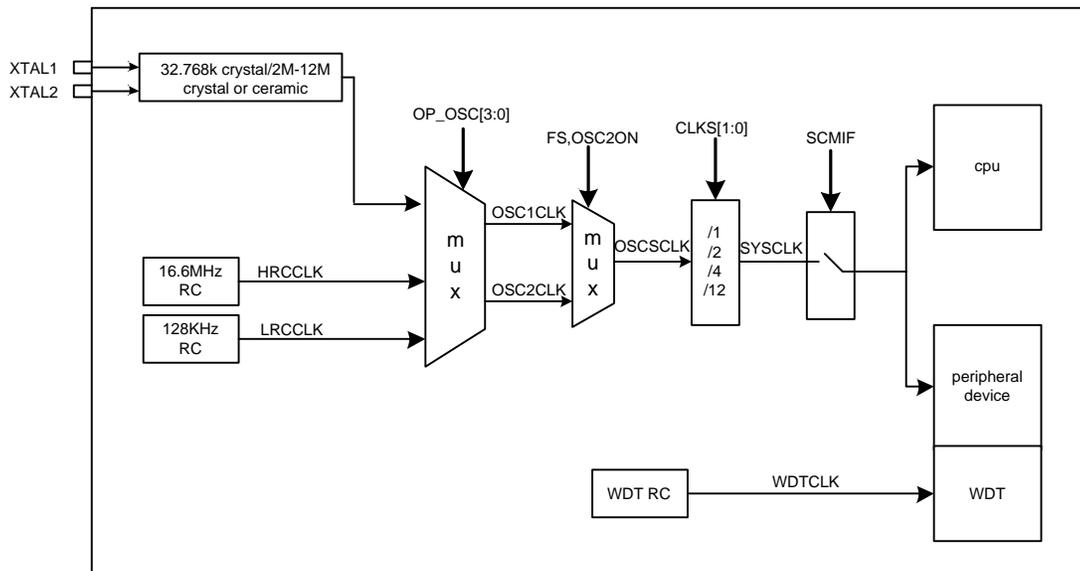
**WDTCLK:** 内部看门狗RC振荡器时钟。 $f_{wdt}$ 定义为WDTCLK的频率。 $t_{wdt}$ 定义为WDTCLK的周期。

**OSCSCLK:** 系统时钟频率分频器的输入时钟。这个时钟可能为OSCCLK或者OSC1CLK。 $f_{oscs}$ 定义为OSCSCLK的频率。 $t_{oscs}$ 定义为OSCSCLK的周期。

**SYSCLK:** 系统时钟，系统频率分频器的输出时钟。这个时钟为CPU指令周期的时钟。 $f_{sys}$ 定义为SYSCLK的频率。 $t_{sys}$ 定义为SYSCLK的周期。

### 7.5.3 概述

SH79F6412支持3种振荡器类型：晶体谐振器（2MHz-12MHz）（32.768KHZ），陶瓷谐振器（2MHz-12MHz）和内部RC振荡器（16.6MHz，128K）。振荡器类型的选择由代码选项OP\_OSC决定（详见代码选项章节）。SH79F6412有2个振荡器引脚（XTAL1，XTAL2），可以从5种振荡器类型中产生1种或者2种时钟。这些都是由代码选项OP\_OSC决定（详见代码选项章节）。由振荡器产生的基本时钟脉冲提供系统时钟支持CPU及片上外围设备。





## 7.5.4 寄存器

Table 7.15 系统时钟控制寄存器

B2H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
CLKCON	32k_SPDUP	CLKS1	CLKS0	SCMIF	HFON	FS	-	-
读/写	读/写	读/写	读/写	读	读/写	读/写	-	-
复位值 (POR/WDT/LVR/PIN)	1	1	1	0	0	0	-	-

位编号	位符号	说明
7	32k_SPDUP	<p><b>32.768kHz晶体谐振器加速模式控制位</b></p> <p>0: 32.768kHz振荡器常规模式，由软件清0。 1: 32.768kHz振荡器加速模式，由软件或者硬件置1。</p> <p>此位在系统发生任何形式的复位，如上电复位，看门狗复位等时，自动由硬件设置1，用以加速32.768kHz振荡器起振，缩短32.768kHz振荡器的起振时间。如果有需要，本位也可以由软件置1或者清0。比如进入掉电模式（Power-down mode）前，可以将此位置1，掉电模式唤醒后再由软件清0。</p> <p>应该注意的是关闭32.768kHz加速模式（此位清0），可以节省系统的耗电。只有代码选项OP_OSC为0111时（选择32.768kHz晶体振荡器，详见代码选项章节），此控制位才有效。</p>
6-5	CLKS[1:0]	<p><b>系统时钟频率分频器</b></p> <p>00: <math>f_{sys} = f_{osc}</math> 01: <math>f_{sys} = f_{osc}/2</math> 10: <math>f_{sys} = f_{osc}/4</math> 11: <math>f_{sys} = f_{osc}/12</math></p> <p>如果选择32kHz crystal振荡器为OSCCLK，此控制位无效。</p>
3	HFON	<p><b>OSC1CLK开关控制寄存器</b></p> <p>0: 关闭OSC1CLK 1: 打开OSC1CLK</p> <p>只有代码选项OP_OSC为0011, 0110, 1010时（128K内部RC与32.768K晶体振荡器，详见代码选项章节），此控制位才有效。</p>
2	FS	<p><b>频率选择位</b></p> <p>0: 选择128KHz或32.768KHz为OSCCLK 1: 选择OSC1CLK为OSCCLK</p> <p>只有代码选项OP_OSC为0011, 0110, 1010时（128K内部RC与32.768K晶体振荡器，详见代码选项章节），此控制位才有效。</p>

**注意:**

(1) 当代码选项OP\_OSC为0000时，OSC1CLK为内建16.6MHzRC；当代码选项OP\_OSC为1110,1010时，OSC1CLK为从XTALX输入的晶体谐振器或陶瓷谐振器。

(2) 当代码选项OP\_OSC为0011, 0110, 1010时，HFON和FS才有效。

(3) 当OSC1CLK作为OSCCLK时（也就是说，HFON = 1和FS = 1），HFON不能软件清0。

(4) 当OSCCLK从128K或32.768K切换到OSC1CLK时，假如当时OSC1CLK为关闭状态，则必须按以下步骤依次设置：

- a. 设置HFON = 1，打开OSC1CLK
- b. 至少等待振荡器预热时间（详见振荡器预热章节）
- c. 设置FS = 1，选择OSC1CLK作为OSCCLK

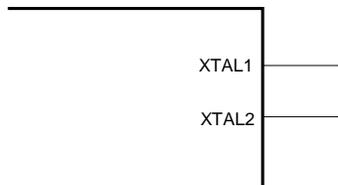
(5) 当OSCCLK从OSC1CLK切回到128K或32.768K时，则必须按以下步骤依次设置：

- a. FS位清0，选择128K或32.768K作为OSCCLK
- b. 添加1个NOP指令
- c. HFON位清0（降低功耗）

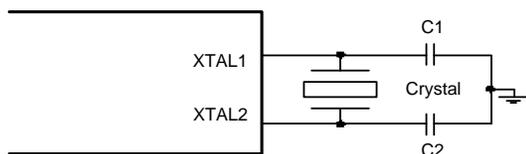


### 7.5.5 振荡器类型

(1) OP\_OSC = 0000, 0011: 内部RC振荡器, XTAL1,2引脚与I/O共用



(2) OP\_OSC = 0110,1010,1110: 从XTAL1输入2M - 12M或32.768K晶体/陶瓷谐振器



### 7.5.6 谐振器负载电容选择

陶瓷谐振器		
频率	C1	C2
3.58MHz	-	-
4MHz	-	-

晶体谐振器		
频率	C1	C2
4MHz	8 - 15pF	8 - 15pF
12MHz	8 - 15pF	8 - 15pF
32.768KHz	10 - 12pF	10 - 12pF

**注意:**

(1) 表中负载电容仅供参考!

(2) 以上电容值可通过谐振器基本的起振和运行测试, 并非最优值。

(3) 请注意印制板上的杂散电容, 用户应在超过应用电压和温度的条件下测试谐振器的性能。

在应用陶瓷谐振器/晶体谐振器之前, 用户需向谐振器生产厂要求相关应用参数以获得最佳性能。

请登陆<http://www.sinowealth.com>以取得更多的推荐谐振器生产厂。



## 7.6 系统时钟监控 (SCM)

为了增强系统的可靠性，SH79F6412含有一个系统时钟监控 (SCM) 模块。如果系统时钟出现故障 (例如：外部振荡器停振等)，内建SCM模块会将OSCCLK自动切换到内部时钟2M RC，同时系统时钟监控标志位 (SCMIF) 被置1。当EA和ESCM位均被置1时，SCM模块将会产生中断。如果外部振荡器恢复工作，SCM将会切换OSCCLK到外部振荡器，然后SCMIF位自动清0。

### 注意：

SCMIF为只读寄存器，只能由硬件清0或者置1。

如果SCMIF清0，SCM将系统时钟自动切换到系统时钟出故障前的状态。

如果代码选项选择内部RC振荡器 (详见代码选项章节) 作为OSCCLK，则系统时钟监控功能不可用。

Table 7.16 系统时钟控制寄存器

B2H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
CLKCON	-	-	-	SCMIF	-	-	-	-
读/写	-	-	-	只读	-	-	-	-
复位值 (POR/WDT/LVR/PIN)	-	-	-	0	-	-	-	-

位编号	位符号	说明
4	SCMIF	系统时钟监控标志位 0: 表示系统时钟正常运行 1: 表示系统时钟故障



## 7.7 I/O端口

### 7.7.1 特性

- 30个双向I/O端口
- I/O端口可与其它功能共享

SH79F6412提供26个可编程双向I/O端口。端口数据在寄存器Px中。26个双向I/O口均有内部上拉电阻。端口控制寄存器(PxCRy)控制端口是作为输入或者输出。当端口作为输入时, 有由PxPCRy控制的内部上拉电阻(x = 0-3, y = 0-7)。

SH79F6412的有些I/O引脚能与选择功能共享。当所有功能都允许时, 在CPU中存在优先权以避免功能冲突。(详见端口共享章节)。

未使用的I/O需要设置输出(保持低电平)或者输入上拉(保持高电平), 避免引脚浮动导致的漏电及干扰。

SOP28封装的P1.3-P1.6, P3.5, P4.0-P4.3需要设置。

SOP20封装的P1.1-P1.6, P2.3, P3.0-P3.1, P3.3-P3.5, P4.0-P4.3需要设置。

### 7.7.2 寄存器

Table 7.17 端口控制寄存器

E1H - E5H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P0CR (E1H, Bank0)	P0CR.7	P0CR.6	P0CR.5	P0CR.4	P0CR.3	P0CR.2	P0CR.1	P0CR.0
P1CR (E2H, Bank0)	P1CR.7	P1CR.6	P1CR.5	P1CR.4	P1CR.3	P1CR.2	P1CR.1	P1CR.0
P2CR (E3H, Bank0)	P2CR.7	P2CR.6	P2CR.5	P2CR.4	P2CR.3	P2CR.2	P2CR.1	P2CR.0
P3CR (E4H, Bank0)	-	P3CR.6	P3CR.5	P3CR.4	P3CR.3	P3CR.2	P3CR.1	P3CR.0
P4CR (91H, Bank1)	-	-	-	-	P4CR.3	P4CR.2	P4CR.1	P4CR.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	PxCRy x = 0-4, y = 0-7	端口输入/输出控制寄存器 0: 输入模式 1: 输出模式

Table 7.18 端口上拉电阻控制寄存器

E9H - ECH, 92H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P0PCR (E9H, Bank0)	P0PCR.7	P0PCR.6	P0PCR.5	P0PCR.4	P0PCR.3	P0PCR.2	P0PCR.1	P0PCR.0
P1PCR (EAH, Bank0)	P1PCR.7	P1PCR.6	P1PCR.5	P1PCR.4	P1PCR.3	P1PCR.2	P1PCR.1	P1PCR.0
P2PCR (EBH, Bank0)	P2PCR.7	P2PCR.6	P2PCR.5	P2PCR.4	P2PCR.3	P2PCR.2	P2PCR.1	P2PCR.0
P3PCR (ECH, Bank0)	-	P3PCR.6	P3PCR.5	P3PCR.4	P3PCR.3	P3PCR.2	P3PCR.1	P3PCR.0
P4PCR (92H, Bank1)	-	-	-	-	P4PCR.3	P4PCR.2	P4PCR.1	P4PCR.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	PxPCRy x = 0-4, y = 0-7	输入端口内部上拉电阻控制 0: 内部上拉电阻关闭 1: 内部上拉电阻开启



Table 7.19 端口数据寄存器

80H - B0H, 90H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P0 (80H, Bank0)	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
P1 (90H, Bank0)	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
P2 (A0H, Bank0)	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
P3 (B0H, Bank0)	-	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
P4 (90H, Bank1)	-	-	-	-	P4.3	P4.2	P4.1	P4.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	Px.y x = 0-4, y = 0-7	端口数据寄存器

Table 7.20 端口模式选择寄存器

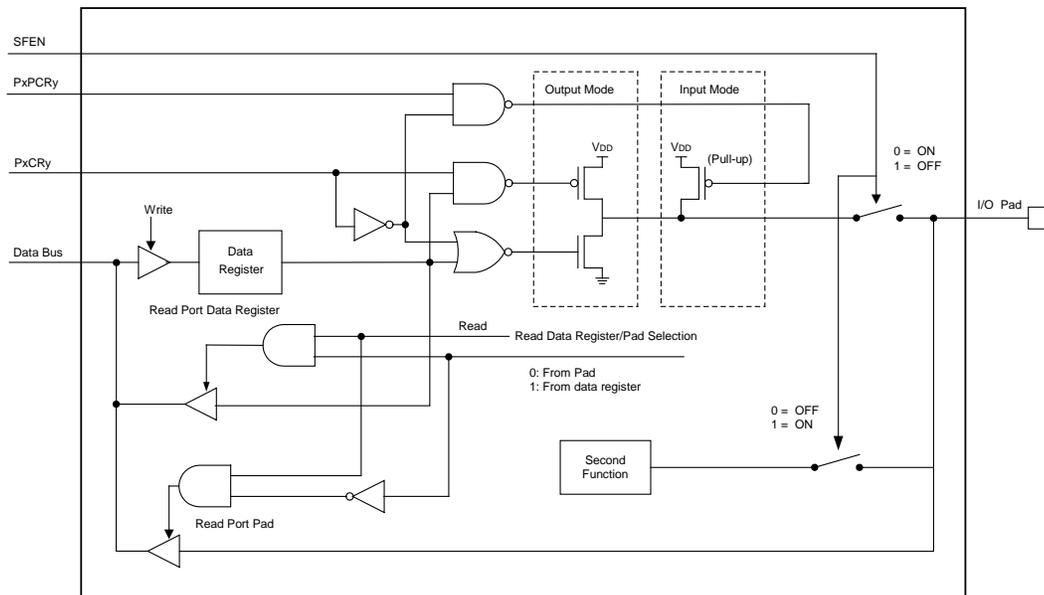
EFH, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P1OS	-	-	-	-	P1OSCR.1	P1OSCR.0	P1OS.1	P1OS.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

位编号	位符号	说明
3-2	P1OSCR.x x = 1-0	N沟道开漏输出模式选择上拉电阻 0: 开漏输出选择外部上拉电阻 1: 开漏输出选择内部10K上拉电阻
1-0	P1OS.x x = 1-0	端口0输出模式选择 0: 引脚输出模式为N沟道开漏输出 1: 引脚输出模式为CMOS推挽输出

注意: P0.7, P1.0端口作为为N-通道的开漏I/O, 但是此时端口电压不得超过VCC\_IN+0.3V。(P1OS.0为P0.7, P1OS.1为P1.0。)



## 7.7.3 端口模块图

**注意:**

- (1) 输入端口读操作直接读引脚电平。
- (2) 输出端口读操作的输入源有两种，一种是从端口数据寄存器读取，另一种是直接读引脚电平。
- (3) 用读取指令来区分：读-改-写指令是读寄存器，而其它指令读引脚电平。不管端口是否共用为其他功能，对端口写操作都是针对端口数据寄存器。



### 7.7.4 端口共享

46个双向I/O端口也能共享作为第二或第三种特殊功能。共享优先级按照外部最高内部最低的规则：

在**引脚配置图**中引脚最外边标注功能享有最高优先级，最里边标注功能享有最低优先级。这意味着一个引脚已经使用较高优先级功能（如果被允许的话），就不能用作较低优先级功能，即使较低优先级功能被允许。只有较高优先级功能由硬件或软件关闭后，相应的引脚才能用作较低优先级功能。上拉电阻也由相同规则控制。

当允许端口复用为其它功能时，用户可以修改PxCR、PxPCR（x = 0-5），但在复用的其它功能被禁止前，这些操作不会影响端口状态。

当允许端口复用为其它功能时，任何对端口的读写操作只会影响到数据寄存器的值，端口引脚值保持不变，直到复用的其它功能关闭。

#### PORT0:

- SEG14 - 15: LED SEG14 - 15显示输出（P0.5 - P0.6）
- AN1 - AN3: ADC输入通道（P0.4 - P0.6）
- COM1: LED双向COM1（P0.7）
- SDA: TWI数据线输出/输入（P0.7）
- RXD: UART数据输入（P0.7）
- INT30: 外部中断3输入（P0.7）
- BUZ1, 2: 蜂鸣器输出（P0.3, P0.6）

Table 7.21 PORT0共享列表

引脚编号	优先级	功能	允许位
12	1	P0.0	I/O
13	1	P0.1	I/O
14	1	P0.2	I/O
15	1	BUZ1	BUZCON寄存器的BZEN1位置1
	2	P0.3	无上述情况
16	1	AN1	ADCH寄存器的CH3-CH0和SCH[4:0]相应位置1
	2	P0.4	无上述情况
17	1	SEG15	SEG02中相应位置1
	2	AN2	ADCH寄存器的CH3-CH0和SCH[4:0]相应位置1
	3	P0.5	无上述情况
18	1	SEG14	SEG02中相应位置1
	2	AN3	ADCH寄存器的CH3-CH0和SCH[4:0]相应位置1
	3	BUZ2	BUZCON寄存器的BZEN2位置1
	4	P0.6	无上述情况
19	1	COM3	LEDCOM中相应位置1
	2	SDA	ENTWI = 1时，对寄存器TWIDAT进行操作
	3	RXD	SCON寄存器的REN位置1（自动上拉）
	4	INT30	IEN1寄存器的EX3位和I EXF1寄存器的EXS30位置1，且P0CR相应位置0
	5	P0.7	无上述情况



**PORT1:**

- INT31: 外部中断3输入 (P1.0)
- TXD: UART数据输出 (P1.1)
- SCK: TWI时钟线输出/输入 (P1.1)
- COM2 - 4: LED双向COM输入输出 (P1.0 - P1.2)
- INT40, INT45, INT47: 外部中断输入 (P1.1 - P1.2, P1.7)
- TCH0: 触摸按键通道0 (P1.7)
- SEG7, SEG13: LED SEG7, SEG13显示输出 (P1.2, P1.7)

**Table 7.22** PORT1共享列表

引脚编号	优先级	功能	允许位
20	1	COM2	LEDCOM中相应位置1
	2	SCK	ENTWI = 1时, 对寄存器TWIDAT进行操作
	3	TXD	对SBUF寄存器写操作
	4	INT31	IEN1寄存器的EX3位和I EXF1寄存器的EXS31位置1, 且P1CR相应位置0
	5	P1.0	无上述情况
21	1	COM3	LEDCOM中相应位置1
	2	INT47	IEN1寄存器的EX4位和I EXF1寄存器的EXS47-EXS40位置1, 且P1CR相应位置0
	3	P1.1	无上述情况
22	1	COM4	LEDCOM中相应位置1
	2	SEG13	SEG02中相应位置1
	3	INT40	IEN1寄存器的EX4位和I EXF1寄存器的EXS47-EXS40位置1, 且P1CR相应位置0
	4	P1.2	无上述情况
23	1	TCH0	P2SS中相应位置1
	2	SEG7	SEG01中相应位置1
	3	INT45	IEN1寄存器的EX4位和I EXF1寄存器的EXS47-EXS40位置1, 且P1CR相应位置0
	4	P1.7	无上述情况

**注意:** 当P1OS = 60H时, 引脚15, 16配置为开漏极端口。



**PORT2:**

- INT46: 外部中断4输入 (P2.2)
- PWM1 - PWM2: PWM输出 (P2.0 - P2.1)
- SEG0 - 6: LED显示输出通道 (P2.0 - P2.7)
- TCH0 - TCH6: 触摸按键输入通道 (P2.0 - P2.7)
- AN4 - AN10: ADC输入通道 (P2.0 - P2.7)
- C: 触摸按键电容引脚 (P2.7)

**Table 7.23** PORT2共享列表

引脚编号	优先级	功能	允许位
24	1	TCH1	<b>P2SS</b> 中相应位置1
	2	SEH6	<b>SEG01</b> 中相应位置1
	3	PWM2	<b>PWMEN</b> 寄存器的相应位位置1
	4	P2.0	无上述情况
25	1	TCH2	<b>P2SS</b> 中相应位置1
	2	SEG5	<b>SEG01</b> 中相应位置1
	3	AN4	<b>ADCH</b> 寄存器的 <b>CH3-CH0</b> 和 <b>SCH[4:0]</b> 相应位置1
	4	PWM1	<b>PWMEN</b> 寄存器的相应位位置1
	5	P2.1	无上述情况
26	1	TCH3	<b>P2SS</b> 中相应位置1
	2	SEG7	<b>SEG01</b> 中相应位置1
	3	AN5	<b>ADCH</b> 寄存器的 <b>CH3-CH0</b> 和 <b>SCH[4:0]</b> 相应位置1
	4	INT46	<b>IEN1</b> 寄存器的 <b>EX4</b> 位和 <b>EXF1</b> 寄存器的 <b>EXS47-EXS40</b> 位置1, 且 <b>P2CR</b> 相应位置0
	5	P2.2	无上述情况
27 - 28 1 - 2	1	TCH4 - TCH7	<b>P2SS</b> 中相应位置1
	2	SEG0 - SEG3	<b>SEG01</b> 中相应位置1
	3	AN6 - AN9	<b>ADCH</b> 寄存器的 <b>CH3-CH0</b> 和 <b>SCH[4:0]</b> 相应位置1
	4	P2.3 - 2.6	无上述情况
3	1	C	<b>TKCON1</b> 寄存器中的 <b>TKCON</b> 位置1
	2	COM7	<b>LEDCOM</b> 中相应位置1
	3	AN10	<b>ADCH</b> 寄存器的 <b>CH3-CH0</b> 和 <b>SCH[4:0]</b> 相应位置1
	4	P2.7	无上述情况



**PORT3:**

- T3: 定时器3外部输入 (P3.1)
- T4: 定时器4外部输入 (P3.0)
- INT2: 外部中断2输入 (P3.0)
- AN11, AN12: ADC输入通道 (P3.3, P3.6)
- INT32: 外部中断输入 (P3.6)
- AN0: ADC输入通道 (P3.4)
- SEG11, 12: LED显示输出通道 (P3.0, P3.1)
- COM7: LED双向COM输入输出 (P3.1)
- SEW: 单线仿真输入/输出引脚 (P3.2)

**Table 7.24** PORT3共享列表

引脚编号	优先级	功能	允许位
4	1	XTAL1	代码选项
	2	COM6	LEDCOM中相应位置1
	3	SEG12	SEG02中相应位置1
	4	T4	T4CON寄存器的TR4和T4CLKS位置1 (自动上拉) 或T4CON寄存器的T4CLKS位清0且TC4位置1或方式2下TR4位置1
	5	P3.0	无上述情况
5	1	XTAL2	代码选项
	2	COM5	LEDCOM中相应位置1
	3	SEG11	SEG02中相应位置1
	4	T3	T3CON寄存器的TR3位置1且T3CLKS[1:0] = 01 (自动上拉)
	5	P3.1	无上述情况
6	1	SWE	单线仿真
	2	INT2	IEN1寄存器的EX2位置1, P3.2输入模式
	3	P3.2	无上述情况
10	1	AN11	ADCH寄存器的CH3-CH0和SCH[4:0]相应位置1
	2	P3.3	无上述情况
11	1	AN0	ADCH寄存器的CH3-CH0和SCH[4:0]相应位置1
	2	P3.4	无上述情况
7	1	AN11	ADCH寄存器的CH3-CH0和SCH[4:0]相应位置1
	2	INT32	IEN1寄存器的EX3位和I EXF1寄存器的EXS32位置1, 且P3CR相应位置0
	3	P3.6	无上述情况



## 7.8 定时器

### 7.8.1 特性

- SH79F6412有5个定时器（定时器0，1，2，3，4）
- 定时器2是16位自动重载定时器
- 定时器3是16位自动重载定时器，且可以工作在掉电模式
- 定时器4是16位自动重载定时器，两个数据寄存器TH4和TL4可作为一个16位寄存器来访问

### 7.8.2 定时器0和定时器1

每个定时器的两个数据寄存器（THx & TLx（x = 0, 1））可作为一个16位寄存器来访问。它们由寄存器TCON和TMOD控制。IEN0寄存器的ET0和ET1位置1能允许定时器0和定时器1中断。（详见中断章节）。

#### 定时器x的方式（x = 0, 1）

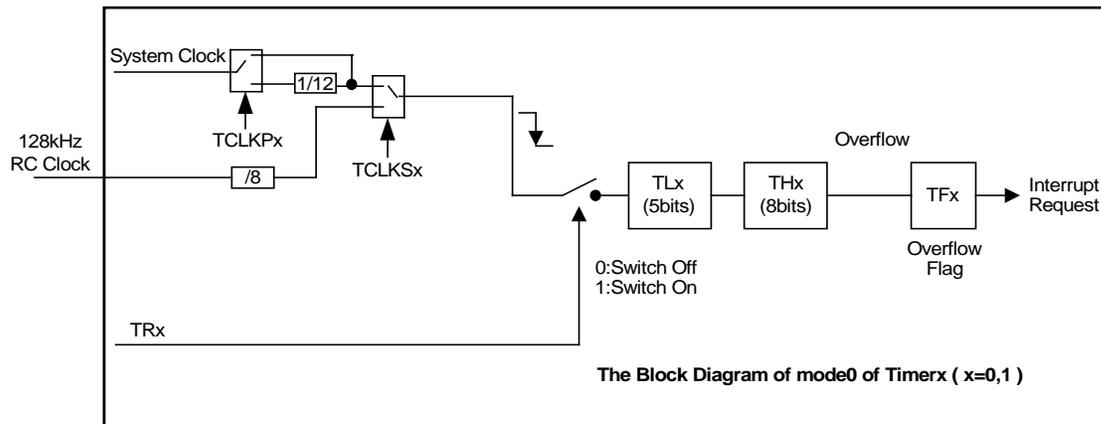
通过定时器方式寄存器（TMOD）的方式选择位Mx1-Mx0，选择定时器工作方式。

#### 方式0：13位定时器

在方式0中，定时器x为13位定时器。THx寄存器存放13位定时器的高8位，TLx存放低5位（TLx.4-TLx.0）。TLx的高三位（TLx.7-TLx.5）是不确定的，在读取时应该被忽略。当13位定时器寄存器递增，溢出时，系统置起定时器溢出标志TFx。如果定时器x中断被允许，将会产生一个中断。

当作为定时器应用时，可配置寄存器TCON1中的TCLKSx（x = 0, 1）位选择系统时钟或低频128kHz的8分频作为定时器x（x = 0, 1）的时钟源。

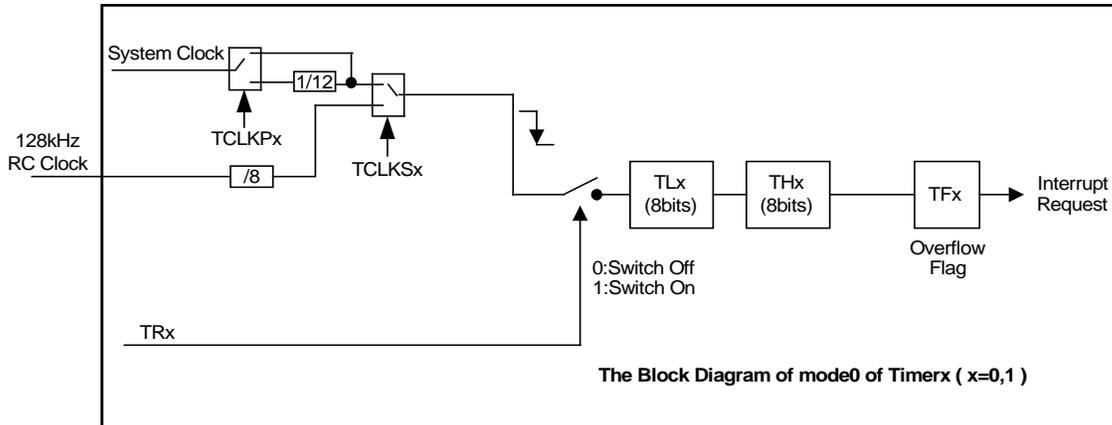
可配置寄存器TCON1中的TCLKPx（x = 0, 1）位选择系统时钟或系统时钟的1/12作为定时器x（x = 0, 1）的时钟源。





### 方式1：16位定时器

除了使用16位定时器之外，方式1的运行与方式0一致。打开和配置计数器/定时器也如同方式0。



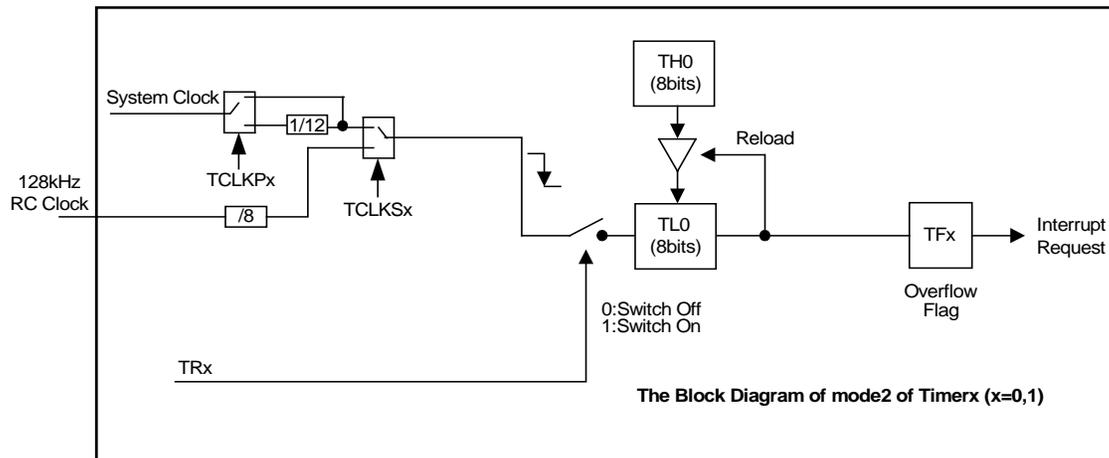
### 方式2：8位自动重载定时器

方式2中，定时器x是8位自动重载定时器。TLx存放计数值，THx存放重载值。当在TLx中的计数器溢出至0x00时，置起定时器溢出标志TFx，寄存器THx的值被重载入寄存器TLx中。如果定时器中断使能，当TFx置1时将产生一个中断。而在THx中的重载值不会改变。在允许定时器正确计数开始之前，TLx必须初始化为所需的值。

除了自动重载功能外，方式2中的定时器的使能和配置与方式1和0是一致的。

当作为定时器应用时，可配置寄存器TCON1中的TCLKSx (x = 0, 1) 位选择系统时钟或128kHz的8分频作为定时器x (x = 0, 1) 的时钟源。

可配置寄存器TCON1中的TCLKPx (x = 0, 1) 位选择系统时钟或系统时钟的1/12作为定时器x (x = 0, 1) 的时钟源。



**方式3：两个8位定时器（只限于定时器0）**

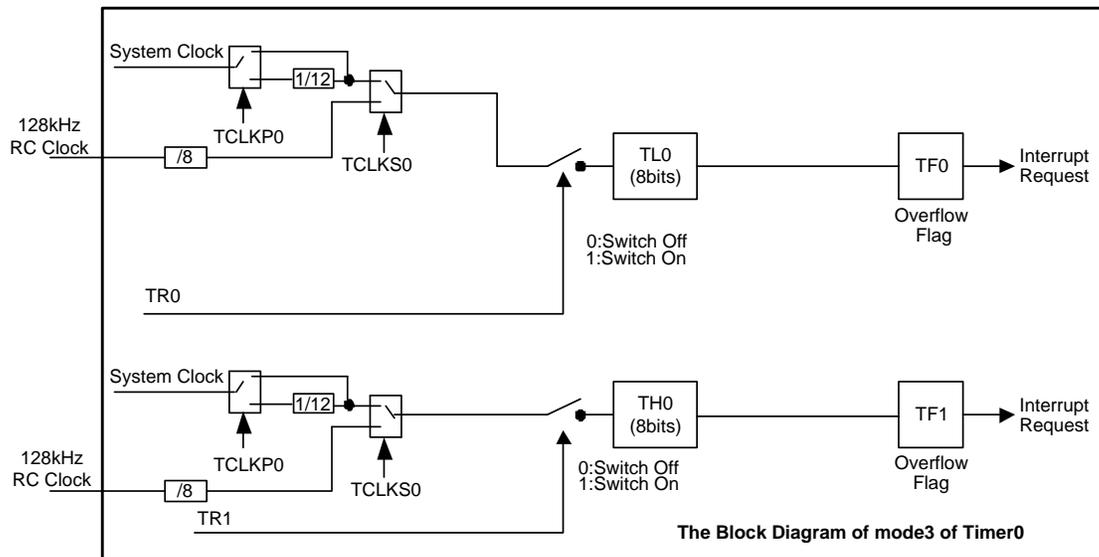
在方式3中，定时器0用作两个独立的8位定时器，分别由TL0和TH0控制。TL0使用定时器0的控制（在TCON中）和状态（在TMOD中）位：TR0，TF0。TL0能用系统时钟或128kHz的8分频或外部输入信号作为时钟源。

TH0只能用作定时器功能，时钟源来自系统时钟。TH0由定时器1的控制位TR1控制使能，溢出时定时器1溢出标志TF1置1，控制定时器1中断。

定时器0工作在方式3时，定时器1可以工作在方式0、1或2，但是不能置1 TF1标志和产生中断。TH1和TL1只能用作定时器功能，时钟源来自系统时钟。定时器1由方式控制使能与否，因为TR1被定时器0占用。定时器1在方式0、1或2时使能，在方式3时被关闭。

当作为定时器应用时，可配置寄存器TCON1中的TCLKSx（x = 0, 1）位选择系统时钟或128kHz的8分频作为定时器x（x = 0, 1）的时钟源。

可配置寄存器TCON1中的TCLKPx（x = 0, 1）位选择系统时钟或系统时钟的1/12作为定时器x（x = 0, 1）的时钟源。



**注意：**此时定时器1溢出率不作为波特率发生器，此功能仅作屏蔽，芯片的可变波特率仅由寄存器SBRTL/H设置。



寄存器

注意：定时器0, 1寄存器未定义的空白位必须清0。

Table 7.25 定时器x控制寄存器 (x = 0, 1)

88H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TCON	TF1	TR1	TF0	TR0	-	-	-	-
读/写	读/写	读/写	读/写	读/写	-	-	-	-
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	-	-	-	-

位编号	位符号	说明
7, 5	TFx x = 0, 1	定时器x溢出标志位 0: 定时器x无溢出, 可由软件清0 1: 定时器x溢出, 由硬件置1; 若由软件置1将会引起定时器中断
6, 4	TRx x = 0, 1	定时器x启动, 停止控制位 0: 停止定时器x 1: 启动定时器x

Table 7.26 定时器x方式寄存器 (x = 0, 1)

89H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TMOD	-	-	M11	M10	-	-	M01	M00
读/写	-	-	读/写	读/写	-	-	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	0	0	-	-	0	0

位编号	位符号	说明
5-4 1-0	Mx[1:0] x = 0, 1	定时器x定时器方式选择位 00: 方式0, 13位向上定时器, 忽略TLx的第7-5位 01: 方式1, 16位向上定时器 10: 方式2, 8位自动重载定时器 11: 方式3 (只用于定时器0), 两个8位定时器



注意：定时器0，1寄存器未定义的空白位必须清0。

Table 7.27 定时器x数据寄存器 (x = 0, 1)

8AH-8DH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TL0 (8AH)	TL0.7	TL0.6	TL0.5	TL0.4	TL0.3	TL0.2	TL0.1	TL0.0
TH0 (8BH)	TH0.7	TH0.6	TH0.5	TH0.4	TH0.3	TH0.2	TH0.1	TH0.0
TL1 (8CH)	TL1.7	TL1.6	TL1.5	TL1.4	TL1.3	TL1.2	TL1.1	TL1.0
TH1 (8DH)	TH1.7	TH1.6	TH1.5	TH1.4	TH1.3	TH1.2	TH1.1	TH1.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TLx.y, THx.y x=0-1, y=0-7	定时器x低及高字节计数器

Table 7.28 定时器x控制寄存器1 (x = 0, 1)

8FH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TCON1	-	TCLK_S1	TCLK_S0	-	TCLKP1	TCLKP0	-	-
读/写	-	读/写	读/写	-	读/写	读/写	-	-
复位值 (POR/WDT/LVR/PIN)	-	0	0	-	0	0	-	-

位编号	位符号	说明
6-5	TCLK_Sx x = 0, 1	定时器x时钟源控制位 0: 系统时钟作为定时器x的时钟源 1: 选择128kHz RC时钟的8分频作为定时器x的时钟源
3-2	TCLKPx x = 0, 1	定时器x时钟源预分频控制位 0: 选择系统时钟的1/12作为定时器x的时钟源 1: 选择系统时钟作为定时器x的时钟源



### 7.8.3 定时器2

两个数据寄存器（TH2和TL2）串联后可作为一个16位寄存器来访问，由寄存器T2CON和T2MOD控制。设置IEN0寄存器中的ET2位能允许定时器2中断。（详见中断章节）

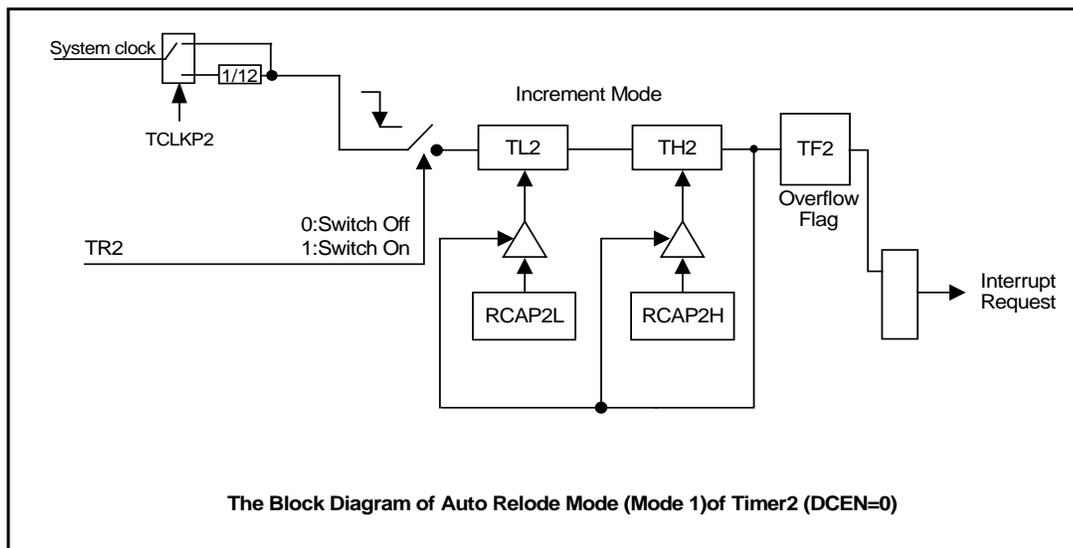
通过所选的引脚设置TR2允许定时器2数据寄存器计数。

#### 定时器2方式

##### 方式0：16位自动重载定时器

在16位自动重载方式下，定时器2可以被选为递增计数。

定时器2递增到0FFFFH，在溢出后置起TF2位，同时定时器自动将用户软件写好的寄存器RCAP2H和RCAP2L的16位值装入TH2和TL2寄存器。



#### 注意:

- (1) 当事件发生时或其它任何时间都能由软件设置TF2为1，只有软件以及硬件复位才能使之清0。
- (2) 当EA = 1且ET2 = 1时，设置TF2为1能引起定时器2中断。



寄存器

注意：定时器2寄存器未定义的空白位必须清0。

Table 7.29 定时器2控制寄存器

C8H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
T2CON	TF2	-	-	-	-	TR2	-	-
读/写	读/写	-	-	-	-	读/写	-	-
复位值 (POR/WDT/LVR/PIN)	0	-	-	-	-	0	-	-

位编号	位符号	说明
7	TF2	定时器2溢出标志位 0: 无溢出（必须由软件清0） 1: 溢出（如果RCLK = 0和TCLK = 0，由硬件设1）
2	TR2	定时器2开始/停止控制位 0: 停止定时器2 1: 开始定时器2

Table 7.30 定时器2模式控制寄存器

C9H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
T2MOD	TCLKP2	-	-	-	-	-	-	-
读/写	读/写	-	-	-	-	-	-	-
复位值 (POR/WDT/LVR/PIN)	0	-	-	-	-	-	-	-

位编号	位符号	说明
7	TCLKP2	分频选择控制位 0: 选择系统时钟的1/12作为定时器2的时钟源 1: 选择系统时钟作为定时器2的时钟源

Table 7.31 定时器2重载/捕获和数据寄存器

CAH-CDH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
RCAP2L	RCAP2L.7	RCAP2L.6	RCAP2L.5	RCAP2L.4	RCAP2L.3	RCAP2L.2	RCAP2L.1	RCAP2L.0
RCAP2H	RCAP2H.7	RCAP2H.6	RCAP2H.5	RCAP2H.4	RCAP2H.3	RCAP2H.2	RCAP2H.1	RCAP2H.0
TL2	TL2.7	TL2.6	TL2.5	TL2.4	TL2.3	TL2.2	TL2.1	TL2.0
TH2	TH2.7	TH2.6	TH2.5	TH2.4	TH2.3	TH2.2	TH2.1	TH2.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	RCAP2L.x	定时器2重载/捕获数据低位高位，x = 0 - 7
	RCAP2H.x	
7-0	TL2.x	定时器2低位/高位计数器，x = 0 - 7
	TH2.x	



### 7.8.4 定时器3

定时器3是16位自动重载定时器，通过两个数据寄存器TH3和TL3访问，由T3CON寄存器控制。IEN1寄存器的ET3位置1允许定时器3中断（详见中断章节）。

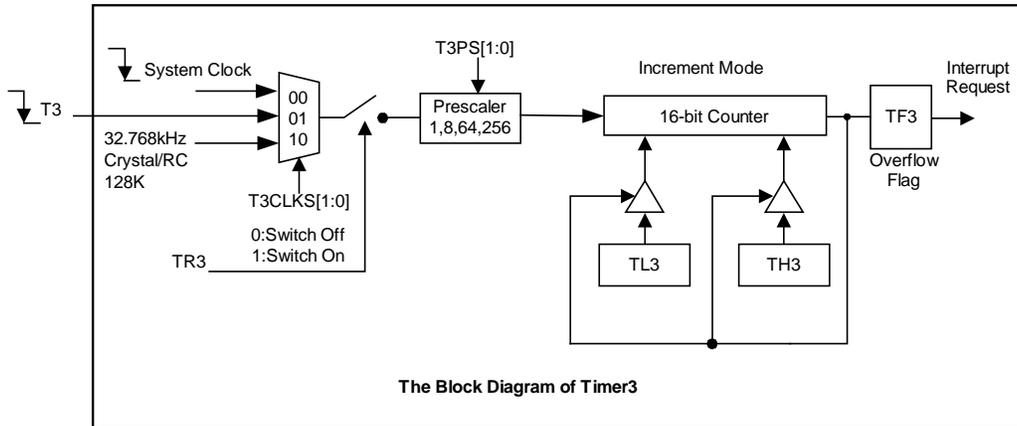
定时器3只有一个工作方式：16位自动重载计数器/定时器，可以设置预分频比，并可以工作在CPU掉电模式。

定时器3有一个16位计数器/定时器寄存器（TH3，TL3）。当TH3和TL3被写时，用作定时器重载寄存器，当被读时，被用作计数寄存器。TR3位置1使定时器3开始递增计数。定时器在0xFFFF到0x0000溢出并置TF3位为1。溢出同时，定时器重载寄存器的16位数据被重新载入计数寄存器中，TH3写操作也导致重载寄存器的数据重新载入计数寄存器。

TH3和TL3读写操作遵循以下顺序：

写操作：先低位后高位

读操作：先高位后低位



定时器3可以工作在掉电模式。

当OP\_OSC[3:0]（详见代码选项章节）为0000，1110可以选为00，01。当OP\_OSC[3:0]为0011，0110，1010时，T3CLKS[1:0]可以选为00，01，10。

如果T3CLKS[1:0]为00，定时器3不能工作在掉电模式下。如果T3CLKS[1:0]为01，T3端口输入外部时钟，定时器3可以工作在普通模式或掉电模式（在系统为高频时进入掉电模式）。当T3CLKS[1:0]为10并且OP\_OSC[2:0]011时，定时器3可以工作在普通模式或掉电模式。当T3CLKS[1:0]为10并且OP\_OSC[3:0]为0000，1110时，定时器3不工作。详见下表：

OP_OSC[3:0]	T3CLKS[1:0]	工作在普通模式	工作在掉电模式
0000 1110	00	YES	NO
	01	YES	YES
	10	NO	NO
0011 0110 1010	00	YES	NO
	01	YES	YES
	10	YES	YES

**注意：**

- (1) 在读或写TH3和TL3时，要确保TR3 = 0。
- (2) 当定时器3用T3端口作为时钟源，而且TR3由0变为1时，定时器3忽略T3的第1个下降沿。



寄存器

Table 7.32 定时器3控制寄存器

C0H, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
T3CON	TF3	-	T3PS.1	T3PS.0	-	TR3	T3CLKS.1	T3CLKS.0
读/写	读/写	-	读/写	读/写	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	-	0	0	-	0	0	0

位编号	位符号	说明
7	TF3	定时器3溢出标志位 0: 无溢出 (硬件清0) 1: 溢出 (硬件置1)
5-4	T3PS[1:0]	定时器3预分频比选择位 00: 1/1 01: 1/8 10: 1/64 11: 1/256
2	TR3	定时器3允许控制位 0: 停止定时器3 1: 开始定时器3
1-0	T3CLKS[1:0]	定时器3定时器/计数器方式选定位 00: 系统时钟, T3引脚用作I/O端口 01: T3端口输入外部时钟, 自动上拉 10: 外部32.768kHz晶体谐振器 11: 保留

Table 7.33 定时器3重载/计数数据寄存器

C2H-C3H, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TL3 (C2H)	TL3.7	TL3.6	TL3.5	TL3.4	TL3.3	TL3.2	TL3.1	TL3.0
TH3 (C3H)	TH3.7	TH3.6	TH3.5	TH3.4	TH3.3	TH3.2	TH3.1	TH3.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TL3.x	定时器3低位高位计数器, x = 0 - 7
	TH3.x	

Table 7.34 定时器3重载/计数数据寄存器

C1H, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
SWTHL	-	-	-	-	-	-	-	TH3LCON
读/写	-	-	-	-	-	-	-	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	-	-	0

位编号	位符号	说明
0	T3HLCON	0: 读TH3和TL3时, 返回值为T3计数器值 1: 读TH3和TL3时, 返回值为T3重载寄存器值



### 7.8.5 定时器4

定时器4是16位自动重载定时器。两个数据寄存器TH4和TL4可作为一个16位寄存器来访问。由T4CON寄存器控制。IEN1寄存器的ET4位置1允许定时器4中断（详见中断章节）。

当TH4和TL4被写时，用作定时器重载寄存器，当被读时，被用作计数寄存器。TR4位置1使定时器4开始递增计数。定时器在0xFFFF到0x0000溢出并置TF4位为1。溢出同时，定时器重载寄存器的16位数据重新载入计数寄存器中，对TH4的写操作也导致重载寄存器的数据重新载入计数寄存器。

TH4和TL4读写操作遵循以下顺序：

写操作：先低位后高位

读操作：先高位后低位

#### 定时器4方式

定时器4有两种工作方式：16位自动重载定时器，有T4边沿触发的16位自动重载定时器。这些方式通过T4CON寄存器的T4M[1:0]设置。

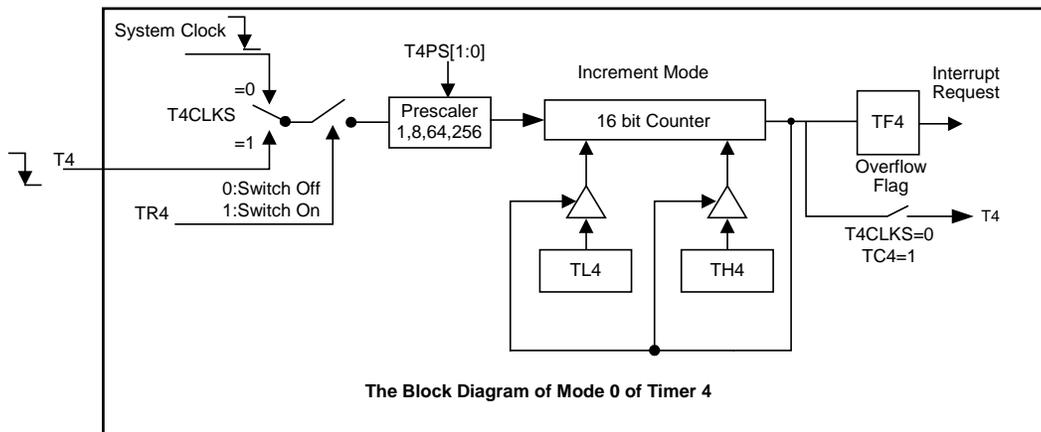
#### 方式0：16位自动重载定时器

定时器4在方式0为16位自动重载定时器。TH4寄存器存放16位计数器/定时器高8位，TL4存放低8位。当16位定时寄存器从0xFFFF到0x0000递增，并溢出时，系统置定时器溢出标志TF4（T4CON.7）为1，16位寄存器的值被重新载入计数器，如果允许定时器4中断则产生中断。

T4CON.0寄存器的T4CLKS位选择时钟源。当T4CLKS = 1时，定时器4的时钟源为外部时钟，预分频后，计数器数据寄存器增加。当T4CLKS = 0，定时器4的时钟源为系统时钟。

T4CON.1寄存器的TR4位置1允许定时器4，且不清定时器4的计数器。在允许定时器4之前，将希望的初始值写入定时器重载寄存器中。

在比较方式中，T4端口自动被硬件设为输出。定时器4从TH4和TL4预设值开始向0xFFFF计数，当计数器溢出时，T4端口输出反转，同时定时器4中断标志位被置起。在比较方式中，定时器4必须工作在定时方式（T4CLKS = 0）。





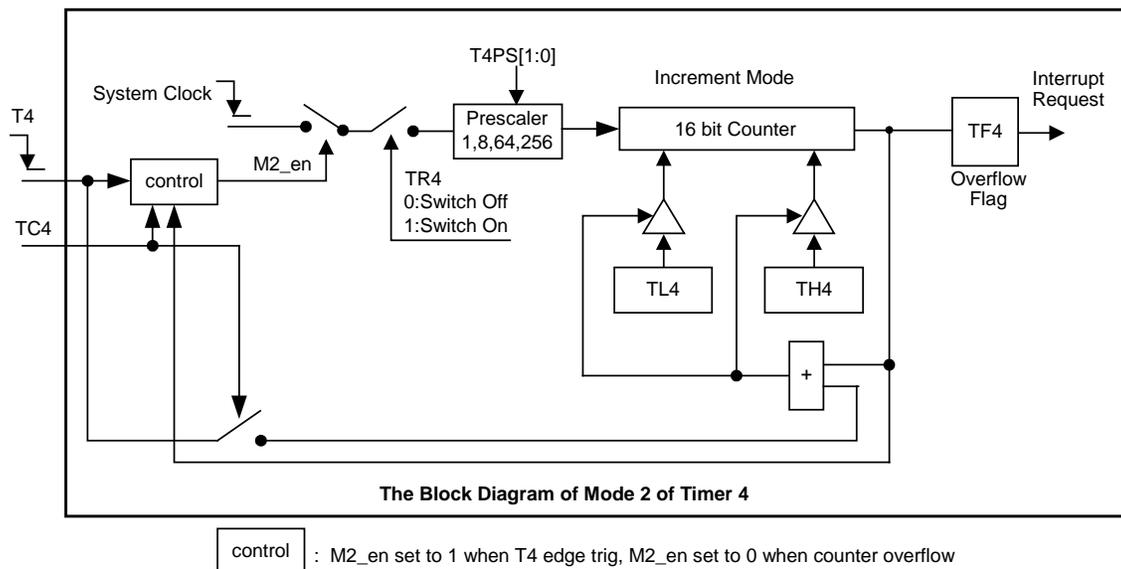
### 方式1：带边沿触发的16位自动重载定时器

定时器4在方式2为16位自动重载定时器。T4CON.0寄存器的T4CLKS位一直为0，定时器4只能选择系统时钟为时钟源，其余设置与方式0一致。

方式2中，T4CON.1的TR4位置1，定时器4等待T4端口的触发信号（由T4M[1:0]控制上升/下降沿）开始计数，一个触发信号使定时器4开始运行。当定时器4从0xFFFF到0x0000溢出时，TF4（T4CON.7）会被置起，如定时器4的中断使能，将产生定时器4中断。T4CLKS位始终为0。定时器4的时钟为系统时钟。定时器重载寄存器的数据重载入TH4和TL4中，保持到下一个触发信号。

在定时器4工作时同时有一个触发信号时，如果TC4 = 0，忽略此信号；如果TC4 = 1，定时器4被触发。

TR4置1不清定时器4的计数器，在允许定时器之前应该把希望的初始化值写入重载寄存器。



#### 注意：

- (1) 当定时器4在波特率发生器方式下作为定时器工作时（TR4 = 1），TH4或TL4不能读取或写入。因为定时器在每个状态时间递增，可能导致读取或写入的结果不精确。因此，在访问TH4/TL4寄存器之前，定时器4必须被关闭（TR4 = 0）。
- (2) 当定时器4用作计数器时，T4引脚的输入信号频率要小于系统时钟的一半。



寄存器

Table 7.35 定时器4控制寄存器

C8H, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
T4CON	TF4	TC4	T4PS1	T4PS0	T4M1	T4M0	TR4	T4CLKS
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	TF4	定时器4溢出标志位 0: 无溢出 (硬件清0) 1: 溢出 (硬件置1)
6	TC4	比较功能允许位 当T4M[1:0] = 00 0: 禁止定时器4比较功能 1: 允许定时器4比较功能 当T4M[1:0] = 10或11 0: 定时器4不能被再触发 1: 定时器4可以被再触发
5-4	T4PS[1:0]	定时器4预分频比选择位 00: 1/1 01: 1/8 10: 1/64 11: 1/256
3-2	T4M[1:0]	定时器4方式选择位 00: Mode0, 16位自动重载定时器 01: 保留 10: Mode2, T4端口上升沿触发 (只用系统时钟, T4CLKS无效) 11: Mode2, T4端口下降沿触发 (只用系统时钟, T4CLKS无效)
1	TR4	定时器4允许控制位 0: 禁止定时器4 1: 允许定时器4
0	T4CLKS	定时器4时钟源选择位 0: 系统时钟, T4端口作为I/O口 1: T4端口输入外部时钟, 自动上拉

Table 7.36 定时器4重载/计数数据寄存器

CCH-CDH, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TL4	TL4.7	TL4.6	TL4.5	TL4.4	TL4.3	TL4.2	TL4.1	TL4.0
TH4	TH4.7	TH4.6	TH4.5	TH4.4	TH4.3	TH4.2	TH4.1	TH4.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TL4.x	定时器4低位高位计数器, x = 0 - 7
	TH4.x	



## 7.9 中断

### 7.9.1 特性

- 15个中断源
- 4层中断优先级

SH79F6412有15个中断源：3个外部中断（INT2，INT3，INT4），5个定时器中断（定时器0，1，2，3，4），1个EUART中断，TWI中断，TK中断，ADC中断，两个PWM中断，ESCM中断。

### 7.9.2 中断允许

任何一个中断源均可通过对寄存器IEN0和IEN1中相应的位置1或清0，实现单独使能或禁止。IEN0寄存器中还包含了一个全局使能位EA，它是所有中断的总开关。一般在复位后，所有中断允许位设置为0，所有中断被禁止。

### 7.9.3 寄存器

Table 7.37 初级中断允许寄存器

A8H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IEN0	EA	EADC	ET2	ES	ET1	EPWM2	ET0	TKIE
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	EA	所有中断允许位 0: 禁止所有中断 1: 允许所有中断
6	EADC	ADC中断允许位 0: 禁止ADC中断 1: 允许ADC中断
5	ET2	定时器2溢出中断允许位 0: 禁止定时器2溢出中断 1: 允许定时器2溢出中断
4	ES	EUART中断允许位 0: 禁止EUART中断 1: 允许EUART中断
3	ET1	定时器1中断允许位 0: 禁止定时器1溢出中断 1: 允许定时器1溢出中断
2	EPWM2	PWM2中断允许位 0: 禁止PWM2中断 1: 允许PWM2中断
1	ET0	定时器0溢出中断允许位 0: 禁止定时器0溢出中断 1: 允许定时器0溢出中断
0	TKIE	触摸按键中断允许位 0: 禁止触摸按键中断 1: 允许触摸按键中断



Table 7.38 次级中断允许寄存器

A9H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IEN1	ESCM/ELPD	EX3	EPWM1/ECRC	ET3	ETWI	EX4	EX2	ET4
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	ESCM/ELPD	<b>SCM/LPD中断允许位</b> 0: 禁止SCM/LPD中断 1: 允许SCM/LPD中断
6	EX3	<b>外部中断3允许位</b> 0: 禁止外部中断3中断 1: 允许外部中断3中断
5	EPWM1/ECRC	<b>PWM1/CRC中断允许位</b> 0: 禁止PWM1中断/CRC中断 1: 允许PWM1中断/CRC中断
4	ET3	<b>定时器3溢出中断允许位</b> 0: 禁止定时器3溢出中断 1: 允许定时器3溢出中断
3	ETWI	<b>TWI中断允许位</b> 0: 禁止TWI中断 1: 允许TWI中断
2	EX4	<b>外部中断4允许位</b> 0: 禁止外部中断4中断 1: 允许外部中断4中断
1	EX2	<b>外部中断2允许位</b> 0: 禁止外部中断2 1: 允许外部中断2
0	ET4	<b>定时器4溢出中断允许位</b> 0: 禁止定时器4溢出中断 1: 允许定时器4溢出中断

注意:

- (1) 打开外部中断2/3/4时, 相应的端口必须设置为输入状态。
- (2) 打开PWM定时器中断, EPWM位和PWM控制寄存器中的PWMxIE (x = 1, 2) 中断允许位必须同时置1。

Table 7.39 外部中断通道允许寄存器

BAH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IENC	EXS47	EXS46	EXS45	-	-	-	-	EXS40
读/写	读/写	读/写	读/写	-	-	-	-	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	-	-	-	-	0

位编号	位符号	说明
7-5, 0	EXS4x (x = 0, 5-7)	<b>外部中断4通道选择位 (x = 0, 5-7)</b> 0: 禁止外部中断4x 1: 允许外部中断4x

注意: 寄存器未定义的空白位必须清0。



Table 7.40 中断通道选择寄存器

BBH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IENC1	-	-	ECRC	EPWM1	EPWM2	-	ESCM1	ELPD
读/写	-	-	读/写	读/写	读/写	-	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	0	0	0	-	0	0

位编号	位符号	说明
5	ECRC	<b>CRC中断允许位</b> 0: 禁止PWM2中断 1: 允许PWM2中断
4	EPWM1	<b>PWM1中断允许位</b> 0: 禁止PWM1中断 1: 允许PWM1中断
3	EPWM2	<b>PWM2中断允许位</b> 0: 禁止PWM2中断 1: 允许PWM2中断
1	ESCM1	<b>SCM中断允许位</b> 0: 禁止SCM中断 1: 允许SCM中断
0	ELPD	<b>LPD中断允许位</b> 0: 禁止LPD中断 1: 允许LPD中断

Table 7.41 外部中断通道允许寄存器

BCH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IENC2	-	-	-	-	-	EXS32	EXS31	EXS30
读/写	-	-	-	-	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	0	0	0

位编号	位符号	说明
2-0	EXS3x (x = 0-2)	<b>外部中断3通道选择位 (x = 0-2)</b> 0: 禁止外部中断3x 1: 允许外部中断3x

注意：寄存器未定义的空白位必须清0。



### 7.9.4 中断标志

每个中断源都有自己的中断标志，当产生中断时，硬件会置起相应的标志位，在中断摘要表中会列出中断标志位。

外部中断源产生外部中断INT2/3/4时，如果中断为边沿触发，CPU在响应中断后，各中断标志位IE<sub>x</sub> (x =2-4) 被硬件清0；如果中断是低电平触发，外部中断源引脚电平直接控制中断标志，而不是由片上硬件控制。

T2CON寄存器的TF2或EXF2标志位置1时，产生定时器2产生中断，CPU在响应中断后，标志位都不会被硬件自动清0。事实上，中断服务程序必须决定是由TF2或是EXF2产生中断，标志必须由软件清0。

定时器3的计数器溢出时，T3CON寄存器的TF3中断标志位置1，产生定时器3中断，CPU在响应中断后，标志被硬件自动清0。

SCON寄存器的标志RI或TI被置1时，产生EUART产生中断，CPU在响应中断后，标志不会被硬件自动清0。事实上，中断服务程序必须判断是收中断还是发中断，标志必须由软件清0。

Table 7.42 外部中断标志寄存器

E8H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
EXF0	IT4.1	IT4.0	IT3.1	IT3.0	IT2.1	IT2.0	IE3	IE2
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-6	IT4[1:0]	外部中断4触发模式位 00: 低电平触发 01: 下降沿触发 10: 上升沿触发 11: 双沿触发 IT4[1:0]控制外部中断4各中断源采用同一触发方式
5-4	IT3[1:0]	外部中断3触发模式位 00: 低电平触发 01: 下降沿触发 10: 上升沿触发 11: 双沿触发
3-2	IT2[1:0]	外部中断2触发模式位 00: 低电平触发 01: 下降沿触发 10: 上升沿触发 11: 双沿触发
1	IE3	外部中断3请求标志位 0: 无中断挂起 1: 中断挂起
0	IE2	外部中断2请求标志位 0: 无中断挂起 1: 中断挂起



Table 7.43 定时器/计数器控制寄存器 (x = 0, 1)

88H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TCON	TF1	TR1	TF0	TR0	-	-	-	-
读/写	读/写	读/写	读/写	读/写	-	-	-	-
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	-	-	-	-

位编号	位符号	说明
7, 5	TFx (x = 0, 1)	定时器x溢出标志 0: 定时器x无溢出 1: 定时器x溢出
6, 4	TRx (x = 0, 1)	定时器x启动, 停止控制 0: 停止定时器x 1: 启动定时器x

Table 7.44 外部中断4标志寄存器

D8H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
EXF1	IF47	IF46	IF45	-	-	-	-	IF40
读/写	读/写	读/写	读/写	-	-	-	-	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	-	-	-	-	0

位编号	位符号	说明
7-5, 0	IF4x (x = 0, 5-7)	外部中断4请求标志 0: 无中断请求 1: 有中断请求 IF4x要由软件清0

Table 7.45 外部中断3标志寄存器

D1H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
EXF3	-	-	-	-	-	IF32	IF31	IF30
读/写	-	-	-	-	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	0	0	0

位编号	位符号	说明
2-0	IF3x (x = 0-2)	外部中断3请求标志 0: 无中断请求 1: 有中断请求 IF3x要由软件清0

注意: 寄存器未定义的空白位必须清0。



### 7.9.5 中断向量

当一个中断产生时，程序计数器内容被压栈，相应的中断向量地址被载入程序计数器。中断向量的地址在**中断汇总表**中详细列出。

### 7.9.6 中断优先级

每个中断源都可被单独设置为4个中断优先级之一，分别通过清0或置1 IPL0, IPH0, IPL1, IPH1中相应位来实现。中断优先级服务程序描述如下：

响应一个中断服务程序时，可响应更高优先级的中断，但不能响应同优先级或低优先级的另一个中断。

响应最高级中断服务程序时，不响应其它任何中断。如果不同中断优先级的中断源同时申请中断时，响应较高优先级的中断申请。

如果同优先级的中断源在指令周期开始时同时申请中断，那么内部查询序列确定中断请求响应顺序。

中断优先级		
优先位		中断优先级
IPHx	IPLx	
0	0	等级0（最低优先级）
0	1	等级1
1	0	等级2
1	1	等级3（最高优先级）

Table 7.46 中断优先级控制寄存器

B8H, B4H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IPL0	-	PADL	PT2L	PESL	PT1L	PW2L	PT0L	PTKL
IPH0	-	PADH	PT2H	PESH	PT1H	PW2H	PT0H	PTKH
读/写	-	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	0	0	0	0	0	0	0
B9H, B5H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
IPL1	PSCL	PX3L	PW1L	PT3L	PTWL	PX4L	PX2L	PT4L
IPH1	PSCH	PX3H	PW1H	PT3H	PTWH	PX4H	PX2H	PT4H
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7-0	PxxxL/H	相应中断源xxx优先级选择						

**注意：**寄存器未定义的空白位必须清0。



### 7.9.7 中断处理

中断标志在每个机器周期都会被采样获取。所有中断都在时钟的上升沿被采样。如果一个标志被置起，那么CPU捕获到后中断系统调用一个长转移指令（LCALL）调用其中断服务程序，但由硬件产生的LCALL会被下列任何条件阻止：

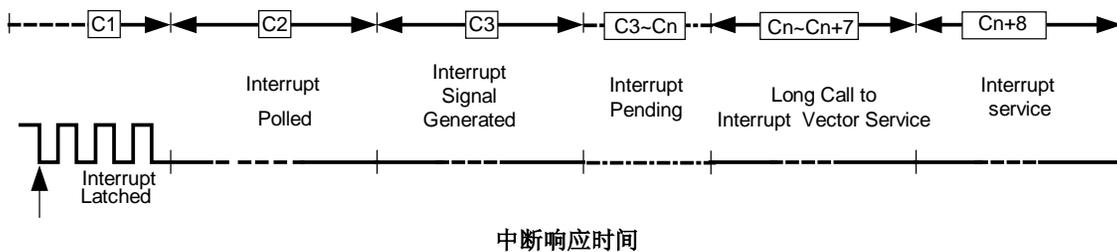
同级或更高级的优先级中断在运行中。

当前的周期不是执行中指令的最后一个周期。换言之，正在执行的指令完成前，任何中断请求都得不到响应。

正在执行的是一条RETI或者访问专用寄存器IEN0\1或是IPL\H的指令。换言之，在RETI或者读写IEN0\1或是IPL\H之后，不会马上响应中断请求，而至少在执行一条其它指令之后才会响应。

**注意：**因为更改优先级通常需要2条指令，在此期间，建议关闭相应的中断以避免在修改优先级过程中产生中断。如果当模块状态改变而中断标志不再有效时，将不会响应此中断。每一个轮询周期只查询有效的中断请求。

轮询周期/LCALL次序如下图所示：



由硬件产生的LCALL把程序计数器中的内容压入堆栈（但不保存PSW），然后将相应中断源的向量地址（参照中断向量表）存入程序计数器。

中断服务程序从指定地址开始，到RETI指令结束。RETI指令通知处理器中断服务程序结束，然后把堆栈顶部两字节弹出，重载入程序计数器中，执行完中断服务程序后程序回到原来停止的地方。RET指令也可以返回到原来地址继续执行，但是中断优先级控制系统仍然认为一个同一优先级的中断被响应，这种情况下，当同一优先级或低优先级中断将不会被响应。

### 7.9.8 中断响应时间

如果检测出一个中断，这个中断的请求标志位就会在被检测后的每个机器周期被置起。内部电路会保持这个值直到下一个机器周期，CPU会在第三个机器周期产生中断。如果响应有效且条件允许，在下一个指令执行的时候硬件LCALL指令将调用请求中断的服务程序，否则中断被挂起。LCALL指令调用程序需要7个机器周期。因而，从外部中断请求到开始执行中断程序至少需要3+7个完整的机器周期。

当请求因前述的三个情况受阻时，中断响应时间会加长。如果同级或更高优先级的中断正在执行，额外的等待时间取决于正执行的中断服务程序的长度。

如果正在执行的指令还没有进行到最后一个周期，假如正在执行RETI指令，则完成正在执行的RETI指令，需要8个周期，加上为完成下一条指令所需的最长时间20个机器周期（如果该指令是16位操作数的DIV，MUL指令），若系统中只有一个中断源，再加上LCALL调用指令7个机器周期，则最长的响应时间是2+8+20+7个机器周期。

所以，中断响应时间一般大于10个机器周期小于37个机器周期。



### 7.9.9 外部中断输入

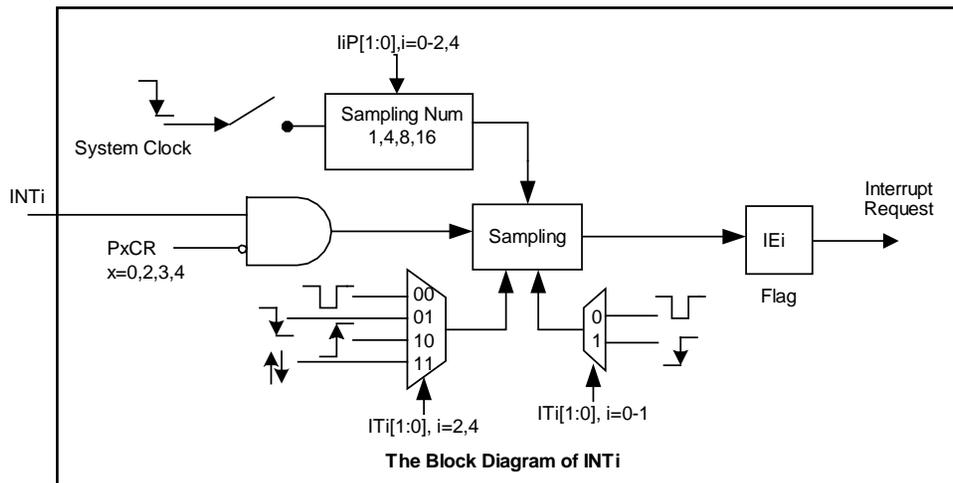
SH79F6412有3个外部中断输入。外部中断2有一个独立的中断源，外部中断4有7个中断源共享一个中断矢量地址，外部中断3有2个中断源共享一个中断矢量地址。外部中断2/3/4可以通过设置EXF0寄存器的ITx (x = 2, 3, 4)，选择是电平触发或是边沿触发或者为双沿触发。当ITx = 00 (x = 2, 3, 4) 时，外部中断INTx (x = 2, 3, 4) 引脚为低电平触发；当ITx (x = 2, 3, 4) = 01, 10，外部中断INTx (x = 3, 4) 为下降沿触发，在这个模式中，一个采样周期内INTx (x = 2, 3, 4) 引脚上连续采样为高电平，而下个周期开始，连续采样SN个周期为低电平 (SN为Sample Num)，EXF0寄存器的中断请求标志位置1，发出一个中断请求。由于外部中断引脚每个周期采样一次，输入高或低电平应当保持至少SN个周期以确保能够被正确采样到。当ITx (x = 2, 3, 4) = 11，外部中断INTx (x = 3, 4) 为双沿触发，任何高低电平的转换都会触发一个中断请求。

如果外部中断为下降沿，上升沿触发，外部中断源应当将中断脚至少保持SN个周期高(低)电平，然后至少保持SN个周期低(高)电平。这样就确保了边沿能够被检测到以使IEx置1。当调用中断服务程序后，CPU自动将IEx清0。

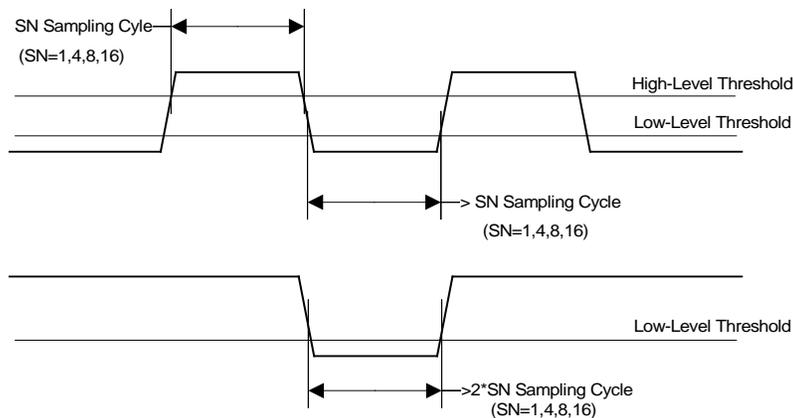
如果外部中断为低电平触发，外部中断源必须一直保持请求有效，直到产生所请求的中断为止，此过程需要2倍SN个采样周期。如果中断服务完成后而外部中断仍旧维持，则会产生下一次中断。当中断为电平触发时不必清除中断标志IEx (x = 2, 3, 4)，因为中断只与输入口电平有关。

中断连续采样次数可以设置EXCON寄存器进行调节。

当SH79F6412进入空闲或是掉电模式，中断会唤醒处理器继续工作，详见电源管理章节。



**注意：**外部中断0-2的中断标志位在执行中断服务程序时被自动硬件清0，但外部中断4标志位IF40-43必须要软件清0。



外部中断检测



Table 7.47 外部中断采样次数控制寄存器

A4H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
EXCON	I1PS1	I1PS0	I1SN1	I1SN0	I0PS1	I0PS0	I0SN1	I0SN0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-6	I1PS[1:0]	外部中断INT4采样时钟预分频比选择位 00: 1/1 01: 1/16 10: 1/64 11: 1/128
5-4	I1SN[1:0]	外部中断INT4连续采样次数选择位 00: 1 01: 2 10: 3 11: 4
3-2	I0PS[1:0]	外部中断INT2, 3采样时钟预分频比选择位 00: 1/1 01: 1/16 10: 1/64 11: 1/128
1-0	I0SN[1:0]	外部中断INT2, 3连续采样次数选择位 00: 1 01: 2 10: 3 11: 4

注意: 若I0SN[1:0] = 11, 则外部中断2, 3 (下降沿触发), 连续采样4次低电平才会产生中断标志。



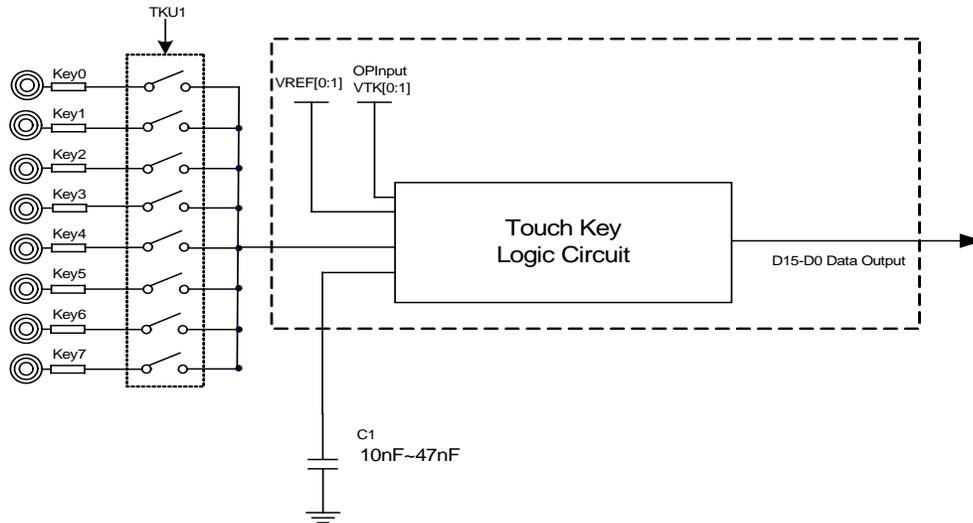
7.9.10 中断汇总

中断源	向量地址	允许位	标志位	轮询优先级	中断号 (C51)
Reset	0000H	-	-	0 (最高级)	-
TK	0003H	TKIE	IFERR+IFGO +IFAVE+IFCOUNT +IFTKOV	1	0
Timer0	000BH	ET0	TF0	2	1
PWM2	0013H	EPWM2	PWM02IF	3	2
Timer1	001BH	ET1	TF1	4	3
EUART	0023H	ES	RI+TI	5	4
Timer2	002BH	ET2	TF2+EXF2	6	5
ADC	0033H	EADC	ADCIF	7	6
Timer4	003BH	ET4	TF4	8	7
INT2	0043H	EX2	IE2	9	8
INT4	004BH	EX4+EXF1	IF47-40	10	9
TWI	0053H	ETWI	TWINT	11	10
Timer3	005BH	ET3	TF3	12	11
PWM1/CRC	0063H	EPWM1/ECRC	PWM01IF/CRCIF	13	12
INT3	006BH	EX3	IF32-IF30	14	13
SCM/LPD	0073H	ESCM+ESCM1/ELPD	SCMIF/LPDF	15	14



## 8. 增强功能

### 8.1 Touch Key 触摸按键功能



系统框图

#### 功能描述

SH79F6412内建触摸按键功能模块，最大能连接8个按键。

SH79F6412内建触摸按键功能模块工作电路精简，应用时仅需外接一个C1电容。C1电容值选择22nF - 44nF之间，要求使用10%或以上精度的涤纶电容、X7R材质电容或NPO材质贴片电容。C1电容可根据实际电路板材质以及触摸按键介质调节合适的灵敏度，电容值越小，灵敏度越高，电容值越大，灵敏度越低。

触摸按键模块最大能连接8个按键，可通过寄存器TKU1来选择，能够选择一轮扫描按键通道个数。

如果不用做触摸按键功能，可通过寄存器设置作为I/O端口或者SEG输出或者COM输出。详见I/O端口章节。

开关电路由寄存器，可通过寄存器FSW1位和FSW0位选择。建议工作频率选择4M或4M以下频率。触摸按键模块内建基准电压，可通过寄存器VREF1位和VREF0位选择。

触摸按键通过调整寄存器TUNE1位和TUNE0位来确保对应不同的C1和工作频率下数据寄存器值的稳定性，

按键采样次数根据实际应用，可选择多次采样，程序只要启动一次采样扫描，硬件会自动执行多次采样的比较值进行平均值后输出结果，例如，选择6次采样输出，寄存器TKGO/DONE位置1，启动按键扫描后，硬件会采样6次按键值，将会将6次的采样值进行比较，去除一个最大值和一个最小值，完成后将剩余四次中值累加的值除以4得到的结果，输出到16位数据寄存器。

28位放大系数寄存器为按键控制器运算结果放大系数，如果运算结果大于16位的数据，即为运算结果高位溢出，标志位IFERR位置1，如果中断允许将响应中断子程序，此时，用户需要将放大系数寄存器的数据值减小后重新启动下一次扫描。一般运算时，16位数据寄存器的值不应大于FFFFH。如果大于FFFFH应将被除数寄存器的数据值减小。

触摸按键有五种情况会产生中断标志位，其中前四种情况中断都会产生，第五种只有中断标志位，但不响应中断，需要判断中断标志位后执行中断子程序：

- (1) 按键扫描结束后，如运算结果高位溢出，将中断标志位IFERR置1，如果是多次采样，系统将停止当前采样状态等待下一次重新启动扫描，而不执行后面未做完的采样。如果发生运算结果高位溢出中断，用户应将28位放大系数寄存器值减小。
- (2) 按键扫描启动标志位TKGO/DONE1置1，系统自动检测比较输出状态是否正常，如果异常会将标志位IFGO置1，此时为按键控制器启动错误，用户应延时10uS后，重新启动下一次扫描。
- (3) 按键扫描结束后，如未发生异常，将中断标志位IFAVE置1。
- (4) 扫描按键计数过程中，当计数寄存器溢出时，将中断标志位IFCOUNT置1，用户需要减小电容或者将28位放大系数寄存器值减小。
- (5) 在与LED共用时，设定TKCOUNT（触摸按键时间宽度），启动触摸按键，如果在TKCOUNT时间内，触摸按键扫描尚未结束，则标志位IFTKOV置1，但是扫描按键和LED会继续扫描而且不会有中断产生，用户需要在发生中断的情况下来判断IFTKOV标志位，调整TKCOUNT时间来确保触摸按键扫描有足够的时间。

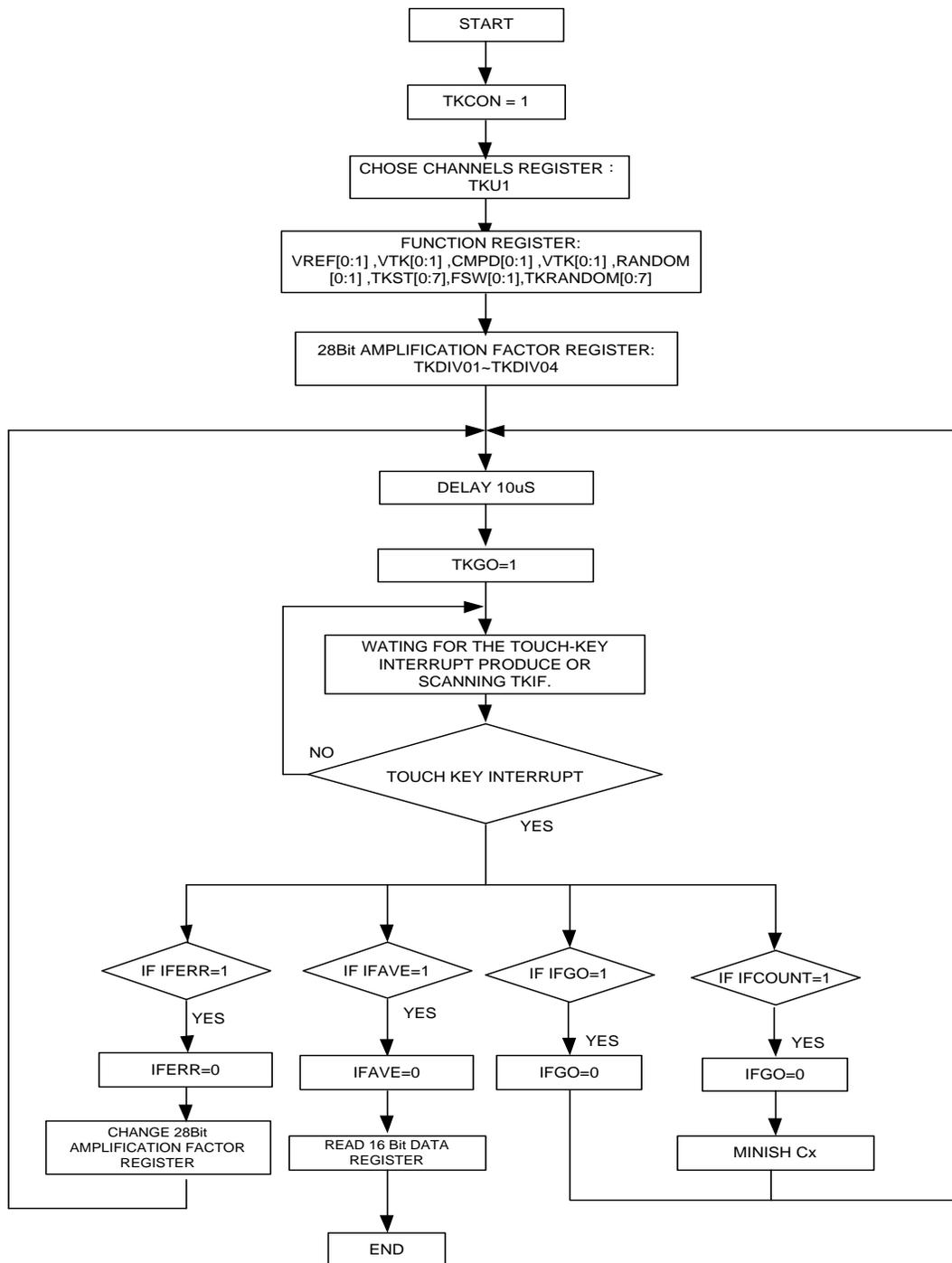


**仅作触摸按键功能时的启动扫描步骤:**

- (1) 选择需要扫描的按键通道;
- (2) 寄存器TKCON位置1, 允许触摸按键模块工作;
- (3) 设置开关频率、参考电压 $V_{REF}$ 和按键采样次数和扫描顺序;
- (4) 设置28位放大系数寄存器;
- (5) 软件延时10uS;
- (6) 寄存器TKGO/DONE位置1, 启动按键扫描;
- (7) 中断产生, TKGO硬件自动清0;
- (8) 中断标志位判断: IFERR, IFGO, IFAVE, IFCOUNT
  - 如果IFAVE = 1, 读数据寄存器201H - 228H, 程序保存数据结果, 执行步骤9;
  - 如果IFERR = 1, 数据寄存器运算溢出错误, 清IFERR和标志位, 重新设置放大系数寄存器值, 减小放大系数值, 返回步骤5重新启动扫描;
  - 如果IFGO = 1, 按键控制器启动错误, 清IFGO和标志位, 返回步骤5重新启动扫描;
  - 如果IFCOUNT = 1, 按键扫描计数溢出错误, 清IFCOUNT和标志位, 减小电容C1或者减少平均次数。返回步骤5, 重新启动扫描。
- (9) 一组按键扫描完成。



操作流程图:





8.1.1 寄存器

Table 8.1 触摸按键功能控制寄存器

A1H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKCON1	TKCON	-	TKGO /DONE	SHARE	MODE	OVDD	FSW1	FSW0
读/写	读/写	-	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	-	0	0	0	0	0	0

位编号	位符号	说明
7	TKCON	触摸按键允许位 0: 禁止触摸按键工作 1: 允许触摸按键工作
5	TKGO/DONE	启动触摸按键允许位 0: 未启动按键扫描或按键扫描结束 1: 启动按键扫描或正在执行按键扫描
4	SHARE	触摸按键LED共享允许位 (详细描述见注3) 0: 禁止共享 1: 启动触摸按键和LED显示共享
3	MODE	触摸按键模式选择位 0: 选择充电次数来作为数据参数 1: 选择充电时间来作为数据参数
2	OVDD	OP输出V <sub>DD</sub> 允许位 0: 选择OP输出电压 1: 由V <sub>DD</sub> 输出电压
1-0	FSW[1:0]	采样次数选择位 00: 按键采样1次输出数据, D15-D0为1次采样的平均值 01: 按键采样3次输出数据, D15-D0为1次采样的平均值 (去除最大值及最小值) 10: 按键采样6次输出数据, D15-D0为4次采样的平均值 (去除最大值及最小值) 11: 按键采样10次输出数据, D15-D0为8次采样的平均值 (去除最大值及最小值)

注意: OVDD = 0: 由VTK来选择OP输出电压。OVDD = 1: 直接由V<sub>DD</sub>输出电压。

Table 8.2 触摸按键功能频率控制寄存器

CFH, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKST	-	ST.6	ST.5	ST.4	ST.3	ST.2	ST.1	ST.0
读/写	-	读/写						
复位值 (POR/WDT/LVR/PIN)	-	0	0	0	0	0	0	0

位编号	位符号	说明
6-0	ST[6:0]	触摸按键功能频率控制位 系统时钟/TKST = 触摸按键功能开关频率

注意: 触摸按键功能频率 = OSC/TKST: TKST最大为127分频, 最小为2, 当寄存器在小于等于2时, 则寄存器默认为系统时钟/2。



Table 8.3 触摸按键频率随机设置寄存器

B6H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKRANDOM	TKRADON	TKOFFSET	TKVDD	TKOUT	-	-	RANDOM1	RANDOM0
读/写	读/写	读/写	读/写	读/写	-	-	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	-	-	0	0

位编号	位符号	说明
7	TKRADON	触摸按键随机频率使能位 0: 禁止触摸按键随机频率功能 1: 打开触摸按键随机频率功能
6	TKOFFSET	触摸按键补偿使能位 0: 禁止触摸按键补偿 1: 打开触摸按键补偿
5	TKV <sub>DD</sub>	触摸按键补偿波形电平选择位 0: 触摸按键补偿波形电平由OP输出 1: 触摸按键补偿波形电平由V <sub>DD</sub> 输出
4	TKOUT	触摸按键补偿波形输出能力选择位 0: 触摸按键补偿波形弱输出 1: 触摸按键补偿波形强输出
1-0	RANDOM[1:0]	随机抖动设置位 TKRADSEL = 0 00: TKST随机抖动±1 01: TKST随机抖动±1, ±2 10: TKST随机抖动±1, ±2, ±3 11: TKST随机抖动±1, ±2, ±3, ±4

注意: Design spec: 随机抖动请在一段时间内保证数学累计为0

- (1) 当TKST为二分频时不能进行抖动时钟, 三分频时只能选择00有效, 四分频时有效选项为00, 01两档。当TKST为六分频以上时, TKST随即抖动四档全部有效。
- (2) 当触摸按键补偿位有效。在触摸按键扫描时, 除当前TK扫描按键通道外, 其他TK扫描按键通道输出补偿波形。当TKV<sub>DD</sub>有效时, 触摸按键补偿波形电平由V<sub>DD</sub>提供, 当TKV<sub>DD</sub>为零时, 由OP输出补偿波形电平。



Table 8.4 触摸按键中断标志位寄存器（该寄存器只能清0）

C1H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKF0	-	IFERR	IFGO	IFAVE	IFCOUNT	IFTKOV	-	-
读/写	-	读/写	读/写	读/写	读/写	读/写	-	-
复位值 (POR/WDT/LVR/PIN)	-	0	0	0	0	0	-	-

位编号	位符号	说明
6	IFERR	运算结果溢出中断标志位 0: 运算结果高位未溢出 1: 运算结果高位溢出产生中断
5	IFGO	启动信号错误中断标志位 0: 启动信号未错误 1: 启动信号错误产生中断
4	IFAVE	按键扫描结束中断标志位 0: 扫描未结束 1: 扫描结束产生中断
3	IFCOUNT	按键扫描计数溢出标志位 0: 按键扫描计数未溢出 1: 按键扫描计数溢出
2	IFTKOV	SHARE状态下LED扫描开始时TK尚未完成时错误信号中断标志位 0: TK扫描在LED扫描前结束 1: TK扫描在LED扫描前未结束产生中断

Table 8.5 放大系数寄存器

C3H-C6H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKDIV01 (C3H)	DIV7	DIV6	DIV5	DIV4	DIV3	DIV2	DIV1	DIV0
TKDIV02 (C4H)	DIV15	DIV14	DIV13	DIV12	DIV11	DIV10	DIV9	DIV8
TKDIV03 (C5H)	DIV23	DIV22	DIV21	DIV20	DIV19	DIV18	DIV17	DIV16
TKDIV04 (C6H)	-	-	-	-	DIV27	DIV26	DIV25	DIV24
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TKDIV0x x = 1- 4,	触摸按键放大系数 DIV0 - DIV27: 填入一个27位数值作为触摸按键的放大系数, 该数值越大, 得到的值越大



Table 8.6 端口功能控制寄存器

9DH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P2SS	P2SS.7	P2SS.6	P2SS.5	P2SS.4	P2SS.3	P2SS.2	P2SS.1	P2SS.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	PxSSy x = 0-5, y = 0-7	端口功能控制 0: 作为I/O 1: 作为触摸按键信道

注意: P2SS.1 - P2SS.7对应触摸按键通道TCH1 - TCH7 (P2.0 - P2.6)。P2SS.0对应触摸按键通道TCH0 (P1.7)。

Table 8.7 触摸按键功能时间控制寄存器

E6H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKCOUNT	COUNT0.7	COUNT0.6	COUNT0.5	COUNT0.4	COUNT0.3	COUNT0.2	COUNT0.1	COUNT0.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
0 7-0	COUNT0[7:0]	TK时间宽度选择位 TK时钟宽度 = LED时钟频率 X TKCOUNT

Table 8.8 触摸按键等待时间控制寄存器

A2H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKWAIT	TKWAIT.7	TKWAIT.6	TKWAIT.5	TKWAIT.4	TKWAIT.3	TKWAIT.2	TKWAIT.1	TKWAIT.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
0 7-0	TKWAIT[7:0]	TK等待时间宽度选择位 TK等待时钟宽度 = LED时钟频率 X TKCOUNT

注意: TKCOUNT TKWAIT寄存器仅在SHARE模式下有效。



在SHARE状态下，LED帧频时间分为三部分，即 $T_A$ 为触摸按键时间宽度， $T_B$ 为LED扫描时间宽度， $T_C$ 为触摸按键扫描等待时间宽度。下面公式中， $T_{LED}$ 为LED时钟宽度， $T_{SYS}$ 为系统时钟宽度。

$$T_{LED} = T_{SYS} \times 8 \times DISPCLK$$

$$T_A = T_{LED} \times TKCOUNT \text{ [时间A]}$$

$$T_B = (T_{LED} \times DISCOM) \times S \text{ [时间B]}$$

S为LED扫描的COM数量：扫描4COM即S = 4，扫描5COM即S = 5，以此类推。

$$T_C = T_{LED} \times TKWAIT \text{ [时间C]}$$

以 $T_A + T_B + T_C$ 扫描64Hz举例：

在SHARE状态下，当系统时钟为RC = 16.6MHz，且需要帧频为64Hz，扫描COM为5COM：

$$\text{那么， } T_{SYS} = 1/16.6M = 60.24ns$$

$$\text{需要结果为 } T_A + T_B + T_C = 1/64s = 15.625ms$$

如果设置DISPCLK为96H（DISPCLK理论上越小越好，但是也要保证TKCOUNT，DISCOM，TKWAIT不会溢出）

$$T_{LED} = T_{SYS} \times 8 \times DISPCLK$$

$$= 60.24ns \times 8 \times 150$$

$$= 72288ns = 72.288us = 0.072288ms$$

需得到 $T_A = 7ms$ 的触摸按键时间宽度：（触摸按键的时间宽度取值需要保证在这一段时间内，所有触摸按键可以采集完成）

$$TKCOUNT = T_A / T_{LED} = 7ms / 0.072288ms = 96 = 60H$$

需得到 $T_B = 6.625ms$ 的LED扫描时间宽度：（LED扫描时间宽度需要尽量多的扫描时间，保证LED亮度）

$$DISCOM = T_B / S / T_{LED} = 6.625ms / 5 / 0.072288ms = 18 = 12H$$

需得到 $T_C = 2ms$ 的触摸按键扫描等待时间宽度：（触摸按键扫描等待时间宽度在不影响触摸按键情况下，需要尽量少的的时间，一般可以设置为0）

$$TKWAIT = T_C / T_{LED} = 2ms / 0.072288ms = 27 = 1BH$$

当时间A，B，C当有一段时间如果发生改变，其他段时间如果没有改动的话，帧频就会发生变化。

在实际使用过程中，触摸按键值会随环境温度湿度的变化而变化，所以，触摸按键时间宽度 $T_A$ 需要预留一部分的空闲时间来作为余量，通常为所有触摸按键采集完成时间加上10-15%。

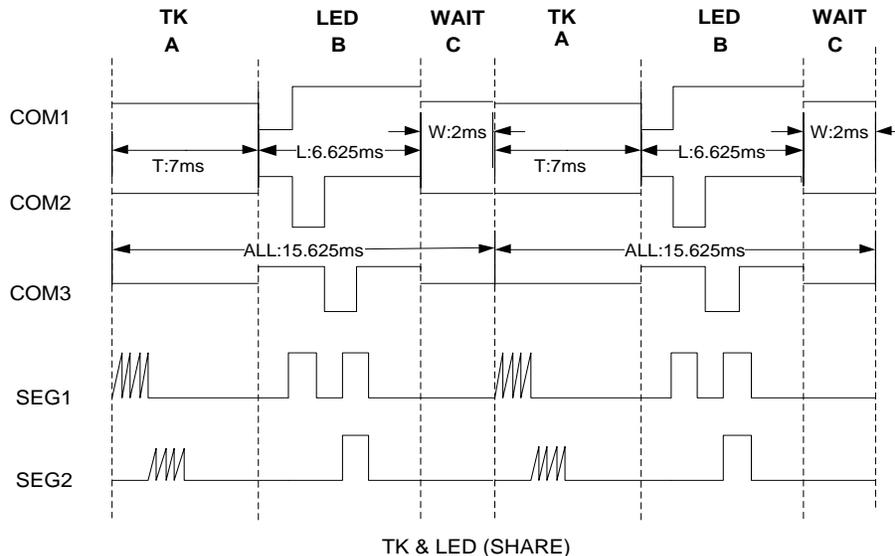


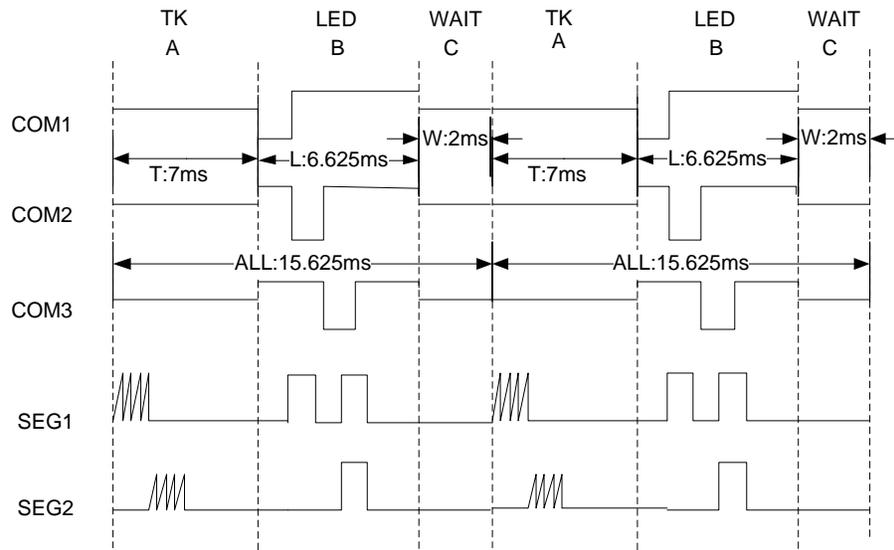


Table 8.9 按键扫描出错寄存器

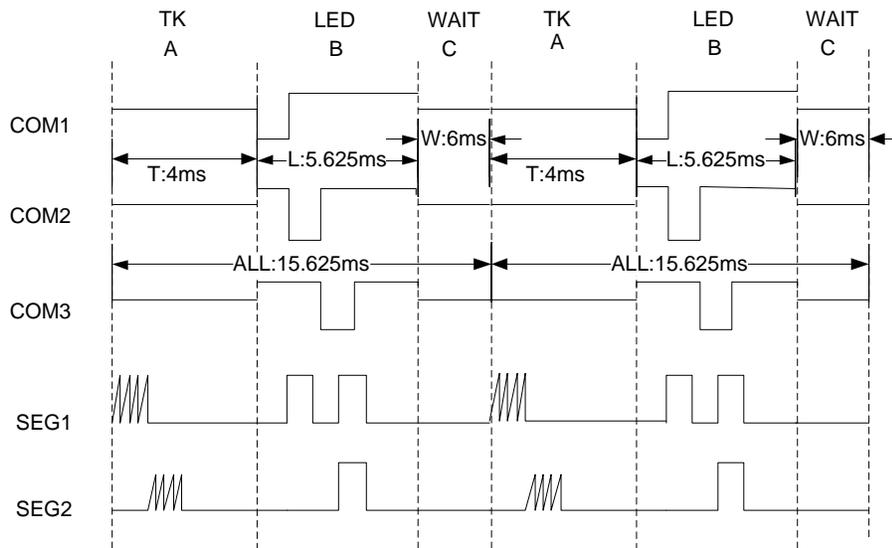
B7H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKW	-	-	-	-	-	TW.2	TW.1	TW.0
读/写	-	-	-	-	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	0	0	0

位编号	位符号	说明
2-0	TW[2:0]	<p>按键扫描出错显示</p> <p>当TKW[2:0] = 000b时, 按键1</p> <p>当TKW[2:0] = 001b时, 按键2</p> <p>当TKW[2:0] = 010b时, 按键3</p> <p>当TKW[2:0] = 011b时, 按键4</p> <p>当TKW[2:0] = 100b时, 按键5</p> <p>当TKW[2:0] = 101b时, 按键6</p> <p>当TKW[2:0] = 110b时, 按键7</p> <p>当TKW[2:0] = 111b时, 按键8</p>

**注意:** 当按键有错误标志位置起时, 除IFTKOV不会停止触摸按键的运行, 其他标志位都会停止触摸按键运行, 并将出错的通道的BIT位置起。该寄存器为只读寄存器。



TK & LED (SHARE) 01



TK & LED (SHARE) 02

图中SEG输出触摸按键波形:

ALL宽度为LED帧频宽度, T宽度为触摸按键设置宽度, L宽度为LED扫描宽度, C宽度为LED COM宽度。W宽度为触摸按键等待时间宽度。

ALL (LED帧频) = L (LED扫描宽度) + T (触摸按键时间宽度) + W (触摸按键等待时间宽度)

例: 当TK宽度调整为7ms, LED宽度调整为6.625ms, 触摸按键等待宽度调整为2ms, 则显示帧频为64Hz。

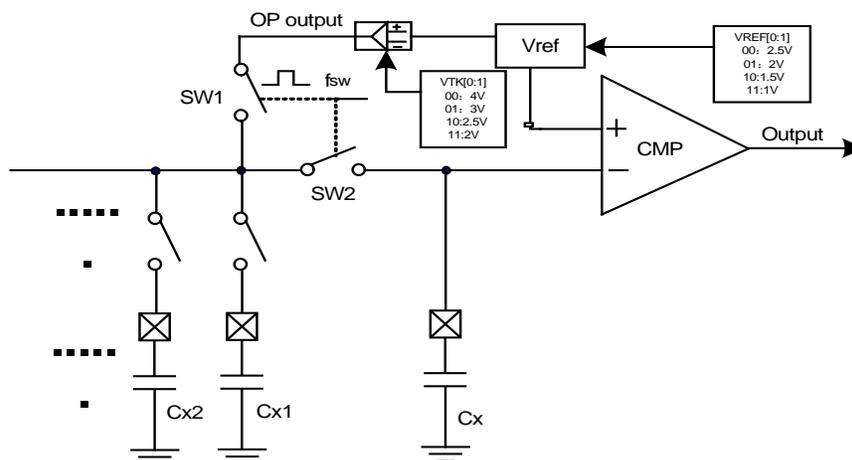
当TK宽度调整为4ms, LED宽度调整为5.625ms, 触摸按键等待宽度调整为6ms, 则显示帧频为64Hz。



Table 8.10 基准电压选择寄存器

CEH, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKVREF	VREF1	VREF0	CMPD1	CMPD0	VTK1	VTK0	TUNE1	TUNE0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-6	VREF[1:0]	内部基准电压选择位 00: $V_{REF} = 2.5V$ 01: $V_{REF} = 2V$ 10: $V_{REF} = 1.5V$ 11: $V_{REF} = 1V$
5-4	CMPD[1:0]	去抖动时间选择位 00: 约 $4 \cdot tsysclk$ 01: 约 $8 \cdot tsysclk$ 10: 约 $16 \cdot tsysclk$ 11: 约 $32 \cdot tsysclk$
3-2	VTK[1:0]	OP输出电压选择位 00: $VTK = 4V$ 01: $VTK = 3.0V$ 10: $VTK = 2.5V$ 11: $VTK = 2V$
1-0	TUNE[1:0]	放电时间调整选择位 00: 延时 $128 \cdot tsysclk$ 01: 延时 $256 \cdot tsysclk$ 10: 延时 $384 \cdot tsysclk$ 11: 延时 $512 \cdot tsysclk$



OP输出电压示意图



Table 8.11 按键扫描顺序寄存器

C2H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TKU1	TK8	TK7	TK6	TK5	TK4	TK3	TK2	TK1
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TK[8:1]	按键扫描顺序控制位 0: 跳过扫描该按键 1: 顺序扫描该按键

**注意:** 当TKU1中的某位置0时, 则当扫描按键开始时, 会跳过该按键扫描信道, 扫描按键顺序为从TKU1的TK1到TKU1的TK8, 分别代表扫描按键的8个信道。

Table 8.12 16位数据寄存器 (触摸按键数据RAM为只读寄存器)

地址		第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
500H	TK01L	D7	D6	D5	D4	D3	D2	D1	D0
501H	TK01H	D15	D14	D13	D12	D11	D10	D9	D8
502H	TK02L	D7	D6	D5	D4	D3	D2	D1	D0
503H	TK02H	D15	D14	D13	D12	D11	D10	D9	D8
504H	TK03L	D7	D6	D5	D4	D3	D2	D1	D0
505H	TK03H	D15	D14	D13	D12	D11	D10	D9	D8
506H	TK04L	D7	D6	D5	D4	D3	D2	D1	D0
507H	TK04H	D15	D14	D13	D12	D11	D10	D9	D8
508H	TK05L	D7	D6	D5	D4	D3	D2	D1	D0
509H	TK05H	D15	D14	D13	D12	D11	D10	D9	D8
50AH	TK06L	D7	D6	D5	D4	D3	D2	D1	D0
50BH	TK06H	D15	D14	D13	D12	D11	D10	D9	D8
50CH	TK07L	D7	D6	D5	D4	D3	D2	D1	D0
50DH	TK07H	D15	D14	D13	D12	D11	D10	D9	D8
50EH	TK08L	D7	D6	D5	D4	D3	D2	D1	D0
50FH	TK08H	D15	D14	D13	D12	D11	D10	D9	D8

**注意:**

- (1) OP输出电压为触摸按键供电电压,  $V_{REF}$ 为触摸按键参考电压源。
- (2) 触摸按键通过设置寄存器TUNE1和TUNE0位调节电容的放电时间, 以确保不同的C1和工作频率下, 数据寄存器值的稳定性。
- (3) 当LED与触摸按键共享开启时: TK1 - TK8作为按键用, 触摸按键功能扫描完成后退出触摸按键模块, 开启LED显示模块, 需要提前设置COM和SEG选项。扫描COM1 - COM7选中项, SEG0 - 15内容需要预存, 扫描一次后结束LED功能重新开始触摸按键功能。



## 8.2 LED驱动器

LED驱动器包含一个控制器，7个COM输出引脚和13个egment输出引脚。支持1/1 - 1/7占空比电压驱动方式。通过LED COM寄存器对驱动模式进行选择。控制器由显示数据RAM存储区和一个占空比发生器组成。

LED\_SEG0-7, 11-15脚还可以当作I/O口使用。P0SS和P2SS寄存器设置无效，SEG01，SEG02寄存器分别用于控制LED\_SEG0-7, 11-15，LED COM寄存器用于控制COM1-COM7和I/O端口模式的选择。

在上电复位、引脚复位、低电压复位或看门狗复位期间，LED被关闭。当LED被关闭时，根据LEDMODE = 0时Common为低电平，Segment输出高电平导通LED，当LEDMODE = 1时Common输出高电平，Segment都输出低电平导通LED。

在上电复位、引脚复位、低电压复位或看门狗复位期间，LED被关闭。

### 8.2.1 寄存器

Table 8.13 LED控制寄存器

D9H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
DISPCON	-	LEDON	-		LEDMODE1	LEDMODE0		MODSW
读/写	-	读/写	-		读/写	读/写		读/写
复位值 (POR/WDT/LVR/PIN)	-	0	-		0	0		0

位编号	位符号	说明
6	LEDON	<b>LED使能控制位</b> 0: 禁止LED驱动器 1: 允许LED驱动器
3-2	LEDMODE[1:0]	<b>LED模式选择位</b> 00: 共阳驱动模式 01: 共阴驱动模式 1X: 共阴/共阳同时驱动模式
0	MODSW	<b>LED共享总选择位</b> 0: SEG0x, LED COM控制位有效 1: 所有LED共享端口设置为IO

#### 注意:

LED显示模块提供三种模式选择，当模式为共阴，LED RAM COMxL (x = 1 - 7) 有效，其他无效；当模式为共阳，LED RAM COMxH (x = 1 - 7) 有效，其他无效；当模式为共阴/共阳模式时，LED RAM都有效，COMxL (x = 1 - 7) 为共阴LED的RAM，COMxH (x = 1 - 7) 为共阳时候的RAM。

**在share模式下，与触摸按键同时进行的仅为共阴模式。**

MODSW位控制了所有LED管脚的共享位；

如果MODSW = 1；则所有的LED SEG/COM口做为IO；

如果MODSW = 0；则所有的LED SEG/COM口做为显示口，继续刚才的LED扫描。

这一位对于按键扫描非常有用；程序可以先启动LED显示控制，当需要扫描按键的时候，设置MODSW = 1，用IO扫描按键，在极短的时间内，扫描按键完成后，设置MODSW = 0，继续显示LED。



Table 8.14 SEG/COM控制寄存器

D7H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
SHARECON	-	-	-	-	-	SHARE.2	SHARE.1	SHARE.0
读/写	-	-	-	-	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	0	0	0

位编号	位符号	说明
2	SHARE.2	<b>P1.2 Drive/sink使能控制位</b> 0: P1.2使用OP_SEG/IO作为使能控制位 1: P1.2使用OP_COM/IO作为使能控制位
1	SHARE.1	<b>P3.0 Drive/sink使能控制位</b> 0: P3.0使用OP_SEG/IO作为使能控制位 1: P3.0使用OP_COM/IO作为使能控制位
0	SHARE.0	<b>P3.1 Drive/sink使能控制位</b> 0: P3.1使用OP_SEG/IO作为使能控制位 1: P3.1使用OP_COM/IO作为使能控制位

Table 8.15 LED时钟控制寄存器

DCH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
DISPCLK	DCK0.7	DCK0.6	DCK0.5	DCK0.4	DCK0.3	DCK0.2	DCK0.1	DCK0.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	DCK0[7:0]	<b>LED时钟选择位</b> LED时钟宽度 = 系统时钟宽度*8*DISPCLK

Table 8.16 COM扫描宽度控制寄存器

DEH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
DISCOM	DCOM.7	DCOM.6	DCOM.5	DCOM.4	DCOM.3	DCOM.2	DCOM.1	DCOM.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
0 7-0	DCOM[7:0]	<b>LED单COM扫描宽度选择位</b> 单COM扫描宽度= LED时钟宽度*DISCOM



**注意:**

$T_{LED}$ 为LED时钟宽度,  $T_{SYS}$ 为系统时钟宽度,  $T_B$ 为LED扫描时间宽度

$$T_{LED} = T_{SYS} \times 8 \times ISPCLK$$

$$T_B = (T_{LED} \times DISCOM) \times S$$

S为LED扫描的COM数量: 扫描4COM即S = 4, 扫描5COM即S = 5, 以此类推。

以需要显示的LED帧频64Hz为例, 当LED为5COM且系统时钟为RC 16.6MHz时:

设置LEDMODE = 01时, LED显示为共阴模式;

如果设置DISPCLK为96H (DISPCLK理论上越小越好, 但是也要保证TKCOUNT, DISCOM, TKWAIT不会溢出)

$$T_{LED} = T_{SYS} \times 8 \times DISPCLK$$

$$= 60.24ns \times 8 \times 150$$

$$= 72288ns = 72.288us = 0.072288ms$$

需要设置 $T_B = 1/64s = 15.625ms$

$$DISCOM = T_B / S / T_{LED} = 15.625ms / 5 / 0.072288 = 43 = 2BH$$

以上举例LEDMODE皆为共阴模式, 其算法共阴模式和共阴/共阳模式都通用, 故不再举例再做叙述

**TK & LED**

触摸按键和LED能够达到share的效果, 效果如下:

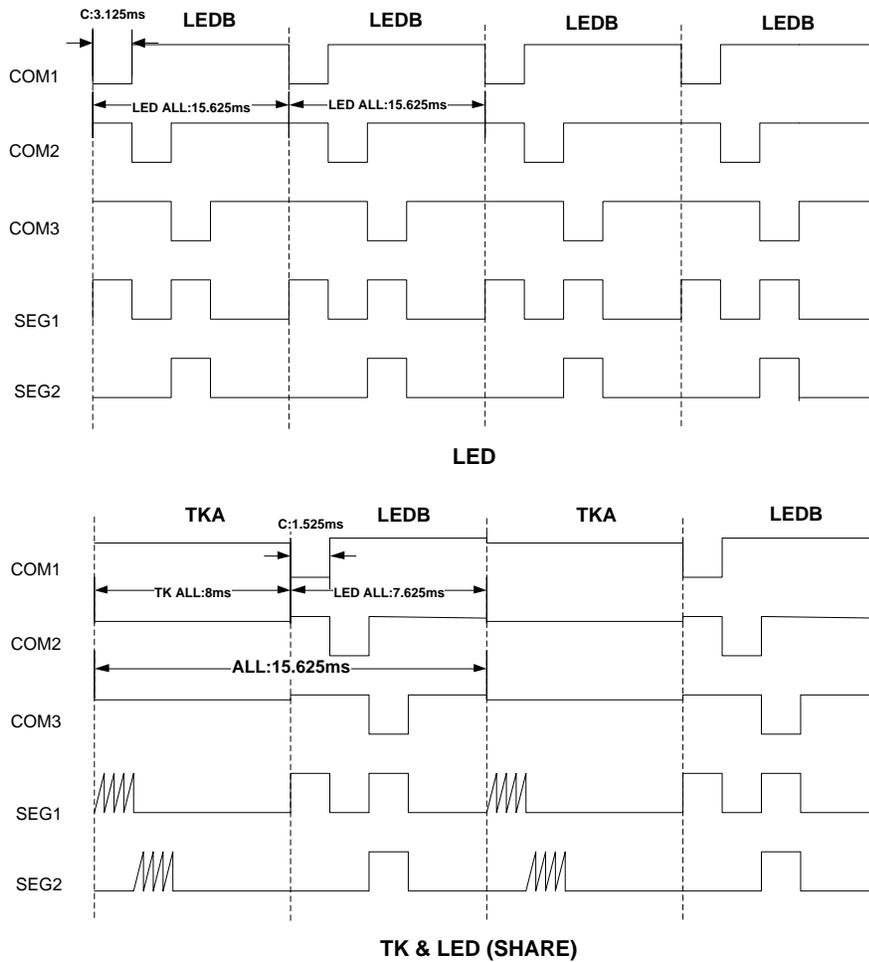




Table 8.17 SEG模式选择寄存器

DAH-ABH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
SEG01 (DAH)	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
SEG02 (DBH)	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	SEG[15:0]	<b>SEG口模式选择位 (x = 0-7)</b> 0: P1.2, P1.7, P2.0-P2.7, P3.1作为I/O 1: P1.2, P1.7, P2.0-P2.7, P3.1作为Segment (LED_S0-LED_S7, LED_S11- LED_S15)

注意: SH79F6412的VCC\_IN有最大流过电流限制(详见电气参数章节), 使能8个以上SEG, 请注意评估流过V<sub>DD</sub>的电流。

Table 8.18 COM模式选择寄存器

DDH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
LEDCOM	-	COM7	COM6	COM5	COM4	COM3	COM2	COM1
读/写	-	读/写						
复位值 (POR/WDT/LVR/PIN)	-	0	0	0	0	0	0	0

位编号	位符号	说明
6-0	COM[7:1]	<b>COM口模式选择位 (x = 0-7)</b> 0: P0.7, P1.0-P1.4, P3.1作为I/O 1: P0.7, P1.0-P1.4, P3.1 (LED_C1 - LED_C7)

注意: SH79F6412的GND有最大流过电流限制(详见电气参数章节), 使能4个以上COM, 请注意评估流过GND电流。

**LED 扫描不支持单COM模式。**

例如:

当LED工作在共阳模式

SEG01 = 0xAA; SEG02 = 0x50; LEDCOM = 0x55; 则SEG1, SEG3, SEG5, SEG7, SEG12, SEG14做为LED显示中的SEG管脚; COM1, COM3, COM5, COM7作为显示的COM管脚;

当LED RAM COM1H, COM3H, COM5H, COM7H中相应的bit为1, 则相对应的LED点亮; 当LED RAM COM1H, COM3H, COM5H, COM7H中相应的bit为0, 则相对应的LED熄灭。

当LED工作在共阴模式

SEG01 = 0xAA; SEG02 = 0x50; LEDCOM = 0x55; 则SEG1, SEG3, SEG5, SEG7, SEG12, SEG14做为LED显示中的SEG管脚; COM1, COM3, COM5, COM7作为显示的COM管脚;

当LED RAM COM1L, COM3L, COM5L, COM7L中相应的bit为1, 则相对应的LED点亮; 当LED RAM COM1L, COM3L, COM5L, COM7L中相应的bit为0, 则相对应的LED熄灭。

当LED工作在共阴/共阳模式

SEG01 = 0xAA; SEG02 = 0x50; LEDCOM = 0x55; 则SEG1, SEG3, SEG5, SEG7, SEG12, SEG14做为LED显示中的SEG管脚; COM1, COM3, COM5, COM7作为显示的COM管脚;

当LED RAM COM1L, COM3L, COM5L, COM7L中相应的bit为1, 则相对应的共阴LED点亮; 当LED RAM COM1L, COM3L, COM5L, COM7L中相应的bit为0, 则相对应的共阴LED熄灭;

当LED RAM COM1H, COM3H, COM5H, COM7H中相应的bit为1, 则相对应的共阳LED点亮; 当LED RAM COM1H, COM3H, COM5H, COM7H中相应的bit为0, 则相对应的共阳LED熄灭。



Table 8.19 辉度选择寄存器

DFH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
LIGHTCOM	-	-	-	-	-	CC3	CC2	CC1
读/写	-	-	-	-	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	0	0	0

位编号	位符号	说明
2-0	CC[3:1]	辉度选择位 000: COM宽度100% 001: COM宽度87.5% 010: COM宽度75% 011: COM宽度62.5% 100: COM宽度50% 101: COM宽度37.5% 110: COM宽度25% 111: COM宽度12.5%



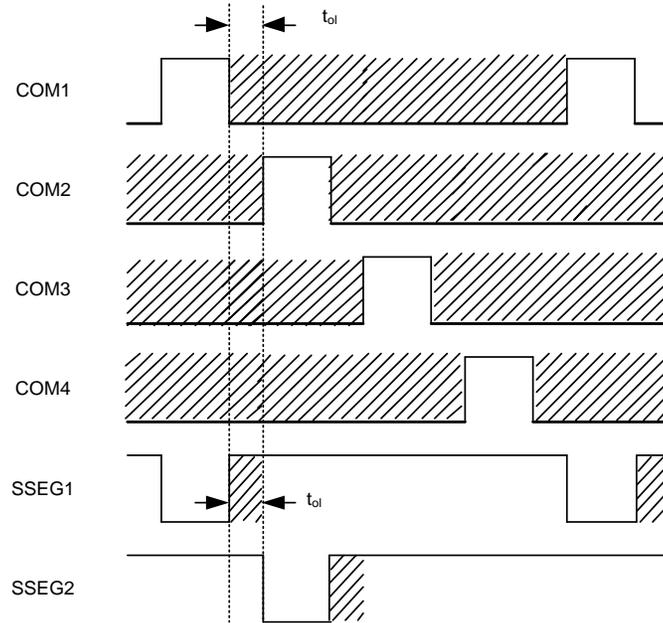
### 8.2.2 LED RAM配置

LED 1/4占空比 (LED\_C1 - C4, LED\_S1 - S8, LED\_S11 - S15)

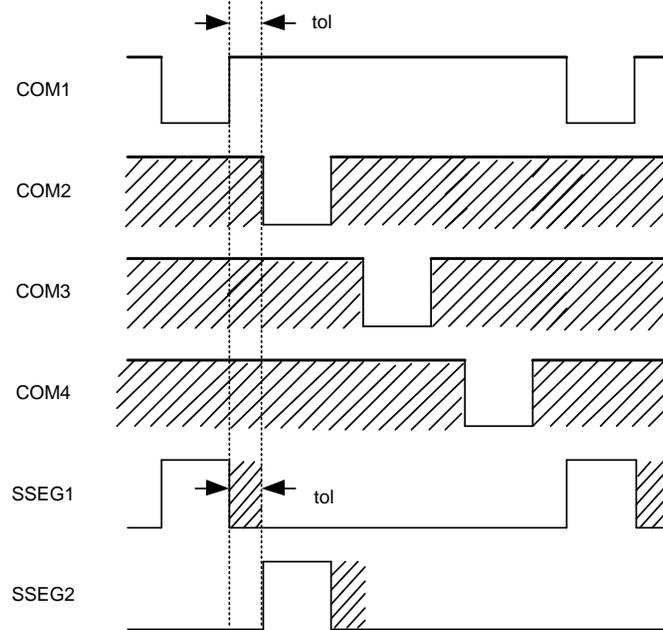
地址		7	6	5	4	3	2	1	0
530H	COM1L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
531H	COM1L	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
532H	COM1H	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
533H	COM1H	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
534H	COM2L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
535H	COM2L	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
536H	COM2H	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
537H	COM2H	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
538H	COM3L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
539H	COM3L	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
53AH	COM3H	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
53BH	COM3H	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
53CH	COM4L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
53DH	COM4L	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
53EH	COM4H	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
53FH	COM4H	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
540H	COM5L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
541H	COM5L	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
542H	COM5H	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
543H	COM5H	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
544H	COM6L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
545H	COM6L	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
546H	COM6H	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
547H	COM6H	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
548H	COM7L	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
549H	COM7L	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-
54AH	COM7H	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
54BH	COM7H	SEG15	SEG14	SEG13	SEG12	SEG11	-	-	-



LEDMODE = 00:

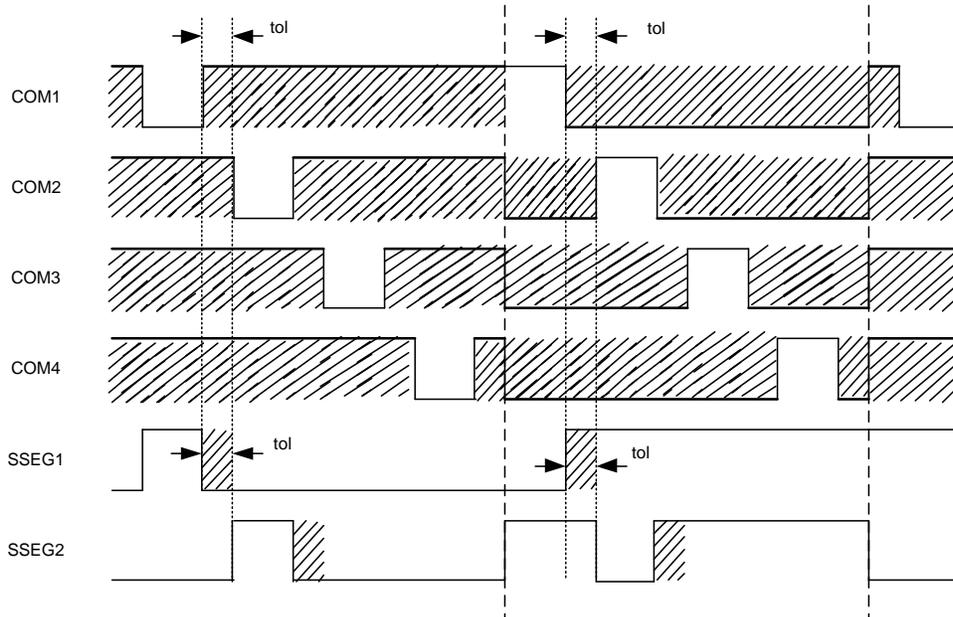


LEDMODE = 01:





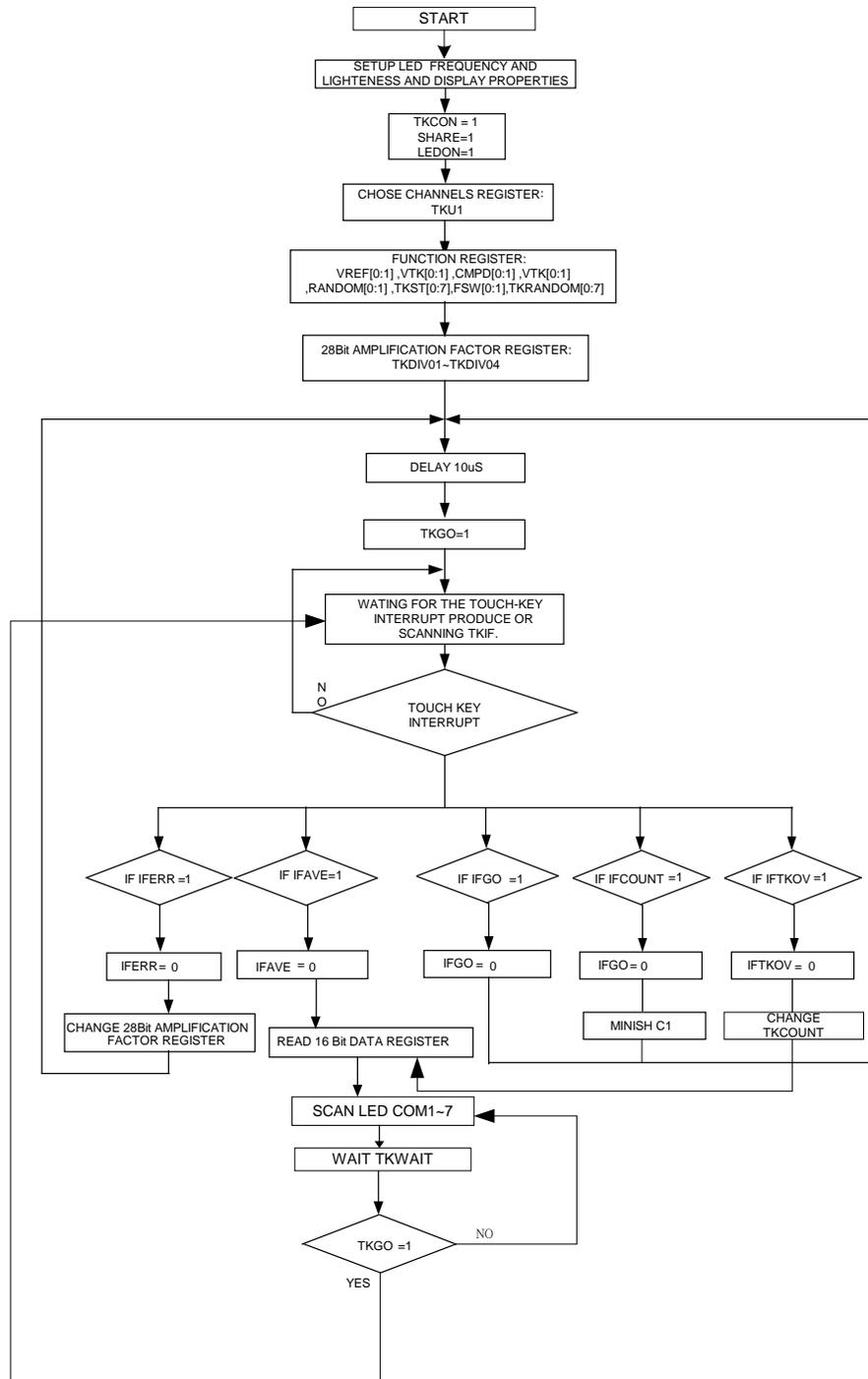
LEDMODE = 1X:



注意:  $t_{OL}$ 为LED Common信号间的重叠时间, 取值范围: 1个LED Clock。图中阴影部分为COM口floating状态。



8.3 Touch Key 触摸按键功能与LED共享功能



寄存器流程图



### 8.3.1 功能描述

SH79F6412内建触摸按键功能模块，最大能连接8个按键。当LED SHARE功能开启时，触摸按键功能最大只能连接8个按键，COM1 - COM7作为LED功能COM用。需要注意，SHARE功能开启，在触摸按键功能将数据存数完毕后，将会切换到LED扫描功能，扫描完COM1 - COM7（选中的4个COM），会有一段设置的TKWAIT时间，如果TKGO在TKWAIT结束前被置1，会自动重新开启触摸按键功能，开始重新一轮的扫描按键及LED。

#### 触摸按键启动扫描步骤：

1. LED设置：SEG01-02 = 0FFH，LEDCOM = 7FH，设置LED显示RAM。
2. 设置LED扫描频率和LED显示辉度和占空比。
3. 寄存器TKCON位置1，允许触摸按键模块工作。
4. 设置开关频率，参考电压 $V_{REF}$ ，OP输出电压，按键采样次数，需要扫描的按键顺序，扫描按键时钟宽度，开启touch key & LED share功能：FSW[0:1]，VREF[0:1]，CMPD[0:1]，VTK[0;1]，TKU1-TKU3，SHARE = 1，LEDON = 1。
5. 设置28位放大系数寄存器：DIV01-04。
6. 软件延时10uS。

7. 寄存器TKGO/DONE位置1，启动按键扫描；

程序查询中断标志位，TKGO硬件自动清0。中断标志位判断：IFERR，IFGO，IFAVE，IFLED，IFTKOV

如果IFAVE = 1，读数据寄存器500H-527H，程序保存数据结果，执行步骤9；

如果IFERR = 1，数据寄存器运算溢出错误，清IFERR和标志位，重新设置放大系数寄存器值，减小放大系数值，返回步骤6重新启动扫描；

如果IFGO = 1，按键控制器启动错误，清IFGO和标志位，返回步骤7重新启动扫描；

如果IFCOUNT = 1，按键扫描计数溢出错误，清IFCOUNT和标志位，减小电容C1或者减少平均次数。返回步骤7，重新启动扫描。

中断产生（如果TKIE = 1），或程序查询中断标志位，TKGO硬件自动清0。

8. 触摸按键功能模块：

触摸按键功能扫描完毕，IFTKOV = 0，在按键时钟宽度时间结束后，执行步骤9；

触摸按键功能扫描未完成，IFTKOV = 1，在扫描按键结束后执行步骤9。

9. LED扫描模块开启：开始扫描COM1-COM7（选中的4个COM）。

10. LED扫描完成后，等待TKWAIT时间。

11. TKWAIT时间完成前，如果TKGO/DONE位置1，在TKWAIT完成后，自动重新触摸按键扫描，如果TKGO/DONE位为0，触摸按键不启动，但是LED扫描和TKWAIT时间仍然循环，直至TKGO/DONE位置1，下个周期再次启动触摸按键。



**TOUCHKEY的工作模式表**

TOUCHKEY和LED按照是否SHARE，可以分成三种工作模式，如下表所示：

TKCON	LED_ON	SHARE	工作模式
1	0	0	TOUCHKEY单独工作
1	1	0	TOUCHKEY和LED各单独工作
1	0	1	TOUCHKEY和LED的SHARE模式，LED不工作
1	1	1	TOUCHKEY和LED的SHARE模式，LED工作
0	1	X	LED单独工作模式
0	0	X	TOUCHKEY和LED都不工作

**SEG口的SHARE列表**

如下表所示，为SEG口的控制信号

TKCON	LED_ON	SHARE	TK_STA 0: TK; 1: LED	PXSS	SEGX	PX
1	X	0	X	0	0	普通IO口
1	X	0	X	0	1	LED口
1	X	0	X	1	X	TK口
0	X	X	X	0	0	普通IO口
0	X	X	X	0	1	LED口
0	X	X	X	1	X	TK口
1	X	1	0	0	X	普通IO口
1	X	1	0	1	X	TK口
1	X	1	1	X	0	普通IO口
1	X	1	1	X	1	LED口



## 8.4 脉冲宽度调制 (PWM)

### 8.4.1 特性

- 两路12位精度PWM模块
- 提供每个PWM周期溢出中断
- 输出极性可选择

内建2路12位PWM模块。PWM模块可以产生2路周期和占空比分别可以调整的脉宽调制波形。PWMxEN (x = 1-2) 位用于使能2路PWM模块。PWMxCON (x = 1-2) 控制PWMx模块的时钟源、输出极性、周期中断等。寄存器PWMxPH/L (x = 1-2) 用于设置PWMx模块的周期，寄存器PWMxDH/L (x = 1-2) 用于设置PWMx模块的占空比。

### 8.4.2 PWM允许寄存器

Table 8.20 PWM[1:2]定时器允许寄存器

E8H, E9H, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWM1CON (E8H)	PWM1EN	PWM1S	PWM1CK1	PWM1CK0	-	PWM1IE	PWM1IF	PWM1SS
PWM2CON (E9H)	PWM2EN	PWM2S	PWM2CK1	PWM2CK0	-	PWM2IE	PWM2IF	PWM2SS
读/写	读/写	读/写	读/写	读/写	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	-	0	0	0

位编号	位符号	说明
7	PWMxEN	<b>PWMx使能位</b> 0: 禁止PWMx模块 1: 允许PWMx模块
6	PWMxS	<b>PWMx输出模式</b> 0: PWMx占空比期间输出高电平，占空比溢出后输出低电平 1: PWMx占空比期间输出低电平，占空比溢出后输出高电平
5-4	PWMxCK[1:0]	<b>PWMx时钟选择位</b> 00: 系统时钟/1 01: 系统时钟/2 10: 系统时钟/4 11: 系统时钟/8
2	PWMxIE	<b>PWMx中断使能位</b> 0: 禁止PWMx周期中断 1: 允许PWMx周期中断
1	PWMxIF	<b>PWMx中断标志位</b> 0: PWMx周期计数器没有溢出 1: PWMx周期计数器溢出，由硬件置1
0	PWMxSS	<b>PWMx引脚输出控制位</b> 0: PWMx输出禁止，用作I/O等功能 1: PWMx输出允许

**注意：**当PWMEN清0后，PWM输出立即关闭。



Table 8.21 PWM1周期寄存器PWM1PH/L

EAH, EBH, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWM1PH (EAH)	-	-	-	-	PWM1P.11	PWM1P.10	PWM1P.9	PWM1P.8
PWM1PL (EBH)	PWM1P.7	PWM1P.6	PWM1P.5	PWM1P.4	PWM1P.3	PWM1P.2	PWM1P.1	PWM1P.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
11-0	PWM1P[11:0]	PWM1数据寄存器

注意：PWM1的计数器计满至PWM1PH/L中的值后归零，若PWM1PH/L为0时，如果PWM1S为0，则PWM1引脚输出低电平；如果PWM1S为1，则PWM1引脚输出高电平。

Table 8.22 PWM2周期寄存器PWM2PH/L

ECH, EDH, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWM2PH (ECH)	-	-	-	-	PWM2P.11	PWM2P.10	PWM2P.9	PWM2P.8
PWM2PL (EDH)	PWM2P.7	PWM2P.6	PWM2P.5	PWM2P.4	PWM2P.3	PWM2P.2	PWM2P.1	PWM2P.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
11-0	PWM2P[11:0]	PWM2数据寄存器

注意：

PWM2的计数器计满至PWM2PH/L中的值后归零，若PWM2PH/L为0时，如果PWM2S为0，则PWM2引脚输出低电平；如果PWM2S为1，则PWM2引脚输出高电平。

修改寄存器PWMxPH将使得PWMx的输出在下一个周期生效。用户需先修改PWMxPL，再修改PWMxPH以修改PWM周期。

Table 8.23 PWM1占空比寄存器PWM1DH/L

E4H, E5H, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWM1DH (E4H)	-	-	-	-	PWM1D.11	PWM1D.10	PWM1D.9	PWM1D.8
PWM1DL (E5H)	PWM1D.7	PWM1D.6	PWM1D.5	PWM1D.4	PWM1D.3	PWM1D.2	PWM1D.1	PWM1D.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
11-0	PWM1D[11:0]	<p><b>PWM1占空比控制，控制PWM1波形占空比的输出时间</b></p> <ol style="list-style-type: none"> <li>当PWM1P ≤ PWM1D时               <ul style="list-style-type: none"> <li>如果PWM1S = 0，则PWM1引脚输出高电平</li> <li>如果PWM1S = 1，则PWM1引脚输出低电平</li> </ul> </li> <li>当PWM1D = 00H时               <ul style="list-style-type: none"> <li>如果PWM1S = 0，则PWM1引脚输出低电平</li> <li>如果PWM1S = 1，则PWM1引脚输出高电平</li> </ul> </li> </ol>



Table 8.24 PWM2占空比寄存器PWM2DH/L

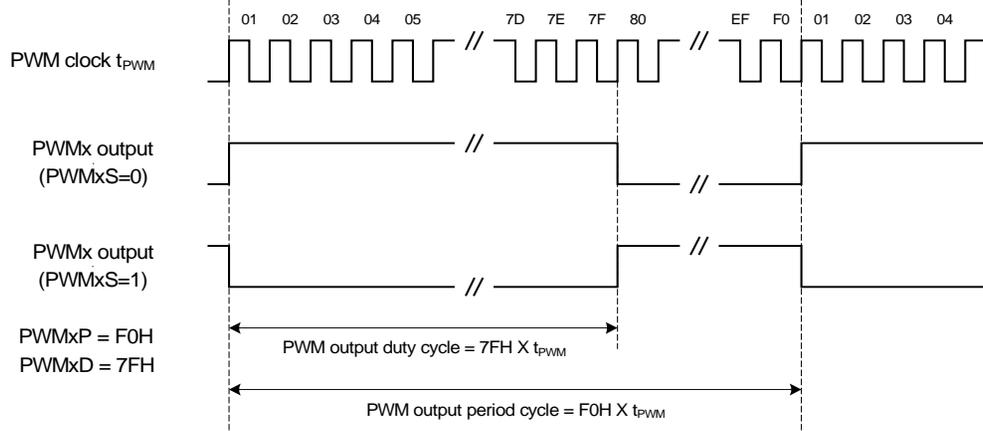
E6H, E7H, Bank1	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWM2DH (E6H)	-	-	-	-	PWM2D.11	PWM2D.10	PWM2D.9	PWM2D.8
PWM2DL (E7H)	PWM2D.7	PWM2D.6	PWM2D.5	PWM2D.4	PWM2D.3	PWM2D.2	PWM2D.1	PWM2D.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
11-0	PWM2D[11:0]	<p><b>PWM2占空比控制，控制PWM2波形占空比的输出时间</b></p> <ol style="list-style-type: none"> <li>当PWM2P ≤ PWM2D时 如果PWM2S = 0，则PWM2引脚输出高电平 如果PWM2S = 1，则PWM2引脚输出低电平</li> <li>当PWM2D = 00H时 如果PWM2S = 0，则PWM2引脚输出低电平 如果PWM2S = 1，则PWM2引脚输出高电平</li> </ol>

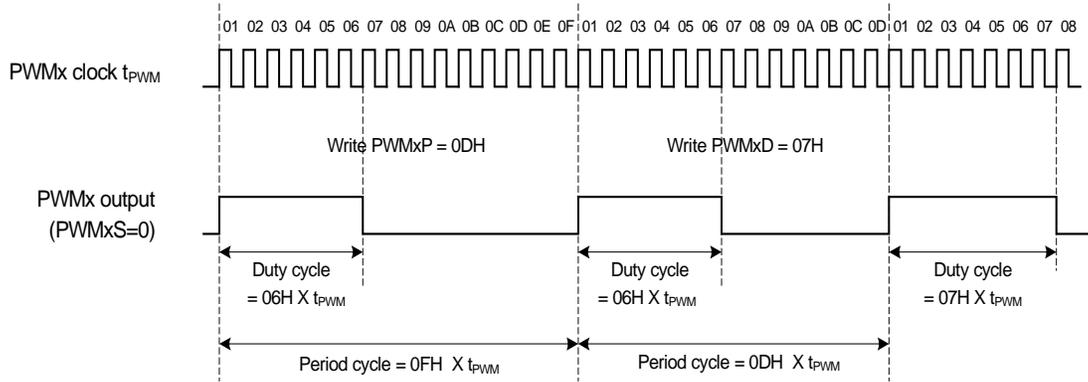
**注意：**修改寄存器PWMxDH将使得PWMx的输出在下一个周期生效。用户需先修改PWMxDL，再修改PWMxDH以修改PWM占空比。

**注意事项：**

- (1) PWMxEN位控制PWMx模块打开。
- (2) PWMxSS (x = 1-2) 位能选择端口是作为I/O端口还是PWM输出端口。
- (3) 在IEN1寄存器中的EPWMx位和PWMxCON寄存器中的PWMxIE位允许/禁止PWMx中断。
- (4) 如果PWMENx置1，PWM模块打开，但PWMxSS (x = 1-2) = 0，PWMx输出关闭，此时PWMx模块可以用作一个12bit timer，此时如果中断控制寄存器IEN1的EPWMx位置1且PWMxIF = 1，则PWMx中断照样发生。



PWM输出范例



PWM输出周期或者占空比改变范例



### 8.5 增强型通用异步收发器 (EUSART)

#### 8.5.1 特性

- 自带波特率发生器的EUART
- 波特率发生器就是一个15位向上计数器
- 增强功能包括帧出错检测及自动地址识别
- EUART有四种工作方式

#### 8.5.2 工作方式

EUART有4种工作方式。在通信之前用户必须先初始化SCON，选择方式和波特率。

在所有四种方式中，任何将SBUF作为目标寄存器的写操作都会启动发送。在方式0中由条件RI = 0和REN = 1初始化接收。这会在TXD引脚上产生一个时钟信号，然后在RXD引脚上移8位数据。在其它方式中由输入的起始位初始化接收（如果RI = 0和REN = 1）。外部发送器通信以发送起始位开始。

#### EUART工作方式列表

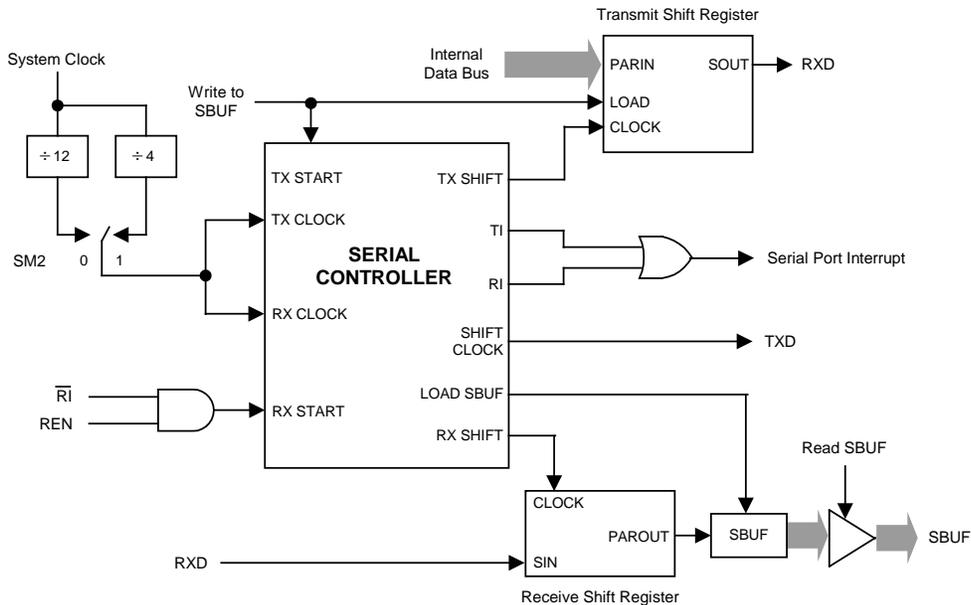
SM0	SM1	方式	类型	波特率	帧长度	起始位	停止位	第9位
0	0	0	同步	$f_{sys}/(4或12)$	8位	无	无	无
0	1	1	异步	自带波特率发生器的溢出率/16	10位	1	1	无
1	0	2	异步	$f_{sys}/(32或64)$	11位	1	1	0, 1
1	1	3	异步	自带波特率发生器的溢出率/16	11位	1	1	0, 1

#### 方式0: 同步, 半双工通讯

方式0支持与外部设备的同步通信。在RXD引脚上收发串行数据。TXD引脚发送移位时钟。SH79F6412提供TXD引脚上的移位时钟。因此这种方式是串行通信的半双工方式。在这个方式中，每帧收发8位，低位先接收或发送。

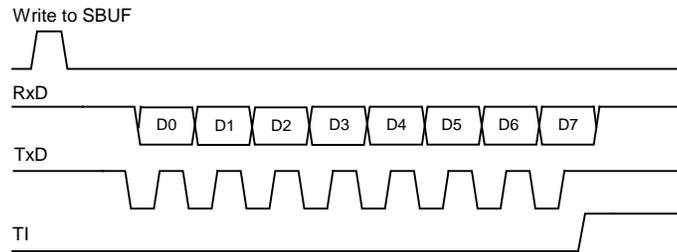
通过置SM2位 (SCON.5) 为0或1，波特率固定为系统时钟的1/12或1/4。当SM2位等于0时，串行端口以系统时钟的1/12运行。当SM2位等于1时，串行端口以系统时钟的1/4运行。与标准8051唯一不同的是，SH79F6412在方式0中有可变波特率。

功能块框图如下图所示。数据通过RXD引脚移入和移出串行端口。移位时钟由TXD引脚输出。



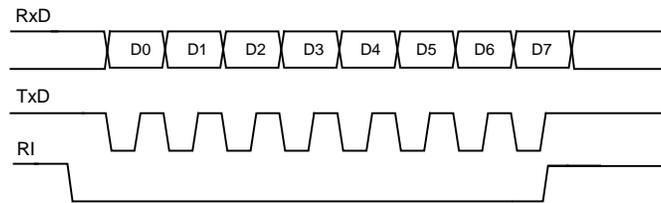


任何将SBUF作为目标寄存器的写操作都会启动发送。下一个系统时钟TX控制块开始发送。数据转换发生在移位时钟的下降沿，移位寄存器的内容逐次从左往右移位，空位置0。当移位寄存器中的所有8位都发送后，TX控制模块停止发送操作，然后在下一个系统时钟的上升沿将TI置位（SCON.1）。



Send Timing of Mode 0

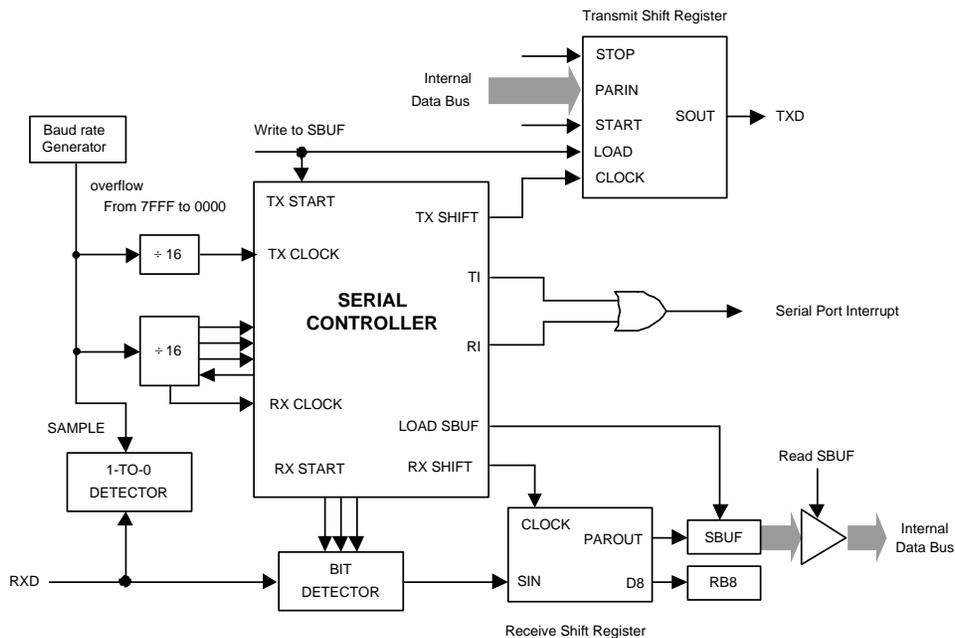
REN（SCON.4）置1和RI（SCON.0）清0初始化接收。下一个系统时钟启动接收，在移位时钟的上升沿锁存数据，接收转换寄存器的内容逐次向左移位。当所有8位数据都移到移位寄存器中后，RX控制块停止接收，在下一个系统时钟的上升沿RI置位，直到被软件清零才允许下一次接收。



Receive Timing of Mode 0

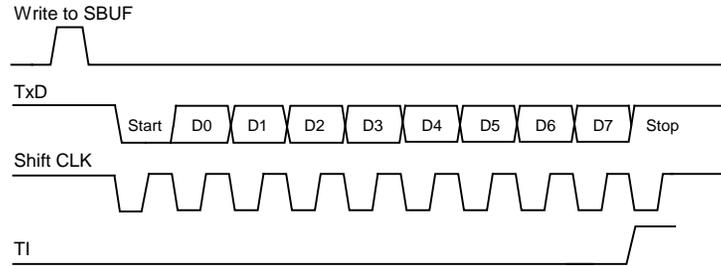
**方式1：8位EUART，可变波特率，异步全双工**

方式1提供10位全双工异步通信，10位由一个起始位（逻辑0），8个数据位（低位为第一位）和一个停止位（逻辑1）组成。在接收时，这8个数据位存储在SBUF中而停止位存储在RB8（SCON.2）中。方式1中的波特率固定为自带波特率发生器溢出率的1/16。功能块框图如下图所示。





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

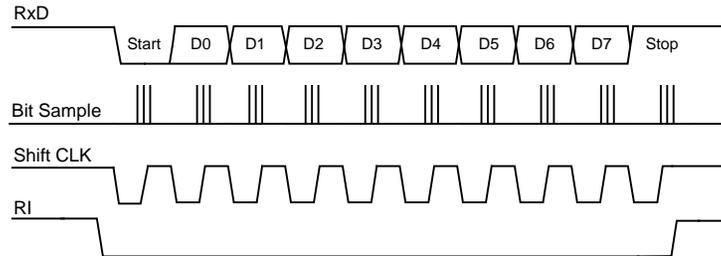


Send Timing of Mode 1

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

- (1) RI = 0
- (2) SM2 = 0或者接收的停止位 = 1

如果这些条件被满足，那么停止位装入RB8，8个数据位装入SBUF，RI被置位。否则接收的帧会丢失。这时，接收器将重新去探测RXD端是否另一个下降沿。用户必须用软件清零RI，然后才能再次接收。

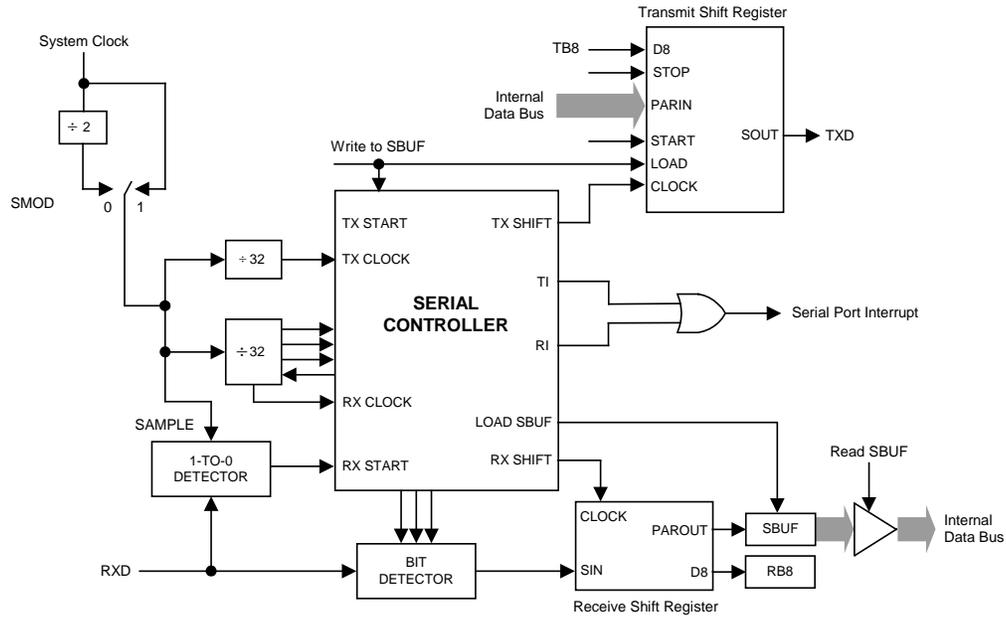


Receive Timing of Mode 1

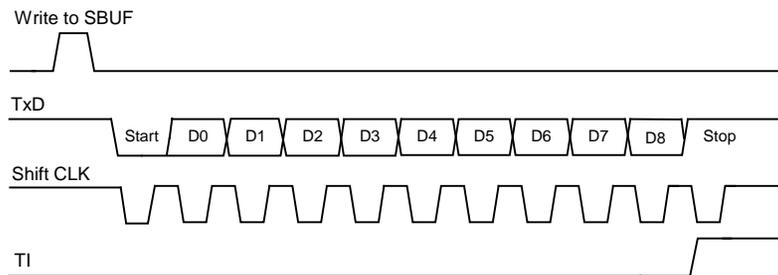


### 方式2: 9位EUART, 固定波特率, 异步全双工

这个方式使用异步全双工通信中的11位。一帧由一个起始位(逻辑0), 8个数据位(低位为第一位), 一个可编程的第9数据位和一个停止位(逻辑1)组成。方式2支持多机通信和硬件地址识别(详见**多机通讯**章节)。在数据传送时, 第9数据位(SCON中的TB8)可以写0或1, 例如, 可写入PSW中的奇偶位P, 或用作多机通信中的数据/地址标志位。当接收到数据时, 第9数据位移入RB8而停止位不保存。PCON中的SMOD位选择波特率为系统工作频率的1/32或1/64。功能块框图如下所示。



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



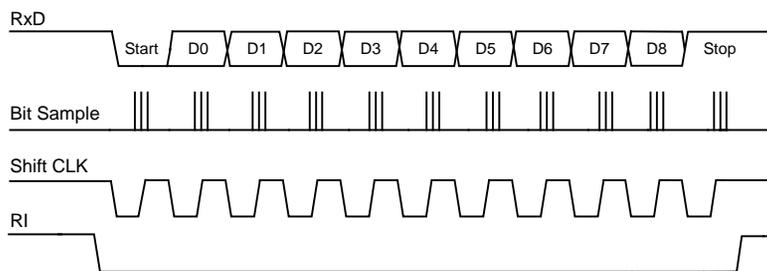
Send Timing of Mode 2



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

- (1) RI = 0
- (2) SM2 = 0或者接收的第9位 = 1，且接收的字节符合实际从机地址

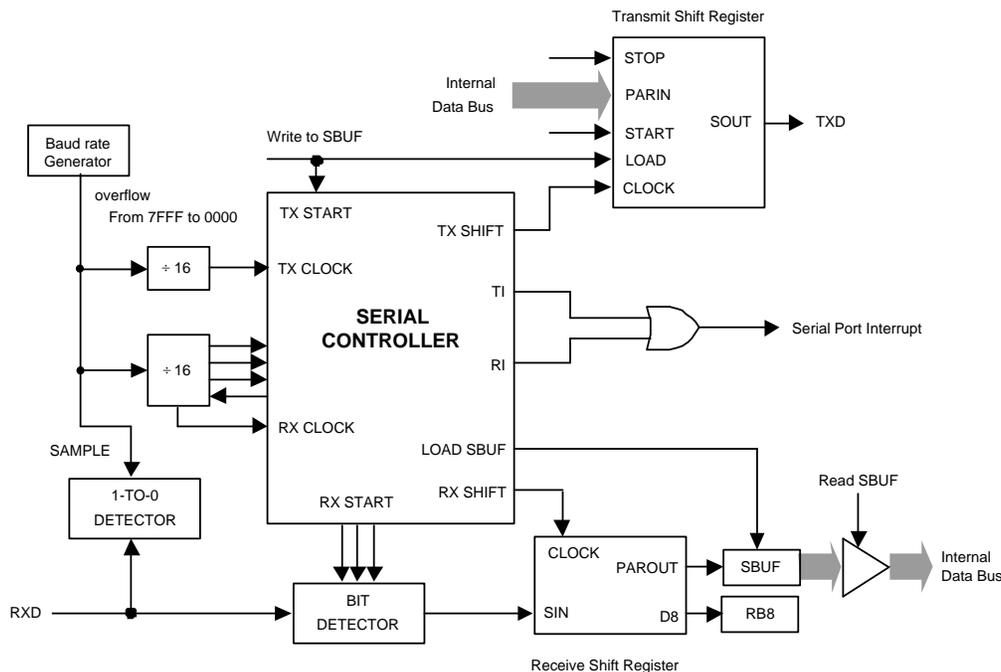
如果这些条件被满足，那么第9位移入RB8，8位数据移入SBUF，RI被置位。否则接收的数据帧会丢失。在停止位的当中，接收器回到寻找RXD引脚上的另一个下降沿。用户必须用软件清除RI，然后才能再次接收。



Receive Timing of Mode 2

### 方式3：9位EUART，可变波特率，异步全双工

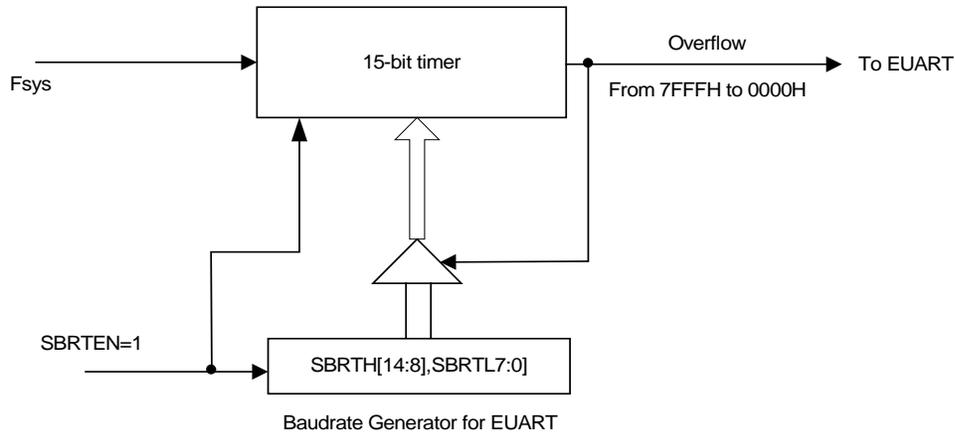
方式3使用方式2的传输协议以及方式1的波特率产生方式。





### 8.5.3 可微调波特率

EUART自带一个波特率发生器，它实质上就是一个15位递增计数器。



由图得到，波特率发生器的溢出率为  $SBRT_{overflowrate} = \frac{F_{sys}}{32768 - SBRT}$ ， $SBRT = [SBRTH, SBRTL]$

因此，EUART在各模式下的波特率计算公式如下。

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

在方式1和方式3中，波特率可微调，精度为一个系统时钟，公式如下：

$$\text{BaudRate} = \frac{F_{sys}}{16 \times (32768 - SBRT) + BFINE}$$

例如：F<sub>sys</sub> = 8MHz，需要得到115200Hz的波特率，SBRT和SFINE值计算方法如下：

$$8000000/16/115200 = 4.34$$

$$SBRT = 32768 - 4 = 32764$$

$$115200 = 8000000/(16 \times 4 + BFINE)$$

$$BFINE = 5.4 \approx 5$$

此微调方式计算出的实际波特率为115942，误差为0.64%；以往方式计算出的波特率误差为8.5%。

在方式2中，波特率固定为系统时钟的1/32或1/64，由SMOD位（PCON.7）中决定。当SMOD位为0时，EUART以系统时钟的1/64运行。当SMOD位为1时，EUART以系统时钟的1/32运行。

$$\text{BaudRate} = 2^{\text{SMOD}} \times \left( \frac{f_{SYS}}{64} \right)$$



## 8.5.4 多机通讯

### 软件地址识别

方式2和方式3具有适用于多机通讯功能。在这两个方式下，接收的是9位数据，第9位移入RB8中，之后是停止位。可以这样设定EUART：当接收到停止位，且RB8 = 1时，串行口中断有效（请求标志RI置位）。此时置位SCON寄存器的SM2，EUART工作在多机通讯模式。

在多机通讯系统中，按如下所述来使用这一功能。当主机要发送一数据块给几个从机中的一个时，先发送一地址字节，以寻址目标从机。地址字节与数据字节可用第9数据位来区别，地址字节的第9位为1，数据字节的第9位为0。

如果从机SM2为1，则不会响应数据字节中断。地址字节可以使所有从机产生中断，每一个从机都检查所接收到的地址字节，以判别本机是不是目标从机。被寻到的从机对SM2位执行清零操作，并准备接收即将到来的数据字节。当接收完毕时，从机再一次将SM2置位。没有被寻址的从机，则保持SM2位为1，不响应数据字节。

**注意：**在方式0中，SM2用来2倍频波特率。在方式1中，SM2用来检测停止位是否有效，如果SM2 = 1，接收中断不会响应直到接收到一个有效的停止位。

### 自动（硬件）地址识别

在方式2和方式3中，SM2置位，EUART运行状态如下：接收到停止位，RB8的第9位为1（地址字节），且接收到的数据字节符合EUART的从机地址，EUART产生一个中断。从机将SM2清零，接收后续数据字节。

第9位为1表明该字节是地址而非数据。当主机要发送一组数据给几个从机中的一个时，必须先发送目标从机地址。所有从机等待接收地址字节，为了确保仅在接收地址字节时产生中断，SM2位必须置位。自动地址识别的特点是只有地址匹配的从机才能产生中断，硬件完成地址比较。

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

使用自动地址识别功能时，主机可以通过调用给定的从机地址选择与一个或多个从机通信。主机使用广播地址可以寻址所有从机。有两个特殊功能寄存器，从机地址（SADDR）和地址屏蔽（SADEN）。从机地址是一个8位的字节，存于SADDR寄存器中。SADEN用于定义SADDR各位的有效与否，如果SADEN中某一位为0，则SADDR中相应位被忽略，如果SADEN中某一位置位，则SADDR中相应位将用于产生约定地址。这可以使用户在不改变SADDR寄存器中的从机地址的情况下灵活地寻址多个从机。

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

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

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

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



### 8.5.5 帧出错检测

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

**注意：**SSTAT位必须为逻辑1是访问状态位（FE，RXOV和TXCOL），SSTAT位为逻辑0时是访问方式选择位（SM0，SM1和SM2）。

#### 发送冲突

如果在一个发送正在进行时，用户软件写数据到SBUF寄存器时，发送冲突位（SCON寄存器中的TXCOL位）置位。如果发生了冲突，新数据会被忽略，不能被写入发送缓冲器。

#### 接收溢出

如果在接收缓冲器中的数据未被读取之前，RI清零，又有新的数据存入接收缓冲器，那么接收溢出位（SCON寄存器中的RXOV位）置位。如果发生了接收溢出，接收缓冲器中原来的数据将丢失。

#### 帧出错

如果检测到一个无效（低）停止位，那么帧出错位（寄存器SCON中的FE）置位。

**注意：**在发送之前TXD引脚必须被设置为输出高电平。

### 8.5.6 寄存器

Table 8.25 电源控制寄存器

87H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PCON	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL
读/写	读/写	读/写	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	-	-	0	0	0	0

位编号	位符号	说明
7	SMOD	<b>波特率加倍器</b> 0: 在方式2中，波特率为系统时钟的1/64 1: 在方式2中，波特率为系统时钟的1/32
6	SSTAT	<b>SCON[7:5]功能选择</b> 0: SCON[7:5]工作方式作为SM0，SM1，SM2 1: SCON[7:5]工作方式作为FE，RXOV，TXCOL
3-2	GF[1:0]	用于软件的通用标志位
1	PD	掉电模式控制位
0	IDL	空闲模式控制位



Table 8.26 EUART控制及状态寄存器

98H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
SCON	SM0 /FE	SM1 /RXOV	SM2 /TXCOL	REN	TB8	RB8	TI	RI
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-6	SM[0:1]	<b>EUART串行方式控制位, SSTAT = 0</b> 00: 方式0, 同步方式, 固定波特率 01: 方式1, 8位异步方式, 可变波特率 10: 方式2, 9位异步方式, 固定波特率 11: 方式3, 9位异步方式, 可变波特率
7	FE	<b>EUART帧出错标志位, 当FE位被读时, SSTAT位必须被设置位</b> 0: 无帧出错, 由软件清零 1: 帧出错, 由硬件置位
6	RXOV	<b>EUART接收溢出标志位, 当RXOV位被读时, SSTAT位必须被置位</b> 0: 无接收溢出, 由软件清零 1: 接收溢出, 由硬件置位
5	SM2	<b>EUART多处理机通讯允许位 (第9位“1”校验器), SSTAT = 0</b> 0: 在方式0下, 波特率是系统时钟的1/12 在方式1下, 禁止停止位确认检验, 任何停止位都会置位RI 在方式2和3下, 任何字节都会置位RI 1: 在方式0下, 波特率是系统时钟的1/4 在方式1下, 允许停止位确认检验, 只有有效的停止位 (1) 才能置位RI 在方式2和3下, 只有地址字节 (第9位 = 1) 才能置位RI
5	TXCOL	<b>EUART发送冲突标志位, 当TXCOL位被读时, SSTAT位必须被置位</b> 0: 无发送冲突, 由软件清零 1: 发送冲突, 由硬件置位
4	REN	<b>EUART接收器允许位</b> 0: 接收禁止 1: 接收允许
3	TB8	在EUART的方式2和3下发送的第9位, 由软件置位或清零
2	RB8	在EUART的方式1, 2和3下接收的第9位 在方式0下, 不使用RB8 在方式1下, 如果接收中断发生, 停止位移入RB8 在方式2和3下, 由第9位接收
1	TI	<b>EUART的传送中断标志位</b> 0: 由软件清零 1: 由硬件置位
0	RI	<b>EUART的接收中断标志位</b> 0: 由软件清零 1: 由硬件置位



Table 8.27 EUART数据缓冲器寄存器

99H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
<b>SBUF</b>	SBUF.7	SBUF.6	SBUF.5	SBUF.4	SBUF.3	SBUF.2	SBUF.1	SBUF.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	<b>SBUF[7:0]</b>	这个寄存器寻址两个寄存器：一个移位寄存器和一个接收锁存寄存器 SBUF的写入将发送字节到移位寄存器中，然后开始传输 SBUF的读取返回接收锁存器中的内容

Table 8.28 EUART从机地址及地址屏蔽寄存器

9AH-9BH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
<b>SADDR (9AH)</b>	SADDR.7	SADDR.6	SADDR.5	SADDR.4	SADDR.3	SADDR.2	SADDR.1	SADDR.0
<b>SADEN (9BH)</b>	SADEN.7	SADEN.6	SADEN.5	SADEN.4	SADEN.3	SADEN.2	SADEN.1	SADEN.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	<b>SADDR[7:0]</b>	寄存器SADDR定义了EUART的从机地址
7-0	<b>SADEN[7:0]</b>	寄存器SADEN是一个位屏蔽寄存器，决定SADDR的哪些位被检验 0: SADDR中的相应位被忽略 1: SADDR中的相应位对照接收到的地址被检验

Table 8.29 EUART波特率发生器寄存器

C7H, BFH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
<b>SBRTH (C7H)</b>	SBRTEN	SBRT.14	SBRT.13	SBRT.12	SBRT.11	SBRT.10	SBRT.9	SBRT.8
<b>SBRTL (BFH)</b>	SBRT.7	SBRT.6	SBRT.5	SBRT.4	SBRT.3	SBRT.2	SBRT.1	SBRT.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	<b>SBRTEN</b>	EUART波特率发生器使能控制位 0: 关闭（默认） 1: 打开
6-0, 7-0	<b>SBRT[14:0]</b>	EUART波特率发生器计数器高7位和低8位寄存器

Table 8.30 EUART波特率发生器微调寄存器

BEH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
<b>SFINE</b>	-	-	-	-	SFINE.3	SFINE.2	SFINE.1	SFINE.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0

位编号	位符号	说明
3-0	<b>SFINE[3:0]</b>	EUART波特率发生器微调数据寄存器



## 8.6 TWI串行通讯接口

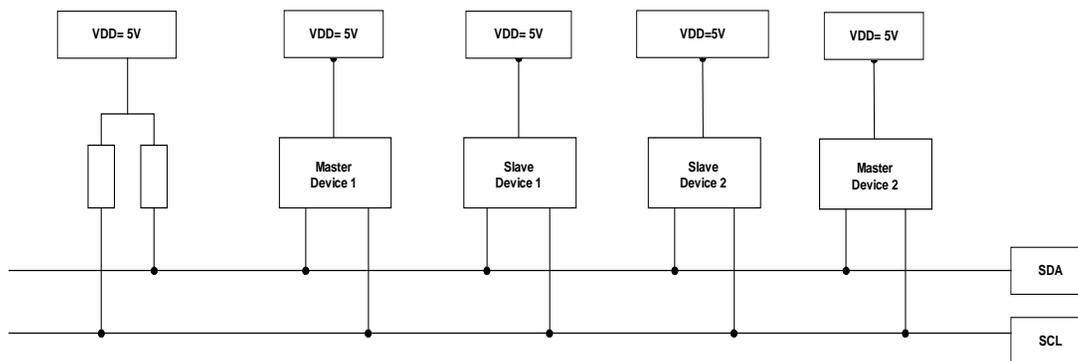
### 8.6.1 特性

- 两线模式，简单快捷
- 支持主机模式（Master）和从机模式（Slave）
- 允许发送数据（Transmitter）和接收数据（Receiver）
- 支持多主机通讯的仲裁功能
- 具有低电平总线超时判断（Timeout）
- 空闲模式下可唤醒系统
- 地址可编程

TWI串行总线采用两根线（SDA和SCL）在总线和装置之间传递信息。SH79F6412完全符合TWI总线规范，自动对字节进行传输进行处理，并对串行通讯进行跟踪。

TWI功能需要16.6MHz系统频率，当前系统周期为128KHz/32.768K时，OSC2 16.6MRC不能关闭，否则TWI无法通讯。

典型TWI通讯如下图所示，最高支持128个不同的器件进行通讯。



### 8.6.2 数据传输格式

#### 数据传输格式

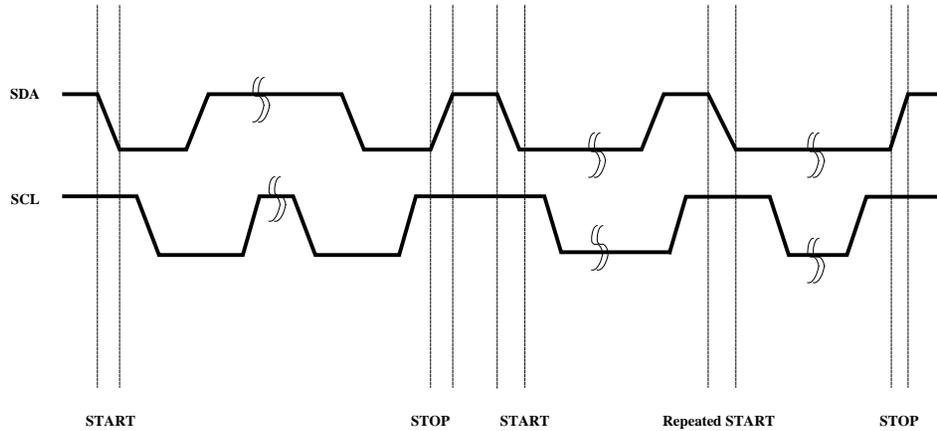
数据传输中数据线上每一位的传输均需要时钟线上一个脉冲。在时钟高电平时数据线应保持稳定。但发送起始条件和终止条件时不需要遵守此规则。

和I2C通讯协议相似，TWI定义了两个特殊的波形：起始条件和终止条件。在时钟线为高电平时数据线的下降沿定义为起始条件；在时钟线为高电平时数据线的上升沿定义为终止条件。起始条件和终止条件均由主机发出。

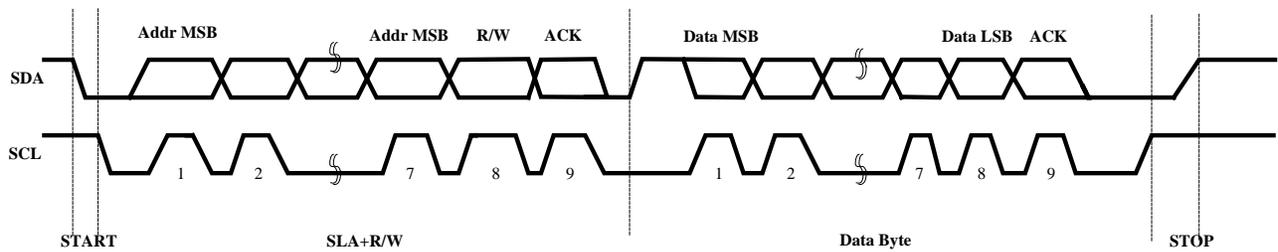
主机可以发起和终结一次传输。当主机发送一个起始条件时开始一次传输，发送一个终止条件时结束本次传输。在起始条件和终止条件之间，总线定义为“忙碌”状态。其它主机不应该去试图发起传输。在“忙碌”状态下，如果主机再次发送起始条件，则定义为“重复起始条件”，表示主机希望不放弃总线的情况下开始一次新的传输。发送重复起始条件后，总线仍处于“忙碌”状态，一直到总线出现终止条件。鉴于重复起始条件和起始条件性质完全一致，除非特别声明，本文中采用起始条件来代替两者。

所有数据包（包括地址包）均有9位组成，包括1个字节和一个应答位。主机负责发出时钟和起始及终止条件，接收者负责给出应答信号。接收者通过在第九个时钟脉冲处将数据线拉低发出“应答（ACK）”信号；或维持第九个脉冲处维持高电平表示“不应答（NACK）”信号。当接收方接收到最后一个字节，或因某种原因无法继续接收数据时，应回应“不应答（NACK）”信号。TWI采用从高到低逐位进行传输。

一次传输通常包括一个起始条件，地址+读/写，一个或多个数据包和一个终止条件。仅包含起始条件和终止条件的数据格式是不合通讯规则的。值得注意的是“线与”结构给主机和从机之间的握手信号提供了方便。当主机相对太快或从机需要处理其它事务时，从机可以通过拉低时钟线来拉长时钟线的低电平时间，从而降低通讯频率。从机可以拉长时钟线低电平周期但不会影响到时钟线高电平的周期。



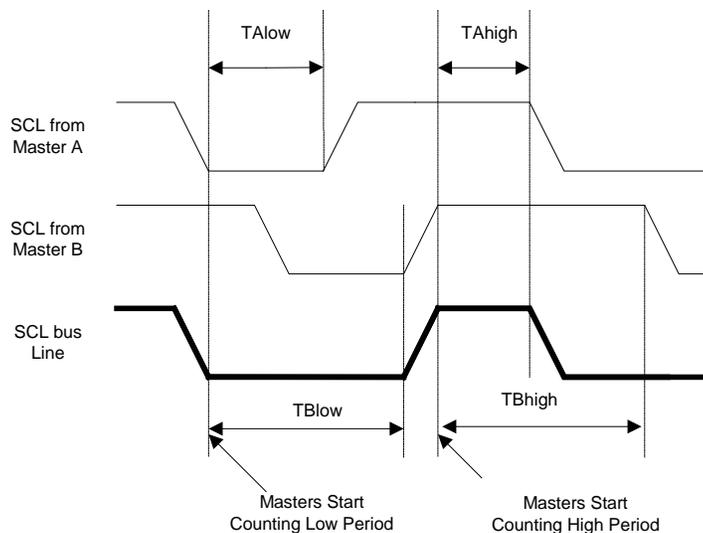
在产生应答信号时，SH79F6412拉低SDA信号线。中断标志位置起期间，SH79F6412拉低SCL信号线，释放SDA信号线。中断处理完毕后清除TWINT标志，释放SCL信号线。



### 时钟同步

当多个主机同时希望控制总线时，总线将依据“线与”原则决定时钟线高低电平。对于所有参与传输的主机来说，定义清楚每一个时钟脉冲的起始是相当重要的。

时钟线电平的由高到低跳变将导致所有参与传输的器件开始低电平计时。每一个器件计时达到自己低电平要求时释放时钟线，在时钟线变为高电平之前进入高电平等待期；当所有器件均计满低电平周期时，时钟线才变为高电平。之后所有器件开始对高电平进行计时，第一个计满高电平周期的器件将拉低时钟线，进入下一个时钟周期。





### 数据仲裁

主机只有在总线处于“空闲”状态时才能开始一次传输。两个或多个主机可能在最小保持时间 ( $t_{HOLD:STA}$ ) 内同时发送起始条件，从而在总线上只看到一个起始条件。

由于发送起始条件的主机无法知道是否有其它主机在竞争总线，只能靠时钟高电平时对数据线的仲裁判断哪个主机占用总线。当有主机传输低电平时，传输高电平的主机将失去仲裁，必须放弃总线。

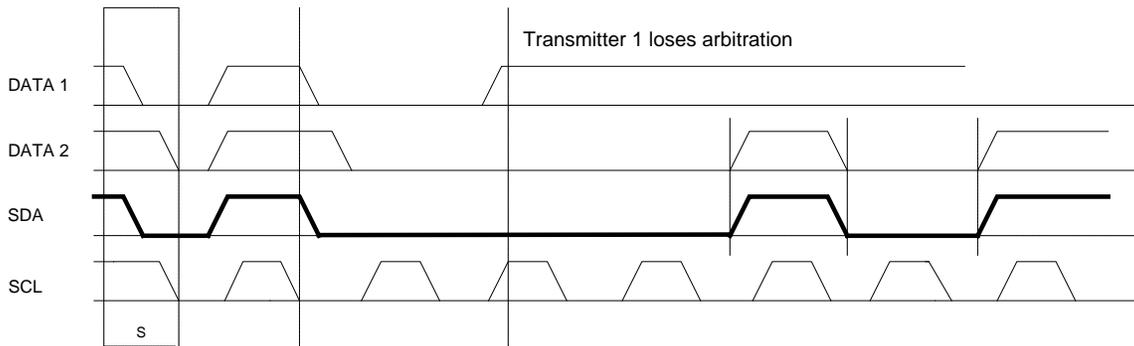
失去仲裁的主机将继续发送时钟，直到当前传输字节发送完毕。当两个主机同时访问一个从机时，可能会顺利通过地址阶段，在传输数据时将继续进行仲裁。这种机制要求所有TWI器件在进行数据传输时可以检测数据线上的真实状态。

如果该主机同时开启了从机模式，在发送地址阶段失去仲裁后应检测线上的地址是否与自己相匹配；如果是对自己的访问，应立即切换到从机模式，接收信息。

每次传输中，仍要检测线上的“重复起始条件”，当检测到并非自己发出的“重复起始条件”时，应立即退出当前传输。

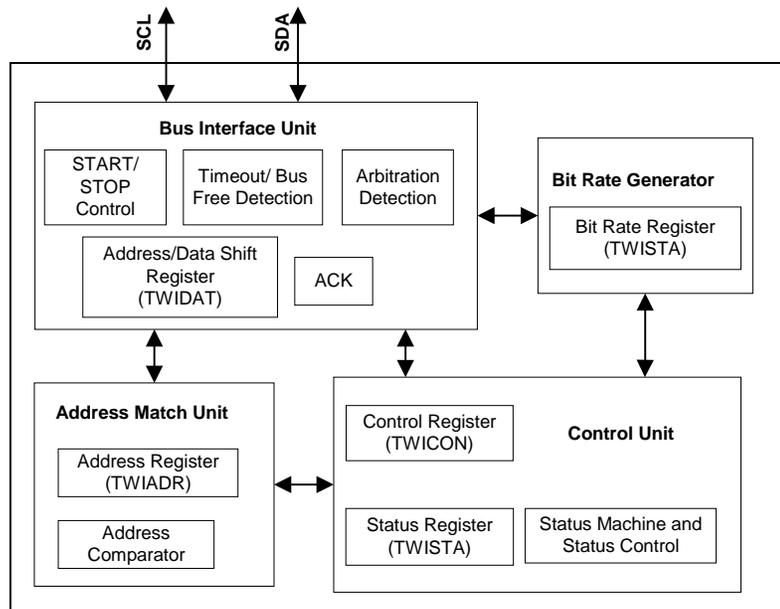
仲裁不应发生在如下情况：

- (1) 重复起始条件和数据
- (2) 终止条件和数据
- (3) 重复起始条件和终止条件



### 8.6.3 功能描述

下图描述了TWI通讯模块的详细结构。





## 总线接口单元

总线接口单元包括数据和地址移位寄存器（TWIDAT），开始/终止条件控制器，仲裁和总线超时检测单元。

寄存器TWIDAT存储了即将发送的数据或地址和接收到的数据和地址。

开始/终止条件控制器负责发送和检测总线上的开始条件，重复开始条件和终止条件。

如果SH79F6412已经作为主机开始一次传输，仲裁单元将始终检测是否有仲裁发生。当失去仲裁时，控制单元可以进行合适的动作，并产生相应的状态码。

SH79F6412在传输数据/地址时，必须在SCL由低跳高前维持数据稳定。

SH79F6412在传输ACK/NACK时，在SCL由低跳高后产生TWINT中断，并在SCL由高跳低时拉低SCL，在TWINT中断清零时释放SCL。

SH79F6412在传输ACK/NACK信号时，若TWINT已被清零，SCL仍为高电平时，SDA产生跳变，则重新产生TWINT中断，状态为00H。SH79F6412当前通讯终止，该状态与普通00H状态处理一致。

SH79F6412在传输ACK/NACK信号时，若TWINT未被清零，SCL仍为高电平时，SDA产生跳变，则状态直接切换到00H，不会再次产生中断。SH79F6412作为从机进入该状态，则当前通讯终止，可发生STA开始主机传输，或重新接受STA+ADR对自己地址的访问。SH79F6412作为主机进入该状态，则当前通讯终止，可发生STA开始主机传输，或重新接受STA+ADR对自己的访问。

SH79F6412在当前通讯终止后，不会再参与当前传输。SH79F6412若作为主机存在，请开启EFREE功能，防止进入逻辑死区。

SH79F6412规定总线维持为高电平超过50us时为“空闲”状态，释放总线。该功能仅适用于一个数据包传输过程中(8+1个位)。SH79F6412处于从机发送模式，且所传输的第一个字节为低电平时适用该功能。起始条件（STA、RSTA）不适用于该功能。SH79F6412产生中断，寄存器TWICON中的TFREE会被置位（如果控制位EFREE已置位）。

如果时钟线SCL被从机拉低时，通讯会暂时中止；而主机也没有办法将时钟线拉高。为解决此问题，TWI协议规定参与传输的所有器件将时钟线低电平超过25ms时定义为“总线超时”，寄存器TWICON中的TOUT会被置位（如果控制位ETOT已置位）。10ms内TWI模块将复位，释放总线。

## 频率生成单元

在主机模式下，可通过寄存器TWICON的CR[1:0]来设定通讯频率。通讯频率共有四个：4KHz、16KHz、64KHz、100KHz（4MHz时钟源）。

## 地址匹配单元

地址匹配单元检验所收到的地址是否与寄存器TWIADR中的7位地址相匹配。如果通用地址使能位GC被置位，也将检测是否与通用地址00H相匹配。当地址匹配时，控制单元将产生合适的动作及相应的状态码。

## 控制单元

控制单元监视TWI总线，并依据控制寄存器TWICON的设置进行相应的回应。当TWI总线有需要应用层注意的事件时，TWI中断标志被置起，标明当前事件的状态码会被写入状态寄存器TWISTA。状态寄存器TWISTA只表示TWI通讯中断产生时的通讯状态信息；其它情况下状态寄存器内是一个用于表示没有有效状态码的状态码。在中断清除之前，时钟线将维持低电平。应用软件可在处理完任务后才允许TWI通讯继续。

## 8.6.4 传输模式

TWI通讯是以字节为基础和中断驱动的通讯总线。诸如接收到一个字节或发送一个开始条件的所有总线事件均会产生一个中断。所以在字节传输期间，应用软件可以进行其它的操作。需注意的是，控制寄存器TWICON中的TWI使能位ENTWI和中断控制寄存器IEN0中的所有中断控制位EA和TWI中断控制位ETWI将共同决定TWI中断标志TWINT被置位时是否会产生中断。如果ETWI或EA未置位，应用软件必须对TWINT标志进行枚举检测才能知道是否有TWI事件发生。

当TWINT位置起时，表示一次TWI传输已完成，等待应用软件的回应，此时状态寄存器TWISTA包含了当前的状态。应用软件可通过寄存器TWICON和TWISTA决定TWI进行哪种通讯。

下面将分别介绍TWI通讯的四种主要模式，并对所有可能的状态码进行了描述。下图中有如下缩写：

S	: 开始条件	$\bar{A}$	: 无应答位
Rs	: 重复开始条件	DATA	: 8位数据
R	: 读控制位	P	: 终止条件
W	: 写控制位	SLA	: 从机地址
A	: 应答位		

圆形用于表示中断标志已被置起。其中的数字表示当前状态寄存器TWISTA中被掩去低三位的状态码。在TWINT被清除之前，TWI通讯会暂停，应用软件必须决定是继续通讯还是终止当前传输。对每一个状态码，所需要的软件动作和随后的传输细节均有描述。



**主机发送模式**

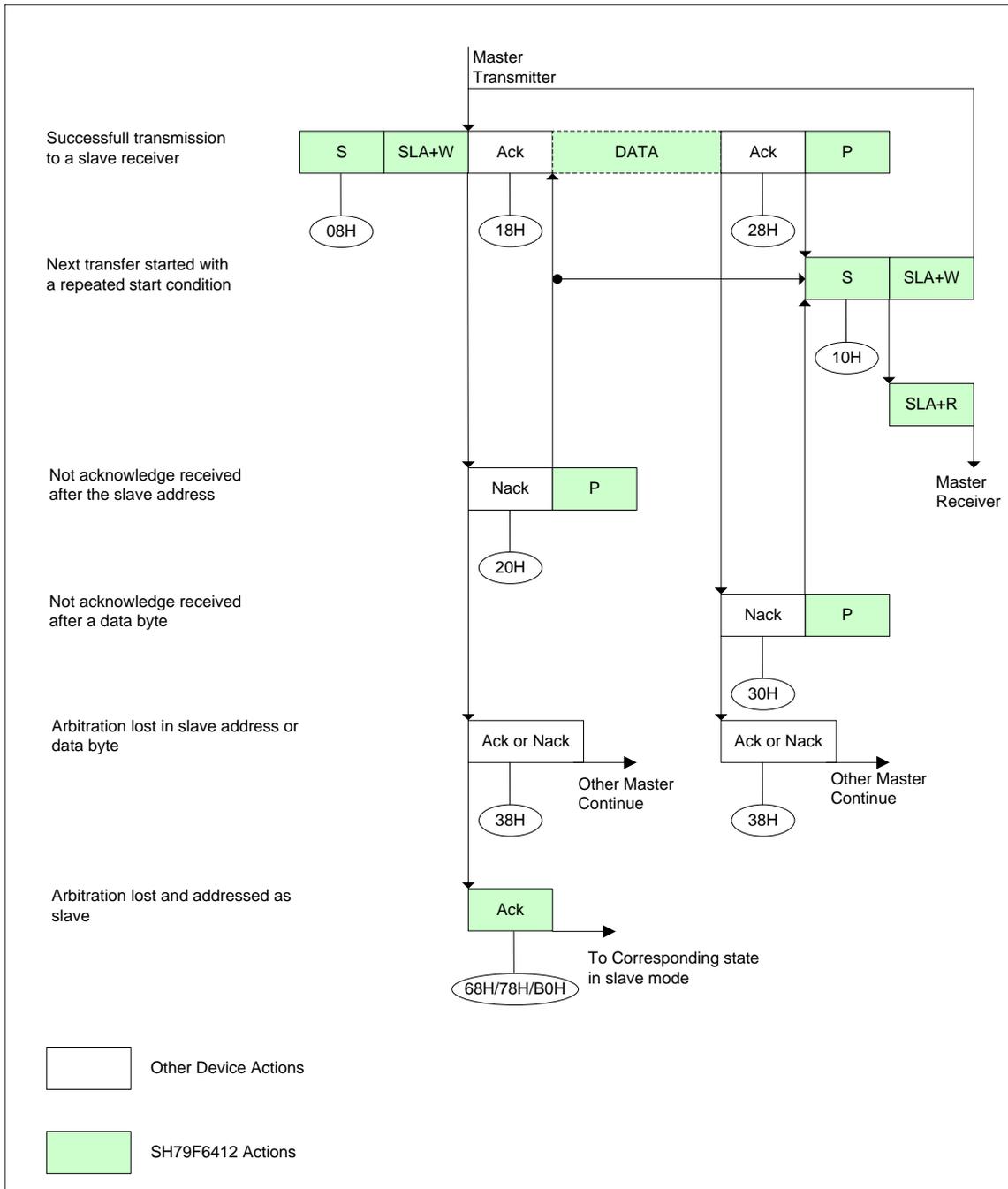
主机发送模式中,主机发送一系列数据到从机。为进入主机发送模式,一个开始条件,随后一个从机地址+写控制字(SLA+W)地址包表示进入主机发送模式(MT)。

通过设置控制寄存器TWICON中的ENTWI和STA,清除STO和TWINT,TWI逻辑将检测TWI总线并在允许时发出一个开始条件(STA)。当开始条件(STA)传输完毕,通讯中断(TWINT)被置起,状态寄存器(TWISTA)为08H,中断服务程序应将 从机地址和写控制字(SLA+W)写入数据寄存器TWIDAT。在开启下一个传输前清除TWINT标志。

当从机地址和写控制字传输完毕并收到一个“应答”信息时,中断(TWINT)被置起,状态寄存器TWISTA中有几个可能的状态:对主机模式有18H,20H和38H,对从机模式有68H,78H和B0H。

**主机发送模式状态码**

状态码	TWI总线和 硬件接口状态	应用软件响应				TWI执行的下一个动作	
		读/写数据寄存器 TWIDAT操作	控制位操作				
			STA	STO	TWINT		AA
08H	已发送开始条件	写入SLA+W	X	0	0	X	发送SLA+W,接收ACK
10H	已发送重复开始条件	写入SLA+W	X	0	0	X	发送SLA+W,接收ACK
		写入SLA+R	X	0	0	X	发送SLA+R,TWI将切换到主机接收模式
18H	已发送SLA+W; 已接收ACK	写入数据字节	0	0	0	X	发送数据,接收ACK
		无TWIDAT动作	1	0	0	X	发送重复开始条件
			0	1	0	X	发送终止条件;清除STO标志
			1	1	0	X	发送终止条件,之后发送起始条件;STO被清除
20H	已发送SLA+W; 已接收NACK	写入数据字节	0	0	0	X	发送数据,接收ACK
		无TWIDAT动作	1	0	0	X	发送重复开始条件
			0	1	0	X	发送终止条件;清除STO标志
			1	1	0	X	发送终止条件,之后发送起始条件;STO被清除
28H	已发送TWIDAT 中数据; 已接收ACK	写入数据字节	0	0	0	X	发送数据,接收ACK
		无TWIDAT动作	1	0	0	X	发送重复开始条件
			0	1	0	X	发送终止条件;清除STO标志
			1	1	0	X	发送终止条件,之后发送起始条件;STO被清除
30H	已发送TWIDAT 中数据; 已接收NACK	写入数据字节	0	0	0	X	发送数据,接收ACK
		无TWIDAT动作	1	0	0	X	发送重复开始条件
			0	1	0	X	发送终止条件;清除STO标志
			1	1	0	X	发送终止条件,之后发送起始条件;STO被清除
38H	在SLA+W或数据 传输中丢失仲裁位	无TWIDAT动作	0	0	0	X	TWI总线被释放;进入非寻址从机模式
			1	0	0	X	在总线空闲时发送开始条件





**主机接收模式**

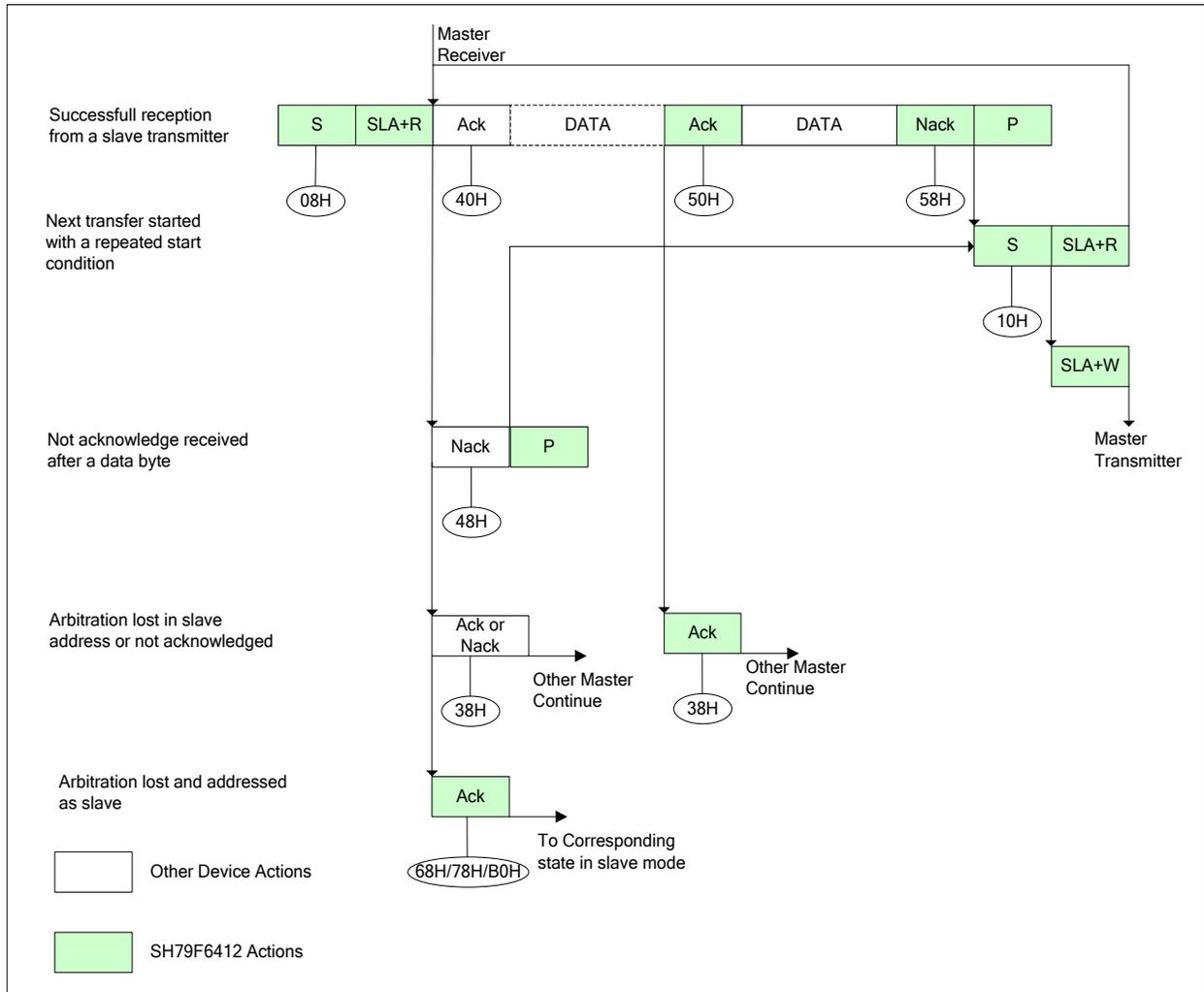
主机接收模式中，主机从从机接收一系列数据。为进入主机接收模式，一个开始条件，随后一个从机地址+读控制字（SLA+R）地址包表示进入主机发送模式（MR）。

通过设置控制寄存器TWICON中的ENTWI和STA，清除STO和TWINT，TWI逻辑将检测TWI总线并在允许时发出一个开始条件（STA）。当开始条件（STA）传输完毕，通讯中断（TWINT）被置起，状态寄存器（TWISTA）为08H，中断服务程序应将从机地址和读控制字（SLA+R）写入数据寄存器TWIDAT。在开启下一个传输前清除TWINT标志。

当从机地址和写控制字传输完毕并收到一个“应答”信息时，中断（TWINT）被置起，状态寄存器TWISTA中有几个可能的状态：对主机模式有40H，48H和38H，对从机模式有68H，78H和B0H。

**主机接收模式状态码**

状态码	TWI总线和硬件接口状态	应用软件响应				TWI执行的下一个动作	
		读/写数据寄存器TWIDAT操作	控制位操作				
			STA	STO	TWINT		AA
08H	已发送开始条件	写入SLA+R	X	0	0	X	发送SLA+R，接收ACK
10H	已发送重复开始条件	写入SLA+R	X	0	0	X	发送SLA+R，接收ACK
		写入SLA+W	X	0	0	X	发送SLA+W，TWI将切换到主机发送模式
38H	发送SLA+R或NACK时失去仲裁	无TWIDAT动作	0	0	0	X	TWI总线被释放；进入非寻址从机模式
			1	0	0	X	在总线空闲时发送开始条件
40H	已发送SLA+R；已接收ACK	无TWIDAT动作	0	0	0	0	接收数据，返回NACK
			0	0	0	1	接收数据，返回ACK
48H	已发送SLA+R；已接收NACK	无TWIDAT动作	1	0	0	X	发送重复开始条件
			0	1	0	X	发送终止条件；清除STO标志
			1	1	0	X	发送终止条件，之后发送起始条件；STO被清除
50H	数据已接收；已回应ACK	读取数据	0	0	0	0	接收数据，返回NACK
			0	0	0	1	接收数据，返回ACK
58H	数据已接收；已回应NACK	读取数据	1	0	0	X	发送重复开始条件
			0	1	0	X	发送终止条件；清除STO标志
			1	1	0	X	发送终止条件，之后发送起始条件；STO被清除





**从机发送模式**

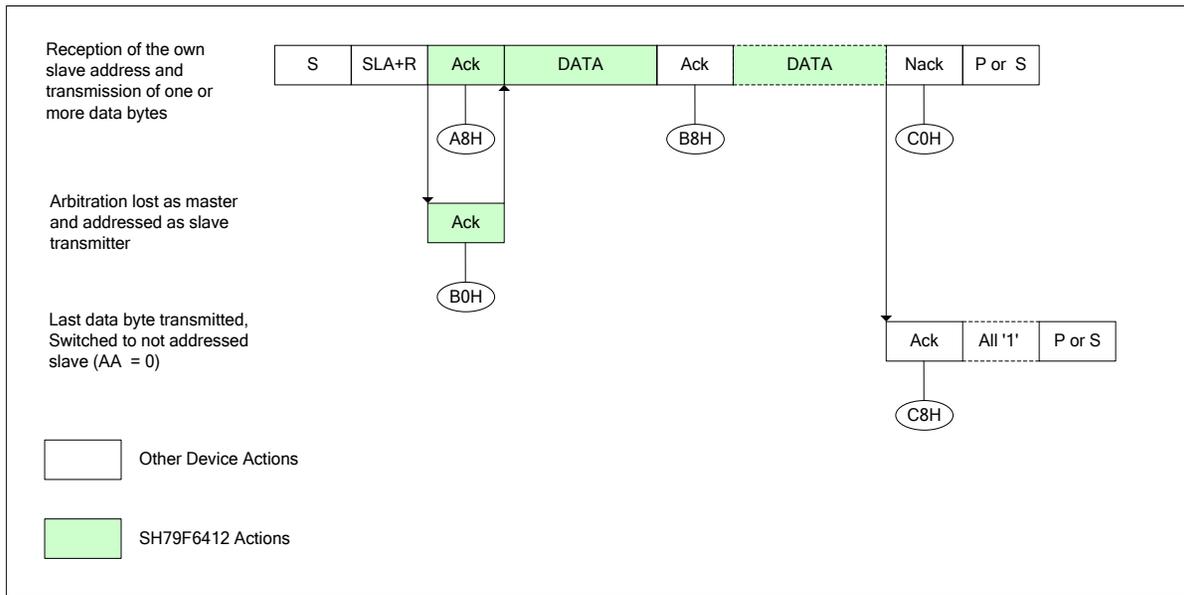
从机发送模式中，从机发送一系列数据到主机。为初始化从机发送模式，必须对控制寄存器TWICON和地址寄存器TWIADR进行初始化：置位控制寄存器TWICON中的ENTWI和AA，清除STA、STO和TWINT；地址寄存器TWIADR中高7位为SH79F6412准备相应的地址。如果GC置位，SH79F6412也将响应通用地址（00H）；否则将不响应通用地址。

在TWIADR和TWICON初始化后，SH79F6412将等待总线对自己地址或通用地址（如果GC被置位）的响应。如果方向标志位是“读”，则TWI进入从机发送模式，否则将进入从机接收模式。在地址和读标志位接收完毕后，中断标志（TWINT）置位，状态寄存器TWISTA有效。

在传输中，如果将应答使能位“AA”清零，TWI将传送最后一个字节，并依据主机接收方发送的应答或不应答信息位进入C0H或C8H状态。总线将切换到非地址从机模式，不在响应主机传输。从而主机接收方将接收到一串“1”。最后一个字节发送完毕后，如果主机仍需额外的数据（传输“应答”信号），则进入C8H状态。

**从机发送模式状态码**

状态码	TWI总线和硬件接口状态	应用软件响应				TWI执行的下一个动作	
		读/写数据寄存器TWIDAT操作	控制位操作				
			STA	STO	TWINT		AA
A8H	已收到自己SLA+R；已经回应ACK	写入数据字节	X	0	0	0	发送最后数据；等待ACK回应
			X	0	0	1	发送数据；等待ACK回应
B0H	作为主机发送SLA+R/W时失去仲裁，收到主机SLA+R；已回应ACK	写入数据字节	X	0	0	0	发送最后数据；等待ACK回应
			X	0	0	1	发送数据；等待ACK回应
B8H	已发送TWIDAT数据；已接收ACK回应	写入数据字节	X	0	0	0	发送最后数据；等待ACK回应
			X	0	0	1	发送数据；等待ACK回应
C0H	已发送TWIDAT数据；已接收NACK回应	无TWIDAT动作	0	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址
			0	0	0	1	切换至非寻址从机模式；响应自己地址，是否响应通用地址依赖于寄存器TWIADR中GC的设置
			1	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址；总线空闲时发送“开始条件”
			1	0	0	1	切换至非寻址从机模式；响应自己地址，是否响应通用地址依赖于寄存器TWIADR中GC的设置；总线空闲时发送“开始条件”
C8H	已发送最后一个TWIDAT数据(AA=0)；已接收ACK回应	无TWIDAT动作	0	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址
			0	0	0	1	切换至非寻址从机模式；响应自己地址，是否响应通用地址依赖于寄存器TWIADR中GC的设置
			1	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址；总线空闲时发送“开始条件”
			1	0	0	1	切换至非寻址从机模式；响应自己地址，是否响应通用地址依赖于寄存器TWIADR中GC的设置；总线空闲时发送“开始条件”



从机接收模式

从机接收模式中，从机从主机接收一系列数据。为初始化从机接收模式，必须对控制寄存器TWICON和地址寄存器TWIADR进行初始化：置位控制寄存器TWICON中的ENTWI和STA，清除STO和TWINT；地址寄存器TWIADR中高7位为SH79F6412准备相应的地址。如果GC置位，SH79F6412也将响应通用地址（00H）；否则将不响应通用地址。

在TWIADR和TWICON初始化后，SH79F6412将等待总线对自己地址或通用地址（如果GC被置位）的响应。如果方向标志位是‘写’，则TWI进入从机接收模式，否则将进入从机发送模式。在地址和写标志位接收完毕后，中断标志（TWINT）置位，状态寄存器TWISTA有效。

在传输中，如果将应答使能位“AA”清零，TWI将接收最后一个字节并回应“不应答”信息。回应“不应答”可以表示当前从机无法接收更多字节。当AA = 0时，SH79F6412无法回应对自己地址的访问；但仍然监视总线状态，并可以通过AA = 1恢复对自己地址的相应。可以通过AA = 0暂时将SH79F6412从总线隔离。

SH79F6412作为从机接收模式时，最小接收频率为4.5KHz，小于4.5KHz时，无法正常接收数据。

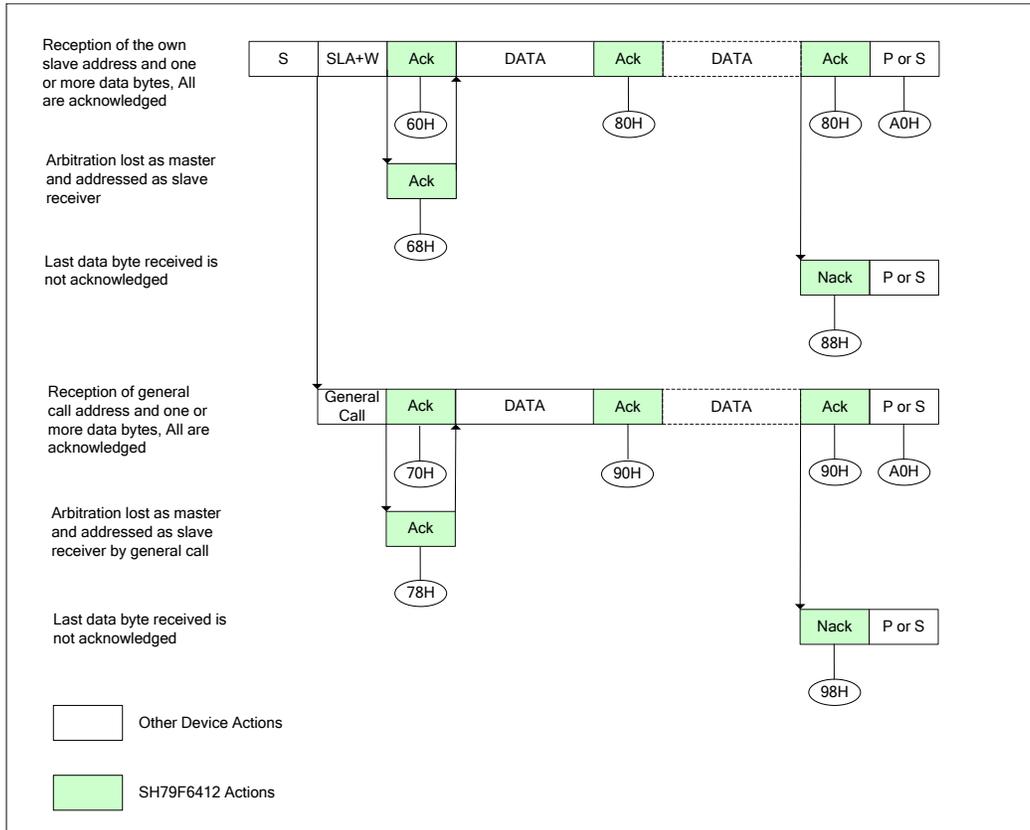
从机接收模式状态码

状态码	TWI总线和硬件接口状态	应用软件响应				TWI执行的下一个动作	
		读/写数据寄存器TWIDAT操作	控制位操作				
			STA	STO	TWINT		AA
60H	已收到自己SLA+W；已回应ACK	无TWIDAT动作	X	0	0	0	接收数据；发送NACK回应
			X	0	0	1	接收数据；发送ACK回应
68H	作为主机发送SLA+R/W时失去仲裁，收到主机SLA+W；已回应ACK	无TWIDAT动作	X	0	0	0	接收数据；发送NACK回应
			X	0	0	1	接收数据；发送ACK回应
70H	收到主机发送通用地址；已回应ACK	无TWIDAT动作	X	0	0	0	接收数据；发送NACK回应
			X	0	0	1	接收数据；发送ACK回应
78H	作为主机发送SLA+R/W时失去仲裁，收到主机发送通用地址；已回应ACK	无TWIDAT动作	X	0	0	0	接收数据；发送NACK回应
			X	0	0	1	接收数据；发送ACK回应



续上表

80H	处于已寻址状态； 已收到数据； 已回应ACK	读取数据	X	0	0	0	接收数据；发送NACK回应
			X	0	0	1	接收数据；发送ACK回应
88H	处于已寻址状态； 已收到数据； 已回应NACK	读取数据	0	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址
			0	0	0	1	切换至非寻址从机模式；响应自己地址，是否响通用地址依赖于寄存器TWIADR中GC的设置
			1	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址；总线空闲时发送“开始条件”
			1	0	0	1	切换至非寻址从机模式；响应自己地址，是否响通用地址依赖于寄存器TWIADR中GC的设置；总线空闲时发送“开始条件”
90H	处于通用地址已寻址状态；已收到数据；已回应ACK	读取数据	X	0	0	0	接收数据；发送NACK回应
			X	0	0	1	接收数据；发送ACK回应
98H	处于通用地址已寻址状态； 已收到数据； 已回应NACK	读取数据	0	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址
			0	0	0	1	切换至非寻址从机模式；响应自己地址，是否响通用地址依赖于寄存器TWIADR中GC的设置
			1	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址；总线空闲时发送“开始条件”
			1	0	0	1	切换至非寻址从机模式；响应自己地址，是否响通用地址依赖于寄存器TWIADR中GC的设置；总线空闲时发送“开始条件”
A0H	作为从机时收到终止条件或重复开始条件	无TWIDAT动作	0	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址
			0	0	0	1	切换至非寻址从机模式；响应自己地址，是否响通用地址依赖于寄存器TWIADR中GC的设置
			1	0	0	0	切换至非寻址从机模式；不响应自己地址和通用地址；总线空闲时发送“开始条件”
			1	0	0	1	切换至非寻址从机模式；响应自己地址，是否响通用地址依赖于寄存器TWIADR中GC的设置；总线空闲时发送“开始条件”



其它模式

除上述状态码外，两个状态码没有明确的TWI状态。状态0F8H表示由于中断标志TWINT未置起，没有相应的状态信息。当中断TWINT未置起，即在清除一个状态后到一个新状态建立前均由0F8H进行填充。

状态00H表示在TWI总线通讯中有错误发生，即传输中有非法的开始条件或终止条件发生。例如在传输地址，数据或回应ACK应答时有起始条件或终止条件发生。当总线扰乱了内部逻辑时也会产生00H状态。当非法状态出现时，会置起中断标志位TWINT。可通过置起STO并清除TWINT标志恢复到正常通讯，SH79F6412将进入非寻址从机模式，并自动清除STO标志。数据线和时钟线将被释放，线上无终止条件传送。

其它模式状态码

状态码	TWI总线和硬件接口状态	应用软件响应				TWI执行的下一个动作	
		读/写数据寄存器 TWIDAT操作	控制位操作				
			STA	STO	TWINT		AA
F8H	没有有效状态码；TWINT=0	无TWIDAT动作	无TWICON动作				等待或处理当前传输
00H	在主机或寻址从机模式下有非法开始条件或终止条件发送；接口导致TWI内部逻辑混乱	无TWIDAT动作	0	1	0	x	只有内部硬件受影响；释放总线；切换到非寻址从机模式；清除STO



## 8.6.5 寄存器

Table 8.31 TWI控制寄存器

F8H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TWICON	TOUT	ENTWI	STA	STO	TWINT	AA	TFREE	EFREE
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	TOUT	<b>总线超时标志位</b> 0: 无超时发生 1: 当TWI总线低电平超过25ms时置位，必须由软件清除
6	ENTWI	<b>TWI使能位</b> 0: 关闭TWI功能 1: 开启TWI功能
5	STA	<b>起始位</b> 0: 不会发送起始条件 1: 当总线空闲时发送起始条件
4	STO	<b>终止位</b> 0: 不会发送终止条件 1: 作为主机时发送终止条件；作为从机时不向总线发送终止条件，但状态恢复到非寻址从机状态。硬件将自动清除该标志位
3	TWINT	<b>TWI串行中断标志位</b> 0: 没有TWI串行中断发生 1: 产生TWI通讯状态中除0F8H之外的状态时置位，必须由软件清除
2	AA	<b>声明应答标志</b> 0: 回复“不应答”信号（SDA高电平） 1: 回复“应答”信号（SDA低电平）
1	TFREE	<b>SCL高电平超时标志位</b> 0: 无超时发生 1: 参与总线传输时如时钟线超过50us高电平时置位，必须由软件清除
0	EFREE	<b>SCL高电平超时使能位</b> 0: 禁止SCL总线高电平超时判断 1: 允许SCL总线高电平超时判断

**特别提示：** TOUT, TWINT, TFREE均将触发TWI中断，三者共享一个中断向量。



Table 8.32 TWI状态寄存器

F9H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TWISTA	TWISTA.7	TWISTA.6	TWISTA.5	TWISTA.4	TWISTA.3	CR.1	CR.0	ETOT
读/写	读	读	读	读	读	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	1	1	1	1	1	0	0	0

位编号	位符号	说明
7-3	TWISTA[7:3]	<b>TWI串行通讯状态位</b> 参见相应的状态描述
2-1	CR[0:1]	<b>TWI发送频率控制位</b> 00: f <sub>OSCS</sub> /4/1024 01: f <sub>OSCS</sub> /4/256 10: f <sub>OSCS</sub> /4/64 11: f <sub>OSCS</sub> /4/42 当SH79F6412处于主机模式时, STA和STO以及重复STA的保持时间与CR[1:0]选中的发送频率相关
0	ETOT	<b>总线超时使能位</b> 0: 禁止总线超时判断 1: 使能总线超时判断

Table 8.33 TWI地址寄存器

FAH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TWIADR	TWA.6	TWA.5	TWA.4	TWA.3	TWA.2	TWA.1	TWA.0	GC
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-1	TWA[6:0]	<b>TWI地址配置位</b> 配置SH79F6412作为从机时的地址
0	GC	<b>通用地址使能位</b> 0: 禁止响应通用地址 1: 允许响应通用地址

Table 8.34 TWI数据寄存器

FDH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TWIDAT	TWIDAT.7	TWIDAT.6	TWIDAT.5	TWIDAT.4	TWIDAT.3	TWIDAT.2	TWIDAT.1	TWIDAT.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TWIDAT[7:0]	<b>TWI通讯数据寄存器</b>



Table 8.35 系统时钟控制寄存器

B2H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
CLKCON	32k_SPDUP	CLKS1	CLKS0	SCMIF	HFON	FS	-	-
读/写	读/写	读/写	读/写	读	读/写	读/写	-	-
复位值 (POR/WDT/LVR/PIN)	1	1	1	0	0	0	-	-

位编号	位符号	说明
7	32k_SPDUP	<p><b>32.768kHz晶体谐振器加速模式控制位</b></p> <p>0: 32.768kHz振荡器常规模式，由软件清0。</p> <p>1: 32.768kHz振荡器加速模式，由软件或者硬件置1。</p> <p>此位在系统发生任何形式的复位，如上电复位，看门狗复位等时，自动由硬件设置1，用以加速32.768kHz振荡器起振，缩短32.768kHz振荡器的起振时间。</p> <p>如果有需要，本位也可以由软件置1或者清0。比如进入掉电模式（Power-down mode）前，可以将此位置1，掉电模式唤醒后再由软件清0。</p> <p>应该注意的是关闭32.768kHz加速模式（此位清0），可以节省系统的耗电。</p> <p>只有代码选项OP_OSC为0111时（选择32.768kHz晶体振荡器，详见代码选项章节），此控制位才有效。</p>
6-5	CLKS[1:0]	<p><b>系统时钟频率分频器</b></p> <p>00: <math>f_{sys} = f_{oscsc}</math></p> <p>01: <math>f_{sys} = f_{oscsc}/2</math></p> <p>10: <math>f_{sys} = f_{oscsc}/4</math></p> <p>11: <math>f_{sys} = f_{oscsc}/12</math></p> <p>如果选择32kHz crystal振荡器为OSCCLK，此控制位无效。</p>
3	HFON	<p><b>OSC1CLK开关控制寄存器</b></p> <p>0: 关闭OSC1CLK</p> <p>1: 打开OSC1CLK</p> <p>只有代码选项OP_OSC为0011, 0110, 1010时（128K内部RC与32.768K晶体振荡器，详见代码选项章节），此控制位才有效。</p>
2	FS	<p><b>频率选择位</b></p> <p>0: 选择128KHz或32.768KHz为OSCCLK</p> <p>1: 选择OSC1CLK为OSCCLK</p> <p>只有代码选项OP_OSC为0011, 0110, 1010时（128K内部RC与32.768K晶体振荡器，详见代码选项章节），此控制位才有效。</p>



## 8.7 模/数转换器 (ADC)

### 8.7.1 特性

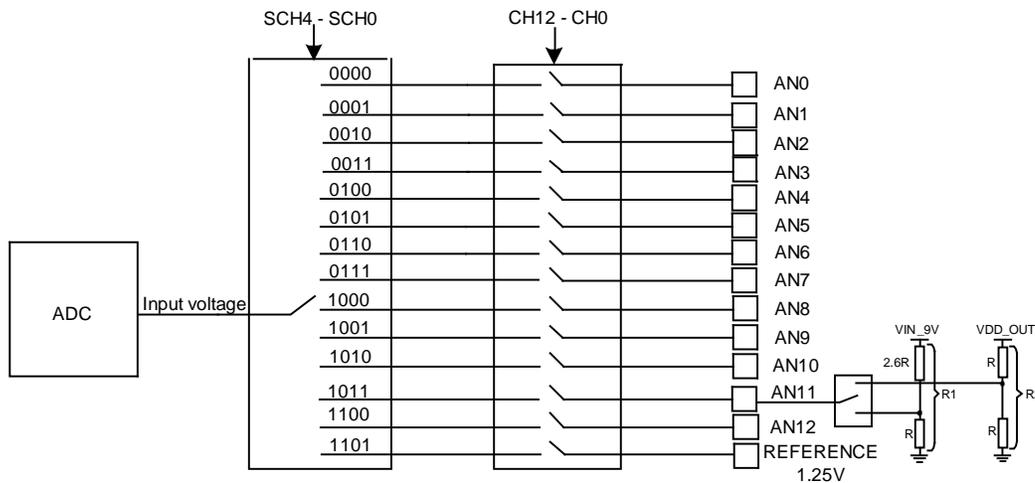
- 12位分辨率
- 内建基准电压
- 可选外接或内建基准电压
- 13通道模拟输入

SH79F6412包含一个单端型、12位逐次逼近型模/数转换器 (ADC)。ADC内建的基准电压 $V_{REF}$ 直接与 $V_{CC}$ 相连, 用户也可以选择 $V_{CC\_OUT}$ 基准电压。13个ADC通道都可以输入独立的模拟信号, 但是每次只能使用一个通道。 $GO/DONE$ 信号控制开始转换, 提示转换结束。当转换完成时, 更新ADC数据寄存器, 设置ADCON寄存器中的ADCIF位, 并产生一个中断 (如果ADC中断被允许)。

ADC模块整合数字比较功能可以比较ADC中的模拟输入的值与数字值。如果允许数字比较功能 (在ADCON寄存器中的EC位置1), 并且ADC模块使能 (在ADCON寄存器中的ADON位置1), 只有当相应的模拟输入的数字值大于寄存器中的比较值 (ADDH/L) 时, 才会产生ADC中断。当 $GO/DONE$ 置1时, 数字比较功能会持续工作, 直到 $GO/DONE$ 清0。这一点与模数转换工作方式不同。

带数字比较功能的ADC模块能在Idle模式下工作, 并且ADC中断能够唤醒Idle模式。但是, 在掉电模式下, ADC模块被禁止。

### 8.7.2 ADC模块图



AD转换器模块图



8.7.3 寄存器

Table 8.36 ADC控制寄存器

93H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ADCON	ADON	ADCIF	EC	SCH3	SCH2	SCH1	SCH0	GO/DONE
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	ADON	<b>ADC控制位</b> 0: 关闭ADC模块 1: 打开ADC模块
6	ADCIF	<b>ADC标志中断标志</b> 0: 无ADC中断, 由软件清0 1: 由硬件置1, 表示已完成AD转换或者模拟输入大于ADDATH/L (如果比较功能打开)
5	EC	<b>比较功能允许位</b> 0: 关闭比较功能 1: 打开比较功能
4-1	SCH[3:0]	<b>ADC通道选择位</b> 0000: ADC通道AN0 0001: ADC通道AN1 0010: ADC通道AN2 0011: ADC通道AN3 0100: ADC通道AN4 0101: ADC通道AN5 0110: ADC通道AN6 0111: ADC通道AN7 1000: ADC通道AN8 1001: ADC通道AN9 1010: ADC通道AN10 1011: ADC通道AN11 1100: ADC通道AN12 1101: 内部1.25V基准 其他: 内部1.25V基准
0	GO/DONE	<b>ADC状态标志位</b> 0: 当完成AD转换时, 该位由硬件自动清0。如在转换期间清0该位会中止当前的转换。如果允许数字比较功能, 该位只能软件清0, 不能硬件清0 1: 设置开始AD转换或者启动数字比较功能



Table 8.37 ADC定时控制寄存器

94H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ADT	TADC2	TADC1	TADC0	-	TS3	TS2	TS1	TS0
读/写	读/写	读/写	读/写	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	-	0	0	0	0

位编号	位符号	说明
7-5	TADC[2:0]	<b>ADC时钟周期选择位</b> 000: ADC时钟周期 $t_{AD} = 2 t_{SYS}$ 001: ADC时钟周期 $t_{AD} = 4 t_{SYS}$ 010: ADC时钟周期 $t_{AD} = 6 t_{SYS}$ 011: ADC时钟周期 $t_{AD} = 8 t_{SYS}$ 100: ADC时钟周期 $t_{AD} = 12 t_{SYS}$ 101: ADC时钟周期 $t_{AD} = 16 t_{SYS}$ 110: ADC时钟周期 $t_{AD} = 24 t_{SYS}$ 111: ADC时钟周期 $t_{AD} = 32 t_{SYS}$
3-0	TS[3:0]	<b>采样时间选择位</b> $2 t_{AD} \leq \text{采样时间} = (\text{TS}[3:0]+1) * t_{AD} \leq 15 t_{AD}$

注意:

- (1) 请确保ADC时钟周期 $t_{AD} \geq 1\mu s$ ;
- (2) 即使 $\text{TS}[3:0] = 0000$ , 最小采样时间为 $2t_{AD}$ ;
- (3) 即使 $\text{TS}[3:0] = 1111$ , 最大采样时间为 $15t_{AD}$ ;
- (4) 选择 $2 * t_{AD}$ 为采样时间时, 请确保连接到ADC输入引脚的串联电阻小于 $10k\Omega$ ;
- (5) 全部转换时间 =  $12t_{AD} + \text{采样时间}$ 。

Table 8.38 ADC通道设置寄存器

95H, 91H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ADC1H (95H)	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
ADC2H (91H)	ADCV	-	-	CH12	CH11	CH10	CH9	CH8
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
ADC1H: 7-0 ADC2H: 4-0	CH[12:0]	<b>信道配置位</b> 0: P0.4-P0.6, P1.7, P2.1-P2.7, P3.3, P3.6作为I/O端口 1: P0.4-P0.6, P1.7, P2.1-P2.7, P3.3, P3.6为ADC输入口
ADC2H: 7	ADCV	<b>基准源开关位</b> 0: 1.25V基准源关闭 1: 1.25V基准源打开



Table 8.39 AD转换数据寄存器（比较值寄存器）

96H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ADDL	-	-	-	-	A3	A2	A1	A0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	0	0	0	0
97H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ADDH	A11	A10	A9	A8	A7	A6	A5	A4
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
3-0 7-0	A11-A0	<b>ADC数据寄存器</b> 采样模拟电压的数字值。当完成转换后，这个值会更新。 如果ADC数字比较功能使能（EC = 1），这个值将与模拟输入进行比较。

**启动ADC转换步骤:**

- (1) 选择模拟输入通道以及基准电压
- (2) 使能ADC模块
- (3) GO/DONE置1开始ADC转换
- (4) 等待GO/DONE = 0或者ADCIF = 1，如果ADC中断使能，则ADC中断将会产生，用户需要软件清0 ADCIF
- (5) 从ADDH/ADDL获得转换数据
- (6) 重复步骤3-5开始另一次转换

**启动数字比较功能步骤:**

- (1) 选择模拟输入通道以及基准电压
- (2) 写入ADDH/ADDL，设置比较值
- (3) EC置1使能数字比较功能
- (4) 使能ADC模块
- (5) GO/DONE置1开始数字比较功能
- (6) 如果模拟输入的值比设置的比较值大，ADIF会被置1。如果ADC中断使能，则ADC中断将会产生，用户需要软件清0 ADCIF
- (7) 数字比较功能会持续工作，直到GO/DONE清0



## 8.8 蜂鸣器

## 8.8.1 特性

- 为音频发生器输出方波信号
- 可输出可调周期的方波信号

## 8.8.2 寄存器

Table 8.40 蜂鸣器控制寄存器

BDH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
BUZCON	BUZEN	BUSLE	-	-	BCA1	BCA0	-	-
读/写	读/写	读/写	-	-	读/写	读/写	-	-
复位值 (POR/WDT/LVR/PIN)	0	0	-	-	0	0	-	-

位编号	位符号	说明
7	BUZEN	蜂鸣器模块允许位 0: 关闭蜂鸣器模块 1: 开启蜂鸣器模块
6	BUSLE	蜂鸣器输出选择位 0: 选择BUZ1作为蜂鸣器输出 (非选口为I/O) 1: 选择BUZ2作为蜂鸣器输出 (非选口为I/O)
3-2	BCA[2:0]	蜂鸣器时钟频率: 00: 系统时钟/256 01: 系统时钟/64 10: 系统时钟/16 11: 系统时钟/4

Table 8.41 蜂鸣器周期控制寄存器 (BUZP)

AFH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
BUZP	BZP.7	BZP.6	BZP.5	BZP.4	BZP.3	BZP.2	BZP.1	BZP.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	BZP[7:0]	蜂鸣器周期8位寄存器

Buzz引脚输出方波频率计算公式: Buzz Frequency = Fsys/BCA[2:0]/BUZP[7:0]



## 8.9 低电压检测 (LPD)

### 8.9.1 特性

- 低电压检测并产生中断
- LPD检测电源电压

低电压检测 (LPD) 功能用来监测电源电压, 如果电压低于指定值时产生内部标志。LPD功能用来通知CPU电源是否被切断或电池是否用尽, 因此在电压低于最小工作电压之前, 软件可以采取一些保护措施。

### 8.9.2 寄存器

Table 8.42 低电压检测控制寄存器

B3H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
LPDCON	LPDEN	LPDF*	-	-	-	LPDS2	LPDS1	LPDS0
读/写	读/写	读/写*	-	-	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	-	-	-	0	0	0

\*: LPDF软件只能够清0, 不能置1。

位编号	位符号	说明
7	LPDEN	<b>LPD允许位</b> 0: 禁止低电压检测 1: 允许低电压检测
6	LPDF	<b>LPD标志位</b> 0: 无LPD发生, 由硬件或软件清0, 即当前电压高于在LPDS[1:0]中设置的LPD电压 1: LPD发生, 由硬件置1, 即当前电压低于在LPDS[1:0]中设置的LPD电压
2-0	LPDS[2:0]	<b>LPD电压设置位</b> 000: 2.8V 001: 3.0V 010: 3.2V 011: 3.4V 100: 3.6V 101: 3.8V 110: 4.0V 111: 4.2V



## 8.10 低电压复位 (LVR)

### 8.10.1 特性

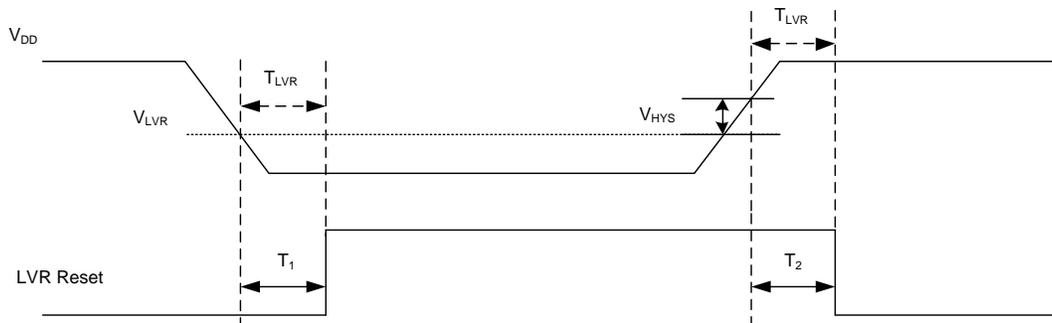
- 通过代码选项选择, LVR设定电压 $V_{LVR}$ 可为3.7V和3.1V, 或2.8V
- LVR去抖动时间 $T_{LVR}$ 为30-60 $\mu$ s
- 当供电电压低于设定电压 $V_{LVR}$ 时, 将产生内部复位

低电压复位 (LVR) 功能是为了监测供电电压, 当供电电压低于设定电压 $V_{LVR}$ 时, MCU将产生内部复位。LVR去抖动时间 $T_{LVR}$ 大约为30 $\mu$ s-60 $\mu$ s。

LVR功能打开后, 具有以下特性 ( $T_1$ 表示电压低于设定电压 $V_{LVR}$ 的时间,  $T_2$ 表示电压高于设定电压 $V_{LVR}+V_{HYS}$ 的时间):

当 $V_{DD} \leq V_{LVR}$ 且 $T_1 \geq T_{LVR}$ 时产生系统复位。

当 $V_{DD} > V_{LVR}+V_{HYS}$ 且 $T_2 \geq T_{LVR}$ 时释放系统复位或 $V_{DD} < V_{LVR}$ , 但 $T_1 < T_{LVR}$ 时不会产生系统复位。其中 $V_{HYS}$ 的范围是0.09V - 0.11V。



这里,  $V_{DD}$ 为电源电压,  $V_{LVR}$ 为LVR检测电压,  $V_{HYS}$ 为低电压复位迟滞电压。

通过代码选项, 可以选择LVR功能的打开与关闭。

在交流电或大容量电池应用中, 接通大负载后容易导致MCU供电暂时低于定义的工作电压。低电压复位可以应用于此, 保护系统在低于设定电压下产生有效复位。



## 8.11 看门狗定时器（WDT），程序超范围溢出（OVL）复位及其它复位状态

### 8.11.1 特性

- 看门狗可以工作在掉电模式下
- 程序超范围溢出后硬件自动检测，并产生OVL复位
- 看门狗溢出频率可选

### 程序超范围溢出复位

SH79F6412为进一步增强CPU运行可靠性，内建程序超范围溢出检测电路，一旦检测到程序计数器的值超出ROM最大值，或者发现指令操作码（不检测操作数）为8051指令集中不存在的A5H，便认为程序跑飞，产生CPU复位信号，同时将WDOF标志位置1。为应用这个特性，用户应该将未使用的Flash ROM用0xA5填满。

### 看门狗

看门狗定时器（WDT）是一个递减计数器，独立内建RC振荡器作为时钟源，因此可以通过代码选项在掉电模式下仍会持续运行。当定时器溢出时，将芯片复位。通过代码选项可以打开或关闭该功能。

WDT控制位（第2 - 0位）用来选择不同的溢出时间。定时器溢出后，WDT溢出标志（WDOF）将由硬件自动置1。通过读写RSTSTAT寄存器，看门狗定时器在溢出前重新开始计数。其它一些复位标志列举如下：

### 8.11.2 寄存器

Table 8.43 复位状态寄存器

B1H, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
RSTSTAT	WDOF	-	PORF	LVRF	-	WDT.2	WDT.1	WDT.0
读/写	读/写	-	读/写	读/写	-	读/写	读/写	读/写
复位值 (POR)	0	-	1	0	-	0	0	0
复位值 (WDT)	1	-	u	u	-	0	0	0
复位值 (LVR)	u	-	u	1	-	0	0	0
复位值 (PIN)	u	-	u	u	-	0	0	0

位编号	位符号	说明
7	WDOF	<b>看门狗溢出或程序超范围溢出标志位</b> 看门狗溢出时由硬件置1，可由软件或上电复位清0 0：未发生WDT溢出或程序超范围溢出 1：发生WDT溢出或程序超范围溢出
5	PORF	<b>上电复位标志位</b> 上电复位后硬件置1，只能由软件清0 0：没有发生上电复位 1：发生过上电复位
4	LVRF	<b>低压复位标志位</b> 低压复位后置1，可由软件或上电复位清0 0：没有发生低压复位 1：发生过低压复位
2-0	WDT[2:0]	<b>WDT溢出周期控制位</b> 000：溢出周期最小值 = 4096ms 001：溢出周期最小值 = 1024ms 010：溢出周期最小值 = 256ms 011：溢出周期最小值 = 128ms 100：溢出周期最小值 = 64ms 101：溢出周期最小值 = 16ms 110：溢出周期最小值 = 4ms 111：溢出周期最小值 = 1ms <b>注意：</b> 应用中如果看门狗打开，程序清看门狗的最大间隔时间不能大于以上所列最小值



## 8.12 电源管理

### 8.12.1 特性

- 空闲模式和掉电模式两种省电模式
- 发生中断和复位可退出空闲（Idle）、掉电（Power-Down）模式

为减少功耗，SH79F6412提供两种低功耗省电模式：空闲（Idle）模式和掉电（Power-Down）模式，这两种模式都由PCON和SUSLO两个寄存器控制。

### 8.12.2 空闲模式（Idle）

空闲模式能够降低系统功耗，在此模式下，程序中止运行，CPU时钟停止，但外部设备时钟继续运行。空闲模式下，CPU在确定的状态下停止，并在进入空闲模式前所有CPU的状态都被保存，如PC，PSW，SFR，RAM等。

两条连续指令：先设置SUSLO寄存器为0x55，随即将PCON寄存器中的IDL位置1，使SH79F6412进入空闲模式。如果不满足上述的两条连续指令，CPU在下一个机器周期清0 SUSLO寄存器或IDL位，CPU也不会进入空闲模式。

IDL位置1是CPU进入空闲模式之前执行的最后一条指令。

两种方式可以退出空闲模式：

(1) 中断产生。恢复CPU时钟，硬件清除SUSLO寄存器和PCON寄存器的IDL位。然后执行中断服务程序，随后跳转到进入空闲模式指令之后的指令。

(2) 复位信号产生后（复位引脚上出现低电平，WDT复位，LVR复位）。CPU恢复时钟，SUSLO寄存器和在PCON寄存器中的IDL位被硬件清0，最后SH79F6412复位，程序从地址位0000H开始执行。此时，RAM保持不变而SFR的值根据不同功能模块改变。

### 8.12.3 掉电模式（Power-Down）

掉电模式可以使SH79F6412进入功耗非常低的状态。掉电模式将停止CPU和外围设备的所有时钟信号。在掉电模式下，如果通过代码选项使能WDT，WDT模块将继续工作。在进入掉电模式前所有CPU的状态都被保存，如PC，PSW，SFR，RAM等。

两条连续指令：先设置SUSLO寄存器为0x55，随即将PCON寄存器中的PD位置1，使SH79F6412进入掉电模式。如果不满足上述的两条连续指令CPU在下一个机器周期清除SUSLO寄存器或的PD位，CPU也不会进入掉电模式。

PD位置1是CPU进入掉电模式之前执行的最后一条指令。

**注意：**如果同时设置IDL位和PD位，SH79F6412进入掉电模式。退出掉电模式后，CPU也不会掉电进入空闲模式，从掉电模式退出后硬件清0 IDL及PD位。

有三种方式可以退出掉电模式：

(1) 有效外部中断（如INT0，INT1，INT2，INT3）和LPD中断使SH79F6412退出掉电模式。在中断发生后振荡器启动，在预热计时结束之后CPU时钟和外部设备时钟恢复，SUSLO寄存器和PCON寄存器中的PD位会被硬件清除，然后继续运行中断服务程序。在完成中断服务程序之后，跳转到进入掉电模式之后的指令继续运行。

(2) 复位信号（复位引脚上出现低电平，WDT复位如果被允许，LVR复位如果被允许）。在预热计时之后会恢复CPU时钟，SUSLO寄存器和PCON寄存器中的PD位会被硬件清除，最后SH79F6412会被复位，程序会从0000H地址位开始运行。RAM将保持不变，而根据不同功能模块SFR的值可能改变。

(3) 定时器3可以在特定模式下工作在掉电模式。并且能够在掉电模式中产生中断并退出掉电模式。（详见定时器3章节）

**注意：**如要进入这两种低功耗模式，必须在置位PCON中的IDL/PD位后增加3个空操作指令（NOP）。



## 8.12.4 寄存器

Table 8.44 电源控制寄存器

87H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PCON	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL
读/写	读/写	读/写	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	-	-	0	0	0	0

位编号	位符号	说明
7	SMOD	UART波特率加倍器
6	SSTAT	SCON[7:5]功能选择位
3-2	GF[1:0]	用于软件的通用标志
1	PD	掉电模式控制位 0: 当一个中断或复位产生时由硬件清0 1: 由软件置1激活掉电模式
0	IDL	空闲模式控制位 0: 当一个中断或复位产生时由硬件清0 1: 由软件置1激活空闲模式

Table 8.45 省电模式控制寄存器

8EH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
SUSLO	SUSLO.7	SUSLO.6	SUSLO.5	SUSLO.4	SUSLO.3	SUSLO.2	SUSLO.1	SUSLO.0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	SUSLO[7:0]	此寄存器用来控制CPU进入省电模式（空闲或掉电）。只有像下面的连续指令才能使CPU进入省电模式，否则在下个周期中SUSLO, IDL或PD位将被硬件清0。

程序举例:

```

IDLE_MODE:
MOV    SUSLO, #55H
ORL    PCON, #01H
NOP
NOP
NOP

POWERDOWN_MODE:
MOV    SUSLO, #55H
ORL    PCON, #02H
NOP
NOP
NOP

```



### 8.13 预热计数器

#### 8.13.1 特性

- 内建电源预热计数器消除电源的上电的不稳定状态
- 内建振荡器预热计数器消除振荡器起振时的不稳定状态

SH79F6412内建有电源上电预热计数器，主要是用来消除上电电压建立时的不稳定态，同时完成内部一些初始化序列，如读取内部客户代码选项等。

SH79F6412还内建振荡器预热计数器，它能消除振荡器在下列情况下起振时的不稳定状态：上电复位，引脚复位，从低功耗模式中唤醒，看门狗复位和LVR复位。

上电后，SH79F6412会先经过电源上电预热计数过程，等待溢出后再进行振荡器的预热计数过程过程，溢出后开始运行程序。

#### 电源上电预热计数时间

上电复位/ 引脚复位/低电压复位		看门狗复位 (不包含掉电模式)		看门狗复位 (唤醒掉电模式)		掉电模式下中断唤醒	
电源上电 预热计数时间	振荡器上电 预热计数时间	电源上电 预热计数时间	振荡器上电 预热计数时间	电源上电 预热计数时间	振荡器上电 预热计数时间	电源上电 预热计数时间	振荡器上电 预热计数时间
11ms	有	≈1ms	无	≈1ms	有	≈800us	有

#### 振荡器上电预热计数时间

代码选项: OP_WMT	00	01	10	11
陶振/晶振	$2^{17} \times T_{osc}$	$2^{14} \times T_{osc}$	$2^{11} \times T_{osc}$	$2^8 \times T_{osc}$
32kHz晶振	$2^{13} \times T_{osc}$			
内部RC	$2^7 \times T_{osc}$			



### 8.14 循环冗余校验CRC (Cyclic Redundancy Check)

#### 8.14.1 特性

- FLASH内容CRC校验，可设置校验Flash ROM范围和初始值
- CRC生成多项式采用CRC-CCITT标准： $X^{16}+X^{12}+X^5+1$ ，高位优先
- 2种模式：高速CRC模式和普通CRC模式

为提高系统可靠性，SH79F6412内建1个CRC校验模块，可用来实时生成code的CRC校验码，采用的生成多项式采用CRC-CCITT标准： $X^{16}+X^{12}+X^5+1$ 。用户可利用此校验码和理论值比较，监测Flash内容是否有变化。CRC理论值可存放在所选ROM范围的最后两个Byte（不参与CRC校验），或其他位置（比如类EEPROM区，序列号区，用户识别码区等）。

设置CRCADR[4:0]控制位可以选择校验空间大小，设置CRC\_GO位启动CRC模块，校验完成后，CRC\_GO自动清0，并置起CRCIF标志，如果中断允许位ESCM\_LPD\_CRC和ECRC都为1，则CPU产生CRC中断，程序响应中断后此标志位被硬件自动清0。

普通CRC模式：在CPU运行程序指令过程中，利用一些特殊指令的执行间隙，进行CRC校验，不会影响CPU的指令执行时间，但校验时间较长，且不可控。

高速CRC模式：为提高CRC的校验时间，可以打开CRC模块，让CPU进入IDLE模式，将执行高速CRC模式，校验时间较短，CRC中断可唤醒IDLE模式。

**注意：**掉电（Power-Down）模式下，且CRC模块处于运行状态，会导致CRC校验结果不正确；请确保CRC运行结束前，系统不进掉电（Power-Down）模式。

#### 8.14.2 寄存器

Table 8.46 CRC控制寄存器

ACH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
CRCCON	CRC_GO	CRCIF	-	CRCADR4	CRCADR3	CRCADR2	CRCADR1	CRCADR0
读/写	读/写	读/写	-	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	-	0	0	0	0	0

位编号	位符号	说明
7	CRC_GO	<b>CRC启动控制位</b> 0: 关闭CRC模块 1: 启动CRC模块，CRC校验完成后自动清0
6	CRCIF	<b>CRC完成中断请求标志位</b> 0: 未启动或启动后未完成 1: CRC校验完成，向CPU请求中断，如果中断允许位ESCM_LPD_CRC和ECRC都为1，则CPU响应中断，程序响应中断后此标志位软件清0
3-0	CRCADR[4:0]	<b>CRC校验范围设置位</b> 0000: 校验地址范围为0000 - 07FDH (2K-2Byte) 00001: 校验地址范围为0000 - 0FFDH (4K-2Byte) ..... 11110: 校验地址范围为0000 - F7FDH (62K-2Byte) 11111: 校验地址范围为0000 - FFFDH (64K-2Byte)



Table 8.47 CRC结果寄存器低位（注意低位地址在前，高位地址在后）

ADH, AEH, Bank0	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
CRCDL (ADH)	CRCD7	CRCD6	CRCD5	CRCD4	CRCD3	CRCD2	CRCD1	CRCD0
CRCDH (AEH)	CRCD15	CRCD14	CRCD13	CRCD12	CRCD11	CRCD10	CRCD9	CRCD8
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	CRCD[15:0]	<b>CRC校验结果寄存器</b> 启动CRC模块前，此寄存器值作为CRC校验初值，校验完成后，校验结果存储于此寄存器中，覆盖之前的数据。

**注意：**

- (1) CRC启动之前，需要对CRC结果寄存器赋初始值。
- (2) CRC\_GO置为1启动CRC校验，在校验过程中，不允许对CRCADR[3:0]重新设置，也不允许对CRC结果寄存器CRCDL/CRCDH有修改动作。
- (3) 所选择的CRC校验范围的最后2个byte，不参与CRC校验，可以存放CRC理论值。

**8.15 代码选项****OP\_WDT:**

- 0: 允许看门狗复位 (默认)
- 1: 禁止看门狗复位

**OP\_WDTPD:**

- 0: 掉电模式下禁止看门狗工作 (默认)
- 1: 掉电模式下允许看门狗工作

**OP\_LVREN:**

- 0: 禁止低电压复位功能 (默认)
- 1: 允许低电压复位功能

**OP\_LVRLE:**

- 00: 低电压复位设定电压为3.7V (默认)
- 01: 低电压复位设定电压为3.1V
- 10: 低电压复位设定电压为2.8V

**OP\_OSC:**

- 0000: 内部16.6MHz RC振荡器作为振荡器1, 振荡器2关闭
- 0011: 内部128kHz RC振荡器作为振荡器1, 16.6MHz内部RC作为振荡器2
- 0110: 内部128kHz RC振荡器作为振荡器1, 2M-12MHz晶体/陶瓷谐振器作为振荡器2
- 1010: 32.768kHz晶体振荡器作为振荡器1, 16.6MHz内部RC作为振荡器2
- 1110: 2M-12MHz晶体/陶瓷谐振器作为振荡器1, 振荡器2关闭

**OP\_SEG/IO: (该选项选择输出电流应用所有模式)**

- 00: P0.5-P0.6, P1.2, P1.7, P2.0-P2.6, P3.0-P3.1,  $V_{DD}=3V$ 时, 输出40mA电流。GND+3V, 输入40mA电流
- 01: P0.5-P0.6, P1.2, P1.7, P2.0-P2.6, P3.0-P3.1,  $V_{DD}=3V$ 时, 输出30mA电流。GND+3V, 输入30mA电流
- 10: P0.5-P0.6, P1.2, P1.7, P2.0-P2.6, P3.0-P3.1,  $V_{DD}=3V$ 时, 输出20mA电流。GND+3V, 输入20mA电流
- 11: P0.5-P0.6, P1.2, P1.7, P2.0-P2.6, P3.0-P3.1,  $V_{DD}=3V$ 时, 输出10mA电流。GND+3V, 输入10mA电流

**OP\_MODSW:**

- 0: 当MODSW = 1时LED功能暂停记录当前扫描通道, MODSW = 0时继续扫描顺序通道 (默认)
- 1: 当MODSW = 1时LED功能不暂停, MODSW = 0时扫描LED当前通道

**OP\_WMT: (不适用于内建RC)**

- 00: 最长预热时间 (默认)
- 01: 长预热时间
- 10: 短预热时间
- 11: 最短预热时间

**OP\_COM/IO: (该选项选择输出电流应用所有模式)**

- 0: 代码选项为P0.7, P1.0-P1.2, P2.7, P3.0-P3.1, sink/drive电流能力加大
- 1: 代码选项为P0.7, P1.0-P1.2, P2.7, P3.0-P3.1, 普通输入/输出电流能力 (默认)

**OP\_SCM:**

- 0: 在预热期间禁止时钟单元检测功能 (默认)
- 1: 在预热期间允许时钟单元检测功能

**OP\_OSCDRIVE1:**

- 0: 当OP\_OSCDRIVER2 = 10: 2M陶振  
当OP\_OSCDRIVER2 = 11: 4M陶振
- 1: 当OP\_OSCDRIVER2 = 10: 8M陶振  
当OP\_OSCDRIVER2 = 11: 12M陶振  
当OP\_OSCDRIVER2 = 00: 4M晶振  
当OP\_OSCDRIVER2 = 01: 8M-12M晶振（默认）

**OP\_OSCDRIVE2:**

- 10: 当OP\_OSCDRIVE1 = 0: 2M陶振  
当OP\_OSCDRIVE1 = 1: 8M陶振
- 11: 当OP\_OSCDRIVE1 = 0: 4M陶振  
当OP\_OSCDRIVE1 = 1: 12M陶振
- 00: 当OP\_OSCDRIVE1 = 1: 4M晶振
- 01: 当OP\_OSCDRIVE1 = 1: 8M-12M晶振（默认）

当使用外部陶振或者晶振时（OP\_OSC为0110或者1110）

	<b>OP_OSCDRIVE2: 00</b>	<b>OP_OSCDRIVE2: 01</b>	<b>OP_OSCDRIVE2: 10</b>	<b>OP_OSCDRIVE2: 11</b>
<b>OP_OSCDRIVE1: 0</b>	-	-	2M陶振	4M陶振
<b>OP_OSCDRIVE1: 1</b>	4M晶振	8M-12M晶振	8M陶振	12M陶振



9. 指令集

算术操作指令				
指令	功能描述	代码	字节	周期
ADD A, Rn	累加器加寄存器	0x28-0x2F	1	1
ADD A, direct	累加器加直接寻址字节	0x25	2	2
ADD A, @Ri	累加器加内部RAM	0x26-0x27	1	2
ADD A, #data	累加器加立即数	0x24	2	2
ADDC A, Rn	累加器加寄存器和进位位	0x38-0x3F	1	1
ADDC A, direct	累加器加直接寻址字节和进位位	0x35	2	2
ADDC A, @Ri	累加器加内部RAM和进位位	0x36-0x37	1	2
ADDC A, #data	累加器加立即数和进位位	0x34	2	2
SUBB A, Rn	累加器减寄存器和借位位	0x98-0x9F	1	1
SUBB A, direct	累加器减直接寻址字节和借位位	0x95	2	2
SUBB A, @Ri	累加器减内部RAM和借位位	0x96-0x97	1	2
SUBB A, #data	累加器减立即数和借位位	0x94	2	2
INC A	累加器加1	0x04	1	1
INC Rn	寄存器加1	0x08-0x0F	1	2
INC direct	直接寻址字节加1	0x05	2	3
INC @Ri	内部RAM加1	0x06-0x07	1	3
DEC A	累加器减1	0x14	1	1
DEC Rn	寄存器减1	0x18-0x1F	1	2
DEC direct	直接寻址字节减1	0x15	2	3
DEC @Ri	内部RAM减1	0x16-0x17	1	3
INC DPTR	数据指针加1	0xA3	1	4
MUL AB	累加器乘寄存器B	0xA4	1	11 20
DIV AB	累加器除以寄存器B	0x84	1	11 20
DA A	十进制调整	0xD4	1	1



逻辑操作指令				
指令	功能描述	代码	字节	周期
ANL A, Rn	累加器与寄存器	0x58-0x5F	1	1
ANL A, direct	累加器与直接寻址字节	0x55	2	2
ANL A, @Ri	累加器与内部RAM	0x56-0x57	1	2
ANL A, #data	累加器与立即数	0x54	2	2
ANL direct, A	直接寻址字节与累加器	0x52	2	3
ANL direct, #data	直接寻址字节与立即数	0x53	3	3
ORL A, Rn	累加器或寄存器	0x48-0x4F	1	1
ORL A, direct	累加器或直接寻址字节	0x45	2	2
ORL A, @Ri	累加器或内部RAM	0x46-0x47	1	2
ORL A, #data	累加器或立即数	0x44	2	2
ORL direct, A	直接寻址字节或累加器	0x42	2	3
ORL direct, #data	直接寻址字节或立即数	0x43	3	3
XRL A, Rn	累加器异或寄存器	0x68-0x6F	1	1
XRL A, direct	累加器异或直接寻址字节	0x65	2	2
XRL A, @Ri	累加器异或内部RAM	0x66-0x67	1	2
XRL A, #data	累加器异或立即数	0x64	2	2
XRL direct, A	直接寻址字节异或累加器	0x62	2	3
XRL direct, #data	直接寻址字节异或立即数	0x63	3	3
CLR A	累加器清零	0xE4	1	1
CPL A	累加器取反	0xF4	1	1
RL A	累加器左环移位	0x23	1	1
RLC A	累加器连进位标志左环移位	0x33	1	1
RR A	累加器右环移位	0x03	1	1
RRC A	累加器连进位标志右环移位	0x13	1	1
SWAP A	累加器高4位与低4位交换	0xC4	1	4



数据传送指令				
指令	功能描述	代码	字节	周期
MOV A, Rn	寄存器送累加器	0xE8-0xEF	1	1
MOV A, direct	直接寻址字节送累加器	0xE5	2	2
MOV A, @Ri	内部RAM送累加器	0xE6-0xE7	1	2
MOV A, #data	立即数送累加器	0x74	2	2
MOV Rn, A	累加器送寄存器	0xF8-0xFF	1	2
MOV Rn, direct	直接寻址字节送寄存器	0xA8-0xAF	2	3
MOV Rn, #data	立即数送寄存器	0x78-0x7F	2	2
MOV direct, A	累加器送直接寻址字节	0xF5	2	2
MOV direct, Rn	寄存器送直接寻址字节	0x88-0x8F	2	2
MOV direct1, direct2	直接寻址字节送直接寻址字节	0x85	3	3
MOV direct, @Ri	内部RAM送直接寻址字节	0x86-0x87	2	3
MOV direct, #data	立即数送直接寻址字节	0x75	3	3
MOV @Ri, A	累加器送内部RAM	0xF6-0xF7	1	2
MOV @Ri, direct	直接寻址字节送内部RAM	0xA6-0xA7	2	3
MOV @Ri, #data	立即数送内部RAM	0x76-0x77	2	2
MOV DPTR, #data16	16位立即数送数据指针	0x90	3	3
MOVC A, @A+DPTR	程序代码送累加器（相对数据指针）	0x93	1	7
MOVC A, @A+PC	程序代码送累加器（相对程序计数器）	0x83	1	8
MOVX A, @Ri	外部RAM送累加器（8位地址）	0xE2-0xE3	1	5
MOVX A, @DPTR	外部RAM送累加器（16位地址）	0xE0	1	6
MOVX @Ri, A	累加器送外部RAM（8位地址）	0xF2-F3	1	4
MOVX @DPTR, A	累加器送外部RAM（16位地址）	0xF0	1	5
PUSH direct	直接寻址字节压入栈顶	0xC0	2	5
POP direct	栈顶弹至直接寻址字节	0xD0	2	4
XCH A, Rn	累加器与寄存器交换	0xC8-0xCF	1	3
XCH A, direct	累加器与直接寻址字节交换	0xC5	2	4
XCH A, @Ri	累加器与内部RAM交换	0xC6-0xC7	1	4
XCHD A, @Ri	累加器低4位与内部RAM低4位交换	0xD6-0xD7	1	4



控制程序转移指令				
指令	功能描述	代码	字节	周期
ACALL addr11	2KB内绝对调用	0x11-0xF1	2	7
LCALL addr16	64KB内长调用	0x12	3	7
RET	子程序返回	0x22	1	8
RETI	中断返回	0x32	1	8
AJMP addr11	2KB内绝对转移	0x01-0xE1	2	4
LJMP addr16	64KB内长转移	0x02	3	5
SJMP rel	相对短转移	0x80	2	4
JMP @A+DPTR	相对长转移	0x73	1	6
JZ rel (不发生转移) (发生转移)	累加器为零转移	0x60	2	3 5
JNZ rel (不发生转移) (发生转移)	累加器为非零转移	0x70	2	3 5
JC rel (不发生转移) (发生转移)	C置位转移	0x40	2	2 4
JNC rel (不发生转移) (发生转移)	C清零转移	0x50	2	2 4
JB bit, rel (不发生转移) (发生转移)	直接寻址位置位转移	0x20	3	4 6
JNB bit, rel (不发生转移) (发生转移)	直接寻址位清零转移	0x30	3	4 6
JBC bit, rel (不发生转移) (发生转移)	直接寻址位置位转移并清该位	0x10	3	4 6
CJNE A, direct, rel (不发生转移) (发生转移)	累加器与直接寻址字节不等转移	0xB5	3	4 6
CJNE A, #data, rel (不发生转移) (发生转移)	累加器与立即数不等转移	0xB4	3	4 6
CJNE Rn, #data, rel (不发生转移) (发生转移)	寄存器与立即数不等转移	0xB8-0xBF	3	4 6
CJNE @Ri, #data, rel (不发生转移) (发生转移)	内部RAM与立即数不等转移	0xB6-0xB7	3	4 6
DJNZ Rn, rel (不发生转移) (发生转移)	寄存器减1不为零转移	0xD8-0xDF	2	3 5
DJNZ direct, rel (不发生转移) (发生转移)	直接寻址字节减1不为零转移	0xD5	3	4 6
NOP	空操作	0	1	1



位操作指令				
指令	功能描述	代码	字节	周期
CLR C	C清零	0xC3	1	1
CLR bit	直接寻址位清零	0xC2	2	3
SETB C	C置位	0xD3	1	1
SETB bit	直接寻址位置位	0xD2	2	3
CPL C	C取反	0xB3	1	1
CPL bit	直接寻址位取反	0xB2	2	3
ANL C, bit	C逻辑与直接寻址位	0x82	2	2
ANL C, /bit	C逻辑与直接寻址位的反	0xB0	2	2
ORL C, bit	C逻辑或直接寻址位	0x72	2	2
ORL C, /bit	C逻辑或直接寻址位的反	0xA0	2	2
MOV C, bit	直接寻址位送C	0xA2	2	2
MOV bit, C	C送直接寻址位	0x92	2	3



### 10. 电气特性

**极限参数\***

直流供电电压.....-0.3V to +6.0V  
 输入/输出电压.....GND-0.3V to V<sub>DD</sub>+0.3V  
 工作环境温度.....-40°C to +85°C  
 存储温度.....-55°C to +125°C

**\*注释**

如果器件的工作条件超过左列“**极限参数**”的范围，将造成器件永久性破坏。只有当器件工作在说明书所规定的范围内时功能才能得到保障。器件在极限参数列举的条件下工作将会影响到器件工作的可靠性。

低压部分直流电气特性 (V<sub>DD</sub> = 2.7 - 5.5V, GND = 0V, T<sub>A</sub> = +25°C, 除非另有说明)

参数	符号	最小值	典型值*	最大值	单位	条件
工作电压	V <sub>DD</sub>	2.7	5.0	5.5	V	400kHz ≤ f <sub>OSC</sub> ≤ 16.6MHz
工作电流	I <sub>OP1</sub>	-	5	10	mA	f <sub>OSC</sub> = 16.6MHz, V <sub>DD</sub> = 5.0V 所有输出引脚无负载（所有数字输入引脚不浮动）； CPU打开（执行NOP指令）；WDT打开，关闭其它所有功能
	I <sub>OP2</sub>	-	40	80	μA	f <sub>OSC</sub> = 128kHz, V <sub>DD</sub> = 5.0V 所有输出引脚无负载（所有数字输入引脚不浮动）； CPU打开（执行NOP指令）；WDT打开，关闭其它所有功能
待机电流（空闲模式）	I <sub>SB1</sub>	-	3	5	mA	f <sub>OSC</sub> = 16.6MHz, V <sub>DD</sub> = 5.0V 所有输出引脚无负载（所有数字输入引脚不浮动）； CPU停止（空闲模式）；WDT关闭；LVR打开；关闭其它所有功能
	I <sub>SB2</sub>	-	20	40	μA	f <sub>OSC</sub> = 128kHz, V <sub>DD</sub> = 5.0V 所有输出引脚无负载（所有数字输入引脚不浮动）； CPU停止（空闲模式）；WDT关闭；LVR打开；关闭其它所有功能
待机电流（掉电模式）	I <sub>SB3</sub>	-	-	15	μA	振荡器停止, V <sub>DD</sub> = 5.0V 所有输出引脚无负载；CPU停止（掉电模式）； LVR打开, WDT关闭, 关闭其它所有功能
WDT电流	I <sub>WDT</sub>	-	1	3	μA	V <sub>DD</sub> = 5.0V, WDT打开
输入低电压1	V <sub>IL1</sub>	GND	-	0.3 X V <sub>DD</sub>	V	I/O端口, V <sub>DD</sub> = 2.7 - 5.5V
输入高电压1	V <sub>IH1</sub>	0.7 X V <sub>DD</sub>	-	V <sub>DD</sub>	V	I/O端口, V <sub>DD</sub> = 2.7 - 5.5V
输入低电压2	V <sub>IL2</sub>	GND	-	0.2 X V <sub>DD</sub>	V	T0, T1, T2, T3, T4, INT2/3/4, SCK, T2EX, RXD, V <sub>DD</sub> = 2.7 - 5.5V
输入高电压2	V <sub>IH2</sub>	0.8 X V <sub>DD</sub>	-	V <sub>DD</sub>	V	T0, T1, T2, T3, T4, INT2/3/4, SCK, T2EX, RXD, V <sub>DD</sub> = 2.7 - 5.5V
输入漏电流	I <sub>IL</sub>	-1	-	1	μA	输入口, V <sub>DD</sub> = 5.0V, V <sub>IN</sub> = V <sub>DD</sub> 或者GND
输出漏电流	I <sub>OL</sub>	-1	-	1	μA	V <sub>DD</sub> = 5.0V, V <sub>OUT</sub> = V <sub>DD</sub> 或者GND
上拉电阻	R <sub>PH</sub>	-	30	-	kΩ	V <sub>DD</sub> = 5.0V, V <sub>IN</sub> = GND
输出高电压1	V <sub>OH1</sub>	V <sub>DD</sub> - 3V	-	-	V	I/O端口, I <sub>OH</sub> = 40mA, V <sub>DD</sub> = 5.0V（需要OP_SEG选择输出40mA选项）
输出高电压2	V <sub>OH2</sub>	V <sub>DD</sub> - 3V	-	-	V	I/O端口, I <sub>OH</sub> = 30mA, V <sub>DD</sub> = 5.0V（需要OP_SEG选择输出30mA选项）
输出高电压3	V <sub>OH3</sub>	V <sub>DD</sub> - 3V	-	-	V	I/O端口, I <sub>OH</sub> = 20mA, V <sub>DD</sub> = 5.0V（需要OP_SEG选择输出20mA选项）
输出高电压4	V <sub>OH4</sub>	V <sub>DD</sub> - 3V	-	-	V	I/O端口, I <sub>OH</sub> = 10mA, V <sub>DD</sub> = 5.0V（需要OP_SEG选择输出10mA选项）



续上表

参数	符号	最小值	典型值*	最大值	单位	条件
输出低电压1	$V_{OL1}$	-	-	GND + 3V	V	I/O端口, $I_{OL} = 40\text{mA}$ , $V_{DD} = 5.0\text{V}$ (需要OP_SEG选择输出40mA选项)
输出低电压2	$V_{OL2}$	-	-	GND + 3V	V	I/O端口, $I_{OL} = 30\text{mA}$ , $V_{DD} = 5.0\text{V}$ (需要OP_SEG选择输出30mA选项)
输出低电压3	$V_{OL3}$	-	-	GND + 3V	V	I/O端口, $I_{OL} = 20\text{mA}$ , $V_{DD} = 5.0\text{V}$ (需要OP_SEG选择输出20mA选项)
输出低电压4	$V_{OL4}$	-	-	GND + 3V	V	I/O端口, $I_{OL} = 10\text{mA}$ , $V_{DD} = 5.0\text{V}$ (需要OP_SEG选择输出10mA选项)
大驱动口灌电流能力	$I_{OL}$	88	100	-	mA	$V_{DD} = 5.0\text{V}$ , $V_{OL} = 0.5\text{V}$ (COM1 - COM7) (需要OP_COM选择输出大电流选项)
大驱动口输出电流能力	$I_{OH}$	-88	-100	-	mA	$V_{DD} = 5.0\text{V}$ , $V_{OH} = 4.4\text{V}$ (COM1 - COM7) (需要OP_COM选择输出大电流选项)

**注意:**

- (1) “\*”表示典型值下的数据是在5.0V, 25°C下测得的, 除非另有说明。
- (2) 流过 $V_{DD}$ 的最大电流值须小于180mA。
- (3) 流过GND的最大电流值须小于250mA。
- (4) IO口输出高或者低时, 相当于IC内部有一个不大于75ohm (3V/0.04A) 电阻接到GND或者 $V_{DD}$ 。
- (5) IO口输出高或者低时, 相当于IC内部有一个不大于100ohm (3V/0.03A) 电阻接到GND或者 $V_{DD}$ 。
- (6) IO口输出高或者低时, 相当于IC内部有一个不大于150ohm (3V/0.02A) 电阻接到GND或者 $V_{DD}$ 。
- (7) IO口输出高或者低时, 相当于IC内部有一个不大于300ohm (3V/0.01A) 电阻接到GND或者 $V_{DD}$ 。
- (8) IO口输出低时, 相当于IC内部有一个不大于5.7ohm (0.5V/0.088A) 电阻接到GND。
- (9) IO口输出高时, 相当于IC内部有一个不大于6.8ohm (0.6V/0.088A) 电阻接到 $V_{DD}$ 。

**5V模/数转换器电气特性** ( $V_{DD} = 2.7\text{V} - 5.5\text{V}$ ,  $GND = 0\text{V}$ ,  $T_A = +25^\circ\text{C}$ , 除非另有说明)

参数	符号	最小值	典型值	最大值	单位	条件
供电电压	$V_{AD}$	2.7	5.0	5.5	V	
精度	$N_R$	-	12	-	bit	$GND \leq V_{AIN} \leq V_{REF}$
A/D输入电压	$V_{AIN}$	GND	-	$V_{REF}$	V	除AN2, AN1外
A/D输入电压	$V_{AIN}$	GND	-	2.0	V	AN1, AN2
A/D输入电阻*	$R_{AIN}$	2	-	-	MΩ	$V_{IN} = 5.0\text{V}$
内建基准源精度	$V_{REF}$	1.2375	1.25	1.2625	V	$T_A = +25^\circ\text{C}$
内建基准源精度	$V_{REF}$	1.218	1.25	1.282	V	$T_A = -40^\circ\text{C} \sim 85^\circ\text{C}$
模拟电压源推荐阻抗	$Z_{AIN}$	-	-	10	kΩ	
A/D转换电流	$I_{AD}$	-	1	3	mA	ADC模块打开, $V_{DD} = 5.0\text{V}$
A/D输入电流	$I_{ADIN}$	-	-	10	μA	$V_{DD} = 5.0\text{V}$
微分非线性误差	$D_{LE}$	-	-	±1	LSB	$V_{DD} = 5.0\text{V}$
积分非线性误差	$I_{LE}$	-	-	±2	LSB	$V_{DD} = 5.0\text{V}$
满刻度误差	$E_F$	-	±1	±3	LSB	$V_{DD} = 5.0\text{V}$
偏移量误差	$E_Z$	-	-	±8	LSB	$V_{DD} = 5.0\text{V}$
总绝对误差	$E_{AD}$	-	-	±8	LSB	$V_{DD} = 5.0\text{V}$
总转换时间**	$T_{CON}$	14	-	-	μs	12 bit精度, $V_{DD} = 5.0\text{V}$

**注意:**

- (1) “\*”表示ADC输入电阻就是直流条件下ADC自身的输入电阻。
- (2) “\*\*”建议与ADC连接的信号源内阻小于10kΩ。



交流电气特性 ( $V_{DD} = 2.7V - 5.5V$ ,  $GND = 0V$ ,  $T_A = +25^{\circ}C$ ,  $f_{OSC} = 16.6MHz$ , 除非另有说明)

参数	符号	最小值	典型值	最大值	单位	条件
RC频率	$F_{RC}$	-	16.6	-	MHz	$V_{DD} = 5V$
频率稳定性 (RC)	$ \Delta F /F$	-	-	1	%	RC振荡器:  F-16.6MHz /16.6MHz ( $V_{DD} = 2.7-5.5V$ , $T_A = +25^{\circ}C$ )
		-	-	2	%	RC振荡器:  F-16.6MHz /16.6MHz ( $V_{DD} = 2.7-5.5V$ , $T_A = -40^{\circ}C \sim 85^{\circ}C$ )
		-	128	-	KHz	$V_{DD} = 5V$
		-	-	5	%	RC振荡器:  F-128kHz /128kHz ( $V_{DD} = 2.7-5.5V$ , $T_A = +25^{\circ}C$ )
		-10	-	+10	%	RC振荡器:  F-128kHz /128kHz ( $V_{DD} = 2.7-5.5V$ , $T_A = -40^{\circ}C \sim 85^{\circ}C$ )

触摸按键电气特性 ( $V_{DD} = 2.7V - 5.5V$ ,  $GND = 0V$ ,  $T_A = +25^{\circ}C$ , 除非另有说明)

参数	符号	最小值	典型值	最大值	单位	条件
工作电压	$V_{DD}$	2.7	-	5.5	V	
输出电压1	$V_{DDR1}$	3.9	4	4.1	V	$V_{DD} = 4.5 - 5.5V$
输出电压2	$V_{DDR2}$	2.9	3.0	3.1	V	$I_{V_{DDR2}} = 0 - 20mA$ , $V_{DD} = 3.5 - 5.5V$
输出电压3	$V_{DDR3}$	2.4	2.5	2.6	V	$I_{V_{DDR3}} = 0 - 20mA$ , $V_{DD} = 3.0 - 5.5V$
输出电压4	$V_{DDR4}$	1.9	2	2.1	V	$I_{V_{DDR4}} = 0 - 20mA$ , $V_{DD} = 2.7 - 5.5V$
输出参考电压1	$V_{REF1}$	0.9	1	1.1	V	$V_{DD} = 2.7 - 5.5V$
输出参考电压2	$V_{REF2}$	1.4	1.5	1.6	V	$V_{DD} = 2.7 - 5.5V$
输出参考电压3	$V_{REF3}$	1.9	2.0	2.1	V	$V_{DD} = 2.7 - 5.5V$
输出参考电压4	$V_{REF4}$	2.4	2.5	2.6	V	$V_{DD} = 3.0 - 5.5V$

触摸按键比较器电气特性 ( $V_{DD} = 2.7 - 5.5V$ ,  $GND = 0V$ ,  $T_A = +25^{\circ}C$ , 除非另有说明)

参数	符号	最小值	典型值	最大值	单位	条件
输入偏移电压	$V_{IO}$	-	10	15	mV	$T_A = 25^{\circ}C$
输入电压随温度变化偏移电压	$V_{I1}$	-	3	5	mV	$T_A = -40^{\circ}C \sim 85^{\circ}C$
输入共模电压范围	$V_{ICM}$	0	-	$V_{DD}-1V$	V	

低电压复位电气特性 ( $V_{DD} = 2.7V - 5.5V$ ,  $GND = 0V$ ,  $T_A = +25^{\circ}C$ , 除非另有说明)

参数	符号	最小值	典型值	最大值	单位	条件
LVR电压设定电压1	$V_{LVR}$	3.0	3.1	3.2	V	LVR使能
LVR电压设定电压2	$V_{LVR}$	3.6	3.7	3.8	V	LVR使能
LVR电压设定电压3	$V_{LVR}$	2.7	2.8	2.9	V	LVR使能
LVR低电压复位宽度	$T_{LVR}$	-	30	-	$\mu s$	

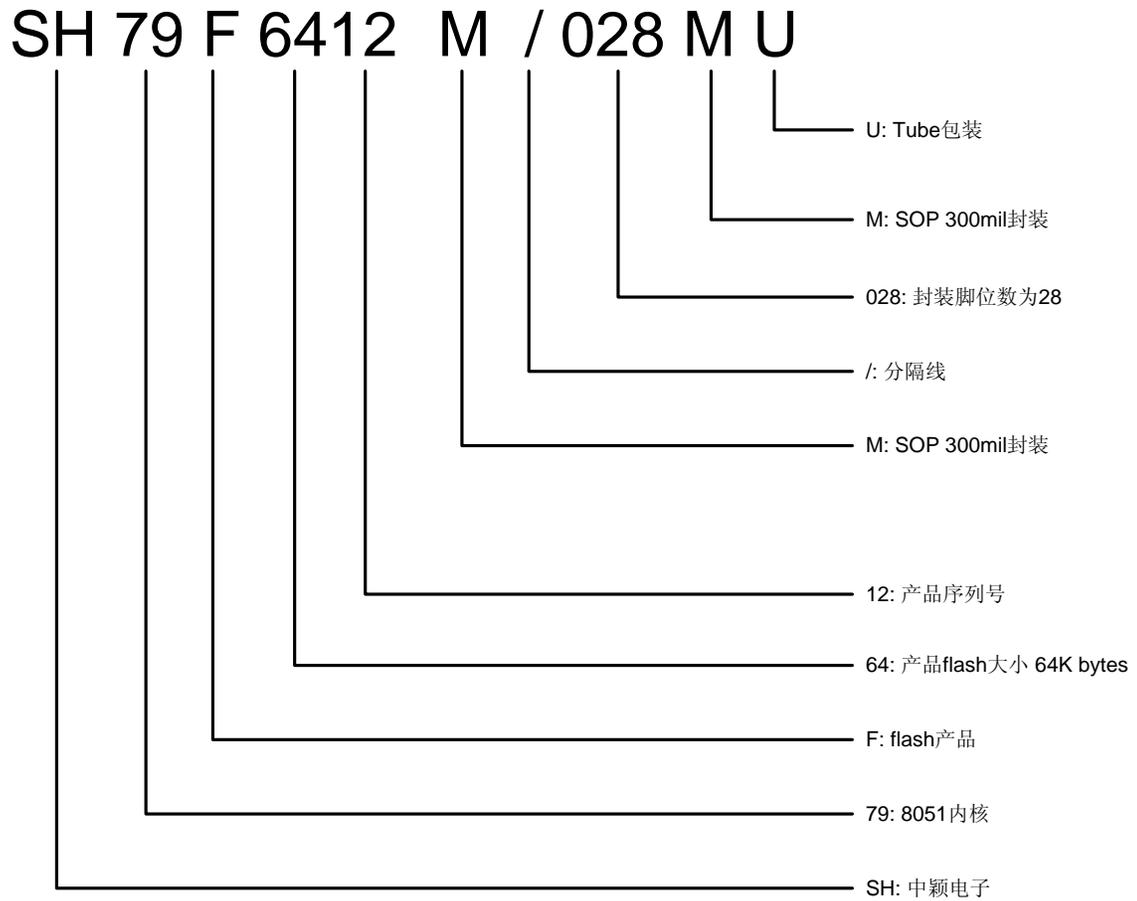


**11. 订购信息**

产品编号	封装
SH79F6412M/028MU	28 SOP
SH79F6412M/020MU	20 SOP



12. 产品命名规则

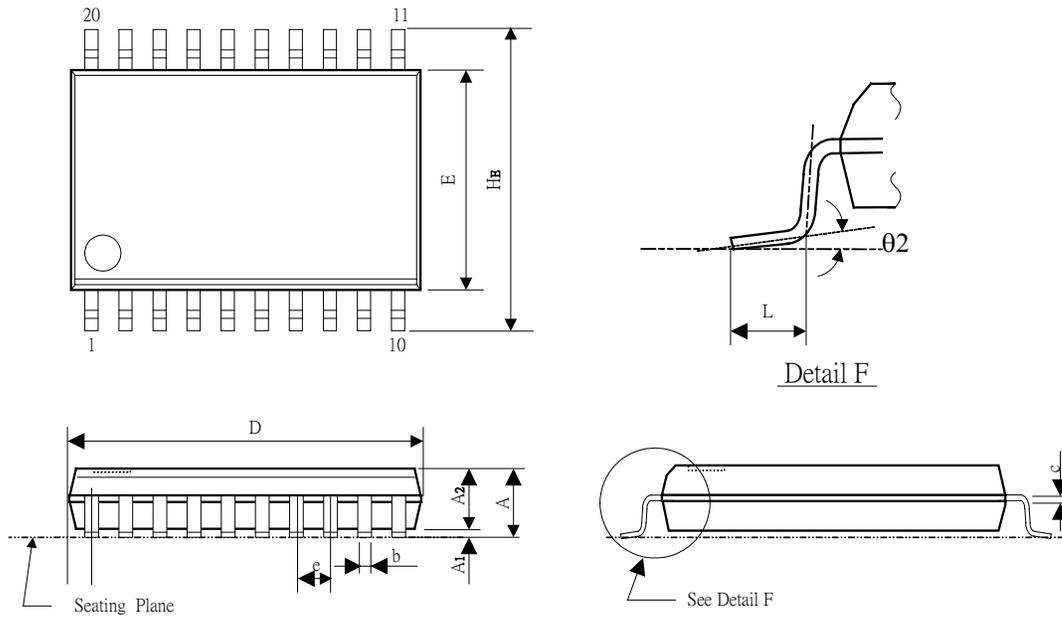




13. 封装信息

SOP 20L Outline Dimensions

unit: inches/mm

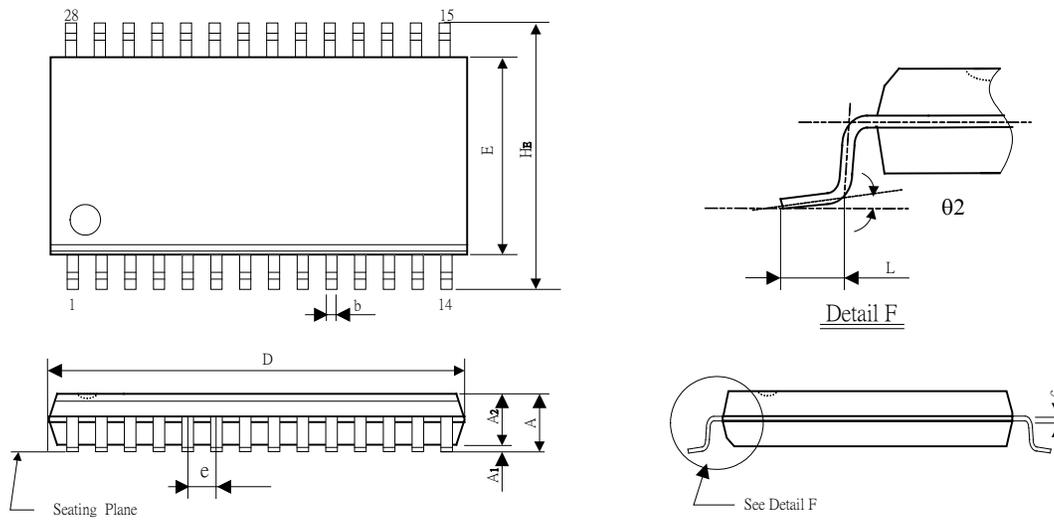


Symbol	Dimensions in inches		Dimensions in mm	
	Min	Max	Min	Max
A	0.093	0.104	2.35	2.65
A1	0.004	0.012	0.10	0.30
A2	0.083	0.098	2.10	2.50
b	0.013	0.020	0.33	0.51
c	0.008	0.013	0.20	0.33
D	0.493	0.516	12.52	13.10
E	0.291	0.299	7.40	7.60
e	0.050(BSC)		1.27(BSC)	
HE	0.398	0.418	10.11	10.61
L	0.016	0.050	0.40	1.27
$\theta 2$	0°	8°	0°	8°



SOP 28L Outline Dimensions

unit: inches/mm



Symbol	Dimensions in inches		Dimensions in mm	
	Min	Max	Min	Max
A	0.085	0.104	2.15	2.65
A1	0.004	0.012	0.10	0.30
A2	0.081	0.098	2.05	2.50
b	0.013	0.02	0.33	0.51
c	0.006	0.014	0.15	0.36
D	0.697	0.715	17.70	18.15
E	0.291	0.303	7.40	7.70
e	0.050(BSC)		1.27(BSC)	
HE	0.402	0.418	10.21	10.61
L	0.016	0.05	0.40	1.27
θ2	0°	8°	0°	8°



14. 规格更改记录

版本	记录	日期
1.0	初始版本	2018年1月



## 目录

1. 特性	1
2. 概述	1
3. 方框图	2
4. 引脚配置	3
5. 引脚描述	5
6. SFR映像	7
7. 标准功能	19
7.1 CPU	19
7.1.1 CPU内核特殊功能寄存器	19
7.1.2 CPU增强内核特殊功能寄存器	20
7.1.3 寄存器	20
7.2 随机数据存储存储器 (RAM)	21
7.2.1 特性	21
7.2.2 寄存器	21
7.3 FLASH程序存储器	22
7.3.1 特性	22
7.3.2 ICP模式下的Flash操作	24
7.4 SSP功能	25
7.4.1 寄存器	25
7.4.2 Flash控制流程图	28
7.4.3 SSP编程注意事项	29
7.4.4 可读识别码	29
7.5 系统时钟和振荡器	30
7.5.1 特性	30
7.5.2 时钟定义	30
7.5.3 概述	30
7.5.4 寄存器	31
7.5.5 振荡器类型	32
7.5.6 谐振器负载电容选择	32
7.6 系统时钟监控 (SCM)	33
7.7 I/O端口	34
7.7.1 特性	34
7.7.2 寄存器	34
7.7.3 端口模块图	36
7.7.4 端口共享	37
7.8 定时器	41
7.8.1 特性	41
7.8.2 定时器0和定时器1	41
7.8.3 定时器2	46
7.8.4 定时器3	48
7.8.5 定时器4	50
7.9 中断	53
7.9.1 特性	53
7.9.2 中断允许	53
7.9.3 寄存器	53
7.9.4 中断标志	56
7.9.5 中断向量	58
7.9.6 中断优先级	58
7.9.7 中断处理	59
7.9.8 中断响应时间	59
7.9.9 外部中断输入	60
7.9.10 中断汇总	62
8. 增强功能	63
8.1 TOUCH KEY触摸按键功能	63



8.1.1 寄存器.....	66
8.2 LED驱动器.....	75
8.2.1 寄存器.....	75
8.2.2 LED RAM配置.....	80
8.3 TOUCH KEY触摸按键功能与LED共享功能.....	83
8.3.1 功能描述.....	84
8.4 脉冲宽度调制 (PWM).....	86
8.4.1 特性.....	86
8.4.2 PWM允许寄存器.....	86
8.5 增强型通用异步收发器 (EUART).....	90
8.5.1 特性.....	90
8.5.2 工作方式.....	90
8.5.3 可微调波特率.....	95
8.5.4 多机通讯.....	96
8.5.5 帧出错检测.....	97
8.5.6 寄存器.....	97
8.6 TWI串行通讯接口.....	100
8.6.1 特性.....	100
8.6.2 数据传输格式.....	100
8.6.3 功能描述.....	102
8.6.4 传输模式.....	103
8.6.5 寄存器.....	112
8.7 模/数转换器 (ADC).....	115
8.7.1 特性.....	115
8.7.2 ADC模块图.....	115
8.7.3 寄存器.....	116
8.8 蜂鸣器.....	119
8.8.1 特性.....	119
8.8.2 寄存器.....	119
8.9 低电压检测 (LPD).....	120
8.9.1 特性.....	120
8.9.2 寄存器.....	120
8.10 低电压复位 (LVR).....	121
8.10.1 特性.....	121
8.11 看门狗定时器 (WDT) , 程序超范围溢出 (OVL) 复位及其它复位状态.....	122
8.11.1 特性.....	122
8.11.2 寄存器.....	122
8.12 电源管理.....	123
8.12.1 特性.....	123
8.12.2 空闲模式 (Idle).....	123
8.12.3 掉电模式 (Power-Down).....	123
8.12.4 寄存器.....	124
8.13 预热计数器.....	125
8.13.1 特性.....	125
8.14 循环冗余校验CRC (CYCLIC REDUNDANCY CHECK).....	126
8.14.1 特性.....	126
8.14.2 寄存器.....	126
8.15 代码选项.....	128
9. 指令集.....	130
10. 电气特性.....	135
11. 订购信息.....	138
12. 产品命名规则.....	139
13. 封装信息.....	140
14. 规格更改记录.....	142