



## SH79F1624A

### 带10位ADC的8051微控制器

#### 1. 特性

- 基于8051兼容流水指令的8位单片机
- Flash ROM: 16K字节
- RAM: 内部256字节, 外部256字节
- 片上512字节类EEPROM存储空间
- 工作电压:  
 $f_{OSC} = 30\text{kHz} - 16.6\text{MHz}$ ,  $V_{DD} = 2.8\text{V} - 5.5\text{V}$
- 振荡器 (代码选项):
  - 晶体谐振器: 32.768kHz
  - 晶体谐振器: 400kHz - 16.6MHz, 或陶瓷谐振器: 2MHz - 16.6MHz
  - 陶瓷谐振器: 400kHz - 2MHz
  - 内部振荡器: 16.6MHz ( $\pm 2\%$ )
  - 外部时钟源: 30kHz - 16.6MHz
- 14个CMOS双向I/O管脚 (四种端口模式)
- 5个大电流驱动口
- 3个16位定时器/计数器T0, T1, T2
- 中断源:
  - 定时器0, 1, 2
  - 外部中断0, 1, 2
  - ADC, EUART, SCM, PWM, LPD
- 1个8位PWM定时器
- 1个EUART
- 内建低电压检测功能 (LPD)
- 6通道10位模数转换器 (ADC), 内建比较功能
- 内建低电压复位功能 (LVR) (代码选项)
  - LVR电压1: 4.1V
  - LVR电压2: 3.7V
  - 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
- 振荡器失效检测功能 (SCM)
- 支持省电运作模式:
  - IDLE模式
  - 掉电模式
- 低功耗
- Flash型
- 封装:
  - 16脚SOP封装

#### 2. 概述

SH79F1624A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。

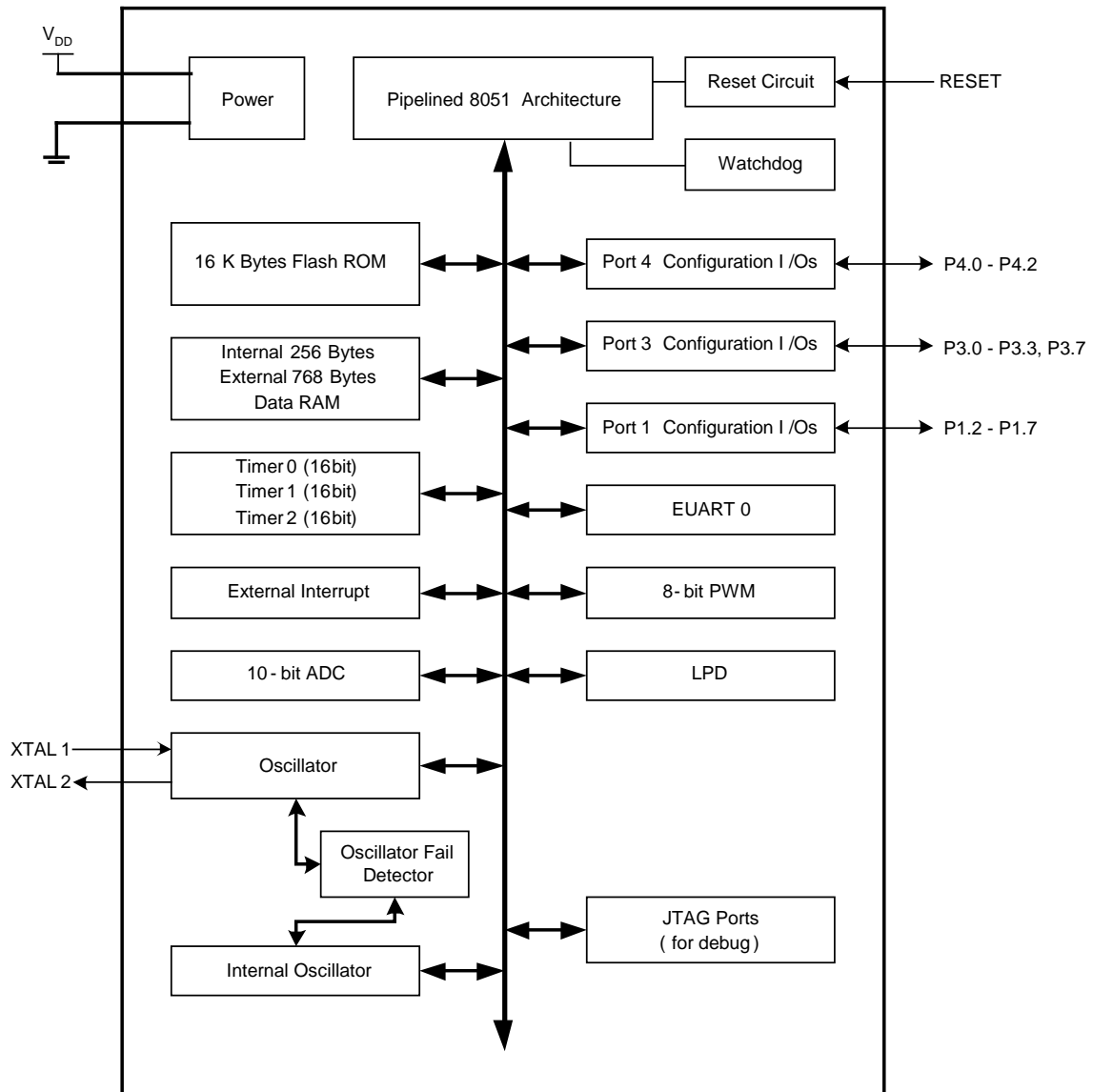
SH79F1624A保留了标准8051芯片的大部分特性。这些特性包括内置256字节RAM和2个16位定时器/计数器, 1个UART和外置中断INT0, INT1, INT2。此外, SH79F1624A还集成了256字节外部RAM, 可兼容8052芯片的16位定时器/计数器 (Timer2)。该单片机还包括适合于程序和数据的16K字节Flash块。

SH79F1624A不仅集成了EUART标准通讯模块, 此外还集成了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, SH79F1624A内建看门狗定时器, 低电压复位功能和低电压检测功能。此外SH79F1624A还提供了2种低功耗省电模式。

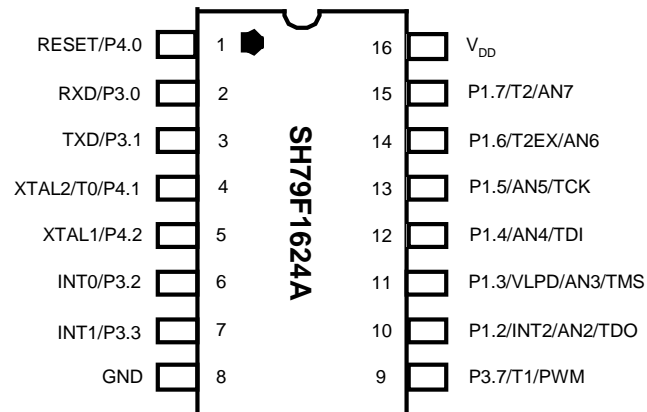


3. 方框图





#### 4. 引脚配置



引脚配置图

总计：16引脚。

**注意：**引脚命名中，写在最外侧的引脚功能具有最高优先级，最内侧的引脚功能具有最低优先级（参见引脚配置图）。当一个引脚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的引脚。只有当软件禁止引脚的高优先级功能，相应引脚才能被释放作为低优先级端口使用。

#### 引脚功能

引脚编号	引脚命名	默认功能
1	P4.0/RESET	RESET
2	RXD/P3.0	P3.0
3	TXD/P3.1	P3.1
4	P4.1/XTAL2	P4.1或振荡器输出引脚，代码选项控制
5	P4.2/XTAL1	P4.2或振荡器输入引脚，代码选项控制
6	INT0/P3.2	P3.2
7	INT1/P3.3	P3.3
8	GND	-----
9	PWM/T1/P3.7	P3.7
10	TDO/AN2/INT2/P1.2	P1.2
11	TMS/AN3/VLPD/P1.3	P1.3
12	TDI/AN4/P1.4	P1.4
13	TCK/AN5/P1.5	P1.5
14	AN6/T2EX/P1.6	P1.6
15	AN7/T2/P1.7	P1.7
16	V <sub>DD</sub>	-----



## 5. 引脚描述

引脚编号	类型	说明
<b>PORT</b>		
P4.0 - P4.2	I/O	3位双向I/O端口
P3.0 - P3.3, P3.7	I/O	5位双向I/O端口
P1.2 - P1.7	I/O	6位双向I/O端口
<b>Timer</b>		
T2	I/O	Timer2外部输入/波特率时钟输出
T2EX	I	Timer2重载/捕捉/方向控制
<b>PWM控制器</b>		
PWM	O	8位PWM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>		
AN2 - AN7	I	ADC输入通道
<b>中断&amp;复位&amp;时钟&amp;电源</b>		
INT0 - INT2	I	外部中断0-2
RESET	I	复位引脚（高电平复位）
XTAL1	I	谐振器输入
XTAL2	O	谐振器输出
V <sub>DD</sub>	P	电源（2.8 - 5.5V）
GND	P	接地
<b>VLPD</b>		
VLPD	I	电源电压输入检测
<b>编程器</b>		
TDO	O	调试接口：测试数据输出
TMS	I	调试接口：测试模式选择
TDI	I	调试接口：测试数据输入
TCK	I	调试接口：测试时钟输入
<b>注意：</b> 当P1.2-1.5作为调试接口时，P1.2-1.5的原有功能被禁止		



**6. 产品信息**

<b>Part Num</b>	<b>RAM (byte)</b>	<b>Flash (byte)</b>	<b>EUART</b>	<b>ADC (10bit)</b>	<b>PWM (8bit)</b>	<b>Timer</b>	<b>ExINT</b>	<b>LPD Pin</b>	<b>Internal RC</b>	<b>IO</b>	<b>Package</b>
<b>SH79F1624A</b>	512	16K	1	6	1	3	3	Y	±2%	14	SOP



## 7. SFR映像

SH79F1624A内置256字节的直接寻址寄存器，包括通用数据存储器和特殊功能存储器（SFR），SH79F1624A的SFR有以下几种：

<b>CPU内核寄存器：</b>	ACC, B, PSW, SP, DPL, DPH
<b>CPU增强内核寄存器：</b>	AUXC, DPL1, DPH1, INSCON, XPAGE
<b>电源和时钟控制寄存器：</b>	PCON, SUSLO
<b>LPD寄存器：</b>	LPDCON
<b>Flash寄存器：</b>	IB_OFFSET, XPAGE, IB_DATA, IB_CON1, IB_CON2, IB_CON3, IB_CON4, IB_CON5, FLASHCON
<b>数据存储页寄存器：</b>	XPAGE
<b>系统时钟控制寄存器：</b>	CLKCON
<b>硬件看门狗定时器寄存器：</b>	RSTSTAT
<b>中断系统寄存器</b>	IEN0, IEN1, IPH0, IPL0, IPH1, IPL1, EXF0
<b>I/O端口寄存器：</b>	P1, P3, P4, P1M0, P1M1, P3M0, P3M1, P4M0, P4M1
<b>定时器寄存器：</b>	TCON, TMOD, TL0, TH0, TL1, TH1, TCON1, T2CON, T2MOD, RCAP2H, RCAP2L
<b>EUART寄存器：</b>	SCON, SBUF, SADEN, SADDR, PCON
<b>ADC寄存器：</b>	ADCON, ADT, ADCH, ADDL, ADDH
<b>PWM寄存器：</b>	PWMCON, PWMP, PWMD



# SH79F1624A

**Table 7.1** CPU core 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	AUXC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1低位字节	00000000	DPL.7	DPL.6	DPL.5	DPL.4	DPL.3	DPL.2	DPL.1	DPL.0
DPH	83H	数据指针1高位字节	00000000	DPH.7	DPH.6	DPH.5	DPH.4	DPH.3	DPH.2	DPH.1	DPH.0
DPL1	84H	数据指针2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2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	----00-0	-	-	-	-	DIV	MUL	-	DPS

**Table 7.2** 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

**Table 7.3** 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	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	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	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	flash控制寄存器2	---00000	-	-	-	IB_CON2.4	IB_CON2.3	IB_CON2.2	IB_CON2.1	IB_CON2.0
IB_CON3	F4H	flash控制寄存器3	----0000	-	-	-	-	IB_CON3.3	IB_CON3.2	IB_CON3.1	IB_CON3.0
IB_CON4	F5H	flash控制寄存器4	----0000	-	-	-	-	IB_CON4.3	IB_CON4.2	IB_CON4.1	IB_CON4.0
IB_CON5	F6H	flash控制寄存器5	----0000	-	-	-	-	IB_CON5.3	IB_CON5.2	IB_CON5.1	IB_CON5.0
FLASHCON	A7H	Flash 访问控制寄存器	-----0	-	-	-	-	-	-	-	FAC



## SH79F1624A

**Table 7.4** 数据存储页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
XPAGE	F7H	flash页寄存器	00000000	XPAGE.7	XPAGE.6	XPAGE.5	XPAGE.4	XPAGE.3	XPAGE.2	XPAGE.1	XPAGE.0

**Table 7.5** 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	看门狗定时器控制寄存器	0-000000	WDOF	-	PORF	LVRF	CLRF	WDT.2	WDT.1	WDT.0

*注意：RSTSTAT初始值根据不同类型的复位而不同。*

**Table 7.6** 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	系统时钟选择	111000--	32K_SPDUP	CLKS1	CLKS0	SCMIF	RCON	FS	-	-

**Table 7.7** 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	中断允许控制0	00000000	EA	EADC	ET2	ES0	ET1	EX1	ET0	EX0
IEN1	A9H	中断允许控制1	0-00-0--	ELPD	-	EPWM	ESCM	-	EX2	-	-
EXF0	E8H	外部中断标志0	---00-0	-	-	-	-	IT2.1	IT2.0	-	IE2
IPL0	B8H	中断优先权控制低位0	-0000000	-	PADCL	PT2L	PSL	PT1L	PX1L	PT0L	PX0L
IPH0	B4H	中断优先权控制高位0	-0000000	-	PADCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
IPL1	B9H	中断优先权控制低位1	0-00-0--	PLPDL	-	PPWML	PSCML	-	PX2L	-	-
IPH1	B5H	中断优先权控制高位1	0-00-0--	PLPDH	-	PPWMH	PSCMH	-	PX2H	-	-





## SH79F1624A

Table 7.8 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
P1	90H	6位端口1	111111--	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	-	-
P3	B0H	5位端口3	1---1111	P3.7	-	-	-	P3.3	P3.2	P3.1	P3.0
P4	C0H	3位端口4	-----111	-	-	-	-	-	P4.2	P4.1	P4.0
P1M0	EAH		00000000	P1M07	P1M06	P1M05	P1M04	P1M03	P1M02	P1M01	P1M00
P1M1	E2H		00000000	P1M17	P1M16	P1M15	P1M14	P1M13	P1M12	P1M11	P1M10
P3M0	ECH		0-000000	P3M07	-	P3M05	P3M04	P3M03	P3M02	P3M01	P3M00
P3M1	E4H		0-000000	P3M17	-	P3M15	P3M14	P3M13	P3M12	P3M11	P3M10
P4M0	EDH		-----000	-	-	-	-	-	P4M02	P4M01	P4M00
P4M1	E5H		-----000	-	-	-	-	-	P4M12	P4M11	P4M10

Table 7.9 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	定时器/计数器0和1控制	00000000	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
TMOD	89H	定时器/计数器0和1模式	00000000	GATE1	C/T1	M11	M10	GATE0	C/T0	M01	M00
TL0	8AH	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	8CH	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	8BH	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	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
T2CON	C8H	定时器2控制	00000000	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
T2MOD	C9H	定时器2模式	0----00	TCLKP2	-	-	-	-	-	T2OE	DCEN
RCAP2L	CAH	定时器2重载/捕获数据低位字节	00000000	RCAP2L.7	RCAP2L.6	RCAP2L.5	RCAP2L.4	RCAP2L.3	RCAP2L.2	RCAP2L.1	RCAP2L.0
RCAP2H	CBH	定时器2重载/捕获数据高位字节	00000000	RCAP2H.7	RCAP2H.6	RCAP2H.5	RCAP2H.4	RCAP2H.3	RCAP2H.2	RCAP2H.1	RCAP2H.0
TL2	CCH	定时器2低位计数位	00000000	TL2.7	TL2.6	TL2.5	TL2.4	TL2.3	TL2.2	TL2.1	TL2.0
TH2	CDH	定时器2高位计数位	00000000	TH2.7	TH2.6	TH2.5	TH2.4	TH2.3	TH2.2	TH2.1	TH2.0
TCON1	CEH	定时器/计数器0和1选择和比较控制	-00-0000	-	TCLKS1	TCLKS0	-	TCLKP1	TCLKP0	TC1	TC0



## SH79F1624A

**Table 7.10 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	串行控制	00000000	SM0/FE	SM1/RXOV	SM2/TXCOL	REN	TB8	RB8	TI	RI
SBUF	99H	串行数据缓冲器	00000000	SBUF.7	SBUF.6	SBUF.5	SBUF.4	SBUF.3	SBUF.2	SBUF.1	SBUF.0
SADDR	9AH	从属地址	00000000	SADDR.7	SADDR.6	SADDR.5	SADDR.4	SADDR.3	SADDR.2	SADDR.1	SADDR.0
SADEN	9BH	从属地址掩码	00000000	SADEN.7	SADEN.6	SADEN.5	SADEN.4	SADEN.3	SADEN.2	SADEN.1	SADEN.0
PCON	87H	电源和串行控制	00--0000	SMOD	SSTAT	-	-	GF1	GF0	PD	IDL

**Table 7.11 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	ADC控制	000-0000	ADON	ADCIF	EC	-	SCH2	SCH1	SCH0	GO/DONE
ADT	94H	ADC时间配置	000-0000	TADC2	TADC1	TADC0	-	TS3	TS2	TS1	TS0
ADCH	95H	ADC通道选择	00000000	CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0
ADDL	96H	ADC数据低位字节	-----00	-	-	-	-	-	-	A1	A0
ADDH	97H	ADC数据高位字节	00000000	A9	A8	A7	A6	A5	A4	A3	A2

**Table 7.12 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
PWMCON	D1H	8位PWM控制	0000--00	PWMEN	PWMS	PWMCK1	PWMCK0	-	-	PWMIF	PWMSS
PWMP	D2H	8位PWM周期控制低位	00000000	PWMP.7	PWMP.6	PWMP.5	PWMP.4	PWMP.3	PWMP.2	PWMP.1	PWMP.0
PWMD	D3H	8位PWM占空比控制低位	00000000	PWMD.7	PWMD.6	PWMD.5	PWMD.4	PWMD.3	PWMD.2	PWMD.1	PWMD.0

**Table 7.13 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	LPD检测控制	000---00	LPDEN	LPDF	LPDV	-	-	-	LPDS1	LPDS0

注意：-：保留位。



SFR映像

	可按位寻址		不可按位寻址						
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8H				IB_OFFSET	IB_DATA				FFH
F0H	B	AUXC	IB_CON1	IB_CON2	IB_CON3	IB_CON4	IB_CON5	XPAGE	F7H
E8H	EXF0		P1M0		P3M0	P4M0			EFH
E0H	ACC		P1M1		P3M1	P4M1			E7H
D8H									DFH
D0H	PSW	PWMCON	PWMP	PWMD					D7H
C8H	T2CON	T2MOD	RCAP2L	RCAP2H	TL2	TH2	TCON1		CFH
C0H	P4								C7H
B8H	IPL0	IPL1							BFH
B0H	P3	RSTSTAT	CLKCON	LPDCON	IPH0	IPH1			B7H
A8H	IEN0	IEN1							AFH
A0H			SPCON	SPDAT				FLASHCON	A7H
98H	SCON	SBUF	SADDR	SADEN					9FH
90H	P1			ADCON	ADT	ADCH	ADDL	ADDH	97H
88H	TCON	TMOD	TL0	TL1	TH0	TH1	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。



## 8. 标准功能

### 8.1 CPU

#### 8.1.1 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。

Table 8.1 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	C	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	C	<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

##### 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。



### 8.1.2 CPU增强内核特殊功能寄存器

#### 特性

- 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

SH79F1624A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。

### 8.1.3 寄存器

Table 8.2 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	-	-	-	-	DIV	MUL	-	DPS
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
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



## 8.2 随机数据存储器 (RAM)

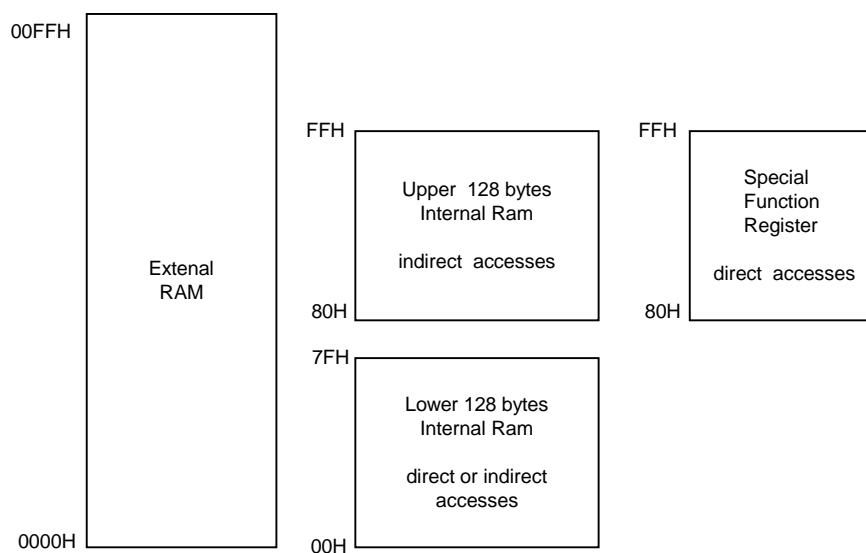
### 8.2.1 特性

SH79F1624A为数据存储提供了内部256字节的RAM和外部256字节的RAM。下列为存储器空间分配:

- 低位128字节的内部RAM (地址从00H到7FH) 可直接或间接寻址
- 高位128字节的内部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256字节的RAM (地址从00H到FFH) 可通过MOVX指令间接寻址

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。当一个指令访问地址高于7FH的内部位置时,CPU可以根据指令的寻址方式来区分是访问高位128字节数据RAM还是访问SFR。

**注意:** 未使用的SFR地址禁止读写。



内部和外部RAM配置

SH79F1624A支持传统的访问外部RAM方法。使用MOVXA, @Ri或MOVX @Ri, A来访问外部低位256字节RAM; 用MOVX A, @DPTR或MOVX @DPTR, A来访问外部256字节RAM。

用户也能用XPAGE寄存器来访问外部RAM, 使用MOVX A, @Ri或MOVX @Ri, A指令即可, 此时用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)。

### 8.2.2 寄存器

Table 8.3 数据存储页寄存器

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.0
读/写	-	-	-	-	-	-	-	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	-	-	0

位编号	位符号	说明
0	XPAGE[0]	RAM页选择控制位

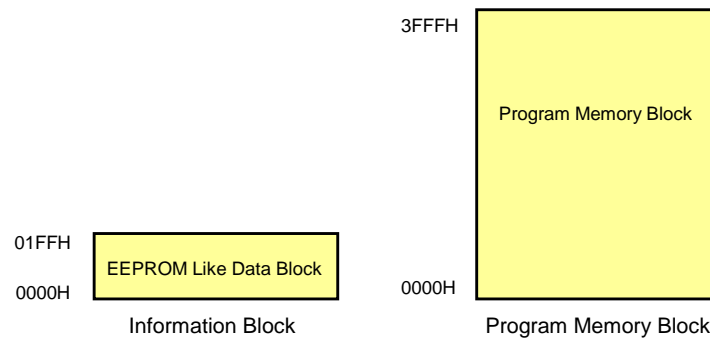
**注意:** 因为只有256字节外部RAM所以, XPAGE[0]只有0有效。



### 8.3 Flash存储器

#### 8.3.1 特性

- Flash 存储器包括 16 X 1KB 扇区，总共 16KB
- 在工作电压范围内都能进行编程和擦除操作
- 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：至少 100000 次
- 数据保存年限：至少 10 年
- 低功耗



SH79F1624A为存储程序代码内置16K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。

SH79F1624A还内置512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2个扇区。

**Flash存储器支持以下操作:****(1) 代码保护控制模式编程**

SH79F1624A的代码保护功能为用户代码提供了高性能的安全措施。每个分区有两种模式可用。

代码保护模式0: 允许/禁止任何编程器的写入/读取操作（不包括整体擦除）。

代码保护模式1: 允许/禁止在其它扇区中通过MOV<sub>C</sub>指令进行读取操作，或通过SSP模式进行擦除/写入操作。

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：**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。**

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：**Flash编程器在ICP模式发出整体擦除指令，进行整体擦除。**

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。

若需Flash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Flash编程器都能执行该操作。

若需用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。不管安全位设置与否，用户程序都能读/写程序自身所在扇区。

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/读代码指令，进行写/读代码。

**(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/读类EEPROM存储区指令，进行写/读类EEPROM存储区。

**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



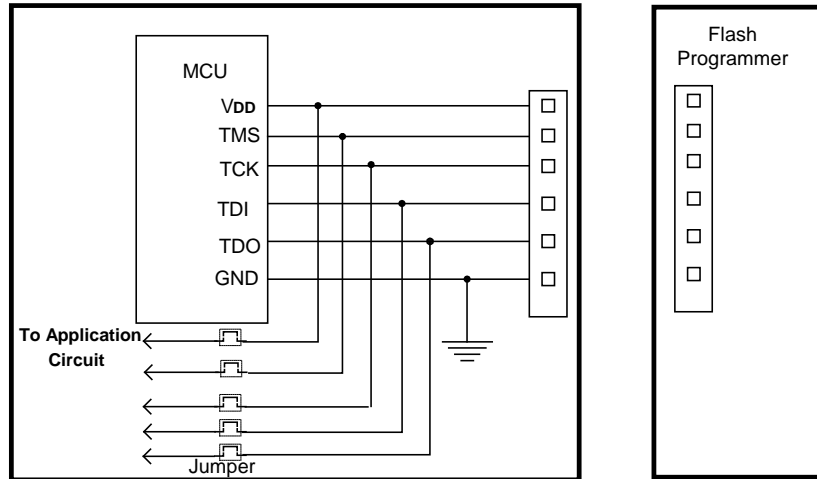


## 8.3.2 ICP模式下的Flash操作

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 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。



### 8.4 扇区自编程 (SSP) 功能

SH79F1624A支持SSP（扇区自编程）功能。如果所选扇区未被保护，用户代码可以擦除所有扇区或对所有扇区执行编程操作。一旦该扇区被编程，则在该扇区被擦除之前不能被再次编程。

SH79F1624A内建一个复杂控制流程以避免误入SSP模式导致代码被误修改。为进入SSP模式，IB\_CON1-5必须满足特定条件。若IB\_CON1-5不满足特定条件，则无法进入SSP模式。

#### 8.4.1 寄存器

Table 8.4 编程用地址选择寄存器

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.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
复位值 (POR/WDT/LVR/PIN)	-	-	0	0	0	0	0	0

对Flash程序存储区，一个扇区为1024字节，该寄存器定义如下：

位编号	位符号	说明
5-2	XPAGE[5:2]	被编程的存储单元扇区号，0000代表扇区0，以此类推
1-0	XPAGE[1:0]	被编程的存储单元高2位地址

对类EEPROM存储区，一个扇区为256字节，该寄存器定义如下：

位编号	位符号	说明
7-1	XPAGE[7:1]	保留位
0	XPAGE[0]	被编程的存储单元扇区号；0代表扇区0，以此类推

注意：

对于程序存储区，一个扇区为1024字节；

对于类EEPROM存储区，一个扇区为256字节。

Table 8.5 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

对Flash程序存储区，一个扇区为1024字节，该寄存器定义如下：

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

对类EEPROM存储区，一个扇区为256字节，该寄存器定义如下：

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



Table 8.6 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						

Table 8.7 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</b> 0xE6: 扇区擦除 0x6E: 存储单元编程						

Table 8.8 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 8.9 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 8.10 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

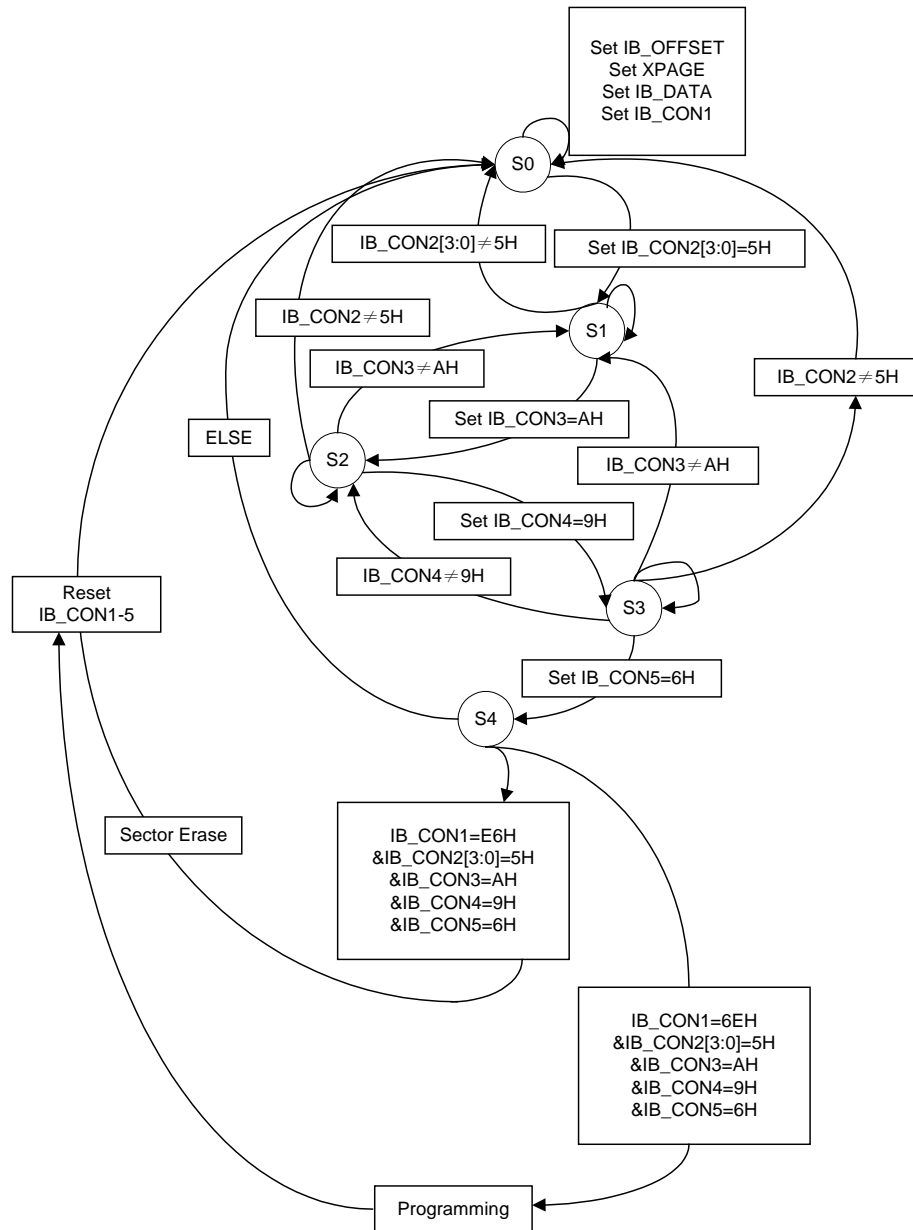
Table 8.11 SSP流程控制寄存器4

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



8.4.2 Flash控制流程图





### 8.4.3 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) 用于代码/数据编程：

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) 用于扇区擦除：

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”或者“MOVC A, @A+PC”。

#### (4) 关于类EEPROM区域

SH79F1624A具有512Byte的类EEPROM，地址是从0000H - 01FFH。对于类EEPROM的操作类似于Flash的操作，即类似于上述部分的描述。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：**当不需要对类EEPROM操作时，必须将FAC位清0。

### 8.4.4 可读识别码

SH79F1624A每颗芯片出厂后都固化有一个8位的可读识别码，它的值为0-255的随机值，它是无法擦除的。它可以由程序或编程工具读出。

读识别码时，首先，设置FAC位为1，然后给DPTR赋值“0A7FH”，将A清0，再使用“MOVC A, @A+DPTR”来读取。

Table 8.12 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
7-1	-	保留位
0	FAC	访问控制 0: MOVC指令或者SSP功能访问Main Block区域 1: MOVC指令或者SSP功能访问类EEPROM区域

**注意：**此识别码虽然采用读程式ROM的指令来读取，但并未存放在程式ROM区，指令靠FAC来区分访问识别码还是访问程式ROM区。因此读完识别码后必须将FAC清0，否则会影响用户程式读程式ROM的指令执行。



## 8.5 系统时钟和振荡器

### 8.5.1 特性

- 支持5种振荡器类型：32.768kHz晶体谐振器，晶体谐振器，陶瓷谐振器，外部时钟和内部16.6M RC振荡器
- 内建16.6MHz（±2%）RC振荡器
- 内建32.768kHz加速电路
- 内建系统时钟分频器

### 8.5.2 时钟定义

SH79F1624A几个内部时钟定义如下：

**OSCCLK:** 从5个可选振荡器类型中（从XTAL输入的32.768kHz晶体谐振器，晶体谐振器，陶瓷谐振器和外部时钟以及内部16.6MHz RC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。

**WDTCLK:** 内部的32kHz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或者内建RC振荡器的频率。 $f_{OSCS}$ 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。

### 8.5.3 概述

SH79F1624A支持5种振荡器类型：32.768kHz晶体谐振器，晶体谐振器（400kHz-16.6MHz），陶瓷谐振（400kHz-16.6MHz），外部时钟（30kHz-16.6MHz）和内部RC振荡器（16.6MHz）。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）。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。



### 8.5.4 寄存器

Table 8.13 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	32K_SPDUP	CLKS1	CLKS0	SCMIF	RCON	FS	-	-
读/写	读/写	读/写	读/写	读/写	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振荡器加速模式控制位</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>应该需要注意的是关闭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011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></p> <p>01: <math>f_{sys} = f_{osc}/2</math></p> <p>10: <math>f_{sys} = f_{osc}/4</math></p> <p>11: <math>f_{sys} = f_{osc}/12</math></p> <p>如果选择32.768kHz振荡器为OSCSCLK，此控制位无效。</p>
3	RCON	<p><b>内建RC振荡器控制</b></p> <p>0: 关闭内建RC振荡器</p> <p>1: 打开内建RC振荡器</p> <p>仅当OP_OSC[2:0]为011时，此控制位才有效。（选择32.768kHz晶体振荡器，详见代码选项章节）</p>
2	FS	<p><b>频率选择位</b></p> <p>0: 选择32.768kHz作为OSCSCLK</p> <p>1: 选择内建RC振荡器作为OSCSCLK</p> <p>只有代码选项OP_OSC为011时，此控制位才有效。（选择32.768kHz晶体振荡器，详见代码选项章节）</p>

**注意:**

RCON和FS位仅当OP\_OSC[2:0]为011时有效；当选择内建RC振荡器作为系统时钟时（即当RCON = 1和FS = 1），RCON不能由软件清除，系统时钟监控功能不可用。

当系统时钟由32.768kHz切换到内建RC振荡器时，操作必须遵循以下顺序：

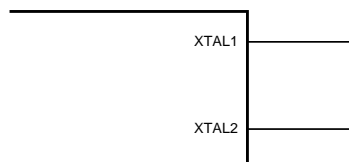
- (1) 置位RCON，开启内建RC振荡器；
- (2) 等待至少2个振荡器周期；
- (3) 置位FS，将系统时钟切换到内建RC振荡器。



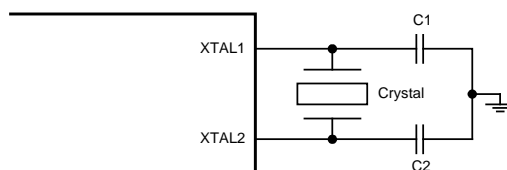


### 8.5.5 振荡器类型

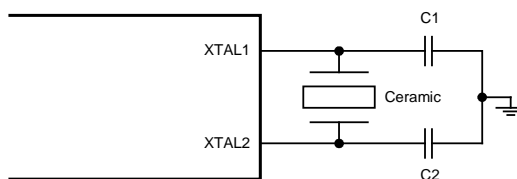
(1) 内建RC振荡器：16.6MHz



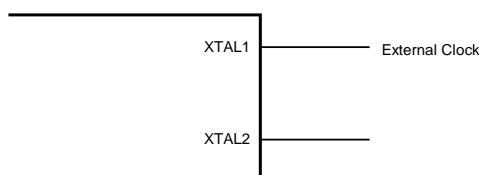
(2) 晶体谐振器：32.768kHz和内部RC振荡器：16.6MHz



(3) 晶体/陶瓷谐振器：400kHz - 16.6MHz



(4) 外部时钟：30kHz - 16.6MHz



### 8.5.6 谐振器负载电容选择

陶瓷谐振器		
频率	C1	C2
455kHz	47 - 100pF	47 - 100pF
3.58MHz	-	-
4MHz	-	-

\* 已经内建负载电容

#### 注意:

(1) 表中负载电容为设计参考数据!

(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。



## 8.6 系统时钟监控 (SCM)

为了增强系统的可靠性，SH79F1624A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WDT时钟 (WDTCLK)，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 8.14 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	32K_SPDUP	CLKS1	CLKS0	SCMIF	RCON	FS	-	-
读/写	读/写	读/写	读/写	只读	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
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



## 8.7 I/O端口

### 8.7.1 特性

- 14个双向I/O端口
- 4种可选IO模式
- I/O端口可与其他功能共享

SH79F1624A提供14个位可编程双向I/O端口。所有I/O可以通过PxMy寄存器设置成以下4种模式中的一种：准双向模式（传统8051模式）、推挽输出模式、开漏输出模式和仅输入模式。

用户可以通过代码选项设置所有I/O复位后默认为准双向模式或者默认为仅输入模式。

为了提高抗干扰能力，每个输入引脚都带有一个施密特触发器。即使处于掉电状态，施密特触发器也不会关闭。

SH79F1624A的I/O引脚能与其它功能选择复用。当所有功能都允许时，在CPU中存在优先级以避免功能冲突。（具体请参考端口共享章节）。注意当I/O工作于其它功能时，即使改写PxMy寄存器也不会改变I/O的模式，也不会改变当前PxMy寄存器中的值。只有当其它功能关闭时，才允许通过改写相应的寄存器来改变I/O的模式。

### 8.7.2 寄存器

Table 8.15 端口控制寄存器

E2H, E4H, E5H EAH, ECH, EDH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P1M0 (EAH)	P1M07	P1M06	P1M05	P1M04	P1M03	P1M02	-	-
P1M1 (E2H)	P1M17	P1M16	P1M15	P1M14	P1M13	P1M12	-	-
P3M0 (ECH)	P3M07	-	-	-	P3M03	P3M02	P3M01	P3M00
P3M1 (E4H)	P3M17	-	-	-	P3M13	P3M12	P3M11	P3M10
P4M0 (EDH)	-	-	-	-	-	P4M02	P4M01	P4M00
P4M1 (E5H)	-	-	-	-	-	P4M12	P4M11	P4M10
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)*	*	*	*	*	*	*	*	*

\*: 复位值根据不同的代码选项有不同。可通过代码选项设置复位后为准双向模式或者仅输入模式（高阻态）

#### 端口模式设置说明

PxM0n	PxM1n	说明
0	0	准双向模式
0	1	推挽输出模式
1	0	仅输入模式（高阻态）
1	1	开漏输出模式

(x = 1, 3或4 n = 7, 6, 5, 4, 3, 2, 1或0)

Table 8.16 端口数据寄存器

90H-C0H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
P1 (90H)	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	-	-
P3 (B0H)	P3.7	-	-	-	P3.3	P3.2	P3.1	P3.0
P4 (C0H)	-	-	-	-	-	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)	*	*	*	*	*	*	*	*

\*: 复位值由代码选项决定。若选择上电后为准双向结构，则复位值为0FFH；若为仅输入结构，则复位值为00H。

位编号	位符号	说明
7-0	Px.y x = 1-4, y = 0-7	端口数据寄存器

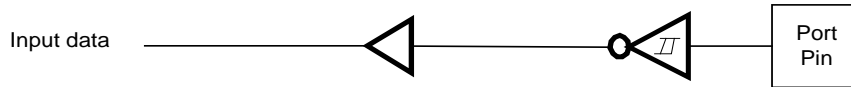
注意：所有端口可作为N-沟道的开漏I/O，但是此时端口电压不得超过V<sub>DD</sub>+0.3V。





**仅输入模式 (Input-Only)**

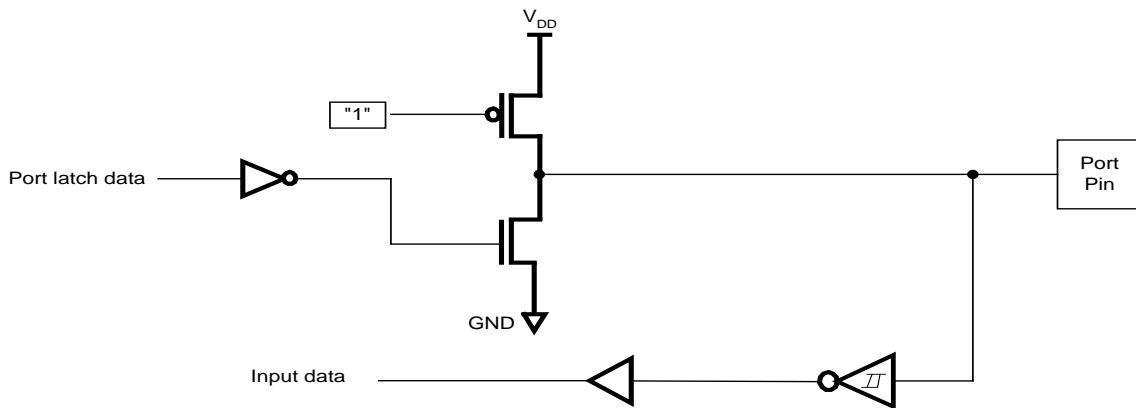
此种模式仅有输入，没有输出能力。仅输入模式的端口结构示意图如下所示。



仅输入模式

**开漏输出模式 (Open-Drain)**

此种模式没有输出高的能力。如果需要输出高，用户必须外接上拉电阻。注意此时外加引脚电压不得超过 $V_{DD}+0.3V$ 。开漏输出模式的端口结构示意图如下所示。



开漏输出模式



### 8.7.4 端口共享

14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：

在引脚配置中引脚最外部标注功能享有最高优先级，而最内部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。只有当较高优先级功能由软件关闭后，相应的引脚才能用作较低优先级功能。

如果第二功能允许，对端口的读写都是针对端口寄存器的操作。

14个双向I/O端口可以提供一些特殊功能：

#### PORT1:

- AN2 - AN7 (P1.2 - P1.7) : ADC模拟输入通道
- T2 (P1.7) : 定时器2外部输入/波特率时钟输出
- T2EX (P1.6) : 定时器2重载/捕捉/方向控制
- VLDP (P1.3) : 电源电压输入检测
- INT2 (P1.2) : 外部中断2

Table 8.17 PORT1共享列表

引脚编号	优先级	功能	允许位
15	1	AN7	ADCH寄存器中ADCH.7位置1且ADCON寄存器中SCH[2:0] = 111
	2	T2	T2CON寄存器中TR2位和T2MOD寄存器中C/T2位置1
	3	P1.7	ADCH寄存器中ADCH.7位, T2CON寄存器中TR2位和T2MOD寄存器中C/T2位清0
14	1	AN6	ADCH寄存器中ADCH.6位置1和ADCON寄存器中SCH[2:0] = 110
	2	T2EX	T2CON寄存器中TR2位, T2MOD寄存器中C/T2位和EXEN2位置1
	3	P1.6	ADCH寄存器中ADCH.6位, T2CON寄存器中TR2位, T2MOD寄存器中C/T2位和EXEN2位清0
13	1	AN5	ADCH寄存器中ADCH.5位置1和ADCON寄存器中SCH[2:0] = 101
	2	P1.5	ADCH寄存器中ADCH.5位清0
12	1	AN4	ADCH寄存器中ADCH.4位置1且ADCON寄存器中SCH[2:0] = 100
	2	P1.4	ADCH寄存器中ADCH.4位清0
11	1	AN3	ADCH寄存器中ADCH.3位置1且ADCON寄存器中SCH[2:0] = 011
	2	VLDP	LPDCON中的LPDV位置1
	3	P1.3	ADCH寄存器中ADCH.3位和LPDV中的LPDEN位清0
10	1	AN2	ADCH寄存器中ADCH.2位置1且ADCON寄存器中SCH[2:0] = 010
	2	INT2	IEN1寄存器中EX2位置1
	3	P1.2	ADCH寄存器中ADCH.2位和IEN1寄存器中EX2位清0

**PORT3:**

- RXD (P3.0) : EUART数据输入
- TXD (P3.1) : EUART数据输出
- INT0 (P3.2) : 外部中断0
- INT1 (P3.3) : 外部中断1
- T1 (P3.7) : 定时器1外部输入
- PWM (P3.7) : PWM输出

**Table 8.18** PORT3共享列表

引脚编号	优先级	功能	允许位
2	1	RXD	SCON寄存器中REN位置1
	2	P3.0	SCON寄存器中REN位清0
3	1	TXD	对SBUF寄存器写操作
	2	P3.1	不对SBUF1寄存器写操作
6	1	INT0	IEN0寄存器中EX0位置1
	2	P3.2	IEN0寄存器中EX0位清0
7	1	INT1	IEN0寄存器中EX1位置1
	2	P3.3	IEN0寄存器中EX1位清0
9	1	PWM	PWMCON寄存器中PWMEN位和PWMSS位置1
	2	T1	TCON寄存器中TR1位和TMOD寄存器中C/T1位置1
	3	P3.7	PWMCON寄存器中PWMEN位和PWMSS位及TCON寄存器中TR1位和TMOD寄存器中C/T1位清0

**PORT4:**

- RESET (P4.0) : 复位
- XTAL2 (P4.1) : 谐振器输出
- XTAL1 (P4.2) : 谐振器输入

**Table 8.19** PORT4共享列表

引脚编号	优先级	功能	允许位
1	1	RESET	代码选项
	2	P4.0	代码选项
4	1	XTAL2	代码选项
	2	P4.1	代码选项
5	1	XTAL1	代码选项
	2	P4.2	代码选项

**注意:** RESET 引脚可与 P4.0 共享, 引脚功能通过代码选项 (OP\_RST) 进行选择。



## 8.8 定时器

### 8.8.1 特性

- SH79F1624A有3个定时器（定时器0，1，2）
- 定时器0兼容标准的8051
- 定时器1兼容标准的8051
- 定时器2兼容标准的8052，且有递增递减计数和可编程输出功能
- 定时器0/1增加了比较输出功能
- 定时器0/1增加了时钟源选择功能
- 定时器0/1/2增加了时钟源分频功能

### 8.8.2 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）。

TCLKP1和TCLKP0两位寄存器用于对系统时钟或12分频进行选择。

当作为定时器应用时，可在定时器x（x = 0, 1）的时钟源选择寄存器中配置TCLKS1和TCLKS0两位分别选择32.768kHz晶体谐振器作为定时器0和定时器1的时钟源。但TCLKS1和TCLKS0两位寄存器仅在代码选项选择32.768kHz晶体谐振器作为时钟源时才可设置操作。

#### 定时器x的方式（x = 0, 1）

通过计数器/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位计数器/定时器

在方式0中，定时器x为13位计数器/定时器。THx寄存器存放13位计数器/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。C/Tx位选择计数器/定时器的时钟源。

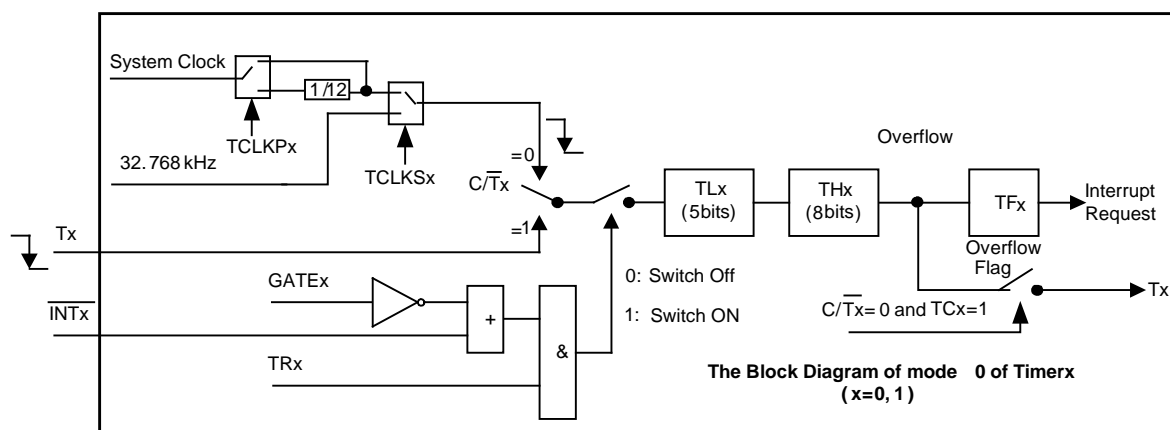
如果C/Tx = 1，定时器x输入引脚（Tx）的电平从高到低跳变，使定时器x数据寄存器加1。如果C/Tx = 0，选择系统时钟为定时器x的时钟源。

当GATEx = 0或GATEx = 1且输入信号INTx有效时，TRx置1打开定时器。GATEx置1允许定时器由外部输入信号INTx控制，便于测量INTx的正脉冲宽度。TRx位置1不强行复位定时器，这意味着如果TRx置1，定时器寄存器将从上次TRx清0时的值开始计数。所以在允许定时器之前，应该设定定时器寄存器的初始值。

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32.768kHz作为定时器x（x = 0, 1）的时钟源。TCLKSx（x = 0, 1）位仅在代码选项选择了32.768kHz晶体谐振器时才有效。

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。

当作为定时器应用时，可配置寄存器TCON1中的TC0/1位使定时器0/1溢出时T0/T1脚自动翻转。如果TC0/1被置1，T0/T1引脚自动设置为输出。

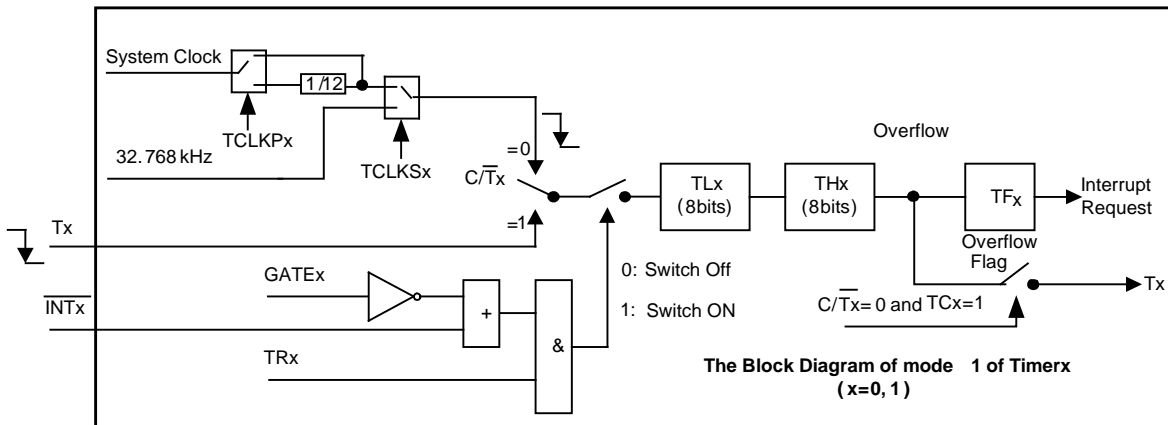






### 方式1：16位计数器/定时器

除了使用16位定时器/计数器之外，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位自动重载计数器/定时器

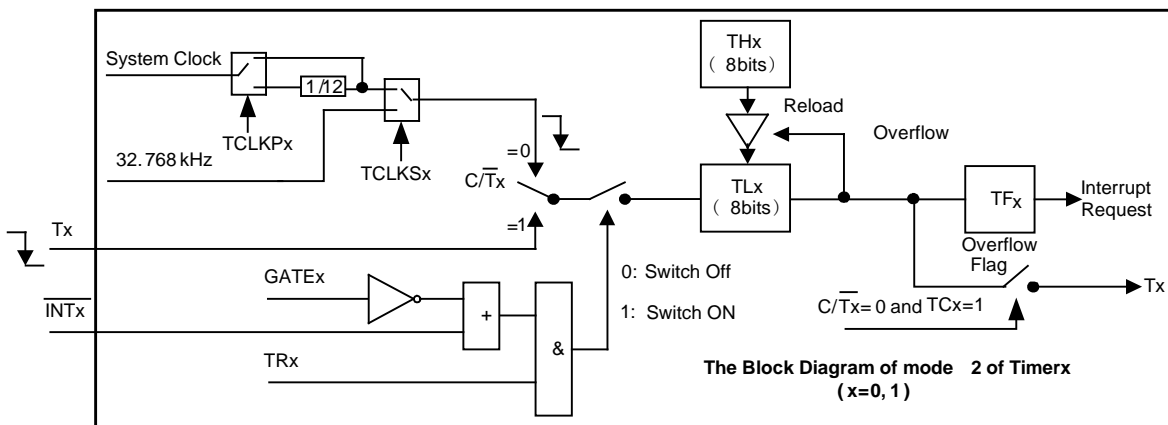
方式2中，定时器x是8位自动重载计数器/定时器。TLx存放计数值，THx存放重载值。当在TLx中的计数器溢出至0x00时，置起定时器溢出标志TFx，寄存器THx的值被重载入寄存器TLx中。如果定时器中断使能，当TFx置1时将产生一个中断。而在THx中的重载值不会改变。在允许定时器正确计数开始之前，TLx必须初始化为所需的值。

除了自动重载功能外，方式2中的计数器/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32.768kHz作为定时器x (x = 0, 1) 的时钟源。TCLKSx (x = 0, 1) 位仅在代码选项选择了32.768kHz晶体谐振器时才有效。

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。

当作为定时器应用时，可配置寄存器TCON1中的TC0/1位使定时器0/1溢出时T0/T1脚自动翻转。如果TC0/1被置1，T0/T1引脚自动设置为输出。





**方式3：两个8位计数器/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位计数器/定时器，分别由TL0和TH0控制。TL0通过定时器0的控制（在TCON中）和状态（在TMOD中）位：TR0，C/T0，GATE0和TF0控制。TL0能用系统时钟或32.768kHz或外部输入信号作为时钟源。

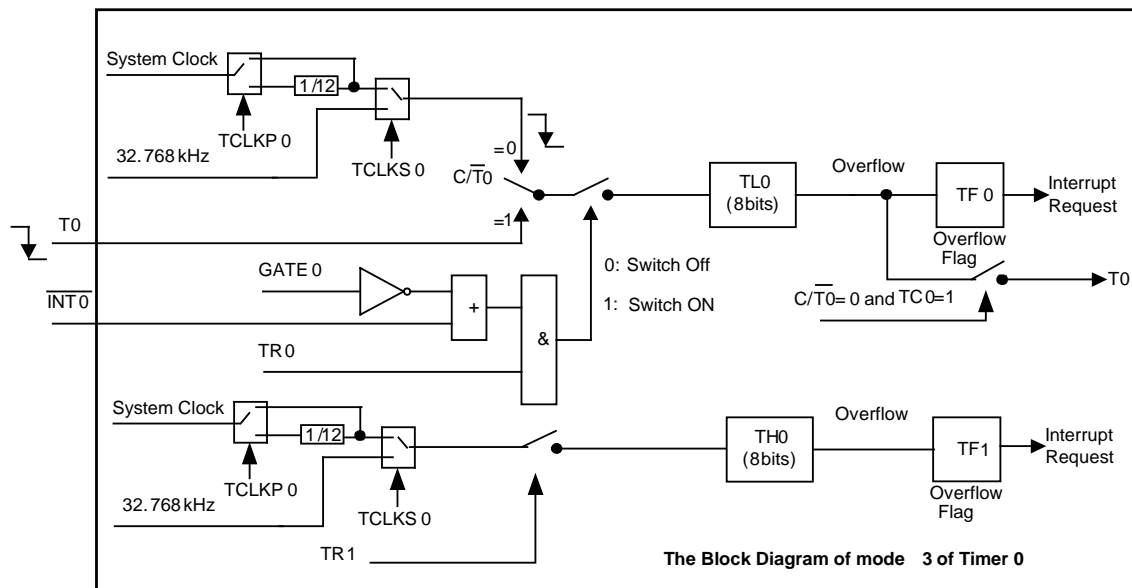
TH0只能用作定时器功能，时钟源来自系统时钟。TH0由定时器1的控制位TR1控制使能，溢出时定时器1溢出标志TF1置1，控制定时器1中断。

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，定时器1溢出可以用来产生串口的波特率。TH1和TL1只能用作定时器功能，时钟源来自系统时钟，GATE1位无效。T1输入脚的上拉电阻也无效。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TCLKS0位选择系统时钟或32.768kHz作为定时器0的时钟源。TCLKS0位仅在代码选项选择了32.768kHz晶体谐振器时才有效。

可配置寄存器TCON1中的TCLKP0位选择系统时钟或系统时钟的1/12作为定时器0的时钟源。

当作为定时器应用时，可配置寄存器TCON1中的TC0位使定时器0溢出时T0脚自动翻转。如果TC0被置1，T0引脚自动设置为输出。



**注意：**当定时器1作为波特率发生器时，读取或写入TH1/TL1会影响波特率的准确性，因此也会引起通信出错。



寄存器

Table 8.20 定时器/计数器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	IE1	IT1	IE0	IT0
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, 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
3, 1	IEx x = 0, 1	外部中断x请求标志位
2, 0	ITx x = 0, 1	外部中断x触发方式选择位

Table 8.21 定时器/计数器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	GATE1	C/T1	M11	M10	GATE0	C/T0	M01	M00
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	GATEx x = 0, 1	定时器x门控位 0: TRx置1, 定时器x即被允许 1: 只有INTx在高电平期间TRx置1, 定时器x才被允许
6, 2	C/Tx x = 0, 1	定时器/计数器方式选择位 0: 定时器方式 1: 计数器方式
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位向上计数计数器/定时器, 忽略TLx的第7-5位 01: 方式1, 16位向上计数计数器/定时器 10: 方式2, 8位自动重载向上计数计数器/定时器 11: 方式3 (只用于定时器0), 两个8位向上计数定时器



Table 8.22 定时器/计数器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 (8CH)	TH0.7	TH0.6	TH0.5	TH0.4	TH0.3	TH0.2	TH0.1	TH0.0
TL1 (8BH)	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 8.23 定时器/计数器x时钟源选择和比较输出寄存器 (x = 0, 1)

CEH	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
TC0N1	-	TCLKS1	TCLKS0	-	TCLKP1	TCLKP0	TC1	TC0
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
6, 5	TCLKSx x = 0, 1	定时器x时钟源控制位 0: 系统时钟作为定时器x时钟源 1: 32.768kHz作为定时器x时钟源
3-2	TCLKPx x = 0, 1	分频选择控制位 0: 选择系统时钟的1/12作为定时器x的时钟源 1: 系统时钟作为定时器x的时钟源
1-0	TCx x = 0, 1	比较输出功能允许位 0: 禁止定时器x比较输出功能 1: 允许定时器x比较输出功能



### 8.8.3 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）

定时器2的工作方式与定时器0和定时器1相似。 $\overline{C/T2}$ 选择系统时钟（定时器）或外部引脚T2（计数器）作为定时器时钟输入。通过所选的引脚设置TR2允许定时器2/计数器2数据寄存器计数。

#### 定时器2方式

定时器2有4种工作方式：捕获/重载，带递增或递减计数器的自动重载方式，波特率发生器和可编程时钟输出。RCLK, TCLK和CP/RL2的组合能选择这些方式。

#### 定时器2方式选择

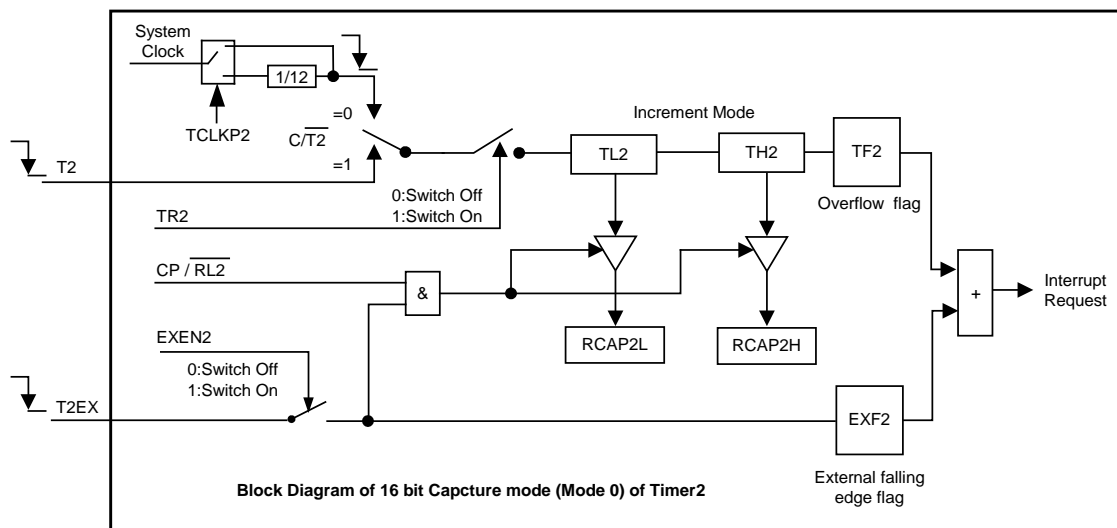
C/ $\overline{T2}$	T2OE	DCEN	TR2	CP/RL2	RCLK	TCLK	方式	
X	0	X	1	1	0	0	0	16位捕获
X	0	0	1	0	0	0	1	16位自动重载定时器
X	0	1	1	0	0	0		
X	0	X	1	X	1	X	2	波特率发生器
					X	1		
0	1	X	1	X	0	0	3	只用于可编程时钟
					1	X		
					X	1		
X	X	X	0	X	X	X	X	定时器2停止，T2EX通路仍旧允许

#### 方式0：16位捕获

在捕获方式中，T2CON的EXEN2位有两个选项。

如果EXEN2 = 0，定时器2作为16位定时器或计数器，如果ET2被允许的话，定时器2能设置TF2溢出产生一个中断。

如果EXEN2 = 1，定时器2执行相同操作，但是在外部输入T2EX上的下降沿也能引起在TH2和TL2中的当前值分别被捕获到RCAP2H和RCAP2L中，此外，在T2EX上的下降沿也能引起在T2CON中的EXF2被置1。如果ET2被允许，EXF2位也像TF2一样也产生一个中断。





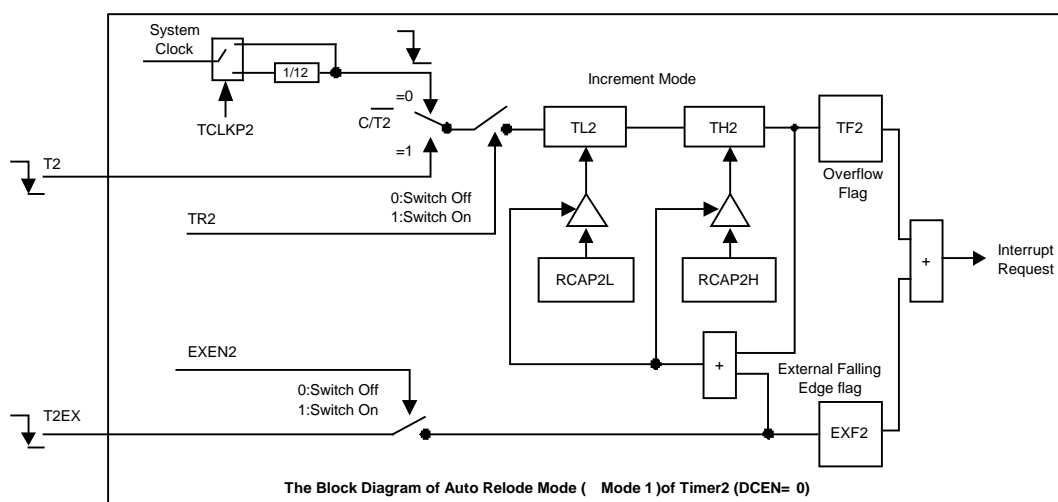
### 方式1：16位自动重载定时器

在16位自动重载方式下，定时器2可以被选为递增计数或递减计数。这个功能通过T2MOD中的DCEN位（递减计数允许）选择。系统复位后，DCEN位复位值为0，定时器2默认递增计数。当设置DCEN时，定时器2递增计数或递减计数取决于T2EX引脚上的电平。

当DCEN = 0，通过在T2CON中的EXEN2位选择两个选项。

如果EXEN2 = 0，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。

如果EXEN2 = 1，溢出或在外部输入T2EX上的下降沿都能触发一个16位重载，置起EXF2位。如果ET2被使能，TF2和EXF2位都能产生一个中断。

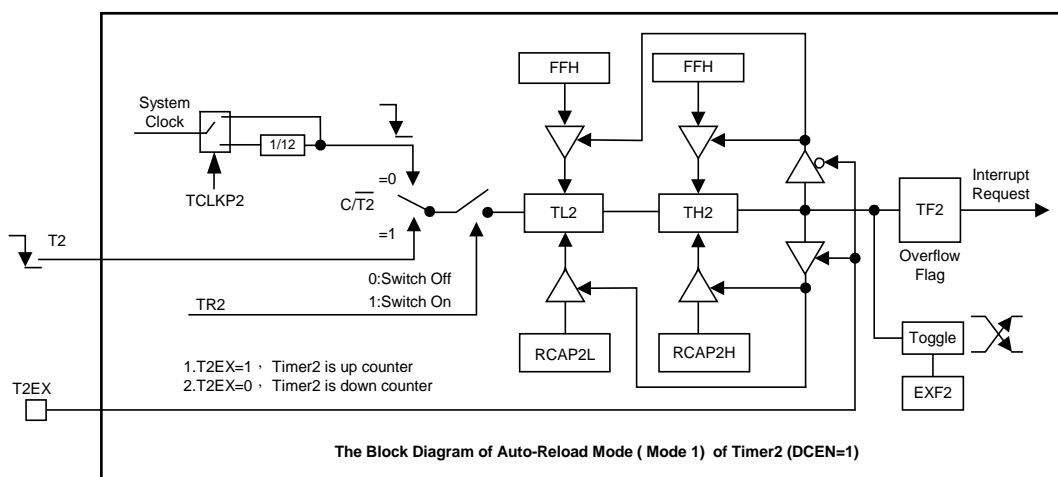


设置DCEN位允许定时器2递增计数或递减计数。当DCEN = 1时，T2EX引脚控制计数的方向，而EXEN2控制无效。

T2EX置1可使定时器2递增计数。定时器向0FFFFH溢出，然后设置TF2位。溢出也能分别引起RCAP2H和RCAP2L上的16位值重载入定时器寄存器。

T2EX清0可使定时器2递减计数。当TH2和TL2的值等于RCAP2H和RCAP2L的值时，定时器溢出。置起TF2位，同时0FFFFH重载入定时器寄存器。

无论定时器2溢出，EXF2位都被用作结果的第17位。在此工作方式下，EXF2不作为中断标志。





**方式2：波特率发生器**

通过设置T2CON寄存器中的TCLK和/或RCLK选择定时器2作为波特率发生器。接收器和发送器的波特率可以不同，如果定时器2作为接收器或发送器则定时器1相应的作为另一种的波特率发生器。

设置RCLK和/或TCLK使定时器2进入波特率发生器方式，该方式与自动重载方式相似。

定时器2的溢出会使RCAP2H和RCAP2L寄存器中的值重载入定时器2计数器，但不会产生中断。

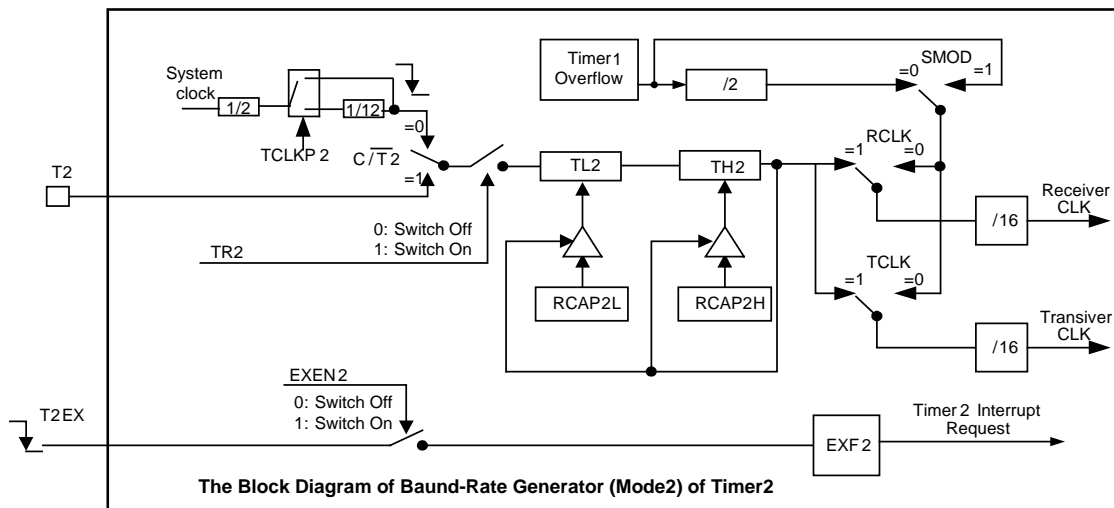
如果EXEN2被置1，在T2EX脚上的下降沿会置起EXF2，但不会引起重载。因此当定时器2作为波特率发生器时，T2EX可作为一个额外的外部中断。

在EUART方式1和3中的波特率由定时器2的溢出率根据下列方程式决定。

$$BaudRate = \frac{1}{2 \times 12 \times 16} \times \frac{f_{SYS}}{65536 - [RCAP2H, RCAP2L]} ; \quad C/T2 = 0, \quad TCLKP2 = 0$$

$$BaudRate = \frac{1}{2 \times 16} \times \frac{f_{SYS}}{65536 - [RCAP2H, RCAP2L]} ; \quad C/T2 = 0, \quad TCLKP2 = 1$$

$$BaudRate = \frac{1}{16} \times \frac{f_{T2}}{65536 - [RCAP2H, RCAP2L]} ; \quad C/T2 = 1$$





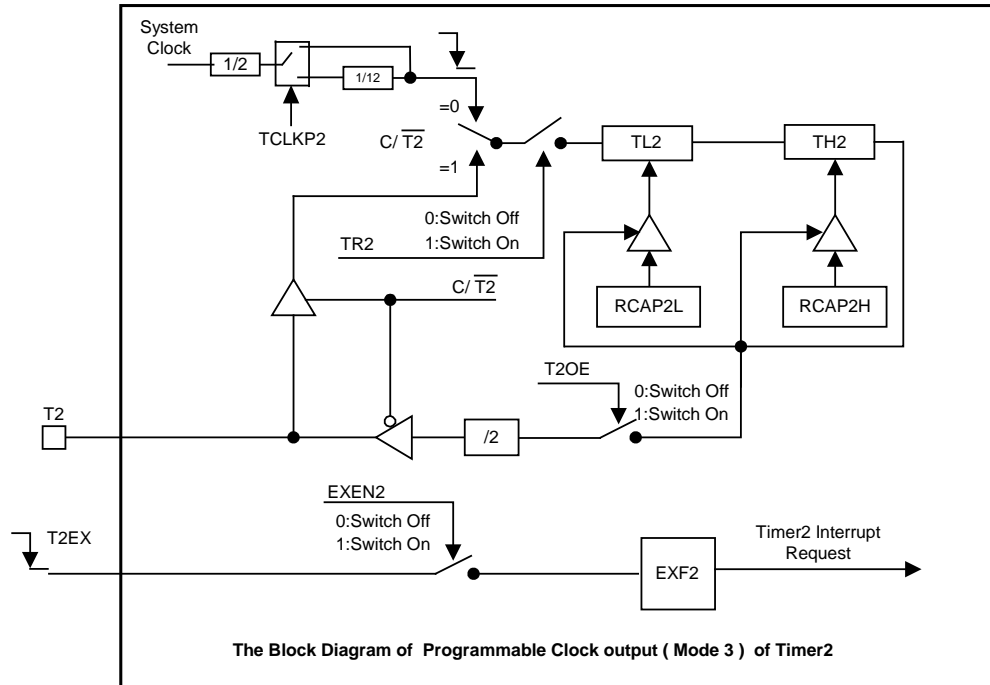
方式3：可编程时钟输出

P1.7可以编程输出50%的占空比时钟周期。清C/T2位和置T2OE位，使定时器2作为时钟发生器。TR2位启动和中止定时器。T2输出占空比为50%的时钟：

$$\text{Clock Out Frequency} = \frac{1}{2 \times 2 \times 12} \times \frac{f_{\text{SYS}}}{65536 - [\text{RCAP2H}, \text{RCAP2L}]} ; \text{TCLKP2} = 0$$

$$\text{Clock Out Frequency} = \frac{1}{2 \times 2} \times \frac{f_{\text{SYS}}}{65536 - [\text{RCAP2H}, \text{RCAP2L}]} ; \text{TCLKP2} = 1$$

定时器2溢出不产生中断，所以定时器2可以同时以相同频率用作波特率发生器和时钟输出。



注意：

- (1) TF2和EXF2都能引起定时器2的中断请求，两者有相同的向量地址。
- (2) 当事件发生时或其它任何时间都能由软件设置TF2和EXF2为1，只有软件以及硬件复位才能使之清0。
- (3) 当EA = 1且ET2 = 1时，设置TF2或EXF2为1能引起定时器2中断。
- (4) 当定时器2作为波特率发生器时，读取或写入TH2/TL2，写入RCAPH2/RCAPL2会影响波特率的准确性，因此也会引起通信出错。





## 寄存器

Table 8.24 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	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
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	TF2	<b>定时器2溢出标志位</b> 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)
6	EXF2	<b>T2EX引脚外部事件输入 (下降沿) 被检测到的标志位</b> 0: 无外部事件输入 (必须由软件清0) 1: 检测到外部输入 (如果EXEN2 = 1, 由硬件设1)
5	RCLK	<b>EUART接收时钟控制位</b> 0: 定时器1产生接收波特率 1: 定时器2产生接收波特率
4	TCLK	<b>EUART发送时钟控制位</b> 0: 定时器1产生发送波特率 1: 定时器2产生发送波特率
3	EXEN2	<b>T2EX引脚上的外部事件输入 (下降沿) 用作重载/捕获触发器允许/禁止控制位</b> 0: 忽略T2EX引脚上的事件 1: 当定时器2不做为EUART时钟 (T2EX始终包括上拉电阻) 时, 检测到T2EX引脚上一个下降沿, 产生一个捕获或重载
2	TR2	<b>定时器2开始/停止控制位</b> 0: 停止定时器2 1: 开始定时器2
1	C/T2	<b>定时器2定时器/计数器方式选定位</b> 0: 定时器方式, T2引脚用作I/O端口 1: 计数器方式, 内部上拉电阻被打开
0	CP/RL2	<b>捕获/重载方式选定位</b> 0: 16位带重载功能的定时器/计数器 1: 16位带捕获功能的定时器/计数器



Table 8.25 定时器2方式控制寄存器

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	-	-	-	-	-	T2OE	DCEN
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
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: 系统时钟作为定时器2的时钟源
1	T2OE	定时器2输出允许位 0: 设置P1.7/T2作为时钟输入或I/O端口 1: 设置P1.7/T2作为时钟输出（波特率发生器方式）
0	DCEN	递减计数允许位 0: 禁止定时器2作为递增/递减计数器，定时器2仅作为递增计数器 1: 允许定时器2作为递增/递减计数器

Table 8.26 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，x = 0 - 7
	RCAP2H.x	
7-0	TL2.x	定时器2高位低位计数器，x = 0 - 7
	TH2.x	



## 8.9 中断

### 8.9.1 特性

- 10个中断源
- 4层中断优先级

### 8.9.2 概述

SH79F1624A有10个中断源：3个外部中断（外部中断0/1/2），3个定时器中断（定时器0/1/2），1个LPD中断，1个EUART中断，1个PWM中断，ADC中断和SCM中断。

### 8.9.3 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，实现单独允许或禁止。IEN0寄存器中还包含了一个全局允许位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。

Table 8.27 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	ES0	ET1	EX1	ET0	EX0
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	ES0	EUART中断允许位 0: 禁止EUART中断 1: 允许EUART中断
3	ET1	定时器1溢出中断允许位 0: 禁止定时器1溢出中断 1: 允许定时器1溢出中断
2	EX1	外部中断1允许位 0: 禁止外部中断1 1: 允许外部中断1
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	EX0	外部中断0允许位 0: 禁止外部中断0 1: 允许外部中断0



Table 8.28 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	ELPD	-	EPWM	ESCM	-	EX2		-
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	ELPD	<b>LPD中断允许位</b> 0: 禁止LPD中断 1: 允许LPD中断
5	EPWM	<b>PWM中断允许位</b> 0: 禁止PWM中断 1: 允许PWM中断
4	ESCM	<b>SCM中断允许位</b> 0: 禁止SCM中断 1: 允许SCM中断
2	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



### 8.9.4 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，在中断汇总表中列出各中断标志位。

外部中断源产生外部中断INTx (x = 0/1/2) 时，如果中断为边沿触发，CPU在响应中断后，各中断标志位 (TCON寄存器的IE0/1位，EXF0寄存器的IE2位) 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。

定时器0/1的计数器溢出时，TCON寄存器的TFx (x = 0, 1) 中断标志位置1，产生定时器0/1中断，CPU在响应中断后，标志被硬件自动清0。

T2CON寄存器的TF2或EXF2标志位置1时，产生**定时器2**中断，CPU在响应中断后，标志不能被硬件自动清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。

SCON寄存器的标志RI或TI置1时，产生EUART中断，CPU在响应中断后，标志不能被硬件自动清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。

ADCON寄存器的ADCIF标志位置1时，产生ADC中断。如果中断产生，ADDH/ADDL中的转换结果是有效的。如果ADC模块的连续比较功能打开，在每次转换中，如果转换结果小于比较值时，ADCIF标志位为0；如果转换结果大于等于比较值时，ADCIF标志位置1，ADCIF中断标志必须由软件清除。

SCM寄存器的SCMIF标志位置1时，产生SCM中断，标志必须由硬件清0。

PWMCON寄存器的PWMIF标志位置1时，产生PWM中断，标志必须由软件清0。

当LPDCON寄存器中的LPDF标志位置1时，产生LPD中断。标志由硬件置位或者清除，软件可以清除此标志，但不能置1。

**Table 8.29** 定时器x/计数器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	IE1	IT1	IE0	IT0
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, 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
3, 1	IEx (x = 0, 1)	外部中断x请求标志 0: 无中断挂起 1: 中断挂起
2, 0	ITx (x = 0, 1)	外部中断x触发方式 0: 低电平触发 1: 下降沿边触发



Table 8.30 外部中断标志寄存器0

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	-	-	-	-	IT2.1	IT2.0	-	IE2
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
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
0	IE2	外部中断2请求标志位 0: 无中断挂起 1: 中断挂起



### 8.9.5 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。

### 8.9.6 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 8.31 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	-	PADCL	PT2L	PSL	PT1L	PX1L	PT0L	PX0L
IPH0	-	PADCH	PT2H	PSH	PT1H	PX1H	PT0H	PX0H
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	PLPDL	-	PPWML	PSCML	-	PX2L		-
IPH1	PLPDH	-	PPWMH	PSCMH	-	PX2H		-
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
7-0	PxxxL/H	相应中断源xxx优先级选择



### 8.9.7 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：

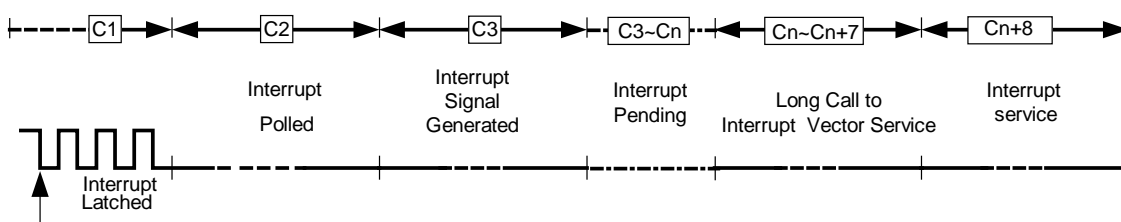
同级或更高级的优先级中断在运行中。

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RET<sub>I</sub>或者访问专用寄存器IEN<sub>0</sub>\1或是IPL\H的指令。换言之，在RET<sub>I</sub>或者读写IEN<sub>0</sub>\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：



中断响应时间

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RET<sub>I</sub>指令结束。RET<sub>I</sub>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。

### 8.9.8 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3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RET<sub>I</sub>指令，则完成正在执行的RET<sub>I</sub>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。





### 8.9.9 外部中断输入

SH79F1624A有3个外部中断输入。外部中断0-2各有一个独立的中断源。外部中断0/1可以通过设置TCON寄存器的IT1, IT0位来选择是电平触发或是边沿触发。当 $ITx=0$  ( $x=0, 1$ )时, 外部中断 $INTx$  ( $x=0, 1$ )引脚为低电平触发; 当 $ITx=1$  ( $x=0, 1$ ), 外部中断 $INTx$  ( $x=0, 1$ )为沿触发, 在这个方式中, 一个周期内 $INTx$  ( $x=0, 1$ )引脚上连续采样为高电平而下个周期为低电平, TCON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机器周期采样一次, 输入高或低电平应当保持至少1个机器周期以确保能够被正确采样到。

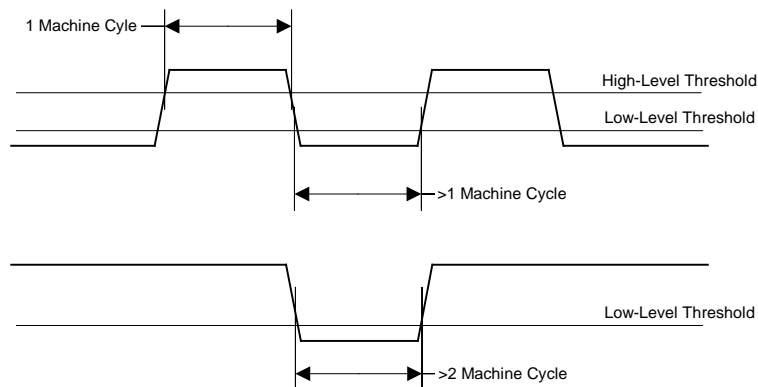
如果外部中断为下降沿触发, 外部中断源应当将中断脚至少保持1个机器周期高电平, 然后至少保持1个机器周期低电平。这样就确保了边沿能够被检测到以使 $IEx$ 置1。当调用中断服务程序后, CPU自动将 $IEx$ 清0。

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个系统时钟周期。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 $IEx$  ( $x=0, 1, 2$ ), 因为中断只与输入口电平有关。

外部中断2除了具有更多的中断触发方式外, 与外部中断0, 1操作类似。

当SH79F1624A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。



外部中断检测

### 8.9.10 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)	-
INT0	0003H	EX0	IE0	1	0
Timer0	000BH	ET0	TF0	2	1
INT1	0013H	EX1	IE1	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
INT2	004BH	EX2	IE2	8	9
SCM	005BH	ESCM	SCMIF	9	11
PWM	0063H	EPWM	PWMIF	10	12
LPD	0073H	ELPD	LPDF	11	14



### 9. 增强功能

#### 9.1 增强型通用异步收发器 (EUART)

##### 9.1.1 特性

- SH79F1624A带有1个EUART，兼容传统8051
- 波特率可选择为系统时钟分频或定时器1/2的溢出率
- 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

##### 9.1.2 EUART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。如果使用方式1或方式3应先初始化定时器1或定时器2。

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。在其他方式中由输入的起始位初始化接收（如果REN = 1）。通过发送起始位，外部发送器开始通信。

##### EUART方式列表

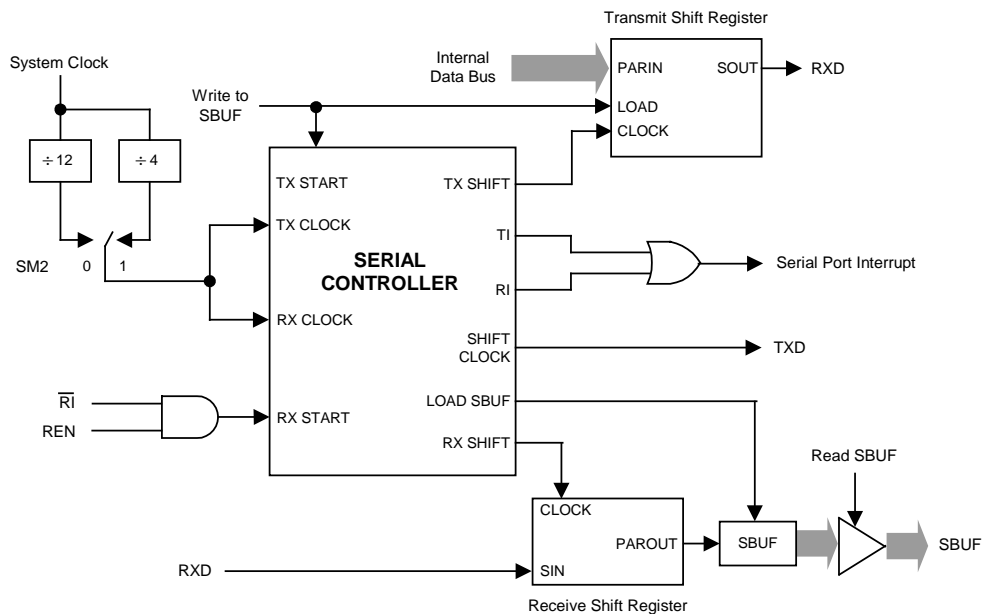
SM0	SM1	方式	类型	波特率	帧长度	起始位	停止位	第9位
0	0	0	同步	SYSCLK/ (4或12)	8位	无	无	无
0	1	1	异步	定时器1或2的溢出率/ (16或32)	10位	1	1	无
1	0	2	异步	SYSCLK/ (32或64)	11位	1	1	0, 1
1	1	3	异步	定时器1或2的溢出率/ (16或32)	11位	1	1	0, 1

##### 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引脚用作发送移位时钟。SH79F1624A提供TxD引脚上的移位时钟。因此这个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。

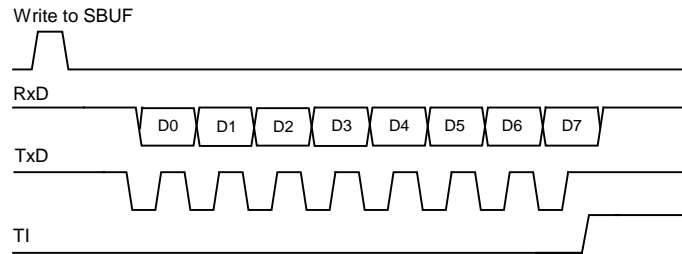
通过置SM2位 (SCON.5) 为0或1，波特率固定为系统时钟的1/12或1/4。当SM2位为0时，串行端口以系统时钟的1/12运行。当置1时，串行端口以系统时钟的1/4运行。与标准8051唯一不同的是，SH79F1624A在方式0中有可变波特率。

功能块框图如下图所示。数据通过RxD引脚进入和移出串行端口。移位时钟由TxD引脚输出，用来移位进出SH79F1624A的数据。



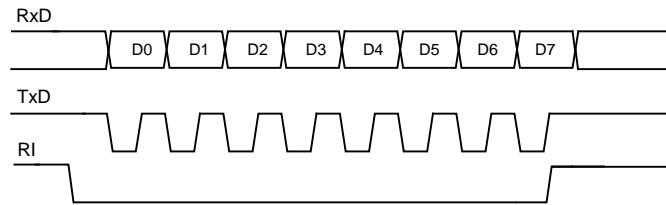


任何将SBUF作为目标寄存器的写操作都会启动发送。下一个系统时钟Tx控制块开始发送。数据转换发生在移位时钟的下降沿，移位寄存器的内容逐次从左往右移位，空位置0。当移位寄存器中的所有8位都发送后，Tx控制模块停止发送操作，然后在下一个系统时钟的上升沿将TI置1（SCON.1）。



Send Timing of Mode 0

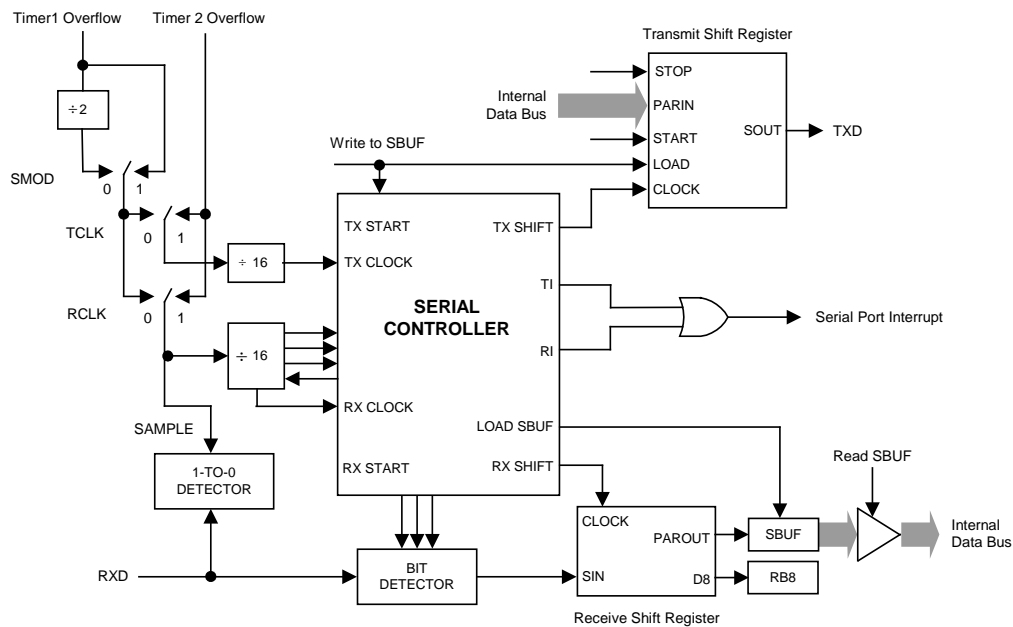
REN（SCON.4）置1和RI（SCON.0）清0初始化接收。下一个系统时钟启动接收，在移位时钟的上升沿锁存数据，接收转换寄存器的内容逐次向左移位。当所有8位都接收到接收移位寄存器中后，Rx控制块停止接收，然后在下一个系统时钟的上升沿上RI置1，直到被软件清0才允许接收。



Receive Timing of Mode 0

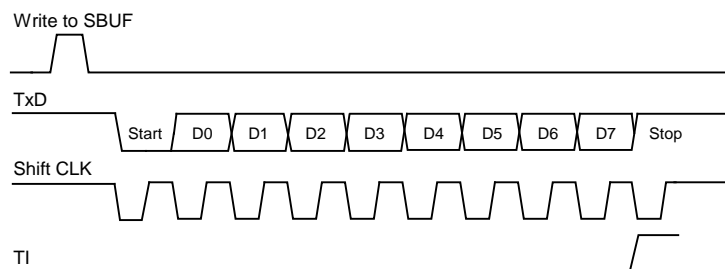
**方式1：8位EUART，可变波特率，异步全双工**

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中而停止位储存在RB8（SCON.2）中。方式1中的波特率是可变的，串行收发波特率可被设置为定时器1溢出率的1/16或1/32，或是定时器2溢出率的1/16。（详见波特率章节）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1。

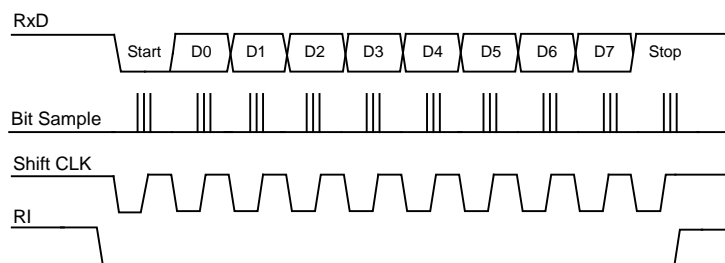


Send Timing of Mode 1

只有REN位置1时才允许接收。当RxD引脚检测到下降沿时串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1，但必须满足下列条件：

1. RI = 0
2. SM2 = 0或者接收的停止位 = 1

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

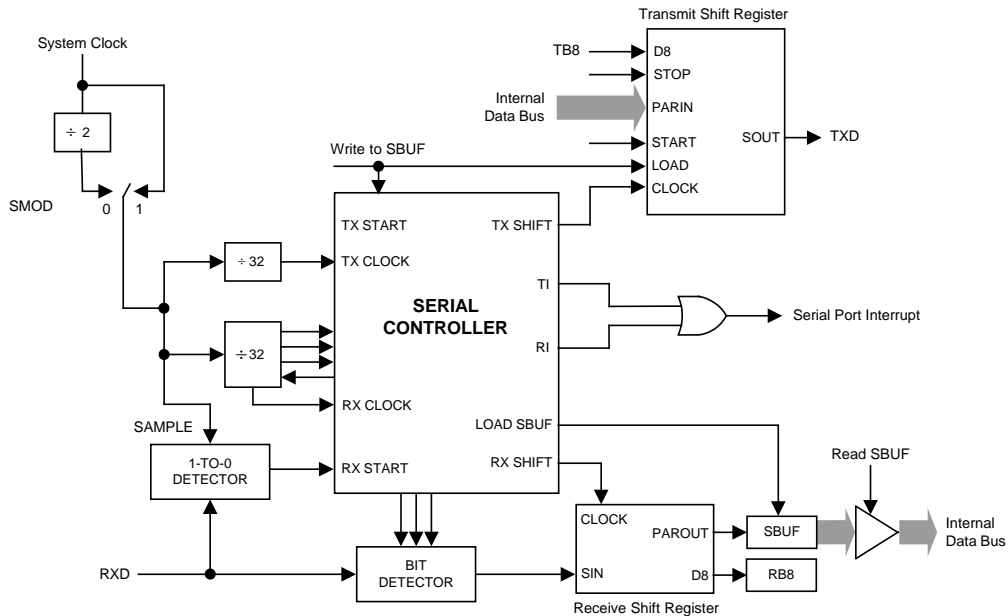


Receive Timing of Mode 1

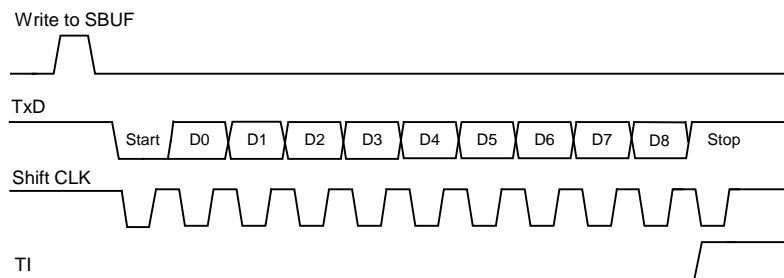


**方式2: 9位EUART, 固定波特率, 异步全双工**

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数据位进入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, 然后是第9数据位。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1。



**Send Timing of Mode 2**

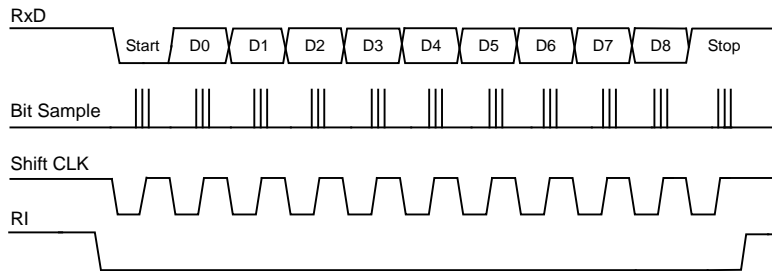


只有REN位置1时才允许接收。当RxD引脚检测到下降沿时串行口开始接收串行数据。为此，CPU对RxD不断采样，采样速率为波特率的16倍。当检测下降沿时，16分频计数器立即复位。这有助于16分频计数器与RxD引脚上的串行数据位同步。16分频计数器把每一位的时间分为16个状态，在第7、8、9状态时，位检测器对RxD端的电平进行采样。为抑制噪声，在这3个状态采样中至少有2次采样值一致数据才被接收。如果所接收的第一位不是0，说明这位不是一帧数据的起始位，该位被忽略，接收电路被复位，等待RxD引脚上另一个下降沿的到来。若起始位有效，则移入移位寄存器，并接着移入其它位到移位寄存器。9个数据位和1个停止位移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

- 1. RI = 0
- 2. SM2 = 0或者接收的第9位 = 1，且接收位符合EUART地址

如果这些条件被满足，那么第9位移入RB8，8位数据移入SBUF，RI被置1。否则接收的数据帧会丢失。

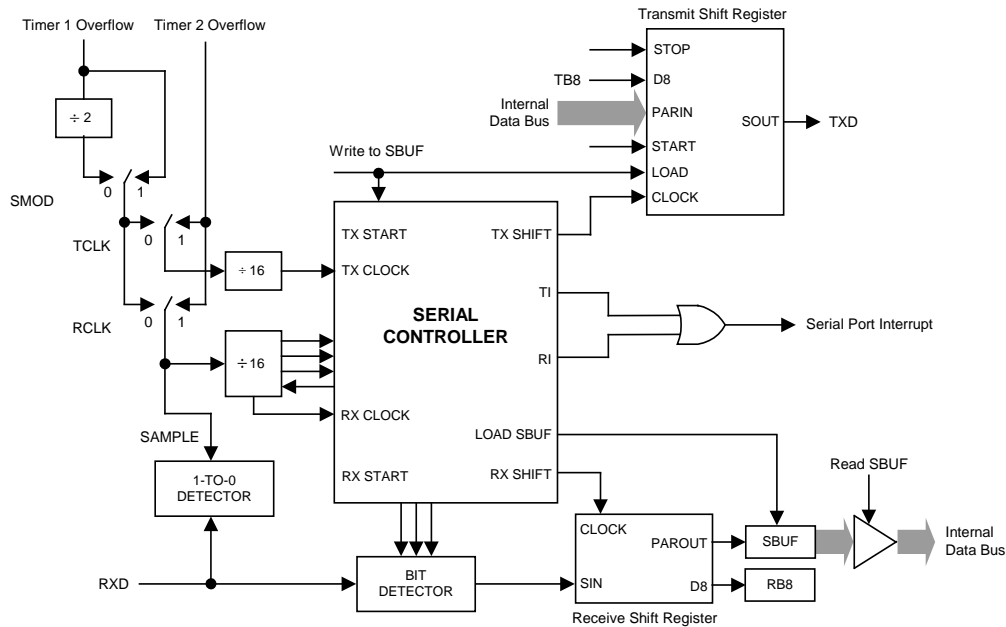
在停止位的当中，接收器回到寻找RxD引脚上的另一个下降沿。用户必须用软件清除RI，然后才能再次接收。



Receive Timing of Mode 2

**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。





### 9.1.3 波特率

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，波特率可选择来自定时器1或定时器2的溢出率。

分别置TCLK (T2CON.4) 和RCLK (T2CON.5) 位为1来选择定时器2作为TX和RX的波特时钟源（详见定时器章节）。无论TCLK还是RCLK为逻辑1，定时器2都为波特率发生器方式。如果TCLK和RCLK为逻辑0，定时器1作为Tx和Rx的波特时钟源。

方式1和方式3波特率公式如下所示，其中TH1是定时器1的8位自动重载寄存器，SMOD为EUART的波特率二倍频器（PCON.7），[RCAP2H, RCAP2L]是定时器2的16位重载寄存器。T1CLK是定时器1的时钟源，T2CLK是定时器2的时钟源。

$$\text{BaudRate} = \frac{2^{\text{SMOD}}}{32} \times \frac{f_{T1}}{256 - \text{TH1}}, \text{ 用定时器1作为波特率发生器, 定时器1工作在方式2}$$

$$\text{BaudRate} = \frac{1}{2 \times 16} \times \frac{f_{T2}}{65536 - [\text{RCAP2H}, \text{RCAP2L}]}, \text{ 用定时器2作为波特率发生器, 定时器2时钟源为系统时钟}$$

$$\text{BaudRate} = \frac{1}{16} \times \frac{f_{T2}}{65536 - [\text{RCAP2H}, \text{RCAP2L}]}, \text{ 用定时器2作为波特率发生器, 定时器2时钟源为时钟源为T2引脚输入时钟}$$

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）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。

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

### 9.1.4 多机通讯

#### 软件地址识别

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

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

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

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

#### 自动（硬件）地址识别

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

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

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

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





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

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

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

系统复位后，SADDR和SADEN两个寄存器初始化为0，这两个结果设定了给定地址和广播地址为XXXXXXXX（所有位都被忽略）。这有效地去除了多处机通讯的特性，禁止了自动寻址方式。这样的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识别地址的多机通讯。

### 9.1.5 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1后，只能通过软件清零，尽管后续接收的帧没有任何错误也不会自动清零。

**注意：**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1。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，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RXOVR位）置1。如果发生了接收超限，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1。

#### 暂停检测

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





9.1.6 寄存器

Table 9.1 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	波特率加倍器 若使用定时器1作为波特率发生器，在方式1和3中置1，EUART的波特率会加倍 如果在方式2中置1，EUART的波特率会加倍
6	SSTAT	SCON[7:5]功能选择 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



EUART相关寄存器

Table 9.2 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位必须被设置为1</b> 0: 无帧出错, 由软件清除 1: 发生帧出错, 由硬件置1
6	RXOV	<b>EUART接收完毕标志位, 当RXOV位被读时, SSTAT位必须被设置为1</b> 0: 无接收溢出, 由软件清除 1: 接收溢出, 由硬件置1
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, 停止位将置RI为1产生中断 在方式2和3下, 任何字节都会置RI为1产生中断 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RI为1产生中断 在方式2和3下, 只有寻址字节 (第9位 = 1) 能置RI为1产生中断
5	TXCOL	<b>EUART发送冲突标志位, 当TXCOL位被读时, SSTAT位必须被设置为1</b> 0: 无发送冲突, 由软件清除 1: 有发送冲突, 由硬件置1
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1或清0
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, RB8的停止位会收到信号 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0 1: 由硬件置1, 在方式0下的第8位最后, 或在其它方式下的停止位开始
0	RI	<b>EUART的接收中断标志位</b> 0: 由软件清0 1: 由硬件置1, 在方式0下的第8位最后, 或在其它方式下的停止位开始



Table 9.3 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>	SFR访问两个寄存器：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 9.4 EUART从机地址及地址掩码寄存器

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</b>	SADDR.7	SADDR.6	SADDR.5	SADDR.4	SADDR.3	SADDR.2	SADDR.1	SADDR.0
<b>SADEN</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>	SFR SADDR定义EUART的从机地址
7-0	<b>SADEN.7-0</b>	SFR SADEN是一个位屏蔽寄存器，决定检验SADDR的哪些位对应接收地址 0：在SADDR中的相应位被忽略 1：SADDR中的相应位被检验是否对应接收地址



## 9.2 模数转换器 (ADC)

### 9.2.1 特性

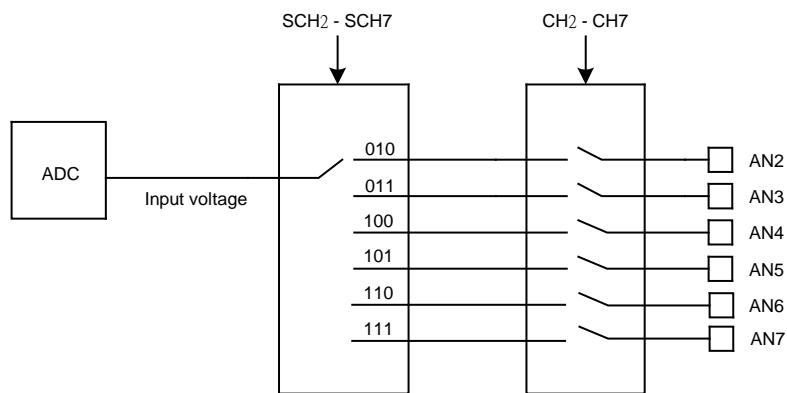
- 10位分辨率
- 内建基准电压
- 6模拟通道输入

SH79F1624A包含一个单端型、10位逐次逼近型模数转换器 (ADC)。ADC内建的基准电压 $V_{REF}$ 和 $V_{DD}$ 相连。6个ADC通道都可以独立输入模拟信号，但是每次转换只能使用一个通道。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数据寄存器与此同时，设置ADCON寄存器中的ADCIF位并且产生一个中断（如果允许ADC中断）。

ADC模块整合数字比较功能可以比较AD转换器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 $EC = 1$ ），并且ADC模块使能（ $ADON = 1$ 在ADCCON寄存器），只有当相应的模拟输入的数字值大于等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。这一点与数模转换工作方式不同。

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Power-Down模式下，ADC模块被禁止。

### 9.2.2 ADC框图



AD转换器模块图



9.2.3 寄存器

Table 9.5 ADC控制寄存器

93H	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	-	SCH2	SCH1	SCH0	GO/DONE
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	ADON	<b>ADC允许位</b> 0: 禁止ADC模块 1: 允许ADC模块
6	ADCIF	<b>ADC中断标志位</b> 0: 无ADC中断 1: 由硬件置1表示已完成AD转换, 或者模拟输入大于ADDH/ADDL (如果允许数字比较模块)
5	EC	<b>比较功能允许位</b> 0: 禁止数字比较功能 1: 允许数字比较功能
3-1	SCH[2:0]	<b>ADC通道选择位</b> 010: ADC通道AN2 011: ADC通道AN3 100: ADC通道AN4 101: ADC通道AN5 110: ADC通道AN6 111: ADC通道AN7
0	GO/DONE	<b>ADC状态标志位</b> 0: 当完成AD转换时, 由硬件自动清0。在转换期间清0这个位会中止AD转换。 如果允许数字比较功能, 该位不会由硬件清0只能由软件清0。 1: 设置开始AD转换或者启动数字比较功能。



Table 9.6 ADC定时控制寄存器

94H	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 $t_{AD} \geq 1\mu\text{s}$ ;
- (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) 在设置 $\text{TS}[3:0]$ 前, 请估算连接到ADC输入引脚的串联电阻;
- (5) 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 $10k\Omega$ ;
- (6) 总共转换时间 =  $12t_{AD} + \text{采样时间}$ 。

**举例说明**

系统时钟(SYSCLK)	TADC[2:0]	$t_{AD}$	TS[3:0]	采样时间	转换时间
32.768kHz	000	$30.5*2=61\mu\text{s}$	0000	$2*61=122\mu\text{s}$	$12*61+122=854\mu\text{s}$
	000	$30.5*2=61\mu\text{s}$	0111	$8*61=488\mu\text{s}$	$12*61+488=1220\mu\text{s}$
	000	$30.5*2=61\mu\text{s}$	1111	$15*61=915\mu\text{s}$	$12*61+915=1647\mu\text{s}$
	111	$30.5*32=976\mu\text{s}$	0000	$2*976=1952\mu\text{s}$	$12*976+1952=13664\mu\text{s}$
	111	$30.5*32=976\mu\text{s}$	0111	$8*976=7808\mu\text{s}$	$12*976+7808=19520\mu\text{s}$
	111	$30.5*32=976\mu\text{s}$	1111	$15*976=14640\mu\text{s}$	$12*976+14640=26352\mu\text{s}$
4MHz	000	$0.25*2=0.5\mu\text{s}$	-	-	( $t_{AD} < 1\mu\text{s}$ , 不推荐)
	001	$0.25*4=1\mu\text{s}$	0000	$2*1=2\mu\text{s}$	$12*1+2=14\mu\text{s}$
	001	$0.25*4=1\mu\text{s}$	0111	$8*1=8\mu\text{s}$	$12*1+8=20\mu\text{s}$
	001	$0.25*4=1\mu\text{s}$	1111	$15*1=15\mu\text{s}$	$12*1+15=27\mu\text{s}$
	111	$0.25*32=8\mu\text{s}$	0000	$2*8=16\mu\text{s}$	$12*8+16=112\mu\text{s}$
	111	$0.25*32=8\mu\text{s}$	0111	$8*8=64\mu\text{s}$	$12*8+64=160\mu\text{s}$
	111	$0.25*32=8\mu\text{s}$	1111	$15*8=120\mu\text{s}$	$12*8+120=216\mu\text{s}$
12MHz	000	$0.083*2=0.166\mu\text{s}$	-	-	( $t_{AD} < 1\mu\text{s}$ , 不推荐)
	100	$0.083*12=1\mu\text{s}$	0000	$2*1=2\mu\text{s}$	$12*1+2=14\mu\text{s}$
	100	$0.083*12=1\mu\text{s}$	0111	$8*1=8\mu\text{s}$	$12*1+8=20\mu\text{s}$
	100	$0.083*12=1\mu\text{s}$	1111	$15*1=15\mu\text{s}$	$12*1+15=27\mu\text{s}$
	111	$0.083*32=2.7\mu\text{s}$	0000	$2*2.7=5.4\mu\text{s}$	$12*2.7+5.4=37.8\mu\text{s}$
	111	$0.083*32=2.7\mu\text{s}$	0111	$8*2.7=21.6\mu\text{s}$	$12*2.7+21.6=54\mu\text{s}$
	111	$0.083*32=2.7\mu\text{s}$	1111	$15*2.7=40.5\mu\text{s}$	$12*2.7+40.5=72.9\mu\text{s}$



Table 9.7 ADC信道配置寄存器

95H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
ADCH	CH7	CH6	CH5	CH4	CH3	CH2	-	-
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
7-0	CH[7:2]	信道配置位 0: P1.2-P1.7作为I/O端口 1: P1.2-P1.7作为ADC输入口

Table 9.8 AD转换数据寄存器（比较值寄存器）

96H	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	-	-	-	-	-	-	A1	A0
读/写	-	-	-	-	-	-	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	-	-	-	-	-	-	0	0
97H	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	A9	A8	A7	A6	A5	A4	A3	A2
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
1-0 7-0	A9-A0	ADC数据寄存器 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；
- (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；
- (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。



### 9.3 脉冲宽度调制 (PWM)

#### 9.3.1 特性

- 8位精度PWM模块
- 提供每个PWM周期溢出中断
- 输出极性可选择

SH79F1624A内建一个8位PWM模块。PWM模块可以产生周期和占空比分别可以调整的脉宽调制波形。寄存器PWMCON用于控制PWM模块的时钟源，寄存器PWMP用于设置PWM模块的周期。寄存器PWMD用于设置PWM模块的占空比。

#### 9.3.2 寄存器

Table 9.9 PWM定时器控制寄存器

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
PWMCON	PWMEN	PWMS	PWMCK1	PWMCK0	-	-	PWMIF	PWMSS
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	PWMEN	<b>PWM模块控制位</b> 0: 禁止PWM 1: 允许PWM
6	PWMS	<b>PWM输出模式</b> 0: 高电平驱动, PWM占空比期间输出高电平, 占空比溢出后输出低电平 1: 低电平驱动, PWM占空比期间输出低电平, 占空比溢出后输出高电平
5-4	PWMCK[1:0]	<b>PWM时钟源选择</b> 00: 系统频率/2 01: 系统频率/4 10: 系统频率/8 11: 系统频率/16
1	PWMIF	<b>PWM中断标志位</b> 0: PWM周期计数器没有溢出 1: PWM周期计数器溢出, 由硬件置1
0	PWMSS	<b>PWM0输出控制位</b> 0: PWM输出禁止, 用作I/O功能 1: PWM输出允许

Table 9.10 PWM周期控制寄存器

D2H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWMP	PWMP.7	PWMP.6	PWMP.5	PWMP.4	PWMP.3	PWMP.2	PWMP.1	PWMP.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	PWMP[7:0]	<b>PWM输出周期 = PWMP * PWM时钟</b> 当PWMP = 00H时, 如果PWMS = 0, PWM引脚输出低 当PWMP = 00H时, 如果PWMS = 1, PWM引脚输出高





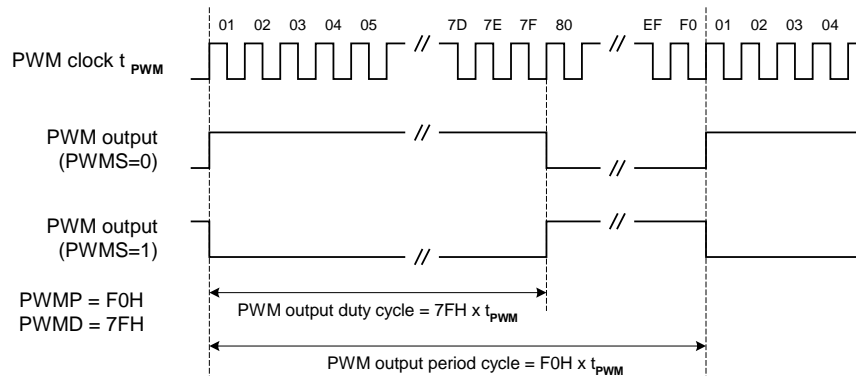
Table 9.11 PWM占空比控制寄存器

D3H	第7位	第6位	第5位	第4位	第3位	第2位	第1位	第0位
PWMD	PWMD.7	PWMD.6	PWMD.5	PWMD.4	PWMD.3	PWMD.2	PWMD.1	PWMD.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

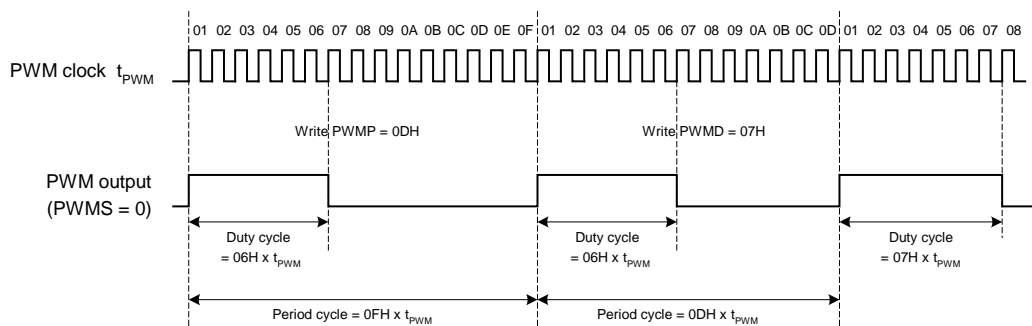
位编号	位符号	说明
7-0	PWMD[7:0]	<p><b>PWM占空比控制，控制PWM波形占空比的输出时间</b></p> <ol style="list-style-type: none"> <li>当PWMP ≤ PWMD时 如果PWMS = 0，则PWM引脚输出高电平 如果PWMS = 1，则PWM引脚输出低电平</li> <li>当PWMD = 00H时 如果PWMS = 0，则PWM引脚输出低电平 如果PWMS = 1，则PWM引脚输出高电平</li> </ol>

**注意:**

- (1) PWMEN位能控制PWM模块打开。
- (2) PWMSS位能选择P3.7端口是作为I/O端口还是PWM输出口。
- (3) 在IEN1寄存器中的EPWM位能允许/禁止PWM中断。
- (4) 如果PWMD置1，PWM模块打开，但PWMS = 0，PWM输出关闭，此时PWM模块可以用作一个8bit timer，此时如果中断控制寄存器IEN1的EPWM位置1，PWM中断照样发生。



**PWM 输出示例**



**PWM输出周期或占空比更改范例**



## 9.4 低电压复位 (LVR)

### 9.4.1 特性

- 通过代码选项选择, LVR 设定电压  $V_{LVR}$  可为 4.1V 或 3.7V
- LVR 去抖动时间  $T_{LVR}$  为 30-100 $\mu$ 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-100 $\mu$ s。

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

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

当  $V_{DD} > V_{LVR}$  或  $V_{DD} < V_{LVR}$ , 但  $t < T_{LVR}$  时不会产生系统复位。

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。



## 9.5 看门狗定时器（WDT），程式超范围溢出（OVL）复位及其它复位状态

### 9.5.1 特性

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

### 程序超范围溢出复位

SH79F1624A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

看门狗定时器是一个递减计数器，独立内建RC振荡器作为其时钟源，因此可以通过代码选项选择在掉电模式下仍持续运行。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：



9.5.2 寄存器

Table 9.12 复位控制寄存器

B1H	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	CLRF	WDT.2	WDT.1	WDT.0
读/写	读/写	-	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR)	0	-	1	0	0	0	0	0
复位值 (WDT)	1	-	u	u	u	0	0	0
复位值 (LVR)	u	-	u	1	u	0	0	0
复位值 (PIN)	u	-	u	u	1	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
3	CLRF	<b>Reset引脚复位标志位</b> 引脚复位后置1，由软件或上电复位清0 0: 没有发生引脚复位 1: 发生过引脚复位
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



## 9.6 电源管理

### 9.6.1 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，SH79F1624A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。

### 9.6.2 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SH79F1624A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除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（复位引脚上出现高电平，WDT复位，LVR复位）。CPU恢复时钟，SUSLO寄存器和在PCON寄存器中的IDL位被硬件清除，最后SH79F1624A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。

### 9.6.3 掉电模式（Power-Down）

掉电模式可以使SH79F1624A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。如果WDT使能，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SH79F1624A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，SH79F1624A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。

有两种方式可以退出掉电模式：

(1) 有效外部中断（如INT0，INT1和INT2）使SH79F1624A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，然后程序运行中断服务程序。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（复位引脚上出现高电平，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SH79F1624A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。

**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）。



## 9.6.4 寄存器

Table 9.13 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除 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除 1: 由软件置1激活空闲模式

Table 9.14 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

```



## 9.7 预热计数器

### 9.7.1 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

SH79F1624A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。

SH79F1624A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，SH79F1624A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，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	电源上电 预热计数时间	振荡器上电 预热计数	电源上电 预热计数时间	振荡器上电 预热计数	电源上电 预热计数时间	振荡器上电 预热计数
11ms	有	1000CKs	无	1000CKs	有	64CKs	有

#### 振荡器上电预热计数时间

OP_WMT选项 振荡器类型	00	01	10	11
陶振	$2^{13} \times T_{osc}$	$2^{11} \times T_{osc}$	$2^9 \times T_{osc}$	$2^7 \times T_{osc}$
晶振	$2^{17} \times T_{osc}$	$2^{15} \times T_{osc}$	$2^{13} \times T_{osc}$	$2^{11} \times T_{osc}$
32kHz晶振	$2^{13} \times T_{osc}$			
内部RC	$2^7 \times T_{osc}$			



## 9.8 低电压检测 (LPD)

### 9.8.1 特性

- 低电压检测并产生中断
- 可选的 LPD 检测电压

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。

### 9.8.2 寄存器

Table 8.15 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*	LPDV*	-	-	-	LPDS1	LPDS0
读/写	读/写	读*	读/写	-	-	-	读/写	读/写
复位值 (POR/WDT/LVR/PIN)	0	0	0	-	-	-	0	0

\*: LPDF软件只能清0, 不能置1。

\*: Program Note: 当配置LPD检测P1.3引脚时, P1.3作为模拟输入引脚, 此时, 不能作为普通IO使用; 当配置LPD检测电源电压时, 此时P1.3引脚作为IO引脚, P1.3不能作为模拟输入引脚, 此时P1.3也不可以作为ADC采集通道。

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
5	LPDV	<b>LPD检测电压源</b> 0: 检测电源电压 1: 检测VLPD (P1.3) 引脚电压
1-0	LPDS[1:0]	<b>LPD电压设置位</b> 00: 3.7V 01: 3.9V 10: 4.2V 11: 4.4V





## 9.9 代码选项

### OP\_WDT[7]:

- 0: 禁止看门狗复位 (默认)
- 1: 允许看门狗复位

### OP\_WDTPD[6]:

- 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

**注意:** 此代码选项仅当OP\_WDT[7] = 1时有效。

### OP\_WMT[4:3]: (不适用于32k晶体振荡器和内建RC)

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

### OP\_OSC[2:0]:

- 000: 内建RC振荡器 (16.6MHz) (默认)
- 010: 外部时钟源 (30kHz - 16.6MHz)
- 011: 32.768kHz晶体振荡器, 内建RC振荡器16.6M (可通过指令打开)
- 101: 晶体振荡器 (400kHz - 16.6MHz)
- 110: 陶瓷振荡器 (400kHz - 16.6MHz)
- Others: 内建RC振荡器 (16.6MHz)

### OP\_RST[5]:

- 0: 允许引脚复位 (默认)
- 1: 选择P4.0作为普通IO

### OP\_LVREN[7]:

- 0: 禁止低电压复位功能 (默认)
- 1: 允许低电压复位功能

### OP\_LVRLE[6:5]:

- 00: 低电压复位设定电压为4.1V (默认)
- 10: 低电压复位设定电压为3.7V
- 01: 低电压复位设定电压为2.8V

### OP\_SCM[3]:

- 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\_IO[0]:

- 0: 上电时IO结构为仅输入结构
- 1: 上电时IO结构为准双向结构 (默认)

### OP\_OSCDRV[4:3]:

- 00: 外部振荡器驱动能力为: Minimum
- 01: 外部振荡器驱动能力为: Middle (默认)
- 10: 外部振荡器驱动能力为: Maximum

**注意:** 此代码选项推荐选择默认值

代码选项OP\_OSC与OP\_OSCDRV组合功能如下:

序号	OP_OSC	OP_OSCDRV	振荡器类型
1	101 (晶体振荡器)	00 (Minimum)	晶体振荡器400KHz - 4MHz
2	101 (晶体振荡器)	01 (Middle)	晶体振荡器4MHz - 16MHz
3	110 (陶瓷振荡器)	00 (Minimum)	陶瓷振荡器 < 2MHz
4	110 (陶瓷振荡器)	01 (Middle)	陶瓷振荡器2MHz - 8MHz
5	110 (陶瓷振荡器)	10 (Maximum)	陶瓷振荡器8MHz - 16MHz



## 10. 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



### 11. 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  
 Flash存储器写/擦除操作.....0°C to +85°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。

直流电气特性 (V<sub>DD</sub> = 2.8 - 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>	2.8	5.0	5.5	V	32.768kHz ≤ f <sub>OSC</sub> ≤ 16.6MHz
工作电流	I <sub>OP</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); 关闭其它所有功能
待机电流 (空闲模式: IDLE)	I <sub>SB1</sub>	-	25	35	μA	f <sub>OSC</sub> = 32768Hz, 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); 关闭其它所有功能
	I <sub>SB2</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); 关闭其它所有功能
待机电流 (掉电模式: Power-down)	I <sub>SB3</sub>	-	-	13	μA	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(掉电模式); 关闭其它所有功能
WDT电流	I <sub>WDT</sub>	-	1	3	μA	V <sub>DD</sub> = 5.0V 所有输出引脚无负载, 看门狗打开
LPD电流	I <sub>LPD</sub>	-	3	5	μA	V <sub>DD</sub> = 5.0V
输入低电压	V <sub>IL</sub>	GND	-	0.2 X V <sub>DD</sub>	V	I/O端口(全部引脚都有schmitt触发器)
输入高电压	V <sub>IH</sub>	0.8 X V <sub>DD</sub>	-	V <sub>DD</sub>	V	I/O端口(全部引脚都有schmitt触发器)
输入漏电流	I <sub>IL</sub>	-1	-	1	μA	输入无上拉 V <sub>IN</sub> = V <sub>DD</sub> or GND (Input Only mode)
输出漏电流	I <sub>OL</sub>	-1	-	1	μA	开漏输出, V <sub>DD</sub> = 5.0V V <sub>OUT</sub> = V <sub>DD</sub> or GND (Open-Drain mode)
极弱上拉电阻	R <sub>PH1</sub>	-	300	-	kΩ	V <sub>DD</sub> = 5.0V, V <sub>IN</sub> = GND
弱上拉电阻	R <sub>PH2</sub>	-	10	-	kΩ	V <sub>DD</sub> = 5.0V, V <sub>IN</sub> = GND, I/O端口
输出高电压1	V <sub>OH1</sub>	V <sub>DD</sub> - 0.7	-	-	V	I/O端口(P1, P3, P4), I <sub>OH</sub> = -10mA, V <sub>DD</sub> = 5.0V (Push-Pull mode, 上拉增强customer option关闭)
输出高电压2	V <sub>OH2</sub>	V <sub>DD</sub> - 0.7	-	-	V	I/O端口(P1, P4), I <sub>OH</sub> = -15mA, V <sub>DD</sub> = 5.0V (Push-Pull mode, 上拉增强customer option打开)
输出低电压	V <sub>OL</sub>	-	-	GND + 0.6	V	I/O端口(P1, P3, P4), I <sub>OL</sub> = 25mA, V <sub>DD</sub> = 5.0V (Push-Pull mode, 下拉增强customer option关闭)
大驱动口灌电流能力	I <sub>OL</sub>	80	100	-	mA	I/O端口(P3), V <sub>DD</sub> = 5.0V, V <sub>OL</sub> = GND + 1.5V, (Push-Pull mode, 下拉增强customer option打开)

#### 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<sub>DD</sub>的最大电流值须小于100mA。
- (3) 流过GND的最大电流值须小于150mA。



模/数转换器电气特性 (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
供电电压	V <sub>AD</sub>	4.5	5.0	5.5	V	
精度	N <sub>R</sub>	-	10	-	bit	GND ≤ V <sub>AIN</sub> ≤ V <sub>REF</sub>
A/D输入电压*	V <sub>AIN</sub>	GND	-	-	V	
A/D输入电阻	R <sub>AIN</sub>	2		-	MΩ	V <sub>IN</sub> = 5.0V
模拟电压源推荐阻抗	Z <sub>AIN</sub>			10	kΩ	
A/D转换电流	I <sub>AD</sub>	-	1	3	mA	ADC模块工作, V <sub>DD</sub> = 5.0V
A/D输入电流	I <sub>ADIN</sub>			10	μA	V <sub>DD</sub> = 5.0V
微分非线性误差	D <sub>LE</sub>	-	-	±1	LSB	f <sub>OSC</sub> = 16.6MHz, V <sub>DD</sub> = 5.0V
积分非线性误差	I <sub>LE</sub>			±2	LSB	f <sub>OSC</sub> = 16.6MHz, V <sub>DD</sub> = 5.0V
满刻度误差	E <sub>F</sub>	-	±1	±3	LSB	f <sub>OSC</sub> = 16.6MHz, V <sub>DD</sub> = 5.0V
偏移量误差	E <sub>Z</sub>	-	±0.5	±2	LSB	f <sub>OSC</sub> = 16.6MHz, V <sub>DD</sub> = 5.0V
总绝对误差	E <sub>AD</sub>	-	-	±3	LSB	f <sub>OSC</sub> = 16.6MHz, V <sub>DD</sub> = 5.0V
总转换时间**	T <sub>CON</sub>	14	-	-	μs	10 bit精度, V <sub>DD</sub> = 5.0V

注意:

(1) “\*”A/D输入电阻就是直流条件下A/D自身的输入电阻。

(2) “\*\*”建议与AD连接的信号源内阻小于10KΩ。

交流电气特性 (V<sub>DD</sub> = 3.3 - 5.5V, GND = 0V, T<sub>A</sub> = 25°C, f<sub>OSC</sub> = 30KHz - 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
振荡器起振时间	T <sub>OSC1</sub>	-	1	2	s	32.768kHz振荡器
	T <sub>OSC2</sub>	-	-	2	ms	16.6MHz时
复位脉冲宽度	t <sub>RESET</sub>	10	-	-	μs	高电平有效
复位引脚下拉电阻	R <sub>RPH</sub>	-	30	-	kΩ	V <sub>DD</sub> = 5.0V, V <sub>IN</sub> = GND
频率稳定性 (RC)	F <sub>RC</sub>	-	-	±2	%	16.6MHz内部RC振荡器, 包含芯片间的漂移 (V <sub>DD</sub> = 2.8 - 5.5V, T <sub>A</sub> = -40°C~+85°C)

低电压复位电气特性 (V<sub>DD</sub> = 2.8 - 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
LVR设定电压1	V <sub>LVR1</sub>	3.95	4.1	4.25	V	LVR使能, V <sub>DD</sub> = 2.8V - 5.5V
LVR设定电压2	V <sub>LVR2</sub>	3.55	3.7	3.85	V	LVR使能, V <sub>DD</sub> = 2.8V - 5.5V
LVR设定电压3	V <sub>LVR2</sub>	2.7	2.8	2.9	V	LVR使能, V <sub>DD</sub> = 2.8V - 5.5V





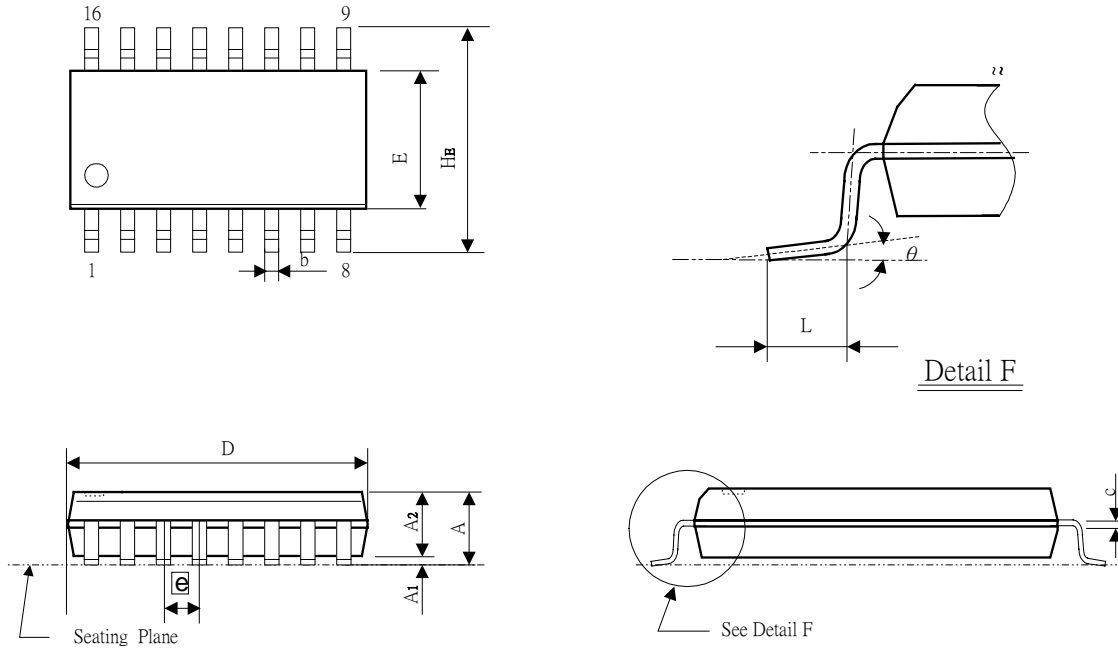
12. 订购信息

产品编号	封装
SH79F1624AL/016LU	SOP16



13. 封装信息

SOP 16L 150mil (W.B.) 外形尺寸



Symbol	Dimensions in inches		Dimensions in mm	
	Min	Max	Min	Max
A	0.053	0.071	1.35	1.8
A1	0.004	0.010	0.1	0.25
A2	0.049	0.061	1.25	1.55
b	0.013	0.020	0.33	0.51
c	0.008	0.014	0.2	0.35
D	0.386	0.402	9.8	10.2
E	0.150	0.157	3.8	4
e	0.050(BSC)		1.27(BSC)	
HE	0.228	0.248	5.8	6.3
L	0.016	0.050	0.4	1.27
□	0°	8°	0°	8°



14. 规格更改记录

版本	记录	日期
2.0	1. Page4页增加对RESET的描述“高电平复位” 2. Page69页修改复位信号的描述“复位引脚上出现高电平，WDT复位，LVR复位” 3. Page65页修改PWM注意事项中描述笔误	2015年11月
1.0	初始版本	2015年7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	4
6. 产品信息	5
7. SFR映像	6
8. 标准功能	12
8.1 CPU	12
8.1.1 CPU内核特殊功能寄存器	12
8.1.2 CPU增强内核特殊功能寄存器	13
8.1.3 寄存器	13
8.2 随机数据存储器 (RAM)	14
8.2.1 特性	14
8.2.2 寄存器	14
8.3 FLASH存储器	15
8.3.1 特性	15
8.3.2 ICP模式下的Flash操作	17
8.4 扇区自编程 (SSP) 功能	18
8.4.1 寄存器	18
8.4.2 Flash控制流程图	21
8.4.3 SSP编程注意事项	22
8.4.4 可读识别码	22
8.5 系统时钟和振荡器	23
8.5.1 特性	23
8.5.2 时钟定义	23
8.5.3 概述	23
8.5.4 寄存器	24
8.5.5 振荡器类型	25
8.5.6 谐振器负载电容选择	25
8.6 系统时钟监控 (SCM)	26
8.7 I/O端口	27
8.7.1 特性	27
8.7.2 寄存器	27
8.7.3 端口结构	28
8.7.4 端口共享	30
8.8 定时器	32
8.8.1 特性	32
8.8.2 定时器0和定时器1	32
8.8.3 定时器2	37
8.9 中断	43
8.9.1 特性	43
8.9.2 概述	43
8.9.3 中断允许	43
8.9.4 中断标志	45
8.9.5 中断向量	47
8.9.6 中断优先级	47
8.9.7 中断处理	48
8.9.8 中断响应时间	48
8.9.9 外部中断输入	49
8.9.10 中断汇总	49
9. 增强功能	50
9.1 增强型通用异步收发器 (EUART)	50
9.1.1 特性	50



9.1.2	EUART工作方式.....	50
9.1.3	波特率.....	55
9.1.4	多机通讯.....	55
9.1.5	帧出错检测.....	56
9.1.6	寄存器.....	57
9.2	模数转换器 (ADC) .....	60
9.2.1	特性.....	60
9.2.2	ADC框图.....	60
9.2.3	寄存器.....	61
9.3	脉冲宽度调制 (PWM) .....	64
9.3.1	特性.....	64
9.3.2	寄存器.....	64
9.4	低电压复位 (LVR) .....	66
9.4.1	特性.....	66
9.5	看门狗定时器 (WDT)，程式超范围溢出 (OVL) 复位及其它复位状态 .....	67
9.5.1	特性.....	67
9.5.2	寄存器.....	68
9.6	电源管理 .....	69
9.6.1	特性.....	69
9.6.2	空闲模式 (Idle) .....	69
9.6.3	掉电模式 (Power-Down) .....	69
9.6.4	寄存器.....	70
9.7	预热计数器 .....	71
9.7.1	特性.....	71
9.8	低电压检测 (LPD) .....	72
9.8.1	特性.....	72
9.8.2	寄存器.....	72
9.9	代码选项 .....	73
10.	指令集 .....	74
11.	电气特性 .....	79
12.	订购信息 .....	81
13.	封装信息 .....	82
14.	规格更改记录 .....	83