



1 特性

- 基于 8051 指令流水线结构的 8 位单片机
- Flash ROM: 32K 字节
- RAM: 内部 256 字节 外部 512 字节
- 工作电压：
 $f_{osc}=32768\text{Hz} - 8.192\text{MHz}$, $V_{DD}=3.0\text{V} - 5.5\text{V}$
 $f_{osc}=32768\text{Hz} - 4.096\text{MHz}$, $V_{DD}=2.4\text{V} - 3.6\text{V}$
(当 $V_{DD}=2.4-3.6\text{V}$ 时, V_{DD} 与 V_{P3} 管脚短接)
- 振荡器 (代码选项)
 - 晶体谐振器: 32768Hz
 - 晶体谐振器: 400kHz - 8MHz
 - 陶瓷谐振器: 400kHz - 8MHz
 - 外部时钟: 400kHz - 8MHz
 - 内部振荡器: 内建 PLL
(提供 1,2,4,8 x 1024kHz 内部时钟, 仅限选择 32768Hz 晶体谐振器时有效)
- 40 个 CMOS 双向 I/O 管脚
- I/O 内建上拉电阻
- 3 个 16 位定时器/计数器 T0、T1、T2
- 中断源：
 - Timer0, Timer1, Timer2,
 - INT0, INT1, INT2
 - EUART0, EUART1, SPI, TWI,
 - HSEC
 - ADC
 - PWM
- 一个 12 位 PWM 定时器
- LCD 驱动器：
 - 4X32 段 (1/4 占空比 1/3 偏置)
 - 5X31 段 (1/5 占空比 1/3 偏置)
 - 6X30 段 (1/6 占空比 1/3 偏置)
- 实时时钟 (5 ppm = 0.5 s/d)
- EUART0 和 EUART1 (内建 IR)
- SPI 接口 (主/从模式)
- 两线串行接口 (主模式)
- 6 通道 10 位模数转换器(ADC), 内建比较器
- 内建低电压检测功能 (LPD)
- 内建低电压复位功能(LVR) (代码选项)
 - LVR 电压 1: 3.1V
 - LVR 电压 2: 2.3V
- CPU 机器周期
 - 1 振荡周期
- 看门狗定时器(WDT)
- 内带振荡器预热计数器
- 支持省电运作模式：
 - 空闲(IDLE)模式
 - 掉电(Power-Down)模式
- 低功耗
- Flash 型
- 64-PIN LQFP 封装



2 概述

SH79F32 是一颗高速高效率 8051 兼容单片机。在同样振荡频率下，较之传统的 8051 芯片它具有运行更快速，性能更优越的特性。

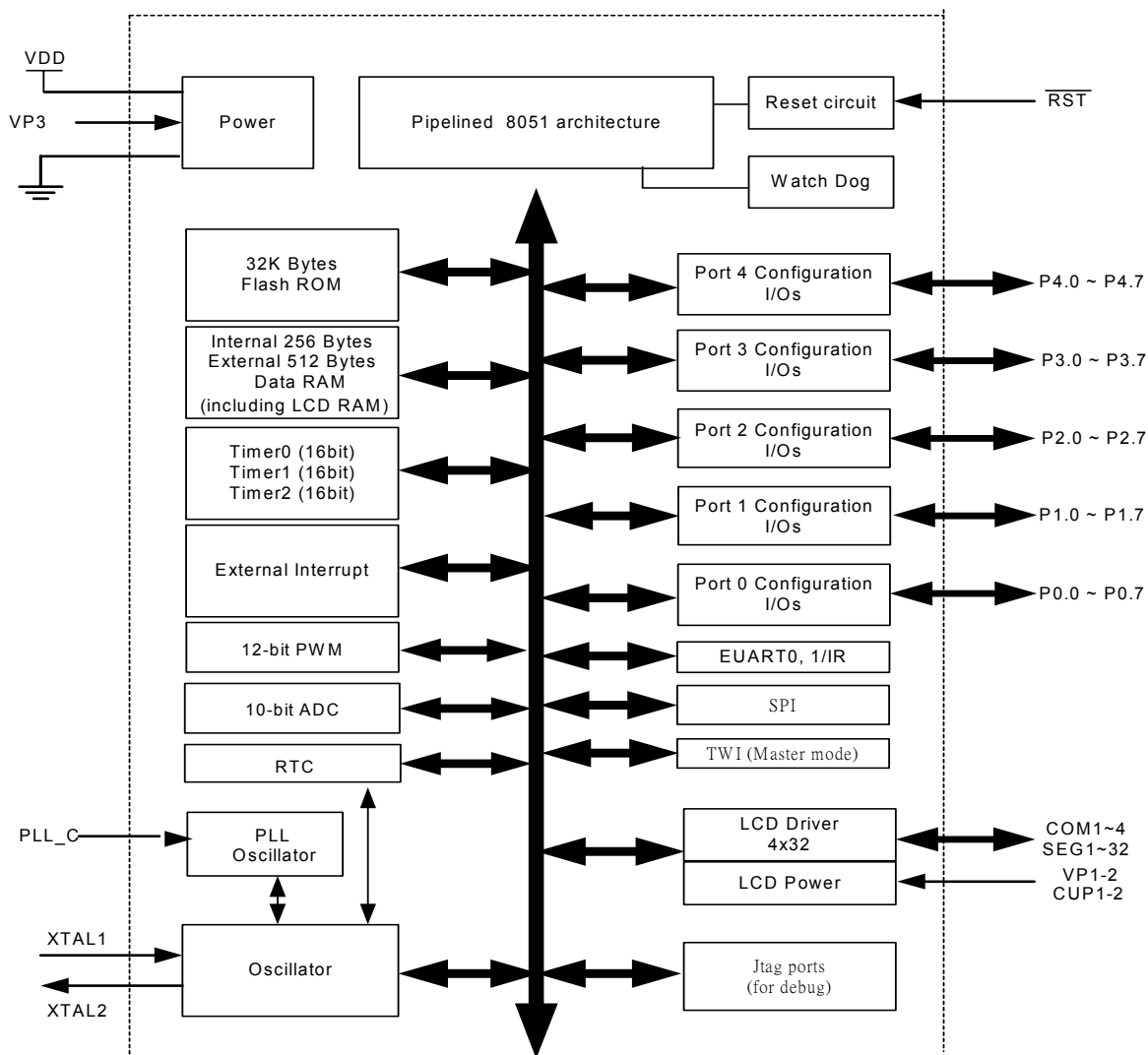
SH79F32 保留了标准 8051 芯片的大部分特性。这些特性包括内置 256 字节 RAM 和 2 个 16 位定时器/计数器，1 个 UART 和外部中断 INT0 和 INT1。此外，SH79F32 还集成了 512 字节外部 RAM，可兼容 8052 芯片的 16 位定时器/计数器（Timer2）和适合存储程序 and 数据的 32K 字节 flash。

SH79F32 不仅集成了如 EUART, IR, TWI 和 SPI 等标准通讯模块，还集成了 LCD 驱动器，具有内建比较功能的 ADC, PWM 定时器和实时时钟等模块。

为了达到高可靠性和低功耗，SH79F32 内建 PLL 时钟，SLP(Super Low Power) LCD 驱动器，看门狗定时器，低电压复位功能，低电压检测功能。此外 SH79F32 还提供了 2 种低功耗省电模式。



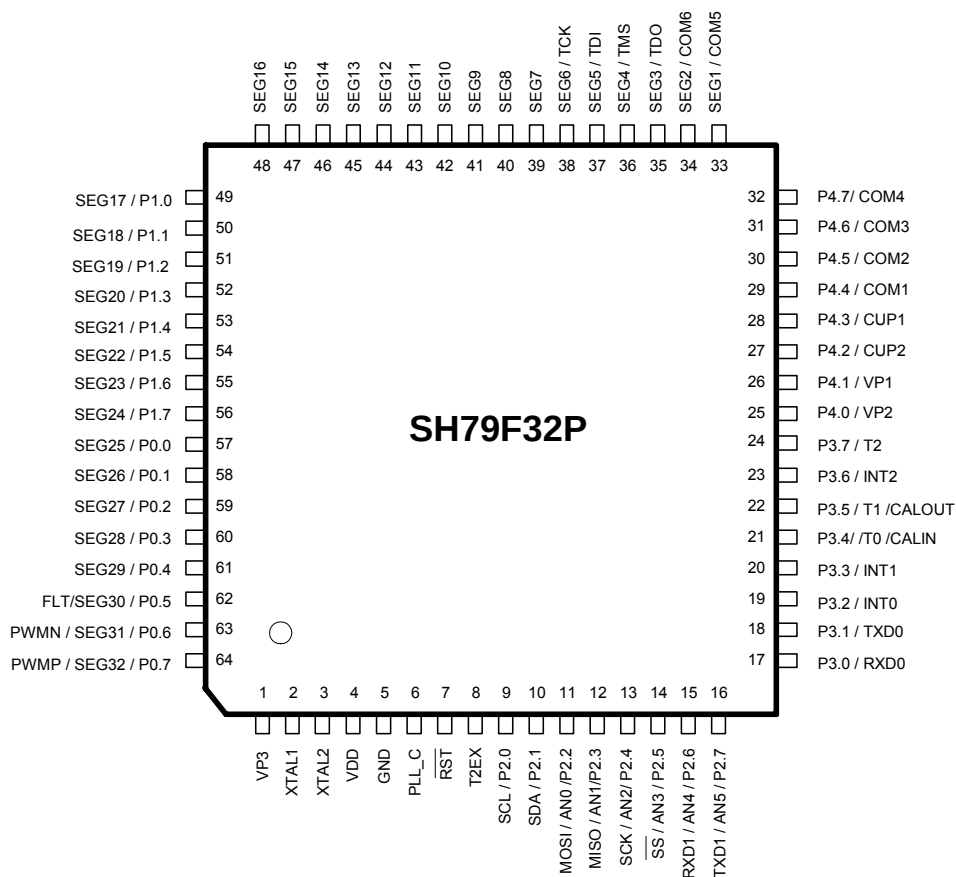
3 方框图





4 引脚配置

LQPF64



引脚配置图

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，例如 CALOUT / T1 / P3.5，CALOUT 优先级最高，T1 次之，P3.5 最低）。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	引脚编号	引脚命名	默认功能
1	VP3	-----	33	SEG1 / COM5	SEG1
2	XTAL1	-----	34	SEG2 / COM6	SEG2
3	XTAL2	-----	35	SEG3	-----
4	V _{DD}	-----	36	SEG4	-----
5	GND	-----	37	SEG5	-----
6	PLL_C	-----	38	SEG6	-----
7	RST	-----	39	SEG7	-----
8	T2EX	-----	40	SEG8	-----
9	SCL / P2.0	P2.0	41	SEG9	-----
10	SDA / P2.1	P2.1	42	SEG10	-----
11	MOSI / AN0 / P2.2	P2.2	43	SEG11	-----
12	MISO / AN1 / P2.3	P2.3	44	SEG12	-----
13	SCK / AN2 / P2.4	P2.4	45	SEG13	-----
14	SS / AN3 / P2.5	P2.5	46	SEG14	-----
15	RXD1 / AN4 / P2.6	P2.6	47	SEG15	-----
16	TXD1 / AN5 / P2.7	P2.7	48	SEG16	-----
17	RXD0 / P3.0	P3.0	49	SEG17 / P1.0	P1.0
18	TXD0 / P3.1	P3.1	50	SEG18 / P1.1	P1.1
19	INT0 / P3.2	P3.2	51	SEG19 / P1.2	P1.2
20	INT1 / P3.3	P3.3	52	SEG20 / P1.3	P1.3
21	CALIN / T0 / P3.4	P3.4	53	SEG21 / P1.4	P1.4
22	CALOUT / T1 / P3.5	P3.5	54	SEG22 / P1.5	P1.5
23	INT2 / P3.6	P3.6	55	SEG23 / P1.6	P1.6
24	T2 / P3.7	P3.7	56	SEG24 / P1.7	P1.7
25	VP2 / P4.0	VP2	57	SEG25 / P0.0	P0.0
26	VP1 / P4.1	VP1	58	SEG26 / P0.1	P0.1
27	CUP2 / P4.2	CUP2	59	SEG27 / P0.2	P0.2
28	CUP1 / P4.3	CUP1	60	SEG28 / P0.3	P0.3
29	COM1 / P4.4	COM0	61	SEG29 / P0.4	P0.4
30	COM2 / P4.5	COM1	62	FLT / SEG30 / P0.5	P0.5
31	COM3 / P4.6	COM2	63	PWMP / SEG31 / P0.6	P0.6
32	COM4 / P4.7	COM3	64	PWMN / SEG32 / P0.7	P0.7



5 引脚描述

引脚描述

引脚编号	类型	说明
PORT		
P4.0 – P4.7	I/O	8 位双向 I/O 端口
P3.0 – P3.7	I/O	8 位双向 I/O 端口
P2.0 – P2.7	I/O	8 位双向 I/O 端口 (P2.0 和 P2.1 是开漏输出端口)
P1.0 – P1.7	I/O	8 位双向 I/O 端口
P0.0 – P0.7	I/O	8 位双向 I/O 端口
Timer		
T0	I/O	Timer0 外部输入
T1	I/O	Timer1 外部输入
T2	I/O	Timer2 外部输入/波特率时钟输出
T2EX	I	Timer 2 重载/捕捉/方向控制
PWM 控制器		
PWMP	O	12 位 PWM 定时器输出引脚
PWMN	O	与 PWMP 有固定相位关系的 12 位 PWM 定时器输出引脚
FLT	I	PWM 故障输入引脚
EUART		
RXD0	I/O	EUART0 数据输入/输出引脚
TXD0	O	EUART0 数据输出引脚
RXD1	I/O	EUART1 或 IR 数据输入/输出引脚
TXD1	O	EUART1 或 IR 数据输出引脚
SPI		
MOSI	I/O	SPI 主输出从输入引脚
MISO	I/O	SPI 主输入从输出引脚
SCK	I/O	SPI 串行时钟引脚
SS	I	SPI 从属选择引脚
ADC		
AN0 – AN5	I	ADC 输入信道
TWI		
SDA	I/O	TWI 串行数据线 (开漏)
SCL	O	TWI 串行时钟线 (开漏)
LCD 控制器		
COM1 - COM4	O	LCD Com 信号输出脚
SEG3-SEG32	O	LCD Segment 信号输出脚
SEG1/COM5	O	LCD Segment1 信号输出或 Com5 信号输出脚
SEG2/COM6	O	LCD Segment2 信号输出或 Com6 信号输出脚
VP1-VP2	P	LCD 电源引脚
CUP1-2	P	LCD 偏置电容连接
RTC		
CALIN	I	RTC 校准输入脚
CALOUT	O	RTC 校准输出脚



引脚说明（续前表）

引脚编号	类型	说明
中断、复位、时钟、电源		
INT0 – INT2	I	外部中断 0,1,2
$\overline{\text{RST}}$	I	该引脚上保持 10us 以上的低电平，CPU 将复位。由于有内建 200k Ω 上拉电阻连接到 V _{DD} ，所以仅接一个外部电容即可实现上电复位。
XTAL1	I	振荡器输入
XTAL2	O	振荡器输出
PLL_C	P	内建 PLL 外部电容连接脚
GND	P	接地
VP3	P	内建 2.9V 电源输出（连接 47uF 电容）
VDD	P	电源
编程接口		
TDO (SEG 3)	O	调试接口：测试数据输出
TMS (SEG 4)	I	调试接口：测试模式选择
TDI (SEG 5)	I	调试接口：测试数据输入
TCK (SEG 6)	I	调试接口：测试时钟输入
注意： 当 SEG3-SEG6 作为调试接口时，SEG3-SEG6 的原有功能被限制。		



6 SFR 映像

SH79F32 内置 256 字节的直接寻址寄存器,包括通用数据存储器 and 特殊功能存储器(SFR), SH79F32 的 SFR 有以下几种 :

CPU 内核寄存器 : ACC, B, PSW, SP, DPL DPH

CPU 内核增强寄存器 : C, DPL1, DPH1, INSCON, XPAGE

电源和时钟控制寄存器 : PCON, SUSLO

LPD 寄存器 : LPDCON

Flash 寄存器 : IB_CLK0, IB_CLK1, IB_OFFSET, XPAGE, IB_DATA, IB_CON1, IB_CON2, IB_CON3, IB_CON4, IB_CON5,

数据页面控制寄存器 : XPAGE

硬件看门狗定时器寄存器 : RSTSTAT

系统时钟控制寄存器: CLKCON

中断寄存器 : IEN0, IEN1, IPH0, IPL0, IPH1, IPL1

I/O 端口寄存器 : P0, P1, P2, P3, P4, P0CR, P1CR, P2CR, P3CR, P4CR, P0PCR, P1PCR, P2PCR, P3PCR, P4PCR, P0SS, P1SS

Timer 寄存器 : TCON, TMOD, TL0, TH0, TL1, TH1, T2CON, T2MOD, EXF0, TL2, TH2, RCAP2L, RCAP2H

EUART0 寄存器 : SCON, SBUF, SADEN, SADDR, PCON

EUART1 寄存器 : SCON1, SBUF1, SADEN1, SADDR1

IR 寄存器 : IRCON, IRF

TWI 寄存器 : TWICON, TWIDAT

SPI 寄存器 : SPCON, SPSTA, SPDAT

ADC 寄存器 : ADCON, ADT, ADCH, ADDL, ADDH

LCD 寄存器 : LCDCON, PUMPCON, LCDCLK, PLCON, P1SS, P0SS,

PLL 寄存器 : CLKCON

RTC 寄存器 : RTCCON, RTCDAT, SEC, MIN, HR, DAY, MTH, YR, DOW

PWM 寄存器 : PWMEN, PWMDT, PWMC, PWMPL, PWMPH, PWMDL, PWMDL



SH79F32

C51 核 SFRs

符号	地址	名称	POR/WDT /LVR 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
C	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 1 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 1 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 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

数据存储器 SFR

符号	地址	名称	POR/WDT /LVR 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

电源时钟控制 SFRs

符号	地址	名称	POR/WDT /LVR 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	000-0000	SMOD	SSTAT	SSTAT1	-	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

LPD 控制 SFR

符号	地址	名称	POR/WDT/LVR 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 控制	00--0000	LPDEN	LPDF	-	-	LPDS3	LPDS2	LPDS1	LPDS0

Flash 控制 SFRs

符号	地址	命名	POR/WDT/L VR 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_CLK0	F9h	flash 编程时 钟寄存器 0	00000000	WTST.7	WTST.6	WTST.5	WTST.4	WTST.3	WTST.2	WTST.1	WTST.0
IB_CLK1	FAh	Flash 编程 时钟寄存器 1	00000000	ETST.7	ETST.6	ETST.5	ETST.4	ETST.3	ETST.2	ETST.1	ETST.0
IB_OFFSET	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_ ON5.3	IB_ CON5.2	IB_ CON5.1	IB_ CON5.0



SH79F32

WDT SFR

符号	地址	命名	POR/WDT/LVR 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	*-***000	WDOF	-	PORF	LVRF	CLRF	WDT.2	WDT.1	WDT.0

*注: RSTSTAT 初始值根据不同类型的复位而不同。

时钟控制 SFR

符号	地址	命名	POR/WDT/LVR 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	100-0000	32K_SPDUP	CLKS1	CLKS0	-	PLLON	FS2	FS1	FS0

中断 SFRs

符号	地址	命名	POR/WDT/LVR 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	ES	ET1	EX1	ET0	EX0
IEN1	A9h	中断允许控制 1	-00-0000	-	ETWI	EPWM	-	ERTC	ES1	EX2	ESPI
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	-00-0000	-	PTWIL	PPWML	-	PRTCL	PS1L	PX2L	PSPIL
IPH1	B5h	中断优先权控制 高位 1	-00-0000	-	PTWIH	PPWMH	-	PRTCH	PS1H	PX2H	PSPIH

端口 SFRs

符号	地址	命名	POR/WDT/LVR 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	8 位端口 0	00000000	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
P1	90h	8 位端口 1	00000000	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
P2	A0h	8 位端口 2	00000000	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0
P3	B0h	8 位端口 3	00000000	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
P4	C0h	8 位端口 4	00000000	P4.7	P4.6	P4.5	P4.4	P4.3	P4.2	P4.1	P4.0
P0CR	E1h	端口 0 输入/输出 方向控制	00000000	P0CR.7	P0CR.6	P0CR.5	P0CR.4	P0CR.3	P0CR.2	P0CR.1	P0CR.0
P1CR	E2h	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	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	端口 3 输入/输出 方向控制	00000000	P3CR.7	P3CR.6	P3CR.5	P3CR.4	P3CR.3	P3CR.2	P3CR.1	P3CR.0
P4CR	E5h	端口 4 输入/输出 方向控制	00000000	P4CR.7	P4CR.6	P4CR.5	P4CR.4	P4CR.3	P4CR.2	P4CR.1	P4CR.0
P0PCR	E9h	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	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	端口 2 内部上拉 允许	000000--	P2PCR.7	P2PCR.6	P2PCR.5	P2PCR.4	P2PCR.3	P2PCR.2	-	-
P3PCR	ECh	端口 3 内部上拉 允许	00000000	P3PCR.7	P3PCR.6	P3PCR.5	P3PCR.4	P3PCR.3	P3PCR.2	P3PCR.1	P3PCR.0
P4PCR	EDh	端口 4 内部上拉 允许	00000000	P4PCR.7	P4PCR.6	P4PCR.5	P4PCR.4	P4PCR.3	P4PCR.2	P4PCR.1	P4PCR.0



SH79F32

定时器 SFRs

符号	地址	命名	POR/WDT/LVR 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/T 1	M11	M10	GATE0	C/T 0	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/计数器 2 控制	00000000	TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T 2	CP/RL 2
T2MOD	C9h	定时器/计数器 2 模式	-----00	-	-	-	-	-	-	T2OE	DCEN
RCAP2L	CAh	定时器/计数器 2 重载/截获低位字 节	00000000	RCAP 2L.7	RCAP 2L.6	RCAP 2L.5	RCAP 2L.4	RCAP 2L.3	RCAP 2L.2	RCAP 2L.1	RCAP 2L.0
RCAP2H	CBh	定时器/计数器 2 重载/截获高位字 节	00000000	RCAP 2H.7	RCAP 2H.6	RCAP 2H.5	RCAP 2H.4	RCAP 2H.3	RCAP 2H.2	RCAP 2H.1	RCAP 2H.0
TL2	CCh	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	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

EUART0 SFRs

符号	地址	命名	POR/WDT/LVR 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	00000000	SMOD	SSTAT	SSTAT1	-	GF1	GF0	PD	IDL

EUART1 SFRs

符号	地址	命名	POR/WDT/LVR 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
SCON1	D8h	串行1 控制	00000000	SM10 /FE1	SM11 /RXOV1	SM12 /TXCOL1	REN1	TB81	RB81	TI1	RI1
SBUF1	D9h	串行1数 据缓冲器	00000000	SBUF1.7	SBUF1.6	SBUF1.5	SBUF1.4	SBUF1.3	SBUF1.2	SBUF1.1	SBUF1.0
SADDR1	DAh	从属 地址1	00000000	SADDR1.7	SADDR1.6	SADDR1.5	SADDR1.4	SADDR1.3	SADDR1.2	SADDR1.1	SADDR1.0
SADEN1	DBh	从属地址 1掩码	00000000	SADEN1.7	SADEN1.6	SADEN1.5	SADEN1.4	SADEN1.3	SADEN1.2	SADEN1.1	SADEN1.0
PCON	87h	电源和串 行控制	00000000	SMOD	SSTAT	SSTAT1	-	GF1	GF0	PD	IDL

IR SFR

符号	Add	命名	POR/WDT/LVR 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
IRCON	A1h	IR 控制	00000000	IRON	IRF6	IRF5	IRF4	IRF3	IRF2	IRF1	IRF0



SH79F32

SPI SFRs

符号	地址	命名	POR/WDT/LVR 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
SPCON	A2h	SPI 控制寄存器	00000000	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0
SPSTA	F8h	SPI 状态寄存器	00000---	SPEN	SPIF	MODF	WCOL	RXOV	-	-	-
SPDAT	A3h	SPI 数据寄存器	00000000	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0

I2C SFRs

符号	地址	命名	POR/WDT/LVR 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
I2CON	A6h	I2C 控制寄存器	00000000	I2IEN	I2IIF	STA	STO	RXDOV	ACK	BR1	BR0
I2DAT	A7h	I2C 数据寄存器	00000000	I2ID7	I2ID6	I2ID5	I2ID4	I2ID3	I2ID2	I2ID1	I2ID0

ADC 和比较器 SFRs

符号	地址	命名	POR/WDT/LVR 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	00--0000	TADC1	TADC0	-	-	TS3	TS2	TS1	TS0
ADCH	95h	ADC 通道选择	0000000--	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	000000000	A9	A8	A7	A6	A5	A4	A3	A2

LCD SFRs

符号	地址	命名	POR/WDT/LVR 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
LCDCON	ABh	LCD 控制	000-0000	LCDON	-	PUMPF	PUMPON	P4L1	P4L0	DUTY1	DUTY0
LCDCLK	ACH	LCD 时钟	00000000	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
P1SS	ADh	P1 功能选择	00000000	P1S7	P1S6	P1S5	P1S4	P1S3	P1S2	P1S1	P1S0
P0SS	Aeh	P0 功能选择	00000000	P0S7	P0S6	P0S5	P0S4	P0S3	P0S2	P0S1	P0S0
PLCON	Afh	SLP LCD 控制	---00000	-	-	-	PLD4	PLD3	PLD2	PLD1	PLD0

RTC SFRs

符号	地址	命名	POR/WDT/LVR 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
RTCCON	BEh	RTC 控制	00u00000	RTCEN	HSECIF	COMEN	CAL	AUTOCAL	OUTF1	OUTF0	EOVL
RTCDAT	BFh	RTC 校准数据	uuuuuuuuu	E7	E6	E5	E4	E3	E2	E1	E0
SEC	C1h	秒和半秒	uuuuuuuuu	HSEC	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
MIN	C2h	分钟	-uuuuuuuu	-	MIN6	MIN5	MIN4	MIN3	MIN2	MIN1	MIN0
HR	C3h	小时	--uuuuuuu	-	-	HR5	HR4	HR3	HR2	HR1	HR0
DAY	C4h	日	--uuuuuuu	-	-	DAY5	DAY4	DAY3	DAY2	DAY1	DAY0
MTH	C5h	月	---uuuuuu	-	-	-	MTH4	MTH3	MTH2	MTH1	MTH0
YR	C6h	年	-----uu	-	-	-	-	-	-	YR1	YR0
DOW	C7h	周	-----uuu	-	-	-	-	-	DOW2	DOW1	DOW0



SH79F32

PWM SFRs

符号	地址	命名	POR/WDT/LVR 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
PWMEN	CFh	PWM 定时器 允许	-0--0-00	-	EFLT	-	-	EPWMN	-	-	EPWM
PWMLO	E7h	PWM 锁定	00000000	PWMLO.7	PWMLO.6	PWMLO.5	PWMLO.4	PWMLO.3	PWMLO.2	PWMLO.1	PWMLO.0
PWMC	D2h	12 位 PWM 控制	00-0000	PWMIE	PWMIF	-	FLTS	FLTC	PWMS	TnCLK1	TnCLK0
PWMPL	D3h	12 位 PWM 周期控制低位	00000000	PP.7	PP.6	PP.5	PP.4	PP.3	PP.2	PP.1	PP.0
PWMPH	D4h	12 位 PWM 周期控制高位	----0000	-	-	-	-	PP.11	PP.10	PP.9	PP.8
PWMDL	D5h	12 位 PWM 占空比控制低位	00000000	PD.7	PD.6	PD.5	PD.4	PD.3	PD.2	PD.1	PD.0
PWMDH	D6h	12 位 PWM 占空比控制高位	----0000	-	-	-	-	PD.11	PD.10	PD.9	PD.8
PWMDT	D1h	PWM 死区时间控制	00000000	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0

外部中断 SFRs

符号	地址	命名	POR/WDT/LVR 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
EXF0	E8h	外部中断 2 控制	-----00	-	-	-	-	-	-	IE2	IT2
TCON	88h	定时器/计数器 0 和 1 控制 外部中断 0,1 控制	00000000	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

注意：-：保留位。

u：复位不影响当前值



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	SPSTA	IB_CLK0	IB_CLK1	IB_OFFSET	IB_DATA				FFh
F0h	B	C	IB_CON1	IB_CON2	IB_CON3	IB_CON4	IB_CON5	XPAGE	F7h
E8h	EXF0	P0PCR	P1PCR	P2PCR	P3PCR	P4PCR			EFh
E0h	ACC	P0CR	P1CR	P2CR	P3CR	P4CR		PWMLO	E7h
D8h	SCON1	SBUF1	SADDR1	SADEN1					DFh
D0h	PSW	PWMDT	PWMC	PWMPL	PWMPH	PWMDL	PWMDH		D7h
C8h	T2CON	T2MOD	RCAP2L	RCAP2H	TL2	TH2		PWMEN	CFh
C0h	P4	SEC	MIN	HR	DAY	MTH	YR	DOW	C7h
B8h	IPL0	IPL1					RTCCON	RTCDAT	BFh
B0h	P3	RSTSTAT	CLKCON	LPDCON	IPH0	IPH1			B7h
A8h	IEN0	IEN1		LCDCON	LCDCLK	P1SS	P0SS	PLCON	AFh
A0h	P2	IRCON	SPCON	SPDAT			TWICON	TWIDAT	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	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	

注意： 未使用的 SFR 地址禁止读写。



7 标准功能

7.1 指令扩展

SH79F32 扩展了'MUL'和'DIV'的指令。使用一个新寄存器-C 寄存器保存运算数据的高 8 位，以实现 16 位运算。

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。设置 INSCON 寄存器的相应位，允许 16 位模式操作。

	操作		结果		
			A	B	C
MUL	INSCON.2 = 0; 8 位模式	$(A) \times (B)$	低位字节	高位字节	---
	INSCON.2 = 1; 16 位模式	$(C A) \times (B)$	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	$(C A) / (B)$	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。使用 INSCON 寄存器中的 DPS 位来选择有效指针。用户能通过切换 DPS 位来切换数据指针。所有 DPTR 相关的指令将会选择最近一次选择的数据指针。

寄存器

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/WDI/LVR)	-	-	-	-	0	0	-	0

位编号	位符号	说明
3	DIV	16 位/ 8 位除选择器 0: 8 位除 1: 16 位除
2	MUL	16 位/ 8 位乘选择器 0: 8 位乘 1: 16 位乘
0	DPS	数据指针选择器 0: 数据指针 1: 数据指针 1



7.2 RAM

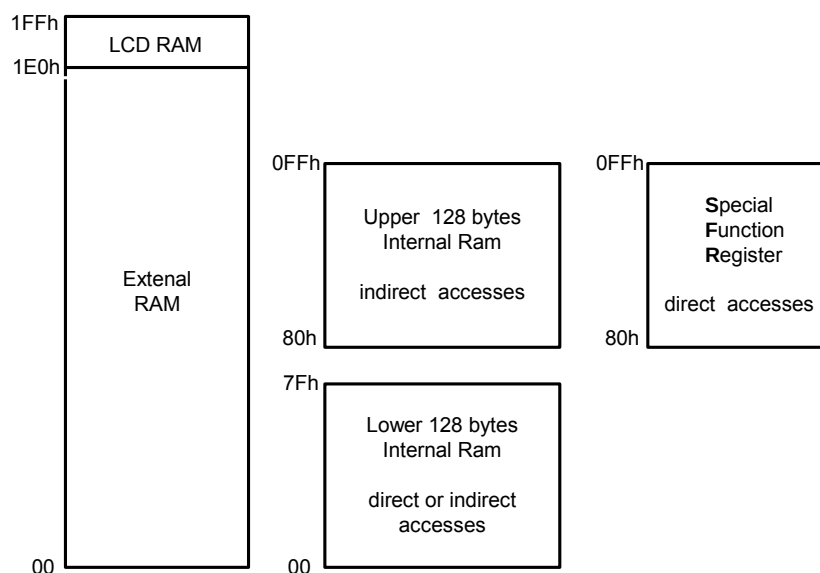
SH79F32 为随机数据存储提供了内部 RAM 和外部 RAM。内部 RAM 分为以下四个的部分:

- 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。

注意: 未使用的 SFR 地址禁止读写

SH79F32 在外部数据空间额外提供了 512 字节 RAM,支持高级语言,并配置 LCD RAM(1E0h – 1FFh)。



内部和外部 RAM 配置

SH79F32 支持传统的访问外部 RAM 方法。可以使用 MOVXA, @Ri 或 MOVX @Ri, A; 来访问外部低位 256 字节 RAM ; 使用 MOVX A, @DPTR 或 MOVX @DPTR, A 来访问外部 512 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 部分)。

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.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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	XPAGE7-0	RAM 页选择器



7.3 Flash 程序存储器

SH79F32 为存储程序代码内置 32K 可编程 flash, 并且提供自编程功能。

注意：最后 64 字节(\$7FC0~\$7FFF)将被保留，不能作为可编程存储器。

在 ICP (在线编程) 模式中, 程序能操作所有 flash, 例如擦除或写入。flash 的读取或写入操作以字节为单位, 但擦除只能以扇区(2K)为单位, 或者整体擦除。

在 ICP 模式中, 扇区擦除操作能擦除除了扇区 15 以外任何区块。在自编程模式(SSP)模式中, 除了扇区 15 不能擦除以外, 包含擦除程序代码的扇区也不能擦除。

在 ICP 模式下, 还可以整体擦除, 这个操作会擦除整个 flash 存储器(包括扇区 15)。

7.3.1 特性

- 可编程存储器包括 16 x 2KB 区块, 总共 32KB。
- 在工作电压范围内都能进行编程和擦除操作。
- ICP 操作支持写入, 读取和擦除操作。
- 快速整体/扇区擦除和编程
- 最小编程/擦除次数: 10000 次
- 数据保存最小年限: 10 年
- 低功耗

7.3.2 ICP 模式下的 Flash 操作

ICP 模式即线上编程模式, 即可以在 CPU 焊在用户板上以后编程。ICP 模式下, 用户系统必须关机后编程器才能通过 ICP 编程接口刷新 flash 存储器。ICP 编程接口包括 6 条导线(VDD, GND, TDO, TDI, TCK, TMS)。

先用 4 个 JTAG 引脚 (TDO, TDI, TCK, TMS) 进入编程模式。只有将指定波形输入 3 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**。

ICP 模式支持以下操作：

代码保护控制模式编程

SH79F32 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/禁止在其他分区中通过 MOVN 指令进行读取操作，或通过 SSP 功能进行擦除/写入操作。

用户必须应用 Flash 编程器设置相应的保护位，以进入所需的保护模式

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编程代码，代码选项，代码保护位和自定义 ID 码的内容。（Flash 编程器为用户提供自定义 ID 码设置功能以区别他们的产品）。

整体擦除只能由 Flash 编程器操作。

扇区擦除

除了扇区 15 外，扇区擦除操作将会擦除所选扇区中内容。用户程式和 Flash 编程都能执行该操作。

若需用户程式执行该操作，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。

注意：最后一个扇区 15(\$7800~\$7FFFh)无法执行扇区擦除功能。

写/读代码

读/写代码操作可以将客户数据写入 Flash 编程存储器或从 Flash 存储器中读取数据。编程器或用户程式都能执行该操作。

若需用户程式执行该操作，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/写程式自身所在扇区。

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。

编程用时钟控制寄存器		
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
写/读	支持 (无安全位)	支持 (无安全位或自身扇区)



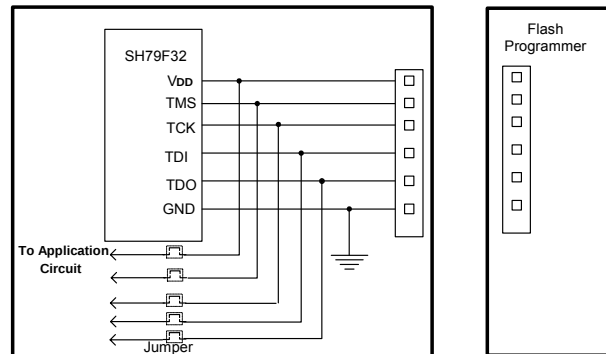
在 ICP 模式中,通过 6 线接口编程器能完成所有 flash 操作。因为编程信号非常灵敏,用户需要用 5 个跳线将编程引脚(VDD, TDO, TDI, TCK, TMS) 从应用电路中分离出。如下图所示。

建议步骤如下：

在程序接口联通前，必须断开跳线，从应用电路中分离编程引脚才能开始编程。

编程器连接编程接口后，开始编程

在编程完成后断开编程器连线并连接跳线。





7.3.3 SSP 功能

SH79F32 提供 SSP (扇区自编程) 功能, 如果所选的扇区未保护, 用户代码可以擦除扇区 15 外的所有扇区或任何扇区执行烧写操作。一旦该扇区被烧写, 在扇区擦除之前不能再烧写。

SH79F32 内建一个复杂控制流程以避免代码被误修改。如果不符合指定条件(IB_CON2~5), SSP 将被终止。

寄存器

编程用定时控制寄存器

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
IB_CLK0	IB_CLK0.7	IB_CLK0.6	IB_CLK0.5	IB_CLK0.4	IB_CLK0.3	IB_CLK0.2	IB_CLK0.1	IB_CLK0.0
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
IB_CLK1	IB_CLK1.7	IB_CLK1.6	IB_CLK1.5	IB_CLK1.4	IB_CLK1.3	IB_CLK1.2	IB_CLK1.1	IB_CLK1.0
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	IB_CLKx[7-0] X=0,1	Flash 烧写时钟选择 IB_CLK1:IB_CLK0 中的值按下列方法计算。 编程： $65536 - \frac{T_{prog}}{8 \times T_{sysck}}, \quad 20us \leq T_{prog} \leq 40us \quad F_{sys} \geq 1MHz$ $65536 - \frac{T_{prog}}{T_{sysck}}, \quad 20us \leq T_{prog} \leq 40us \quad F_{sys} < 1MHz$ 典型值 Tprog = 30us 扇区擦除： $65536 - \frac{T_{prog}}{8 \times T_{sysck}}, \quad 50ms \leq T_{prog} \leq 90ms \quad F_{sys} \geq 1MHz$ $65536 - \frac{T_{prog}}{T_{sysck}}, \quad 50ms \leq T_{prog} \leq 90ms \quad F_{sys} < 1MHz$ 典型值 Terase = 60ms

编程用偏移寄存器

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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-3	XPAGE[7-3]	被编程的存储单元扇区号 00000 代表扇区 0
2-0	XPAGE[2-0]	被编程的存储单元高 3 位地址



编程用 flash 存储器偏移寄存器

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)	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

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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	IB_DATA [7:0]	编程数据

SSP 型选择寄存器

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_CON 1.7	IB_CON 1.6	IB_CON 1.5	IB_CON 1.4	IB_CON 1.3	IB_CON 1.2	IB_CON 1.1	IB_CON 1.0
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	IB_CON1[7-0]	SSP 操作选择 0xE6 = 扇区擦除 0x6E = 烧写存储单元

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_CON 2.4	IB_CON 2.3	IB_CON 2.2	IB_CON 2.1	IB_CON 2.0
读/写	-	-	-	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	0	0	0	0	0

位编号	位符号	说明
4	IB_CON2. 4	系统时钟确认 0: $F_{sys} > 1 \text{ MHz}$ 1: $F_{sys} < 1 \text{ MHz}$
3-0	IB_CON2 [3:0]	必须为 05H, 否则 Flash 编程将会终止



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_CON 3.3	IB_CON 3.2	IB_CON 3.1	IB_CON 3.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	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

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_CON 4.3	IB_CON 4.2	IB_CON 4.1	IB_CON 4.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	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

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	Bit0
IB_CON5	-	-	-	-	IB_CON 5.3	IB_CON 5.2	IB_CON 5.1	IB_CON 5.0
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	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



Flash 控制流程图



**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：

A. 烧写：

- 1) 禁止中断；
- 2) 填写 IB_CLK1, IB_CLK0；
- 3) 按相应的待编程扇区号填写 XPAGE, IB_OFFSET；
- 4) 按编程需要，填写 IB_DATA；
- 5) 填写 IB_CON1~5；
- 6) 开始烧写，CPU 将进入 IDLE 模式，烧写完成后自动退出 IDLE 模式；
- 7) 如需需要继续写入数据，跳转至第 3 步；
- 8) 恢复中断设置；
- 9) 将 XPAGE 寄存器清 0。

注意:在第 6 步与第 7 步之间必须添加 4 个空操作指令。

B.擦除：

- 1) 禁止中断；
- 2) 填写 IB_CLK1, IB_CLK0；
- 3) 按相应的扇区填写 XPAGE；
- 4) 填写 IB_CON1~5；
- 5) 开始擦除，CPU 将进入 IDLE 模式，烧写完成后自动退出 IDLE 模式；
- 6) 如果需要更多擦除扇区，跳转至第 3 步继续操作。
- 7) 恢复中断设置；
- 8) 将 XPAGE 寄存器清 0。

在第 5 步与第 6 步之间必须添加 4 个空操作指令以更稳定的操作，

C.读取：

使用“MOVC A, @A+DPTR” or “MOVC A, @A+PC”

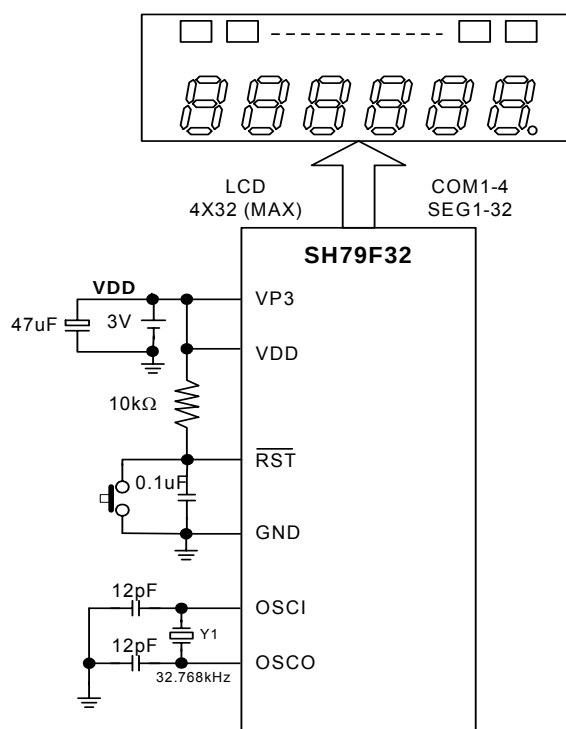


7.4 低电压运行

SH79F32 可以在高电压模式(3.6~5.5V)和低电压模式(2.4V – 3.6V)下运行。在高电压模式下，接通内部稳压源能为 CPU 以及包括 LCD 驱动器在内的外部设备建立稳定的 3V 电源。V_{DD} 作为稳压源输入引脚，VP3 引脚作为稳压源输出。用户必须选择高 LVR 功能确保稳压源有足够的输入电压。

对于低电压的应用 (2.4V – 3.6V)，因为内部稳压源电压比 V_{DD} 低，所以要求短接 V_{DD} 和 VP3 使内部稳压源短路。考虑到可靠性，用户须选择低 LVR 功能避免电池耗尽。

在低电模式下 LCD 的性能会有所降低，当电源低于 3V 时，因为大部分 LCD 面板为 3V 应用，所以 LCD 的性能会有所降低。



应用框图

注意：在低电压模式下，V_{DD} 不能超过 3.6V，否则会对芯片造成永久损坏。



7.5 系统时钟和振荡器

7.5.1 特性

- 内建 8.192MHz 索相环(PLL).
- 晶体振荡器: 32768Hz 或 400kHz – 8MHz.
- 陶瓷振荡器: 32768Hz 或 400kHz – 8MHz.

通过代码选项, SH79F32 可以选择 5 种振荡器为系统提供时钟:32768Hz 晶振, 400kHz-8MHz 晶振,400kHz-8MHz 陶瓷振荡器, 400kHz-8MHz 外灌时钟或 8MHz 内部 RC 电路. 振荡器产生的基本时钟脉冲作为系统时钟提供给 CPU 和片上外围模块。

SH79F32 内建一个锁相环(PLL), 当选择 32768Hz 晶振时, 锁相环能提供最高 8.192MHz 的时钟。但通过代码选项选择 32768Hz 晶振以外的其他振荡器时, PLL 功能不起作用。

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	-	PLLCON	FS2	FS1	FS0
读/写	读/写	读/写	读/写	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	1	0	0	-	0	0	0	0

位编号	位符号	说明
7	32K_SPDUP	32768Hz 振荡器加速模式控制位 0: 32768Hz 振荡器常规模式, 清'0'操作由软件实现。 1: 32768Hz 振荡器加速模式, 置'1'操作由硬件或软件实现。 本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, 自动由硬件设置为'1', 用以进入加速模式, 缩短 32768Hz 振荡器的起振时间。 如果有应用需要, 本位也可以由软件置'1'。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'。 应该注意的是将此位置清'0'可以节省系统的耗电(1uA 左右), 因此, CPU 稳定后应将此位置清'0'。
6-5	CLKS[1: 0]	系统时钟预分频器 (对 FS[2:0]选择的时钟源分频) 00 = / 1 (默认) 01 = / 2 10 = / 4 11 = / 12
3	PLLCON	PLL 锁相环控制 0 = 关闭 PLL 锁相环 1 = 打开 PLL 锁相环(仅选择 32768Hz 晶振时有效)
2-0	FS [2:0]	系统时钟源选择 0xx = 32768Hz 作为系统时钟源 100 = PLL 提供 8.192MHz 系统时钟源 101 = PLL 提供 4.096MHz 系统时钟源 110 = PLL 提供 2.048MHz 系统时钟源 111 = PLL 提供 1.024MHz 系统时钟源

PLL 作为系统时钟源, 必须按以下步骤依次设置:

- 1) 在 PLL 控制寄存器中设置 FS1 和 FS0;
- 2) 设置 PLLCON = 1, 打开 PLL;
- 3) 至少等待 2ms;
- 4) 设置 FS2 = 1, 选择 PLL 作为系统时钟源。

PLL 切换到 32768Hz 作为系统时钟源, 必须按以下步骤依次设置:

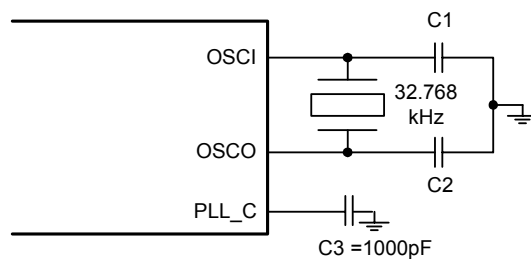
- 1) 设置 FS[2:0]=0xx, 选择 32768Hz 作为系统时钟源;
- 2) 执行一个 NOP 指令;
- 3) 设置 PLLCON=0, 关闭 PLL;

注意: 不要同时执行关 PLL 和切换到 32768Hz 的操作。

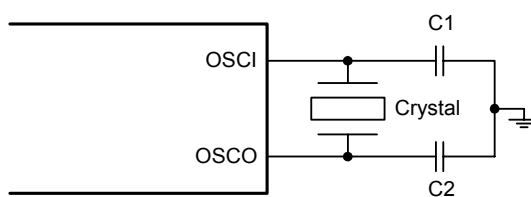


7.5.2 振荡器类型

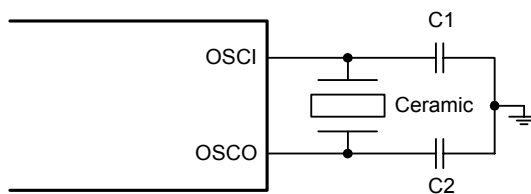
32768Hz 晶体振荡器



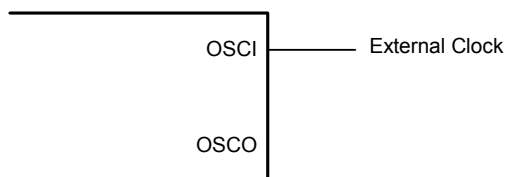
400kHz – 8MHz 晶体振荡器



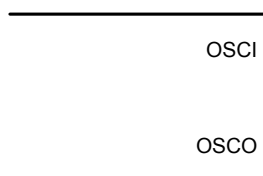
400kHz – 8MHz 陶瓷振荡器



400kHz – 8MHz 外部时钟



8MHz 内部 RC





7.5.3 谐振器负载电容选择

陶瓷谐振器			推荐型号	生产厂
频率	C1	C2		
455kHz	47~100pF	47~100pF	ZTB 455kHz	威克创通讯器材有限公司
			ZT 455E	深圳东光晶博电子有限公司
3.58MHz	-	-	ZTT 3.580M	威克创通讯器材有限公司
			ZT 3.58M*	深圳东光晶博电子有限公司
4MHz	-	-	ZTT 4.000M	威克创通讯器材有限公司
			ZT 4M*	深圳东光晶博电子有限公司

*- 已经内建有负载电容

晶体谐振器			推荐型号	生产厂
频率	C1	C2		
32768Hz	5~12.5pF	5~12.5pF	DT 38 (φ 3x8)	KDS
			φ 3x8 - 32768Hz	威克创通讯器材有限公司
4MHz	8~15pF	8~15pF	HC-49U/S 4.000MHz	威克创通讯器材有限公司
			49S-4.000M-F16E	深圳东光晶博电子有限公司
8MHz	8~15pF	8~15pF	HC-49U/S 8.000MHz	威克创通讯器材有限公司
			49S-8.000M-F16E	深圳东光晶博电子有限公司

注意事项:

- 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, 但并非最优值。
- 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。
- 4) 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 I/O 端口

7.6.1 特性

- 5 个 8 位双向 I/O 端口.
- I/O 端口可与其他选择功能共用.

SH79F32 提供 5 个 8 位可编程双向 I/O 端口。端口数据在寄存器 Px 中。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，每个 I/O 端口带有由 PxPCRy(x=0~4,y=0~7)控制的内部上拉电阻。

SH79F32 的有些 I/O 引脚能与其他选择功能共用。当所有功能都允许时，CPU 内优先级仲裁电路能够避免功能冲突。（具体参考端口共用章节）。

7.6.2 寄存器

端口控制寄存器

	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)	P0CR.7	P0CR.6	P0CR.5	P0CR.4	P0CR.3	P0CR.2	P0CR.1	P0CR.0
P1CR (E2H)	P1CR.7	P1CR.6	P1CR.5	P1CR.4	P1CR.3	P1CR.2	P1CR.1	P1CR.0
P2CR (E3H)	P2CR.7	P2CR.6	P2CR.5	P2CR.4	P2CR.3	P2CR.2	P2CR.1	P2CR.0
P3CR (E4H)	P3CR.7	P3CR.6	P3CR.5	P3CR.4	P3CR.3	P3CR.2	P3CR.1	P3CR.0
P4CR (E5H)	P4CR.7	P4CR.6	P4CR.5	P4CR.4	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)	0	0	0	0	0	0	0	0

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

端口上拉电阻控制寄存器

	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)	P0PCR.7	P0PCR.6	P0PCR.5	P0PCR.4	P0PCR.3	P0PCR.2	P0PCR.1	P0PCR.0
P1PCR (EAH)	P1PCR.7	P1PCR.6	P1PCR.5	P1PCR.4	P1PCR.3	P1PCR.2	P1PCR.1	P1PCR.0
P2PCR (EBH)	P2PCR.7	P2PCR.6	P2PCR.5	P2PCR.4	P2PCR.3	P2PCR.2	-	-
P3PCR (ECH)	P3PCR.7	P3PCR.6	P3PCR.5	P3PCR.4	P3PCR.3	P3PCR.2	P3PCR.1	P3PCR.0
P4PCR (EDH)	P4PCR.7	P4PCR.6	P4PCR.5	P4PCR.4	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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	PxPCR.y 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（默认） 1: 内部上拉电阻开启



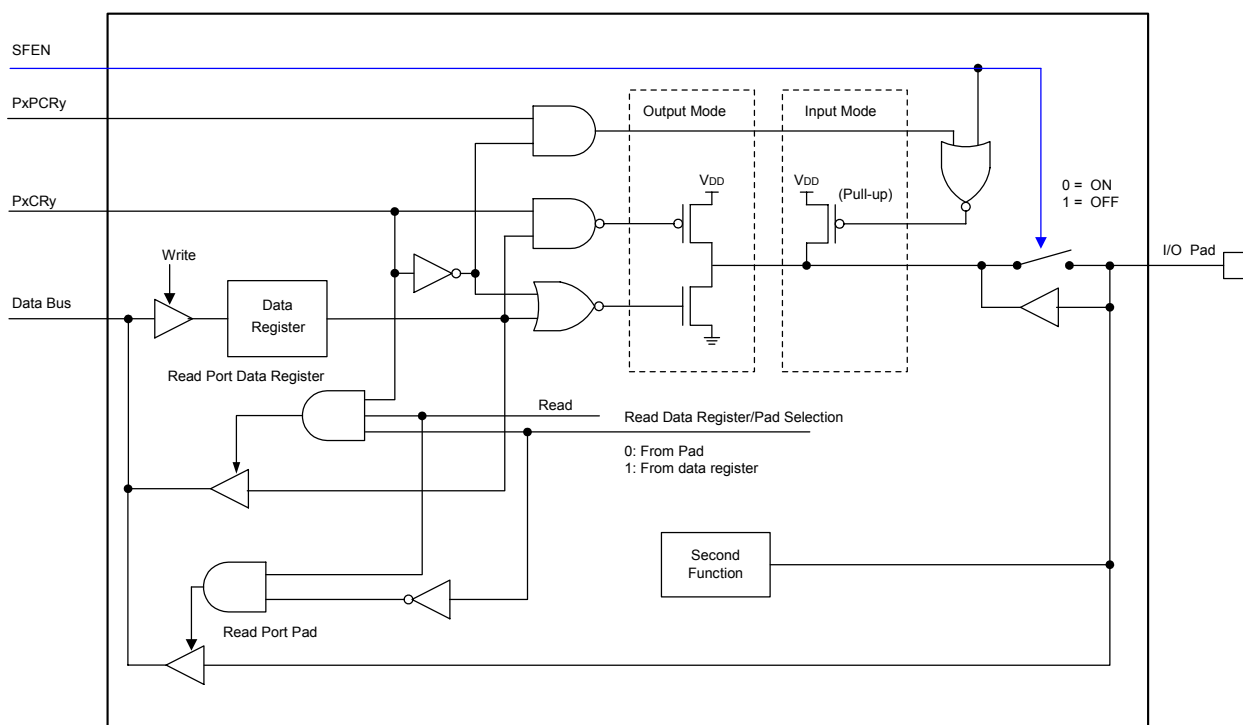
端口数据寄存器

	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)	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
P1 (90H)	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
P2 (A0H)	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1*	P2.0*
P3 (B0H)	P3.7	P3.6	P3.5	P3.4	P3.3	P3.2	P3.1	P3.0
P4 (C0H)	P4.7	P4.6	P4.5	P4.4	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)	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

*注意：这些端口作为 N-沟道的开漏 I/O。

7.6.3 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。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。
- 3) 输入/输出端口写操作都是针对端口数据寄存器。



7.6.4 端口共用

40 个双向 I/O 端口也能共用作第二或第三种特殊功能。共用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部标注功能享有最高优先级，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。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 PxyCR，PxyPCR，但在复用功能被禁止前，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，对任何输出端口进行写入操作时，会影响到数据寄存器的值，端口引脚值保持不变，直到其它复用功能关闭。根据指令的不同，对输出端口进行读取操作时，将读取数据寄存器中的值或引脚电平。

当允许端口复用为其它功能时，任何写入输入端口操作只会影响到数据寄存器，任何从输入端口读取的操作只会返回引脚电平。

如果第二功能允许模拟模块例如 ADC，不论实际引脚的电压电平或 I/O 状态，引脚只能读回 0。

PORT0:

- LCD Segment 25-32 (P0.0 – P0.7)
- PWMN: PWM 负输出(P0.6)
- PWMP: PWM 正输出(P0.7)

PORT0 共用功能列表			
引脚编号	优先级	功能	允许位
引脚 64	1	PWMP	PWMEN 寄存器中的 EPWM 位设置为'1'
	2	SEG32	P0SS 寄存器中的 P0S7 位设置为'1'
	3	P0.7	EPWM 位和 P0S7 位都清'0'
引脚 63	1	PWMN	PWMEN 寄存器中的 EPWMN 位设置为'1'
	2	SEG31	P0SS 寄存器中的 P0S6 位设置为'1'
	3	P0.6	EPWMN 位和 P0S6 位都清'0'
引脚 62	1	Flt	PWMEN 寄存器中的 EFLT 设置为'1'
	2	SEG30	P0SS 寄存器中的 P0S5 位设置为'1'
	3	P0.5	EFLT 位和 P0S5 位都清'0'
引脚 57~61	1	SEG25~SEG29	P0SS 寄存器中的相应位(P0S0-P0S4)设置为'1'
	2	P0.0~P0.4	P0SS 寄存器中的相应位(P0S0-P0S4)清'0'

PORT1:

- LCD Segment 17-24 (P1.0 – P1.7)

PORT1 共用功能列表			
引脚编号	优先级	功能	允许位
引脚 49~56	1	SEG17~SEG24	P1SS 寄存器中的相应位(P1S0-P1S7)设置为'1'
	2	P1.0~P1.7	P1SS 寄存器中的相应位(P1S0-P1S7)清'0'



PORT2:

- SCL (P2.0): TWI 串行时钟线 (开漏)
- SDA (P2.1): TWI 串行数据线 (开漏)
- MOSI (P2.2): SPI 主输出从属输入
- MISO (P2.3): SPI 主输入从属输出
- SCK (P2.4): SPI 串行时钟
- SS (P2.5): SPI 从属选择
- RXD1 (P2.6): EUART1 数据输入
- TXD1 (P2.7): EUART1 数据输出
- AN0~AN5 (P2.2 – P2.7): ADC 输入通道

PORT2 共用功能列表			
引脚编号	优先级	功能	允许位
引脚 9	1	SCL	TWICON 寄存器中的 TWIEN 位设置为'1'
	2	P2.0	TWICON 寄存器中的 TWIEN 位清'0'
引脚 10	1	SDA	TWICON 寄存器中的 TWIEN 位设置为'1'
	2	P2.1	TWICON 寄存器中的 TWIEN 位清'0'
引脚 11	1	MOSI	在从属模式下将 SPSTA 寄存器中的 SPEN 位设置为'1' (当 SPEN,CPHA,SSDIS 位在从属模式下都设置为'1'时,自动上拉)
	2	AN0	ADCH 寄存器中的 CH0 位和 ADCON 寄存器中的 ADON 位都设置为'1', 并且 SCH [2:0] = 000
	3	P2.2	SPSTA 寄存器中的 SPEN 位和 ADCH 寄存器中的 CH0 位都清'0'
引脚 12	1	MISO	在 SPSTA 寄存器中的 SPEN 位设置为'1' (在主模式下将 SPSTA 寄存器中的 SPEN 位设置为'1'时,自动上拉)
	2	AN1	ADCH 寄存器中的 CH1 位和 ADCON 寄存器中的 ADON 位都设置为'1', 并且 SCH [2:0] = 001
	3	P2.3	SPSTA 寄存器中的 SPEN 位和 ADCH 寄存器中的 CH1 位都清'0'
引脚 13	1	SCK	SPSTA 寄存器中的 SPEN 位设置为'1' (当 SPEN,CPHA,SSDIS 位在从属模式下都设置为'1'时,自动上拉)
	2	AN2	ADCH 寄存器中的 CH2 位和 ADCON 寄存器中的 ADON 位都设置为'1', 并且 SCH [2:0] = 010
	3	P2.4	SPSTA 寄存器中的 SPEN 位和 ADCH 寄存器中的 CH2 位都清'0'
引脚 14	1	SS	当 SPEN=1 时, 在 SPI 主模式下将 SPCON 寄存器中的 SSDIS 位清'0', 或者在 SPI 从模式下当 CPHA=1 时将 SPCON 寄存器中的 SSDIS 位清'0', 或者在 SPI 从模式下将 SPCON 寄存器中的 CPHA 位清'0'. (当 SPEN =1 & Master=1 & SSDIS=0 时,自动上拉, 当 SPEN=1 & Master=0 时,自动上拉)
	2	AN3	ADCH 寄存器中的 CH3 位和 ADCON 寄存器中的 ADON 位都设置为'1', 并且 SCH [2:0] = 011
	3	P2.5	上述情况两种情况未发生
引脚 15	1	RXD1	SCON1 寄存器中的 REN1 位设置为'1' (当 REN1=1 时,自动上拉)
	2	AN4	ADCH 寄存器中的 CH4 位和 ADCON 寄存器中的 ADON 位都设置为'1', 并且 SCH [2:0] = 100
	3	P2.6	SCON1 寄存器中的 REN1 位和 ADCH 寄存器中的 CH4 位都清'0'
引脚 16	1	TXD1	写入 SBUF1 寄存器
	2	AN5	ADCH 寄存器中的 CH5 位和 ADCON 寄存器中的 ADON 位都设置为'1', 并且 SCH [2:0] = 101
	3	P2.7	不写 SBUF1 寄存器,并且 ADCH 寄存器中的 CH5 位清'0'



PORT3:

- RXD0 (P3.0): EUART 0 数据输入
- TXD0 (P3.1): EUART 0 数据输出
- INT0 (P3.2): 外部中断 0
- INT1 (P3.3): 外部中断 1
- T0 (P3.4): 定时器 0 外部输入
- T1 (P3.5): 定时器 1 外部输入
- INT2 (P3.6): 外部中断 2
- T2 (P3.7): 定时器 2 外部输入/波特率时钟输出
- CALIN (P3.4): RTC 校准输入
- CALOUT (P3.5): RTC 校准输出

PORT3 共用功能列表			
引脚编号	优先级	功能	允许位
引脚 17	1	RXD0	SCON 寄存器中的 REN 位设置为'1' (自动上拉)
	2	P3.0	SCON 寄存器中的 REN 位清'0'
引脚 18	1	TXD0	写入 SBUF 寄存器
	2	P3.1	无上述情况
引脚 19	1	INT0	IE0 寄存器中的 EX0 位设置为'1', 并且 Port3.2 设置为输入模式 (上拉由软件设置)
	2	P3.2	无上述情况
引脚 20	1	INT1	IE0 寄存器中的 EX1 位设置为'1', 并且 Port3.3 设置为输入模式 (上拉由软件设置)
	2	P3.3	无上述情况
引脚 21	1	CALIN	当 RTCCON 寄存器中的 RTCEN 和 CAL 位均为'1'时, 将 AUTOCAL 位设置为'1'.
	2	T0	TCON 寄存器中的 TR0 位和 TMOD 寄存器中的 C/T0 位都设置为'1' (自动上拉)
	3	P3.4	无上述情况
引脚 22	1	CALOUT	RTCCON 寄存器中的 RTCEN 位设置为'1', 并且置 OUTF [1:0] = 01 或 OUTF [1:0] = 10
	2	T1	TCON 寄存器中的 TR1 位和 T1MOD 寄存器中的 C/T1 位都设置为'1' (自动上拉)
	3	P3.5	无上述情况
引脚 23	1	INT2	IE0 寄存器中的 EX2 位设置为'1', 并且 Port3.6 设置为输入模式 (上拉由软件设置)
	2	P3.6	无上述情况
引脚 24	1	T2	T2CON 寄存器中的 TR2 位和 C/T2 位都设置为'1' (自动上拉)
	2	P3.7	无上述情况

**Port 4:**

- VP2 (P4.0): LCD 电源引脚
- VP1 (P4.1): LCD 电源引脚
- CUP2 (P4.2): LCD 偏置电容器连接引脚
- CUP1 (P4.3): LCD 偏置电容器连接引脚
- COM0 (P4.4): Common 信号输出
- COM1 (P4.5): Common 信号输出
- COM2 (P4.6): Common 信号输出
- COM3 (P4.7): Common 信号输出

PORT4 共用功能列表			
引脚编号	优先级	功能	允许位
引脚 25	1	VP2	LCDCON 寄存器中的 P4L0 位清'0'
	2	P4.0	LCDCON 寄存器中的 P4L0 位置'1'
引脚 26	1	VP1	LCDCON 寄存器中的 P4L0 位清'0'
	2	P4.1	LCDCON 寄存器中的 P4L0 位置'1'
引脚 27	1	CUP2	LCDCON 寄存器中的 P4L0 位清'0'
	2	P4.2	LCDCON 寄存器中的 P4L0 位置'1'
引脚 28	1	CUP1	LCDCON 寄存器中的 P4L0 位清'0'
	2	P4.3	LCDCON 寄存器中的 P4L0 位置'1'
引脚 29	1	COM1	LCDCON 寄存器中的 P4L1 位清'0'
	2	P4.4	LCDCON 寄存器中的 P4L1 位置'1'
引脚 30	1	COM2	LCDCON 寄存器中的 P4L1 位清'0'
	2	P4.5	LCDCON 寄存器中的 P4L1 位置'1'
引脚 31	1	COM3	LCDCON 寄存器中的 P4L1 位清'0'
	2	P4.6	LCDCON 寄存器中的 P4L1 位置'1'
引脚 32	1	COM4	LCDCON 寄存器中的 P4L1 位清'0'
	2	P4.7	LCDCON 寄存器中的 P4L1 位置'1'



7.7 定时器

SH79F32 有 3 个独立的 16 位定时器，Timer0，Timer1 和 Timer 2，这些定时器可兼容传统型 8052 芯片。

7.7.1 特性

- 可分别设置为定时器或计数器。
- 作为定时器时，每一个系统时钟定时器数据寄存器加 1。
- 作为计数器时，当相应的外部输入脚(T0,T1,T2)产生由 1→0 的跳变时，相应的数据寄存器加 1，
- Timer0, Timer1 和 Timer2 都有 4 种工作模式(模式 0,1,2,3)

7.7.2 Timer 0 & Timer 1

两个数据寄存器(THx & TLx (x = 0, 1))串联后可作为一个 16 位寄存器来访问。这两个定时器都由寄存器 TCON 和 TMOD 控制。设置 IEN0 寄存器的 ET0 和 ET1 位能允许 Timer 0 和 Timer 1 中断。(详情请见 “ 中断 ” 章节的相关描述)

寄存器

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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7,5	TFx x = 0, 1	Timer x 溢出标志 0 = Timer x 无溢出，由软件清除 1 = Timer x 溢出，由硬件设置；若由软件设置将会引起定时器中断
6,4	TRx x = 0, 1	Timer x 启动，停止控制 0 = 停止 timer x 1 = 启动 timer x
3,1	IEx x = 0, 1	外部中断 x 请求标记 0 = 无中断 1 = 有中断
2,0	ITx x = 0, 1	外部中断 x 触发模式 0 = 电平触发 1 = 沿边触发



SH79F32

定时器 x / 计数器 x 模式寄存器 (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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7,3	GATE _x x = 0, 1	定时器 x 门控 0 = 禁止 1 = 允许(如果 TR _x 和 GATE _x 位均为 1, INT _x 引脚将被配置为输入状态, 自动打开上拉)
6,2	C/T _x x = 0, 1	定时器 x 定时器/计数器模式选定位 0 = 定时器模式, 如果 RTC 功能被禁止, T0 或 T1 引脚用作 I/O 端口; 如果 RTC 模块允许 CALIN 或 CALOUT, 作为 CALIN 或 CALOUT。 1 = 计数器模式(如果 TR _x 和 C/T _x 位均为 1, T0/T1 引脚将被配置为输入状态, 自动打开上拉)
5-4 1-0	M _x [1:0] x = 0, 1	定时器 x 定时器模式选定位 00 = 模式 0, 13 位向上计数计数器/定时器, 忽略 TL _x 的第 7~5 位 01 = 模式 1, 16 位向上计数计数器/定时器 10 = 模式 2, 8 位自动重载向上计数计数器/定时器 11 = 模式 3 (只用于定时器 0), 两个 8 位向上计数定时器

定时器 x / 计数器 x 数据寄存器 (x = 0,1)

	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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TL _x .y, TH _x .y x=0~1, y=0~7	定时器 x 低及高字节计数器



Timer 0 和 Timer 1 模式

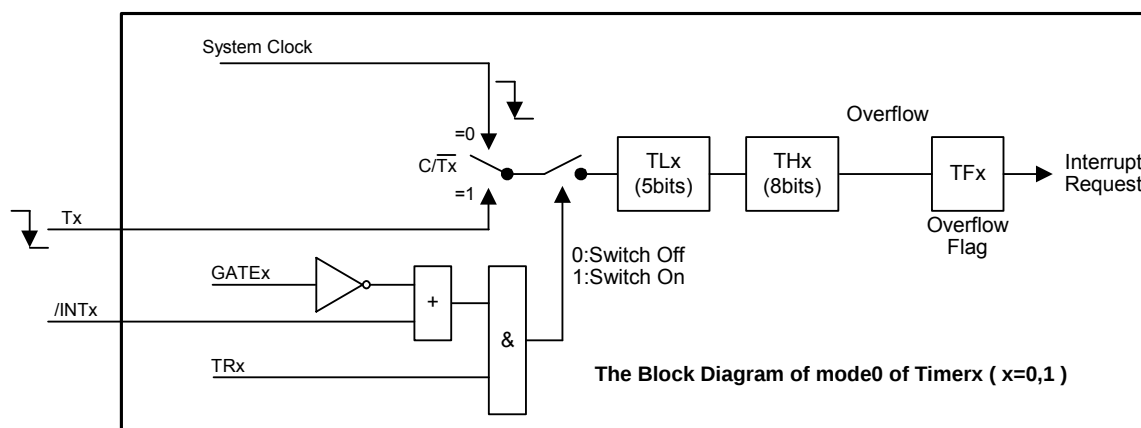
两个定时器都能通过计数器/定时器模式寄存器(TMOD)下的模式选定位 Mx1-Mx0 (x = 0, 1)在 4 种主要模式下运转。下文只对 Timer 0 进行描述。因为 Timer 1 除了使用自己的寄存器位外,与模式 0/1/2 下的 Timer 0 是一致的。

模式 0 : 13 位计数器/定时器

在模式 0 中,Timer 0 和 Timer 1 为 13 位计数器/定时器。TH0 寄存器存放 13 位计数器/定时器的高 8 位, TL0 存放低 5 位 (TL0.4-TL0.0)。TL0 (TL0.7-TL0.5)的高三位是不确定的,在读取时被忽略。当 13 位定时器寄存器递增并从 0xFF1F 溢出到 0x0000 时,系统置起定时器溢出标记 TF0 (TCON.5)。如果 Timer 0 中断被允许,将会产生一个中断。C/T0位(TMOD.2)用于选择计数器/定时器的时钟源。

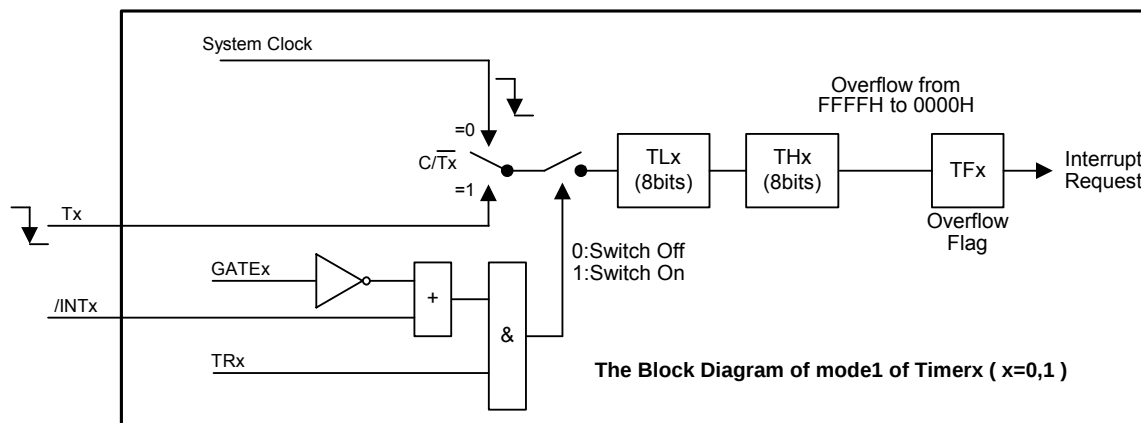
如果 C/T0 = 1, Timer 0 引脚(T0)的输入电平从高到低跳变,使定时器/计数器 0 数据寄存器加 1。如果 C/T0 = 0, 选择系统时钟为定时器/计数器 0 的时钟源。

无论 GATE0 (TMOD.3)= 0 或 GATE0 = 1, 设置 TR0 位(TCON.4)都能打开定时器。GATE0 置高允许定时器由外部输入信号 INT0 控制, 这样可测量 INT0 的正脉冲宽度。置位 TR0 时不会复位计数器, 意味着如果 TR0 置位, 定时器寄存器将从计数器上次值开始计数。所以在开启定时器之前, 定时器寄存器应该重新设置为所需的初始值。



模式 1: 16 位计数器/定时器

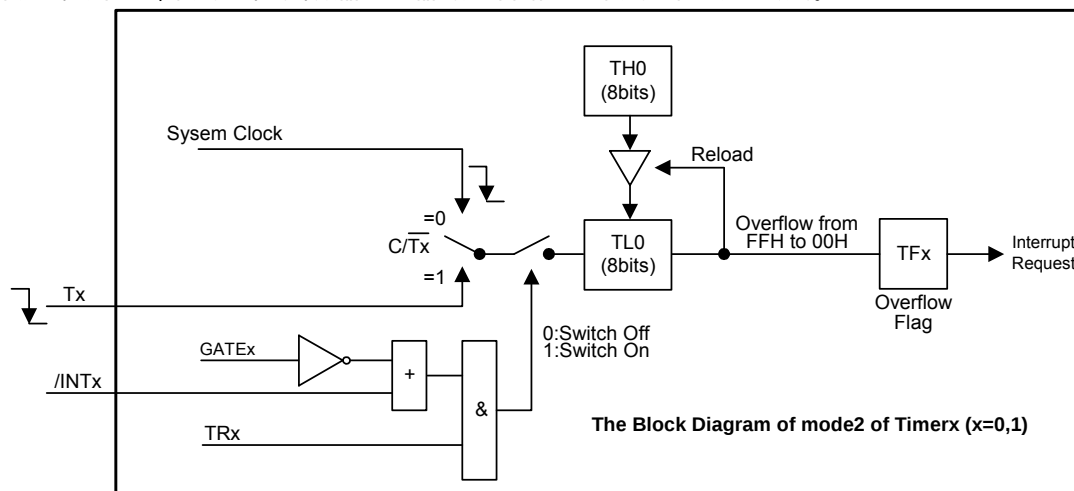
在模式 1 中, 除了使用 16 位定时器/计数器之外, 其余和模式 0 一致, 允许和配置也如同模式 0。



**模式 2: 8 位带自动重载的计数器/定时器**

模式 2 中, Timer 0 和 Timer 1 配置为可自动重载初始值的 8 位计数器/定时器。TL0 存放计数值, TH0 存放重载值。当在 TL0 中的计数器溢出至 0x00 时, 定时器溢出标记 TF0 (TCON.5)置 1, TH0 中的重载值重新存入 TL0 中。如果允许 Timer 0 中断, 当 TF0 置位时将产生一个中断。而在 TH0 中的重载值不会改变。在允许定时器正确计数开始之前, TL0 必须初始化为所需的值。

除了自动重载功能外, 模式 2 中的计数器/定时器的允许和配置与模式 1 和 0 是一致的。

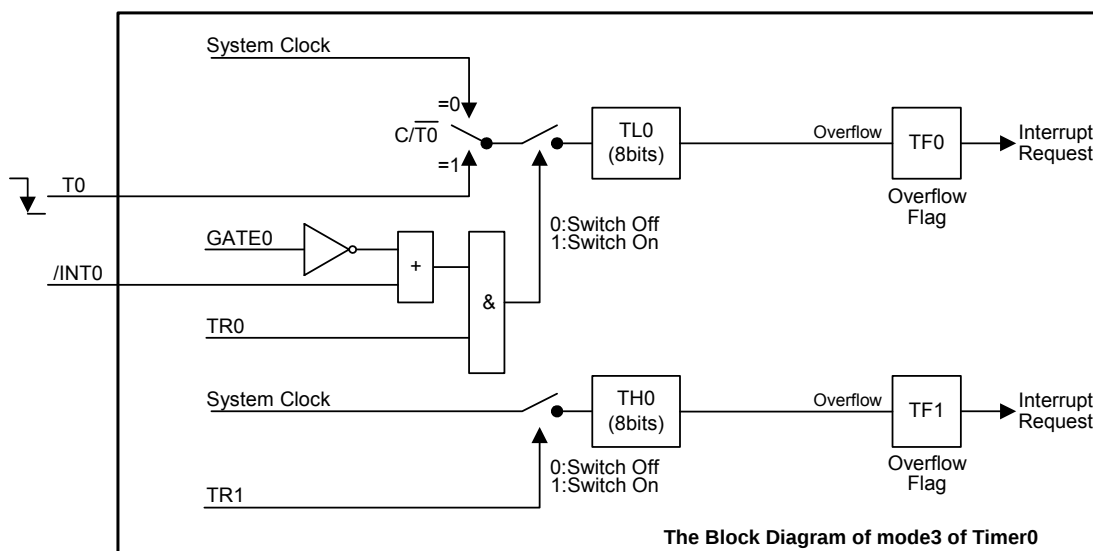
**模式 3: 两个 8 位计数器/定时器 (仅限 Timer 0)**

在模式 3 中, Timer 0 用作两个独立的 8 位计数器/定时器, 计数值分别存入 TL0 和 TH0。寄存器 TCON 和 TMOD (TR0, C/T0, GATE0 和 TF0) 是 Timer 0 的控制/状态位, 用来控制在 TL0 的计数器/定时器。

TL0 能用系统时钟或外部输入信号作为时钟。TH0 只能以系统时钟为时钟源用于定时器功能。Timer 1 控制位 TR1 允许 TH0, TH0 溢出时, 置位 Timer 1 标志 TF1, 控制 Timer 1 中断。

当定时器 0 工作在模式 3 时, 定时器 1 可以工作在模式 0, 模式 1 或是模式 2, 但是不能选用外部时钟源和置位 TF1 产生中断。然而定时器 1 可用作 EUART 的波特率发生器。

当定时器 0 工作在模式 3 时, 因为 TR1 被定时器 0 占用, 定时器 1 的启动运行可由模式寄存器控制, T1 引脚上的上拉电阻也被禁止, 设置定时器 1 工作在模式 0, 模式 1 或模式 2 时则定时器 1 工作; 设置工作在模式 3 时, 定时器 1 被禁止。



注意: 当定时器 0/1 用于计数器模式时, 所有输入信号需与系统时钟同步, 因此 T0/T1 输入频率必须低于系统时钟频率的 $1/2$, /INT0 or /INT1 必须低于系统时钟频率的 $1/4$ 。



7.7.3 Timer 2

SH79F32 的 Timer 2 由两个数据寄存器 (TH2 和 TL2) 串联后构成一个 16 位定时器, 由寄存器 T2CON 和 T2MOD 控制。设置 IEN0 寄存器中的 ET2 位能允许 Timer2 中断 (详情请见“中断”章节)。

Timer2 的工作模式和与 Timer 0 和 Timer 1 相似。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。置位 TR2 允许 Timer 2/Counter 2 按所选的时钟源使计数器递增计数。

Timer2 默认为 Euart1 的波特率, 无论 Timer2 工作在什么模式。

寄存器

Timer 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
R/W	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDI/LVR)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	TF2	Timer2 溢出标记位 0 = 无溢出 (必须由软件清除) 1 = 溢出 (如果 RCLK = 0 且 TCLK = 0, 由硬件设置)
6	EXF2	检测到 T2EX 引脚上有外部输入 (下降沿), 置起的标志位。 0 = 无外部信号输入 (必须由软件清除) 1 = 检测到外部信号输入 (如果 EXEN2 = 1, 由硬件设置)
5	RCLK	EUART0 接收时钟控制位 0 = 来自 Timer 1 1 = 来自 Timer 2
4	TCLK	EUART0 发送时钟控制位 0 = 来自 Timer 1 1 = 来自 Timer 2
3	EXEN2	T2EX 引脚输入的外部信号 (下降沿) 产生重载/捕获的允许/禁止控制位 0 = T2EX 引脚上的信号 1 = T2EX 引脚上检测到一个下降沿, 产生一个捕获或重载
2	TR2	Timer2 开始/停止控制位 0 = 停止 Timer 2 1 = 开始 Timer 2
1	C/T2	Timer 2 定时器/计数器模式选定位 0 = 定时器模式, T2 引脚用作 I/O 端口 1 = 计数器模式, 内部上拉电阻被打开
0	CP/RL2	捕获/重载模式选定位 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



SH79F32

Timer 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	-	-	-	-	-	-	T2OE	DCEN
读/写	-	-	-	-	-	-	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	-	-	0	0

位编号	位符号	说明
1	T2OE	Timer 2 输出允许位 0 = 设置 P3.7/T2 作为时钟输入或 I/O 端口。 1 = 设置 P3.7/T2 作为时钟输出 (波特率发生器模式)
0	DCEN	下降计数器允许位 0 = 禁止 Timer 2 作为向上/向下计数器, Timer 2 仅作为向上计数器。 1 = 允许 Timer 2 作为向上/向下计数器。

Timer 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(CAH)	RCAP2L. 7	RCAP2L. 6	RCAP2L. 5	RCAP2L. 4	RCAP2L. 3	RCAP2L. 2	RCAP2L. 1	RCAP2L. 0
RCAP2H(CBH)	RCAP2H. 7	RCAP2H. 6	RCAP2H. 5	RCAP2H. 4	RCAP2H. 3	RCAP2H. 2	RCAP2H. 1	RCAP2H. 0
TL2(CCH)	TL2.7	TL2.6	TL2.5	TL2.4	TL2.3	TL2.2	TL2.1	TL2.0
TH2(CDH)	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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	RCAP2L.x RCAP2H.x	Timer 2 重载/捕获数据, x=0~7
7-0	TL2.x TH2.x	Timer 2 高位低位计数器, x=0~7



Timer 2 模式

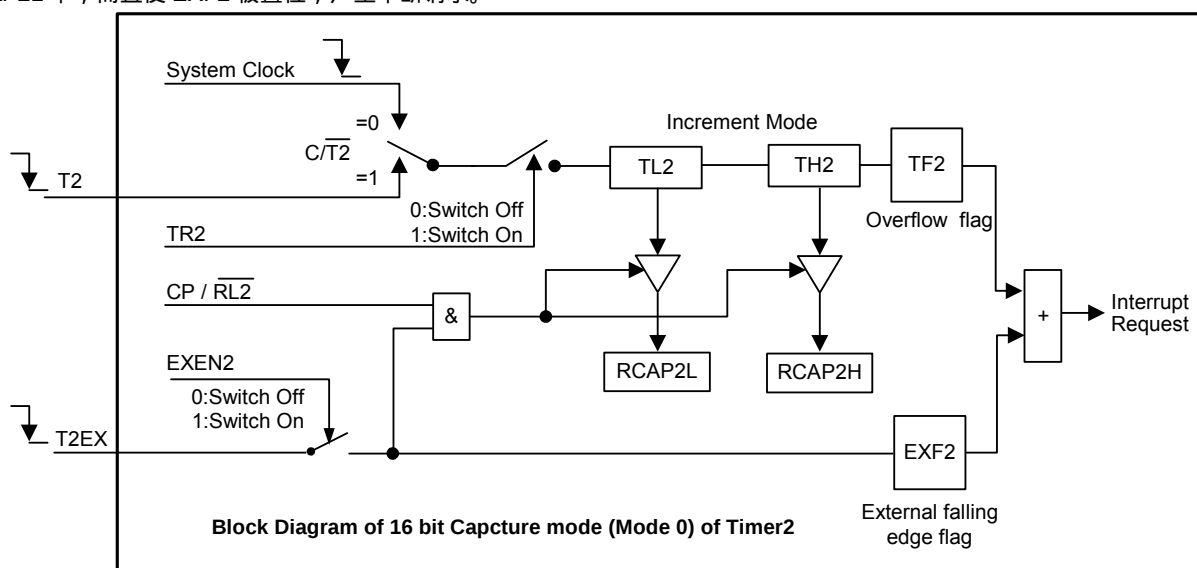
Timer 2 有 4 种工作模式：捕捉，自动重载，波特率发生器和可编程时钟输出。由下图各位内容组合决定。

Timer 2 模式								模式
C/T2	T2OE	DCEN	TR2	CP/RL2	RCLK	TCLK		
X	0	X	1	1	0	0	0	16 位捕捉
X	0	0	1	0	0	0	1	16 位自动重载定时器
X	0	1	1	0	0	0		波特率发生器
X	0	X	1	X	1	X	2	
					X	1		只用于可编程时钟输出
0	1	X	1	X	0	0	3	
					1	X		带波特率发生器的可编程时钟输出
					X	1	3	
X	X	X	0	X	X	X	X	Timer2 停止

模式 0: 16 位捕捉

由图可见，如果 EXEN2 = 0，Timer 2 作为 16 位定时器或计数器，如果 IET2 被允许的话，Timer2 能设置 TF2 溢出产生一个中断。

如果 EXEN2 = 1，Timer 2 执行相同操作，但是 T2EX 端的下降沿会把 TH2 和 TL2 中的当前值分别捕捉到 RCAP2H 和 RCAP2L 中，而且使 EXF2 被置位，产生中断请求。



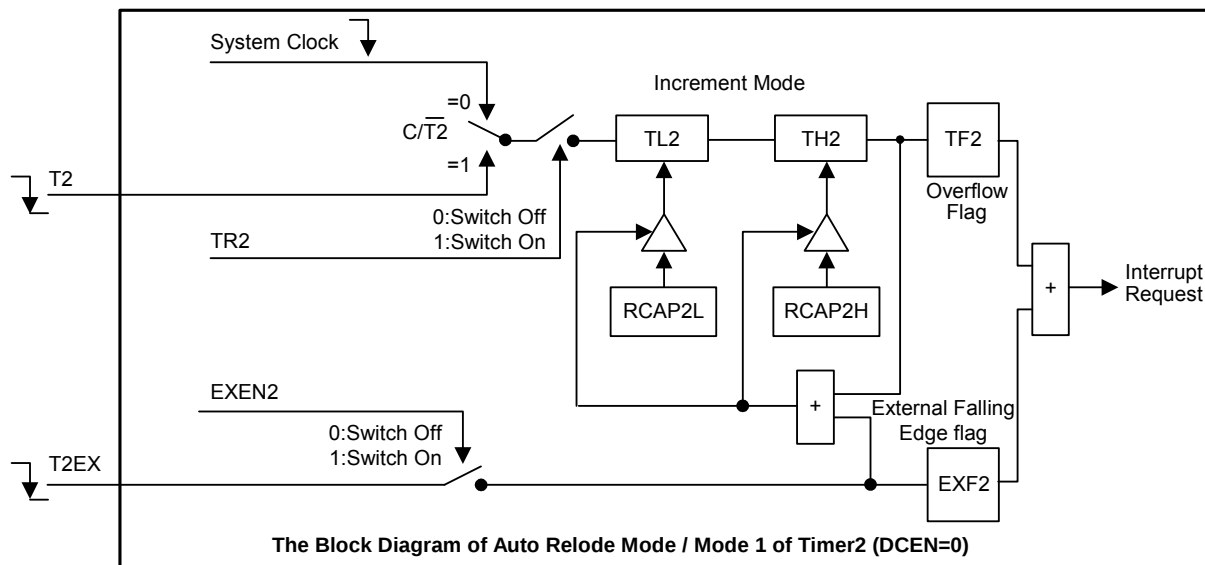
模式 1: 16 位自动重载定时器

在 16 位自动重载模式下，Timer2 可以被编成为为递增计数或递减计数，这个特征由 DCEN 位设定。系统复位后，DCEN 位复位值为 0，Timer 2 默认递增计数。当 DCEN 置位时，Timer 2 递增计数或递减计数取决于 T2EX 引脚上的电平。

当 DCEN=0 时，Timer2 的逻辑关系见下图。

如果 EXEN2 = 0，Timer 2 递增计数到 0FFFFH，在溢出后置位 TF2，RCAP2H 和 RCAP2L 中的值会自动被重载入定时器寄存器 TH2 和 TL2 中。

如果 EXEN2 = 1，定时器溢出或 T2EX 端上的下降沿都能产生自动重载，置位 EXF2，产生中断请求。

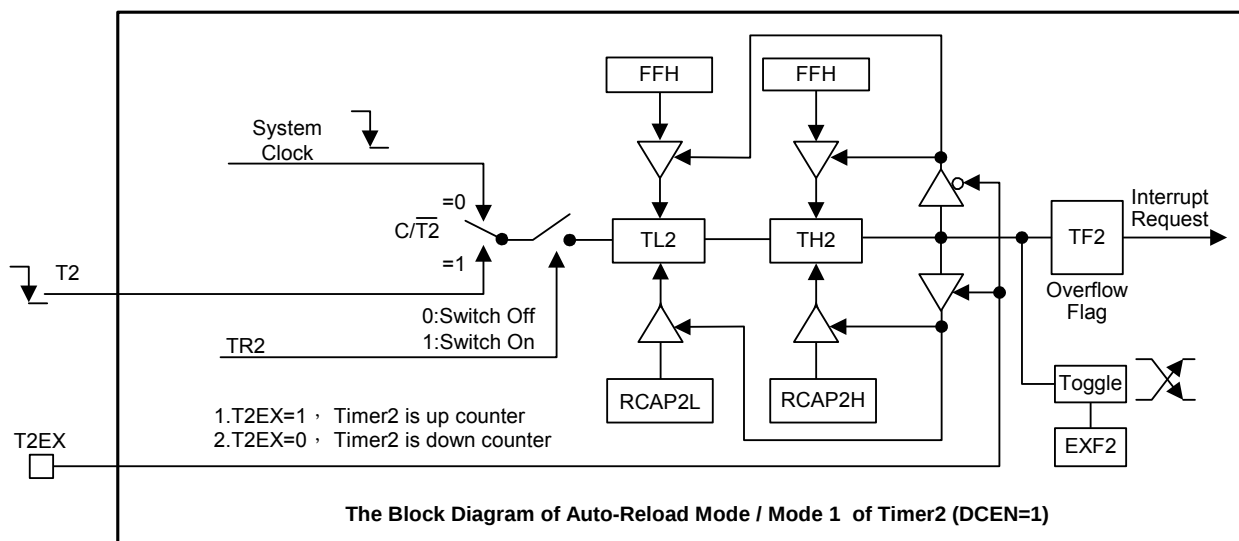


当 DCEN=0 时, Timer2 的逻辑关系见下图。T2EX 引脚电平控制计数的方向, 而 EXEN2 控制无效。

T2EX 输入高电平时, Timer 2 递增计数。定时器直到计数到 0FFFFH 上溢, 然后设置 TF2 位。并使 RCAP2H 和 RCAP2L 的当前值被重载入定时器寄存器。

T2EX 输入低电平时, Timer 2 递减计数。当减到等于 RCAP2H 和 RCAP2L 中的值时, 定时器下溢, TF2 置位。并把 0FFFFH 重载入定时器寄存器。

无论什么时候定时器 2 上溢或者下溢, EXF2 位被触发, 所以可以作为定时器的第 17 位。在此工作模式下, EXF2 不标志中断。





定时器 2 通过设置 TCLK 和 RCLK 作为 UART0 和 UART1 的波特率发生器。

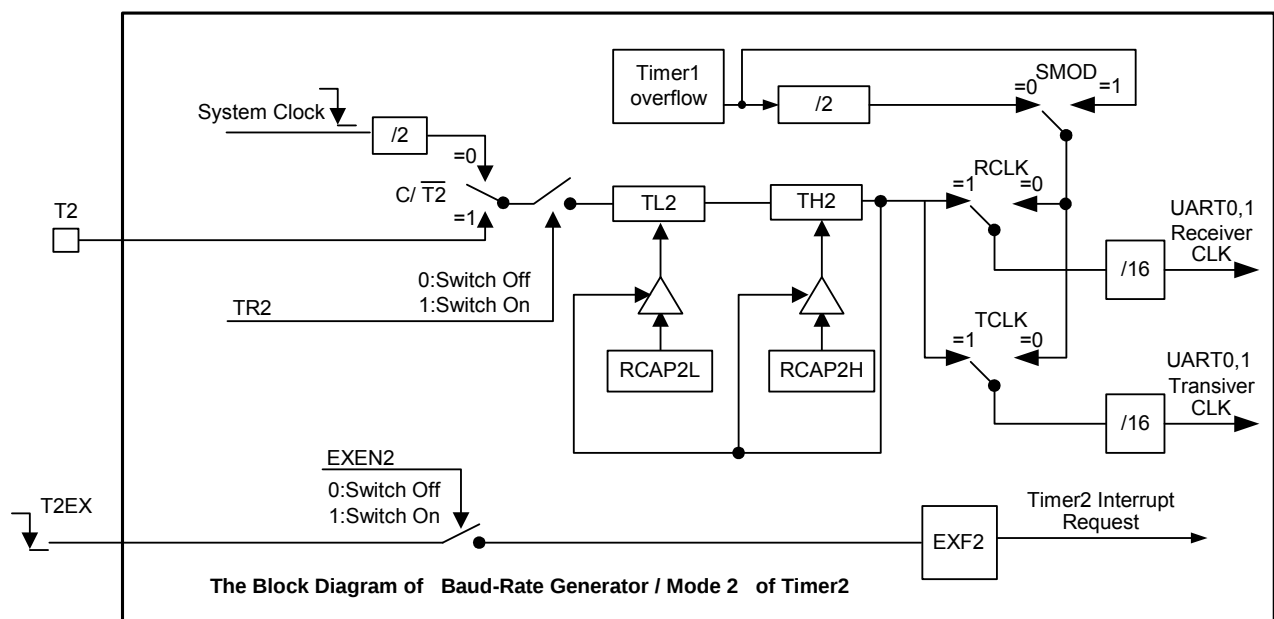
$$BaudRate = \frac{1}{16} \times \frac{System\ Clock}{65536 - [RCAP2H, RCAP2L]}.$$

Timer 2 的溢出会使 RCAP2H 和 RCAP2L 寄存器中的值重载入 Timer 2 計數器，但不会产生中断。

在 EUART 模式 1 和 3 中的波特率由 Timer2 的溢出率控制，公式如下：

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

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



- 1) 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。
- 2) RCAP2 寄存器能读取不能写入，因为写入会覆盖重载而且会引起写入和重载出错。因此，在访问 TH2 / TL2 或 RCAP2H / RCAP2L 寄存器之前，Timer2 必须被关闭（TR2 = 0）。

**模式 3:可编程时钟输出**

T2(P3.7)可以编程输出 50%的占空比时钟周期。清 C/T2位和置 T2OE 位,使 Timer2 作为时钟发生器。TR2 位启动和中止定时器。时钟输出频率取决于系统时钟和 Timer2 捕获寄存器(RCAP2H, RCAP2L)的重载值,如下面的公式所示。

在此种模式下可以分成两种子模式:

1) 如果 RCLK=1 或者 TCLK=1, 这种情况类似于模式 2, 不同的是 T2 引脚将输出一个占空比为 50%的方波。

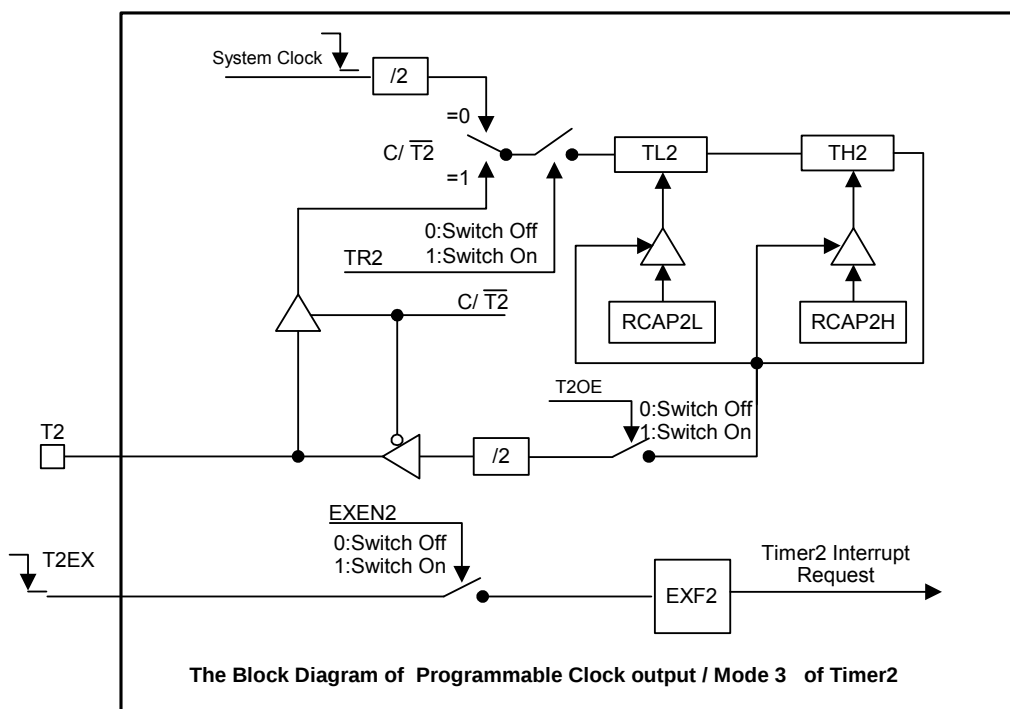
$$\text{Clock Out Frequency} = \frac{1}{2 \times 2} \times \frac{\text{System Clock}}{65536 - [\text{RCAP2H}, \text{RCAP2L}]}$$

此种子模式下, Timer2 溢出时不会产生中断, 此时可以将 Timer2 用作波特率发生器, 同时 T2 脚上输出同频方波。

2) 如果 RCLK 和 TCLK 同时为 0 且 DCEN=0, CP/RL2=0, T2 引脚输出占空比为 50%的方波。

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

此种子模式下, Timer2 将在每次溢出时产生中断。如果此时 EXEN2 被置位, 则当 T2EX 引脚上发生 1 到 0 的变化时, EXF2 将被置位。此时, T2Ex 的功能类似于一个外部中断。

**7.7.4 关于定时器的补充说明:**

- 1) TF2 和 EXF2 都能引起 Timer2 的中断请求, 两者有相同的向量地址。
- 2) 当事件发生时 TF2 和 EXF2 硬件置 1, 在其他任何时间也能由软件设置为 1。但只有软件或硬件复位才能使之清 0。
- 3) 当 EA=1 且 ET2=1 时, 置位 TF2 或 EXF2 能引起 Timer2 中断。
- 4) 当用作计数器使用时, 从 Tx 管脚灌入的信号频率不能大于系统时钟频率的一半。
- 5) 当 Timer1/2 作为波特率发生器时, 读取或写入 TH1/TL1, TH2/TL2, 写入 RCAP2H/RCAP2L 会影响波特率的准确性, 因此也会引起通信出错。



7.8 中断

7.8.1 特性

- 14 个中断源
- 4 级中断优先级
- 程序跑飞中断(OVL)

SH79F32有14个中断源：1个OVL NMI中断，3个外部中断(INT0, INT1和INT2)，3个定时器中断(timer 0, 1和 2)，2个EUART中断，ADC中断，TWI中断，SPI中断，HSEC中断和PWM中断。SH79F32采用4级中断优先级结构。这为SH79F32的14个中断源的处理提供了极大的灵活性。

7.8.2 程序超范围中断(OVL)

SH79F32 有一个 NMI 中断源(OVL)，其向量定位在 0x007B 中，NMI 用以防止 CPU 超出有效程序范围。为应用这个特性，用户应该用常量 0xA5 填满未使用的 flash ROM。

如果 PC 超过了有效的程序范围，则运算代码会显示不存在在 8051 指令集中的 0xA5，CPU 因此获知 PC 已经超出了有效的程序范围，同时 OVL NMI 会发生。如果 PC 超过 32K ROM 范围，NMI 同样会发生。

OVL NMI 享有最高优先级（除 RESET 外），不会被其他中断源中断。同样 OVL NMI 能自身嵌套，但堆栈不会因此增加。当 OVL NMI 发生后，其他中断仍旧被允许，如果满足设定的条件，将置位他们的标志。

用户必须处理 OVL 中断以免保护系统受到不必要的影响。用户可以在中断服务程序中设定继续运行的指定地址，通过 NMI 中断向量服务程式末端的 RETI 指令来修改栈顶地址（因为进入 OVL 中断时，压入堆栈顶端的地址是无用的），这两个操作会使程序调整到用户指定的代码，诸如复位入口或保护程序入口。

```
OVL_NMI_SERVICE:
    MOV DPTR, #(PROTECT_CODE)
    POP    A
    POP    A
    PUSH   DPL
    PUSH   DPH
    RETI
```



7.8.3 中断允许控制

任何一个中断源均可通过对寄存器 IEN0 和 IEN1 中相应的位置位或清零，实现单独使能或禁能。IEN0 寄存器中还包含了一个全局使能位 EA，它可使能所有的中断。一般在复位后，所有中断允许位设置为 0，所有中断被禁止。

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/WDI/LVR)	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	Timer2 溢出中断允许位 0 = 禁止 timer 2 溢出中断 1 = 允许 timer 2 溢出中断
4	ES0	EUART0 中断允许位 0 = 禁止 EUART0 中断 1 = 允许 EUART0 中断
3	ET1	Timer1 溢出中断允许位 0 = 禁止 timer 1 溢出中断 1 = 允许 timer 1 溢出中断
2	EX1	External 中断 1 允许位 0 = 禁止外部中断 1 1 = 允许外部中断 1
1	ET0	Timer0 溢出中断允许位 0 = 禁止 timer 0 溢出中断 1 = 允许 timer 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



SH79F32

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	-	ETWI	EPWM	-	EHSEC	ES1	EX2	ESPI
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

位编号	位符号	说明
6	ETWI	TWI 溢出中断允许位 0 = 禁止 TWI 中断 1 = 允许 TWI 中断
5	EPWM	PWM 中断允许位 0 = 禁止 PWM 中断 1 = 允许 PWM 中断
3	ESEC	半秒中断允许位 0 = 禁止半秒中断 1 = 允许半秒中断
2	ES1	EUART1 中断允许位 0 = 禁止 EUART1 中断 1 = 允许 EUART1 中断
1	EX2	外部中断 2 允许位 0 = 禁止外部中断 2 1 = 允许外部中断 2
0	ESPI	SPI 中断允许位 0 = 禁止 SPI 中断 1 = 允许 SPI 中断



7.8.4 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 INTx(x = 0,1,2)**时，如果中断为边沿触发，CPU 在响应中断后，标志 IEx(x = 0, 1, 2)被硬件清零；如果中断是电平触发，外部中断源直接控制中断标志，而不是由片上硬件控制。

当 TCON 寄存器的 TFx(x = 0, 1)中断标志位被置位时，**定时器 0/1 产生中断**，CPU 在响应中断后，标志被硬件自动清除。

当 T2CON 寄存器的 TF2 或 EXF2 标志位被置位时，**定时器 2 产生中断**，CPU 在响应中断后，两个标志位都不会被硬件自动清除。事实上，中断服务程序必须决定是由 TF2 或是 EXF2 产生中断，标志必须由软件清除。

当 SCON 寄存器的标志 RI 或 TI 被置位时，**EUARTx (x = 0, 1)产生中断**，CPU 在响应中断后，两个标志位都不会被硬件自动清除。事实上，中断服务程序必须决定是接收中断或是发送中断，标志必须由软件清除。

当 ADCON 寄存器的 ADCIF 标志位被置位时，**ADC 产生中断**。如果中断产生，ADCDH/ADCDL 中的修改结果是有效的。如果 ADC 模块的连续比较功能使能，那么每次转换时，ADCIF 标志位都将被置位。中断标志必须由软件清除。

当 SPSTA 寄存器的 SPIF 位或 MODF 位(当 SSDIS 清零)置位时，**SPI 产生中断**。标志必须由软件清除。

RTCCON 寄存器的 HSECIF 标志位被置位时，**Half-Second 产生中断**。标志必须由软件清除。

PWMC 寄存器的 PWMIF 标志位被置位时，**PWM 产生中断**。标志必须由软件清除。

TWICON 寄存器的 TWIIF 标志位被置位时，**TWI 产生中断**。标志必须由软件清除。



外部中断标记寄存器

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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
3,1	IE _x , 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	IT _x , x=0,1	外部中断 x 触发方式 0: 低电平触发 1: 下降沿触发

外部中断 2 标记寄存器

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	-	-	-	-	-	-	IE2	IT2
读/写	-	-	-	-	-	-	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	-	-	0	0

位编号	位符号	说明
1	IE2	外部中断 2 请求标志 0: 无中断挂起 1: 中断挂起
0	IT2	外部中断 2 触发方式 0: 低电平触发 1: 下降沿触发

7.8.5 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摘要表中详细列出。

7.8.6 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，分别通过清零或置位IPL0，IPH0，IPL1，IPH1中相应位来实现。但OVL NMI中断无需IPH/IPL控制，在所有中断源中享有最高优先级（除复位外）。中断优先级服务程序描述如下：

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

最高级中断服务程序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同时申请中断，那么一个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
IPH _x	IPL _x	
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)



中断优先级控制寄存器

	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 (B8H)	-	PADCL	PT2L	PS0L	PT1L	PX1L	PT0L	PX0L
IPH0 (B4H)	-	PADCH	PT2H	PS0H	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)	-	0	0	0	0	0	0	0

	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 (B9H)	-	PTWIL	PPWML	-	PHSECL	PS1L	PX2L	PSPIL
IPH1 (B5H)	-	PTWIH	PPWMH	-	PHSECH	PS1H	PX2H	PSPIH
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

位编号	位符号	说明
7-0	PxxxL/H	相应中断源 xxx 优先级选择

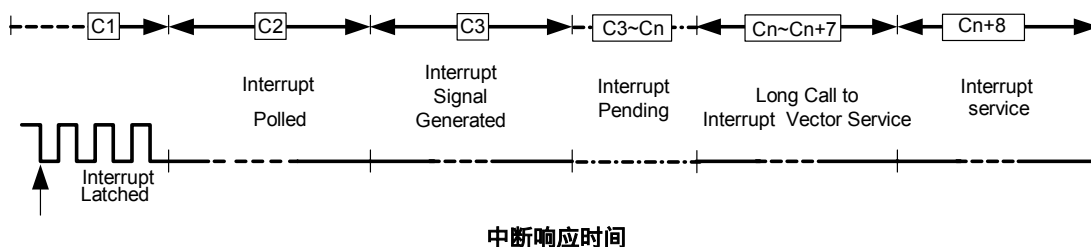
7.8.7 中断处理

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。如果一个标志被置位，那么 CPU 会发现而中断系统会产生一个 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：

- 1) 同级或更高级的优先级中断在运行中。
 - 2) 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。
 - 3) 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。这段时间保证 CPU 能观察到中断状态的变化。
- 注意：因为优先级的更改通常需要 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。

注意：如果当条件改变而标记不再有效，将不会进入此中断服务程序。如果当模块状态改变而标志不再有效时，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 (PC) 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存入程序计数器 (PC)。

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结束，然后把堆栈顶部两字节弹出，重载入程序计数器 (PC) 中，返回到进入中断服务程序之前程序点继续执行。特别注意的是 RETI 指令非常重要，它会通知处理器该优先级中断服务结束。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。



7.8.8 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，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。

当请求因上述的三个情况导致硬件调用 LCALL 指令受阻时，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，额外的等待时间不能超过 21 个机器周期，因为最长的指令 (DIV & MUL) 只有 20 个机器周期 (16 位乘除) 或 11 个机器周期 (8 位乘除)。如果正在执行 RETI 指令，则额外的等待时间不会超过 8+21 个机器周期 (为完成正在执行的 RETI 指令，还需要 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 16bit 操作数的 DIV, MUL 指令)。因而，若系统中只有一个中断源，包括 LCALL 调用指令 7 个机器周期，则响应时间一般大于 10 个机器周期小于 8+21+7 个机器周期。

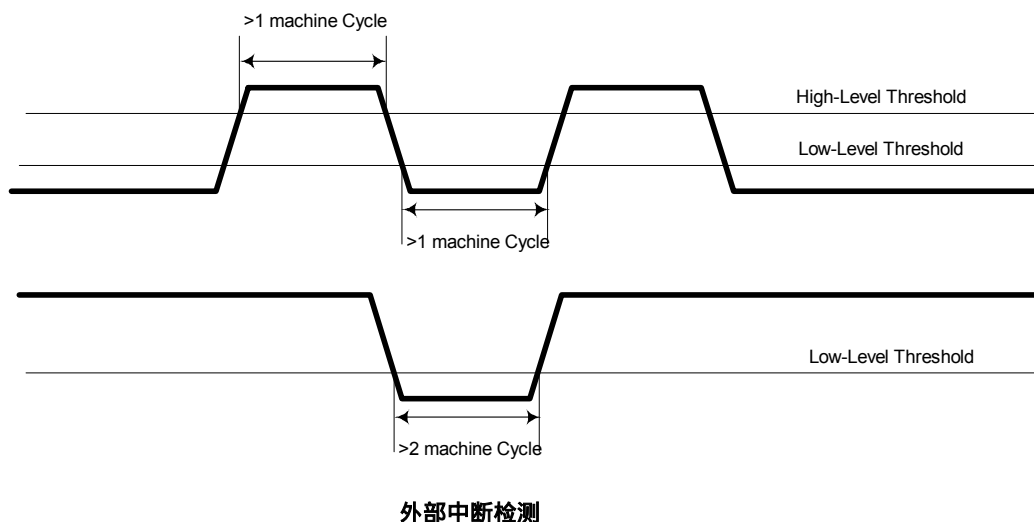
7.8.9 外部中断输入

SH79F32 有三个外部中断输入。这些外部中断可以通过设置 TCON 寄存器的 IT1, IT0 位和 EXF0 寄存器的 IT2 位来选择是电平触发或是边沿触发。当 $ITx=0(x=0,1,2)$ 时， $INTx(x=0,1,2)$ 脚为低电平，外部中断 x 触发。当 $ITx(x=0,1,2)=1$ ，外部中断 x 为沿触发。在这个模式中，一个周期内 $INTx$ 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起，发出一个中断请求。

由于外部中断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。如果外部中断为边沿触发，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 置位。当调用中断服务程序后，CPU 自动将 IEx 清零。

如果外部中断为电平触发，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 个机器周期。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 ，因为中断只与输入口电平有关。

如果外部中断被允许，当 SH79F32 进入空闲或掉电模式，产生中断后可以唤醒处理器恢复工作。详见电源管理章节。



**7.8.10 中断汇总**

中断源	向量地址	允许位	标记位	轮询优先级
复位	0000h			0（最高级）
INT0	0003h	EX0	IE0	2
Timer0	000Bh	ET0	TF0	3
INT1	0013h	EX1	IE1	4
Timer1	001Bh	ET1	TF1	5
EUART0	0023h	ES	RI+TI	6
Timer2	002Bh	ET2	TF2+EXF2	7
ADC	0033h	EADC	ADCIF	8
SPI	003Bh	ESPI	SPIF	9
INT2	0043h	EX2	TF2	10
EUART1	004Bh	ES1	RI1+TI1	11
HESC	0053h	EHSEC	HSECIF	12
PWM	0063h	EPWM	PWMIF	14
TWI	006Bh	ETWI	TWIIF	15
OVL NMI	007Bh	-	-	1



8 增强功能

8.1 LCD 驱动器

8.1.1 特性

- 两种工作模式：电容型 LCD 和 SLP(Super Low Power)型 LCD
- 电容型 LCD 模式
32个segment 输出 (S0-S31) 和6 个common 输出(COM0-COM5)

三种显示模式：

1/4 占空比和1/3 偏置电压

1/5 占空比和1/3 偏置电压

1/6 占空比和1/3 偏置电压

LCD 工作时钟和帧频都可以用软件选择，当 V_{DD} 电压高于 3.2V 时，LCD 工作电压为 2.9V，否则工作电压低于 2.9V。

- SLP 型 LCD 模式
只支持 1/4 占空比。其他与电容型一致

LCD 驱动器包含一个控制器，一个电压发生器，一个占空比发生器，及 4/5/6 COM 驱动器引脚和 32/31/30SEG 驱动器引脚。驱动器可编程为三种驱动模式：1/4 占空比和 1/3 偏置电压，1/5 占空比和 1/3 偏置电压，1/6 占空比和 1/3 偏置电压。驱动模式可通过 LCDCON 寄存器 DUTY[1:0]控制。LCD 显示数据存放于 1E0h - 1FFFh 的 32 字节 RAM 中，它也可用作普通的数据寄存器使用。LCD SEG17-32 可以共用为 P1 和 P0，通过 P0SS 和 P1SS 寄存器来选择。

SH79F32 包括 2 个完全不同的工作模式，可由 Flash 选项来选择。一个是传统电容型 LCD，使用端口 4 来连接 LCD 内部电源的外接电容。另一个是 SLP(Super Low Power)型 LCD，不需要 LCD 内部电源，因此端口 4 可作为常规 I/O 端口使用。但是 SLP LCD 只支持 1/4 占空比驱动模式。

当 SH79F32 进入省电模式后，LCD 被关闭，但 LCD RAM 仍然会保持数据。当 LCD 被关闭时，common 和 segment 都输出低电平。

为更有效的使用 LCD 驱动器，用户必须先设置 LCDCON 寄存器中除 PUMPON 和 LCDON 位之外的所有控制位，然后设置 PUMPON 位，在延迟之后打开 LCDON 位（延迟时间由应用决定），点亮 LCD 面板。

8.1.2 寄存器

LCD 控制寄存器

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
LCDCON	LCDON	-	PUMPF	PUMPON	P4L1	P4L0	DUTY1	DUTY0
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

位编号	位符号	说明
7	LCDON	LCD 打开/关闭开关 0: 禁止 LCD 驱动器 1: 允许 LCD 驱动器
5	PUMPF	PUMP 频率 0: LCDCLK/8 1: LCDCLK/4
4	PUMPON	PUMP 打开/关闭开 0: 禁止 LCD PUMP 1: 允许 LCD PUMP
3	P4L1	端口 4 共用电源配置 0: 共用作为 VP2-1, CUP2-1. 1: 共用作为 P4.0 – P4.3
2	P4L0	端口 4 共用 COM 0: P4.4~P4.7 作为 COM1-4 1: P4.4~P4.7 作为 I/O
1-0	DUTY [1:0]	占空比选择 00: 1/4 占空比, 帧频=LCD Clock / 512



SH79F32

		01: 1/5 占空比, 帧频=LCD Clock / 510 1x: 1/6 占空比, 帧频=LCD Clock / 510
--	--	--

LCD 时钟控制寄存器

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
LCDCLK	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
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	LCK [7:0]	LCD 时钟 0x00: LCD 时钟 = 通过代码选择的振荡器频率 其他: LCD 时钟 = 通过代码选择的振荡器频率 / LCDCLK

P1 口模式选择寄存器

ADH	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
P1SS	P1S7	P1S6	P1S5	P1S4	P1S3	P1S2	P1S1	P1S0
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	P1Sx (x=0-7)	P1 口模式选择 0: 作为 P1.x 1: 作为 Segment (SEG17 - SEG24)

P0 口模式选择寄存器

AEH	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
P0SS	P0S7	P0S6	P0S5	P0S4	P0S3	P0S2	P0S1	P0S0
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	P0Sx(x=0-7)	P0 口模式选择 0: 作为 P0.x 1: 作为 Segment (SEG25 - SEG32)

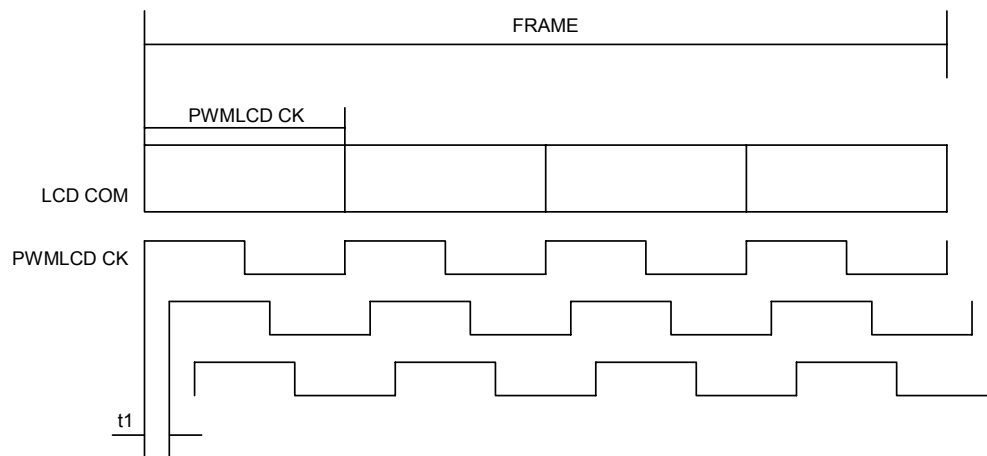


SH79F32

SLP LCD 控制寄存器

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
PLCON	-	-	-	PLD4	PLD3	PLD2	PLD1	PLD0
读/写	-	-	-	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	0	0	0	0	0

位编号	位符号	说明
4-0	PLD [4:0]	SLP LCD 占空比选择 SLP LCD 时钟源 $F_{pl} = \frac{LCDCLK}{128}$ 帧频 : $F_{frame} = \frac{LCDCLK}{128 \times 4}$ $t1 = \frac{PLD[4:0] + 1}{F_{pl} \times 2 \times 2 \times 32}$



SLP LCD 模式波形



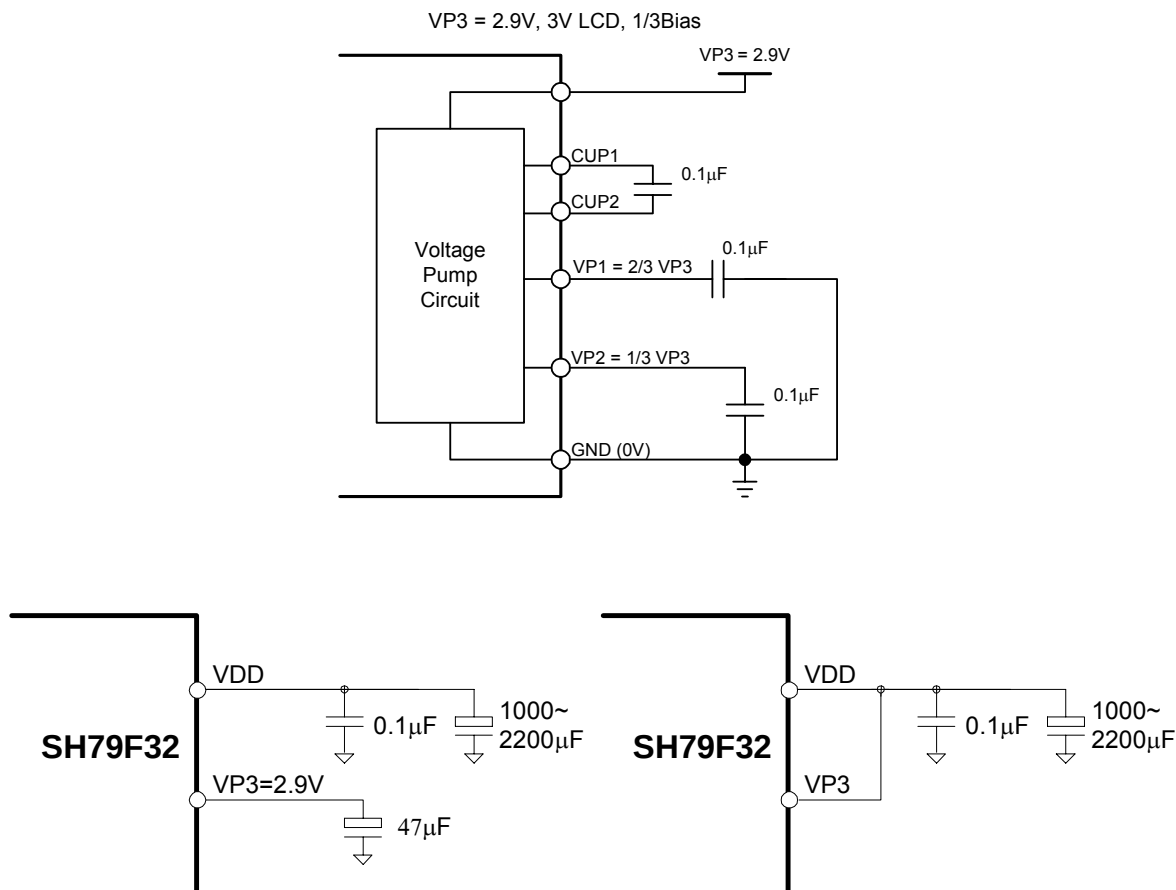
8.1.3 LCD 电源

SH79F32 内建一个稳压源给 LCD 供电。

如果 V_{DD} 超过 3.2V，内部稳压源会产生稳定电压(2.9V)。

如果 V_{DD} 低于 3.2V，内部稳压源输出低于 2.9V，此时 LCD 不能显示在最佳状态。

下图所示即为 LCD 电源电路在传统布局下的应用电路图。



$V_{DD}=3.6V\sim5.5V$ 应用

a. 设置代码选项：

设置 OP_LVREN, OP_LVRLE

=00 或 01 或 11 (参考 8.14 节)；

b. VP3、 V_{DD} 管脚不短接；

c. LCD 电源 = 2.9V。

$V_{DD} < 3.6V$ 应用

a. 设置代码选项：

设置 OP_LVREN/LVRLE=10

(参考 8.14 节)；

b. VP3、 V_{DD} 管脚短接；

c. LCD 电源 = V_{DD}

**8.1.4 LCD RAM 配置**

LCD 1/4 占空比, 1/3 偏置(COM1 - 4, SEG1 - 32)

地址	7	6	5	4	3	2	1	0
	-	-	-	-	COM4	COM3	COM2	COM1
\$1E0h	-	-	-	-	SEG1	SEG1	SEG1	SEG1
\$1E1h	-	-	-	-	SEG2	SEG2	SEG2	SEG2
\$1E2h	-	-	-	-	SEG3	SEG3	SEG3	SEG3
\$1E3h	-	-	-	-	SEG4	SEG4	SEG4	SEG4
\$1E4h	-	-	-	-	SEG5	SEG5	SEG5	SEG5
\$1E5h	-	-	-	-	SEG6	SEG6	SEG6	SEG6
\$1E6h	-	-	-	-	SEG7	SEG7	SEG7	SEG7
\$1E7h	-	-	-	-	SEG8	SEG8	SEG8	SEG8
\$1E8h	-	-	-	-	SEG9	SEG9	SEG9	SEG9
\$1E9h	-	-	-	-	SEG10	SEG10	SEG10	SEG10
\$1EAh	-	-	-	-	SEG11	SEG11	SEG11	SEG11
\$1EBh	-	-	-	-	SEG12	SEG12	SEG12	SEG12
\$1ECh	-	-	-	-	SEG13	SEG13	SEG13	SEG13
\$1EDh	-	-	-	-	SEG14	SEG14	SEG14	SEG14
\$1EEh	-	-	-	-	SEG15	SEG15	SEG15	SEG15
\$1EFh	-	-	-	-	SEG16	SEG16	SEG16	SEG16
\$1F0h	-	-	-	-	SEG17	SEG17	SEG17	SEG17
\$1F1h	-	-	-	-	SEG18	SEG18	SEG18	SEG18
\$1F2h	-	-	-	-	SEG19	SEG19	SEG19	SEG19
\$1F3h	-	-	-	-	SEG20	SEG20	SEG20	SEG20
\$1F4h	-	-	-	-	SEG21	SEG21	SEG21	SEG21
\$1F5h	-	-	-	-	SEG22	SEG22	SEG22	SEG22
\$1F6h	-	-	-	-	SEG23	SEG23	SEG23	SEG23
\$1F7h	-	-	-	-	SEG24	SEG24	SEG24	SEG24
\$1F8h	-	-	-	-	SEG25	SEG25	SEG25	SEG25
\$1F9h	-	-	-	-	SEG26	SEG26	SEG26	SEG26
\$1FAh	-	-	-	-	SEG27	SEG27	SEG27	SEG27
\$1FBh	-	-	-	-	SEG28	SEG28	SEG28	SEG28
\$1FCh	-	-	-	-	SEG29	SEG29	SEG29	SEG29
\$1FDh	-	-	-	-	SEG30	SEG30	SEG30	SEG30
\$1FEh	-	-	-	-	SEG31	SEG31	SEG31	SEG31
\$1FFh	-	-	-	-	SEG32	SEG32	SEG32	SEG32



LCD 1/5 占空比，1/3 偏置 (COM1 - 5, SEG2 - 32)

地址	7	6	5	4	3	2	1	0
	-	-	-	COM5	COM4	COM3	COM2	COM1
\$1E0h	-	-	-	-	-	-	-	-
\$1E1h	-	-	-	SEG2	SEG2	SEG2	SEG2	SEG2
\$1E2h	-	-	-	SEG3	SEG3	SEG3	SEG3	SEG3
\$1E3h	-	-	-	SEG4	SEG4	SEG4	SEG4	SEG4
\$1E4h	-	-	-	SEG5	SEG5	SEG5	SEG5	SEG5
\$1E5h	-	-	-	SEG6	SEG6	SEG6	SEG6	SEG6
\$1E6h	-	-	-	SEG7	SEG7	SEG7	SEG7	SEG7
\$1E7h	-	-	-	SEG8	SEG8	SEG8	SEG8	SEG8
\$1E8h	-	-	-	SEG9	SEG9	SEG9	SEG9	SEG9
\$1E9h	-	-	-	SEG10	SEG10	SEG10	SEG10	SEG10
\$1EAh	-	-	-	SEG11	SEG11	SEG11	SEG11	SEG11
\$1EBh	-	-	-	SEG12	SEG12	SEG12	SEG12	SEG12
\$1ECh	-	-	-	SEG13	SEG13	SEG13	SEG13	SEG13
\$1EDh	-	-	-	SEG14	SEG14	SEG14	SEG14	SEG14
\$1EEh	-	-	-	SEG15	SEG15	SEG15	SEG15	SEG15
\$1EFh	-	-	-	SEG16	SEG16	SEG16	SEG16	SEG16
\$1F0h	-	-	-	SEG17	SEG17	SEG17	SEG17	SEG17
\$1F1h	-	-	-	SEG18	SEG18	SEG18	SEG18	SEG18
\$1F2h	-	-	-	SEG19	SEG19	SEG19	SEG19	SEG19
\$1F3h	-	-	-	SEG20	SEG20	SEG20	SEG20	SEG20
\$1F4h	-	-	-	SEG21	SEG21	SEG21	SEG21	SEG21
\$1F5h	-	-	-	SEG22	SEG22	SEG22	SEG22	SEG22
\$1F6h	-	-	-	SEG23	SEG23	SEG23	SEG23	SEG23
\$1F7h	-	-	-	SEG24	SEG24	SEG24	SEG24	SEG24
\$1F8h	-	-	-	SEG25	SEG25	SEG25	SEG25	SEG25
\$1F9h	-	-	-	SEG26	SEG26	SEG26	SEG26	SEG26
\$1FAh	-	-	-	SEG27	SEG27	SEG27	SEG27	SEG27
\$1FBh	-	-	-	SEG28	SEG28	SEG28	SEG28	SEG28
\$1FCh	-	-	-	SEG29	SEG29	SEG29	SEG29	SEG29
\$1FDh	-	-	-	SEG30	SEG30	SEG30	SEG30	SEG30
\$1FEh	-	-	-	SEG31	SEG31	SEG31	SEG31	SEG31
\$1FFh	-	-	-	SEG32	SEG32	SEG32	SEG32	SEG32



LCD 1/6 占空比，1/3 偏置 (COM1 - 6, SEG3 - 32)

地址	7	6	5	4	3	2	1	0
	-	-	COM6	COM5	COM4	COM3	COM2	COM1
\$1E0h	-	-	-	-	-	-	-	-
\$1E1h	-	-	-	-	-	-	-	-
\$1E2h	-	-	SEG3	SEG3	SEG3	SEG3	SEG3	SEG3
\$1E3h	-	-	SEG4	SEG4	SEG4	SEG4	SEG4	SEG4
\$1E4h	-	-	SEG5	SEG5	SEG5	SEG5	SEG5	SEG5
\$1E5h	-	-	SEG6	SEG6	SEG6	SEG6	SEG6	SEG6
\$1E6h	-	-	SEG7	SEG7	SEG7	SEG7	SEG7	SEG7
\$1E7h	-	-	SEG8	SEG8	SEG8	SEG8	SEG8	SEG8
\$1E8h	-	-	SEG9	SEG9	SEG9	SEG9	SEG9	SEG9
\$1E9h	-	-	SEG10	SEG10	SEG10	SEG10	SEG10	SEG10
\$1EAh	-	-	SEG11	SEG11	SEG11	SEG11	SEG11	SEG11
\$1EBh	-	-	SEG12	SEG12	SEG12	SEG12	SEG12	SEG12
\$1ECh	-	-	SEG13	SEG13	SEG13	SEG13	SEG13	SEG13
\$1EDh	-	-	SEG14	SEG14	SEG14	SEG14	SEG14	SEG14
\$1EEh	-	-	SEG15	SEG15	SEG15	SEG15	SEG15	SEG15
\$1EFh	-	-	SEG16	SEG16	SEG16	SEG16	SEG16	SEG16
\$1F0h	-	-	SEG17	SEG17	SEG17	SEG17	SEG17	SEG17
\$1F1h	-	-	SEG18	SEG18	SEG18	SEG18	SEG18	SEG18
\$1F2h	-	-	SEG19	SEG19	SEG19	SEG19	SEG19	SEG19
\$1F3h	-	-	SEG20	SEG20	SEG20	SEG20	SEG20	SEG20
\$1F4h	-	-	SEG21	SEG21	SEG21	SEG21	SEG21	SEG21
\$1F5h	-	-	SEG22	SEG22	SEG22	SEG22	SEG22	SEG22
\$1F6h	-	-	SEG23	SEG23	SEG23	SEG23	SEG23	SEG23
\$1F7h	-	-	SEG24	SEG24	SEG24	SEG24	SEG24	SEG24
\$1F8h	-	-	SEG25	SEG25	SEG25	SEG25	SEG25	SEG25
\$1F9h	-	-	SEG26	SEG26	SEG26	SEG26	SEG26	SEG26
\$1FAh	-	-	SEG27	SEG27	SEG27	SEG27	SEG27	SEG27
\$1FBh	-	-	SEG28	SEG28	SEG28	SEG28	SEG28	SEG28
\$1FCh	-	-	SEG29	SEG29	SEG29	SEG29	SEG29	SEG29
\$1FDh	-	-	SEG30	SEG30	SEG30	SEG30	SEG30	SEG30
\$1FEh	-	-	SEG31	SEG31	SEG31	SEG31	SEG31	SEG31
\$1FFh	-	-	SEG32	SEG32	SEG32	SEG32	SEG32	SEG32



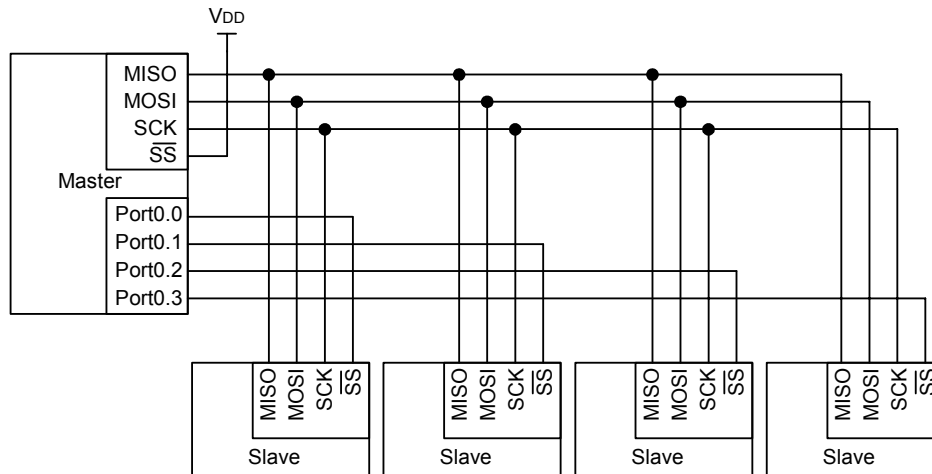
8.2 SPI（串行外部设备接口）控制器

串行外部设备接口(简称 SPI)是一种高速串行通信接口，允许 MCU 与外围设备包括其它 MCU 进行全双工，同步串行通讯。

8.2.1 特征：

- 全双工，三线同步传输
- 主从机操作
- 6 个可编程主时钟频率
- 极性相位可编程的串行时钟
- 带 MCU 中断的主模式故障出错标记
- 写入冲突标记保护
- 可选择 LSB 或 MSB 传输

下图所示即为典型的 SPI 总线配置，通过 3 条线连接所有设备，包括一个主设备和若干从属外部设备。主设备通过控制连接从属设备 SS 引脚的 4 个并行端口选中专用的从属设备。



8.2.2 信号描述

主输出从输入 (MOSI)

该路信号连接主设备和一个从设备。数据通过 MOSI 从主设备串行传送到从设备，主设备输出，从设备输入。

主输入从输出(MISO)

该路信号连接从设备和主设备。数据通过 MISO 从从设备串行传送到主设备，从设备输出，主设备输入。当 SPI 配置为从设备并未被选中(SS高)，MISO 引脚处于高阻状态。

在 SS 引脚上的静态高电平使从设备的 MISO 引脚处于高阻状态。

SPI 串行时钟(SCK)

SCK 信号用作控制 MOSI 和 MISO 线上输入输出数据的同步移动。每 8 时钟周期线上传送一个字节。如果从设备未被选中 (SS = 1)，SCK 信号被此从设备忽略。

从选择 (SS)

每个从属外围设备由一个从选择引脚(SS)选择，对于被选中的从设备，这个信号必须处于低电平。很明显，只有一个主设备可以驱动通讯网络。主设备可以通过软件控制端口电平选择每个从设备。为了防止 MISO 总线冲突，同一时间只允许一个从设备与主设备通讯。在主设备模式中，SS 引脚可以用来关联 SPI 状态寄存器中 MODF 标志位，防止多个主设备驱动 MOSI 和 SCK。

下列情况，SS 引脚可以作为其它功能使用：

- 1) 设备作为主设备，SPCON 寄存器的 SSDIS 控制位置 1。这种配置仅仅存在于通讯网络中只有一个主设备的情况，因此，SPI 状态寄存器中 MODF 标志位不会被置 1。
- 2) 设备配置为从设备，SPI 控制寄存器的 CPHA 位和 SSDIS 位置 1。这种配置情况存在于只有一个主设备一个从设备的通



讯网络中，因此，设备总是被选中的，主设备也不需要控制从设备的 $\overline{SS}$ 引脚选择其作为通讯目标。

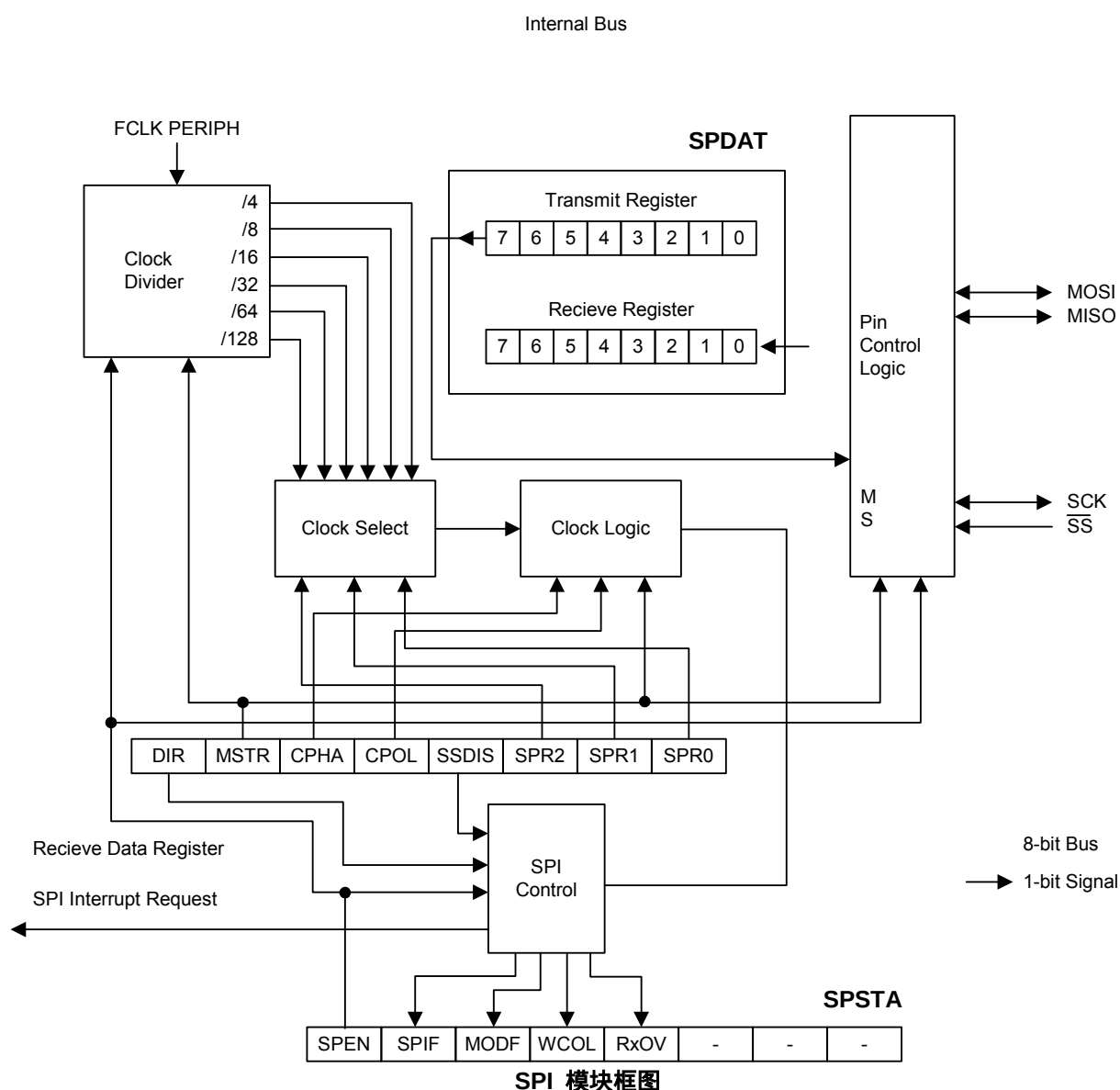
注意：当 $CPHA = 0$ 时， $\overline{SS}$ 引脚电平拉低表示启动发送。

8.2.3 波特率

SPI 的波特率有六种可选择的频率，分别是内部时钟的 4, 8, 16, 32, 64, 或 128 分频。设备在主模式下，通过设定 SPCON 寄存器的 SPR[2:0] 进行选择。

8.2.4 功能描述

下图所示是 SPI 模块的详细结构。



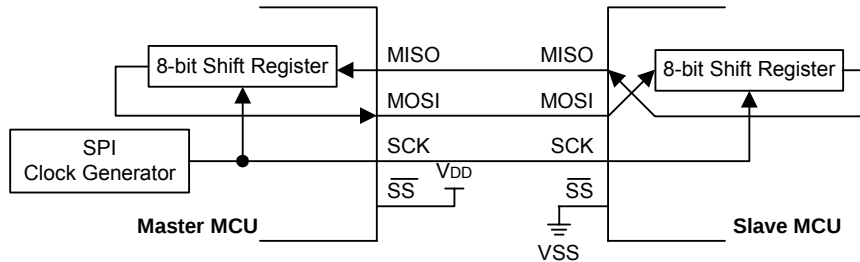
8.2.5 工作模式

串行外围设备可配置为主模式或从属模式中的一种。SPI 模块的配置和初始化通过设置 SPCON 寄存器（串行外围设备控制寄存器）和 SPSTA（串行外围设备状态寄存器）来完成。配置完成后，通过设置 SPCON，SPSTA，SPDAT（串行外围设备数据寄存器）来完成数据传送。



在 SPI 通讯期间，数据同步地被串行的移进移出。串行时钟线（SCK）使两条串行数据线(MOSI 和 MISO)上数据的移动和采样保持同步。从属选择线（SS）可以独立地选择 SPI 从属设备；如果 SPI 从设备不被选中，它就不能参与 SPI 总线上的活动。

当 SPI 主设备通过 MOSI 线传送数据到从设备时，从设备通过 MISO 线发送数据到主设备作为响应，这就实现了在同一时钟下数据发送和接收的同步全双工传输。发送移位寄存器和接收移位寄存器使用相同的 SFR 地址，对 SPDATA 进行写操作将写入发送移位寄存器，对 SPDATA 进行读操作将获得接收移位寄存器的数据。



全双工主从互联图

主模式：

允许：SPI 主设备控制 SPI 总线上所有数据传送的启动。当 SPCON 寄存器中的 MSTR 位置 1 时，SPI 在主模式下运行。只有一个主设备可以启动传送。

发送：在 SPI 主模式下，写一个字节数据到 SPI 数据寄存器（SPDAT），数据将会写入发送移位缓冲器。如果发送移位寄存器已经存在一个数据，那么主 SPI 产生一个 WCOL 信号以表明写入太快。但是在发送移位寄存器中的数据不会受到影响，发送也不会中断。另外如果发送移位寄存器为空，那么主 SPI 立即按照 SCK 上的 SPI 时钟频率串行地移出发送移位寄存器中的数据到 MOSI 线上。当传送完毕，SPSTA 寄存器中的 SPIF 位被置 1。如果 SPI 中断被允许，当 SPIF 置 1 时，也会产生一个中断。

接收：当主设备通过 MOSI 线传送数据给从设备时，相对应的从设备同时也通过 MISO 线将其发送移位寄存器的内容传送给主设备的接收移位寄存器，实现全双工操作。因此，SPIF 标志位置 1 即表示传送完成也表示接收数据完毕。从设备接收的数据按照 MSB 优先的传送方向存入主设备的接收移位寄存器。当一个字节的数据完全被移入接收寄存器时，处理器可以通过读 SPDAT 寄存器获得该数据。如果发生超限（SPIF 标志未被清零，就试图开始下一次传送），RXOV 置 1，表示发生数据超限，此时接收移位寄存器保持原有数据并且 SPIF 位置 1，这样 SPI 主设备将不会接收任何数据直到 SPIF 清零。

从模式

允许：当 SPCON 寄存器中的 MSTR 位清零，SPI 在从模式下运行。在数据传送之前，从设备的 SS 引脚必须被置低，而且必须保持低电平直到一个字节数据传送完毕。

发送与接收：从属模式下，按照主设备控制的 SCK 信号，数据通过 MOSI 引脚移入，MISO 引脚移出。一个位计数器记录 SCK 的边沿数，当接收移位寄存器移入 8 位数据（一个字节）同时发送移位寄存器移出 8 位数据（一个字节），SPIF 标志位被置 1。数据可以通过读取 SPDAT 寄存器获得。如果 SPI 中断被允许，当 SPIF 置 1 时，也会产生一个中断。

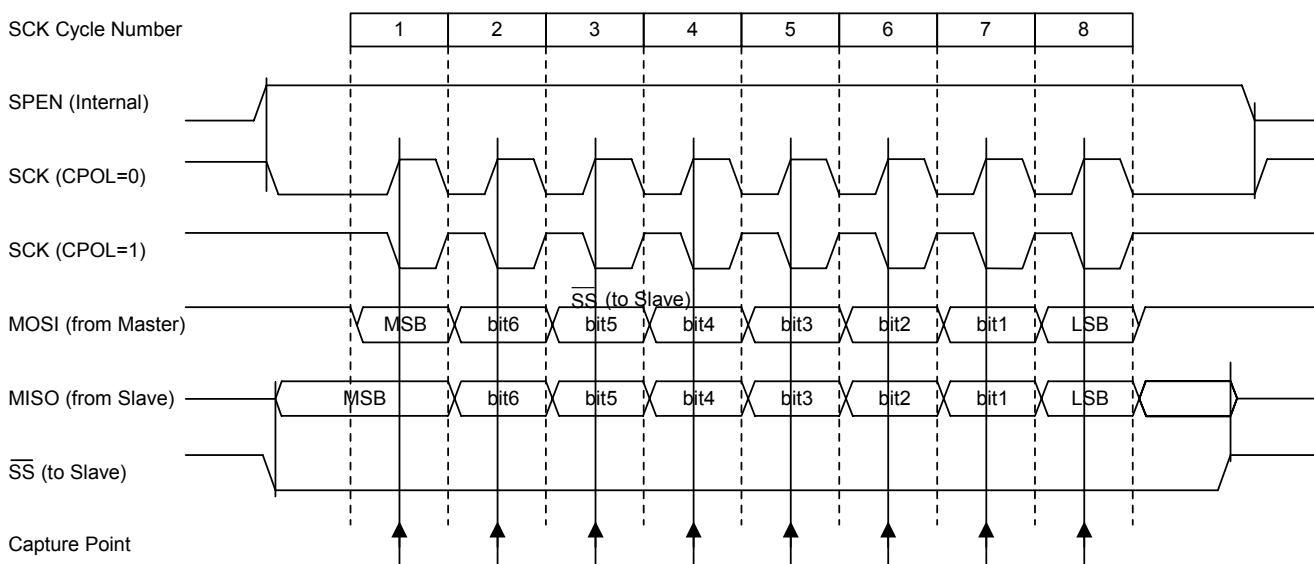
为防止超限，SPI 从设备在向移位寄存器写入数据之前必须先读取 SPDAT 寄存器，软件清除 SPIF 标志，否则 RXOV 置 1，表示发生数据超限，此时接收移位寄存器保持原有数据并且 SPIF 位置 1，这样 SPI 从设备将不会接收任何数据直到 SPIF 清零。

SPI 从设备不能启动数据传送，所以 SPI 从设备必须在主设备开始一次新的数据传送之前将要传送的数据写入发送移位寄存器。如果在开始发送之前未写入数据，从设备将传送“00”字节给主设备。如果写 SPDAT 操作发生在传送过程中，那么 SPI 从设备的 WCOL 标志位置 1，即如果传送移位寄存器已经含有数据，SPI 从设备的 WCOL 标志位置 1，表示写 SPDAT 冲突。但是移位寄存器的数据不受影响，传送也不会被中断。



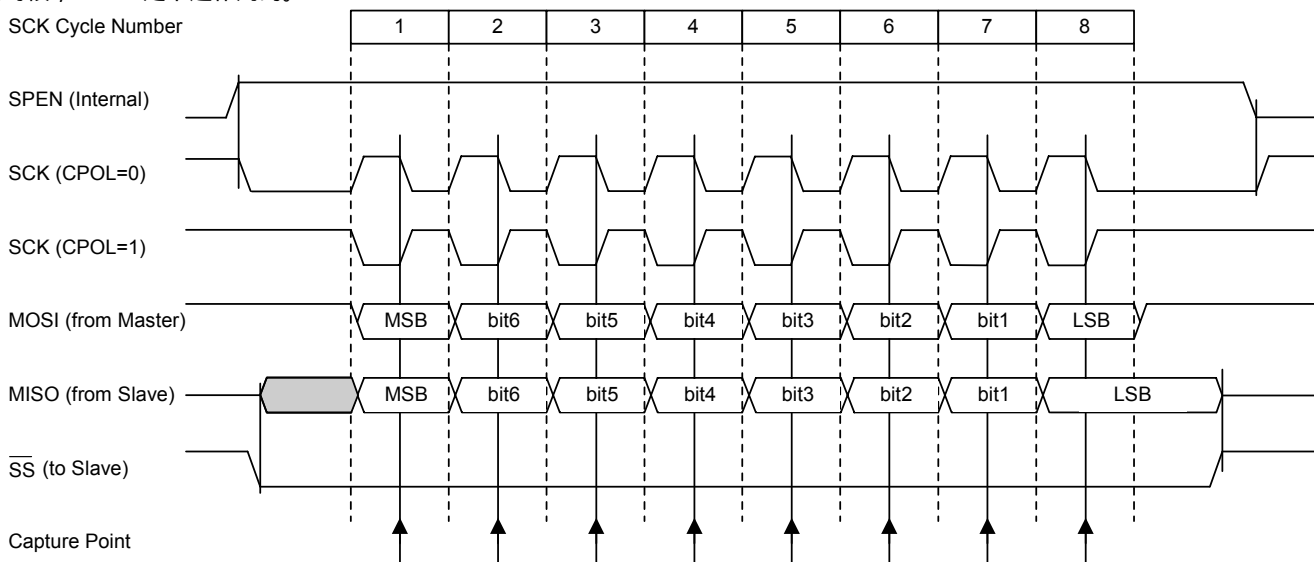
传送形式

通过软件设置 SPCON 寄存器的 CPOL 位和 CPHA 位，可以选择 SPI 时钟极性和相位的四种组合方式。CPOL 位定义时钟的极性，即空闲时的电平状态，它对 SPI 传输格式影响不大。CPHA 定义时钟的相位，即定义允许数据采样移位的时钟边沿。在主从通讯的两个设备中，时钟极性相位的设置应一致。



数据传送形式 (CPHA = 0)

如果 CPHA = 0，SCK 的第一个沿捕获数据，从设备必须在 SCK 的第一个沿之前将数据准备好，因此， $\overline{SS}$ 引脚的下降沿从设备开始发送数据。 $\overline{SS}$ 引脚在每次传送完一个字节之后必须被拉高，在发送下一个字节之前重新设置为低电平，所以 CPHA=0 的时候，SSDIS 是不起作用的。

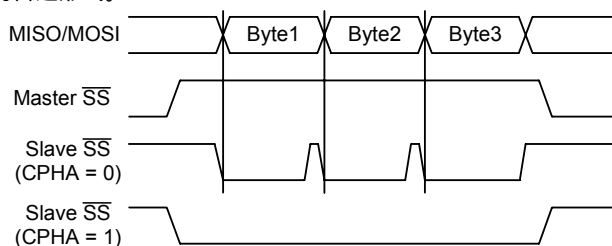


数据发送形式 (CPHA = 1)

如果 CPHA=1，主设备在 SCK 的第一个沿将数据输出到 MOSI 线上，从设备把 SCK 的第一个沿作为开始发送信号。用户必须在第一个 SCK 的两个沿内完成写 SPDAT 的操作。 $\overline{SS}$ 引脚在每个字节数据的传送过程始终保持低电平。这种数据传输形式是



一个主设备一个从设备之间通信的首选形式。



CPHA / SS 时序

注意：当 SPI 配置为从机模式，SPCON 寄存器的 CPOL 位为 0 时，在 SPSTA 寄存器中的 SPEN 位被置 1 之前，SCK 引脚必须被设置为输入状态并且上拉打开

8.2.6 出错检测

SPSTA 寄存器中的标志位表示在 SPI 通讯中的出错情况：

模式故障(MODF): SPI 主模式下的模式故障出错表明 SS 引脚上的电平状态与实际的设备模式不一致。MODF 置 1 后，表明系统控制存在多主设备冲突的问题。这种情况下，SPI 系统受到如下影响：

产生 SPI 接收/错误 CPU 中断请求；

SPSTA 寄存器的 SPEN 位清零，SPI 被禁止；

SPCON 寄存器的 MSTR 位清零。

当 SPCON 寄存器的 SS 引脚禁止位(SSDIS)清零，SS 引脚信号为 0 时，MODF 标志位置 1。然而，对于只有一个主设备的系统来说，主设备的 SS 引脚被拉低，那决不是另外一个主设备试图驱动网络。这种情况下，为防止 MODF 置 1，可使 SPCON 寄存器中的 SSDIS 位置 1，SS 引脚作为普通 I/O 口或是 FLT。

重新启动串行通信时，用户必须软件清零 MODF 位，置位 SPCON 寄存器中的 MSTR 位和 SPSTA 寄存器的 SPEN 位，重新启动主模式。

写冲突 (WCOL): 在发送数据序列期间写入 SPDAT 寄存器会引起的写冲突，SPSTA 中 WCOL 位置 1。WCOL 置 1 不会引起中断，发送也不会中止。WCOL 位需由软件清零。

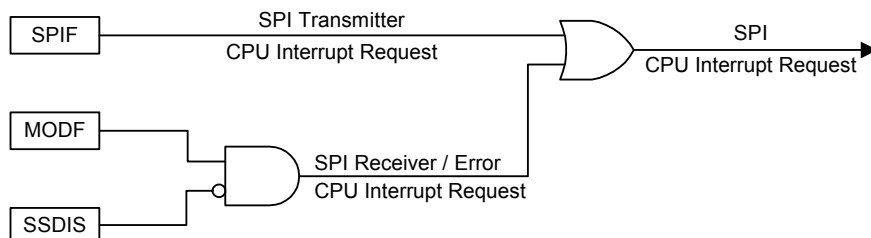
超限情况(RXOV):主设备或从设备尚未清除 SPIF 位时，主或从设备试图发送几个数据字节时，超限情况发生。在这种情况下，接收移位寄存器保持原有数据，SPIF 置 1，同样 SPI 设备直到 SPIF 被清除后才会再接收数据。在 SPIF 位被清除之前继续调用中断，但是还是由 SCK 驱动发送。RXOV 不会引起中断，RXOV 位由软件清除。

8.2.7 中断

两种 SPI 状态标志 SPIF & MODF 能产生一个 CPU 中断请求。

串行外围设备数据发送标志，SPIF：完成一个发送后由硬件置 1。

模式故障标志，MODF：该位被置 1 表示 SS 引脚上的电平与 SPI 模式不一致的。SSDIS 位为 0 并且 MODF 置 1 将产生 SPI 接收器/出错 CPU 中断请求。当 SSDIS 置 1 时，无 MODF 中断请求产生。



SPI 中断请求



8.2.8 寄存器

串联外部设备控制寄存器

A2H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
SPCON	DIR	MSTR	CPHA	CPOL	SSDIS	SPR2	SPR1	SPR0
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	DIR	传送方向选择 5 0 = MSB 优先 1 = LSB 优先
6	MSTR	主串行外部设备 0 = 配置 SPI 作为从属设备 1 = 配置 SPI 作为主设备
5	CPHA	时钟相位 0 = SCK 周期的第一沿采集数据 1 = SCK 周期的第二沿采集数据
4	CPOL	时钟极性 0 = 在 idle 状态下 SCK 处于低电平 1 = 在 idle 状态下 SCK 处于高电平
3	SSDIS	SS引脚控制位 0 = 在主和从属模式下，打开SS引脚 1 = 在主和从属模式下，关闭SS引脚 如果 SSDIS 置 1，不产生 MODF 中断请求。 在从属模式下，如果 CPHA = 0，该位不起作用。
2-0	SPR [2:0]	串行外部设备比率 0 0 0 = 系统时钟 /4 (默认) 0 0 1 = 系统时钟/8 0 1 0 = 系统时钟 /16 0 1 1 = 系统时钟 /32 1 0 0 = 系统时钟/64 其他 = 系统时钟 /128



SPI 状态寄存器

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
SPSTA	SPEN	SPIF	MODF	WCOL	RXOV	-	-	-
读/写	读/写	读/写	读/写	读/写	读/写	-	-	-
复位值 (POR/WDT/LVR)	0	0	0	0	0	-	-	-

位编号	位符号	说明
7	SPEN	SPI 控制位 0 = 关闭 SPI 1 = 打开 SPI 接口。
6	SPIF	串行外部设备数据传送标志 0 = 由软件清除 1 = 表明已完成数据传输, 由硬件置 1
5	MODF	故障模式 0 = 由软件清除 1 = 表明 SS 引脚在不正确的逻辑电平下, 由硬件置 1
4	WCOL	写入冲突标志 0 = 有软件清除, 表明已处理写入冲突 1 = 由硬件置 1, 表明检测到一个冲突
3	RXOV	接收超限 0 = 表明已处理接收超限, 由软件清除 1 = 表明已检测到接收超限, 由硬件置 1

SPI 数据寄存器

A4H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
SPDAT	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
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	SPD7-0	写入 SPDAT 的数据被放置到发送移位寄存器中。读取 SPDAT 中的值将获得接收移位寄存器的数据。

注意：当 SPI 功能禁止时，SPDAT 寄存器中的数据无效。因此当数据传输结束时，必须先读出数据再清除 SPEN。



8.3 两线串行接口 (TWI)

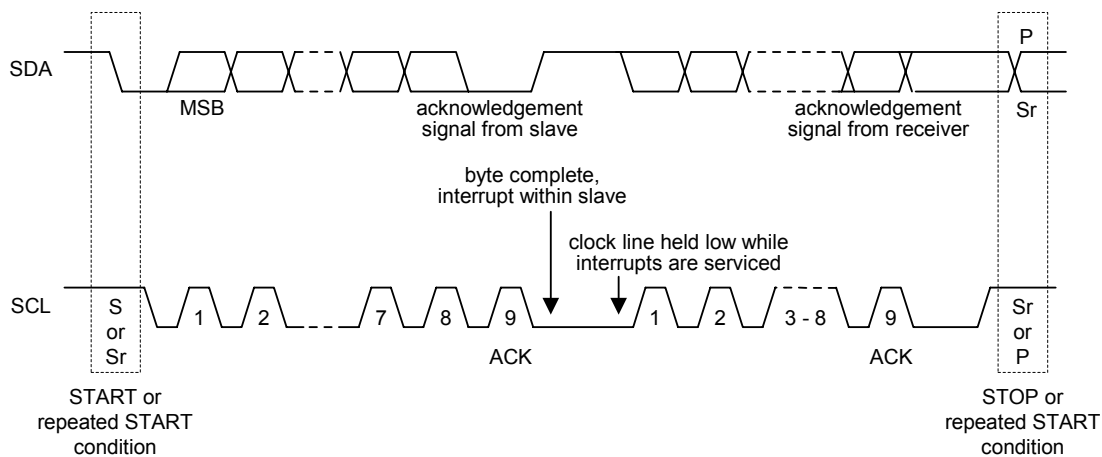
8.3.1 特性

- 只支持主模式
- 数据传输速率可选择

主机到从机的数据传输：首先主机发送一个控制字节，从机返回一个“应答位 (ACK)”。接着数据字节由主机传输到从机中，从机在每一个字节接收完毕后返回一个“应答位”。

从机到主机的数据传输：首先主机发送一个控制字节，从机返回一个“应答位”。接着数据字节由从机传输到主机中，主机在每一个字节接收完毕后返回一个“应答位”。主机在最后一个字节接收完毕后返回一个“负应答位 (NACK)”。“ACK”或“NACK”信号由 TWICON 寄存器中的 ACK 位控制。

START 和 STOP 信号由主机产生，一次数据传输以一个 STOP 信号或一个重复 START 信号中止，一个重复 START 信号也是从下一次数据传输开始的，TWI 总线不被释放。



TWI 总线上的数据传输

8.3.2 TWI 工作模式

SH79F32 的 TWI 仅支持主模式，主模式又包括主发送模式和主接收模式。TWIE 的 SCL 时钟频率可达 400kHz。

主发送模式

在主发送模式中，数据传输由主机到从机，流程如下：

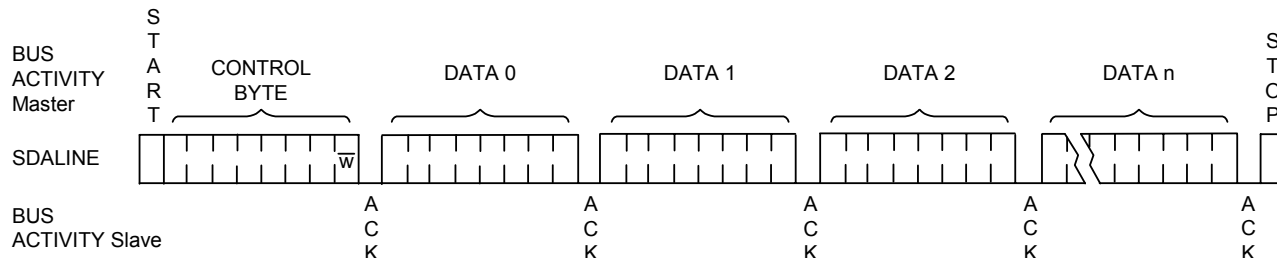
- 1) 初始 TWI
 - a. 通过设置 BR[1:0]，设置 TWI 波特率
 - b. 通过设置 ETWI = 1，使能 TWI 中断，且清除 TWIIF
 - c. 通过设置 TWIEN = 1，使能 TWI
 - d. 设置 STA = 1 来启动 TWI，产生一个 START 条件
- 2) 当发送一个 START 条件时，TWIIF 被设置且产生 TWI 中断，软件必须首先写控制字节到 TWIDAT 中，然后清除 TWIIF 标志以允许下一次中断。在控制字节发送之后，STA 会被清除。
- 3) 当从机收到控制字节后会返回 ACK 或 NACK 信号，第 2 个中断产生。

如果 ACK = 0，表明通信出错，软件必须设置 STA = 1 以开始第二次通信，或设置 STO = 1 以中止通信。

如果 ACK = 1，表明通信成功，软件必须为下一次发送新字节到 TWIDAT 寄存器做准备。然后退出中断服务程序，同时 TWIIF 被清除。



- 4) 通信重复上述 3 个步骤直到所有字节都被发送完毕，然后主发送器必须设置 $STO = 1$ 来中止发送。如果总线中止，硬件清除 STO 。



主接收模式

在主接收模式中，由从发送器接收到数据字节的数字。在重复 START 条件后，TWI 引擎转换到主发送器模式。流程如下：

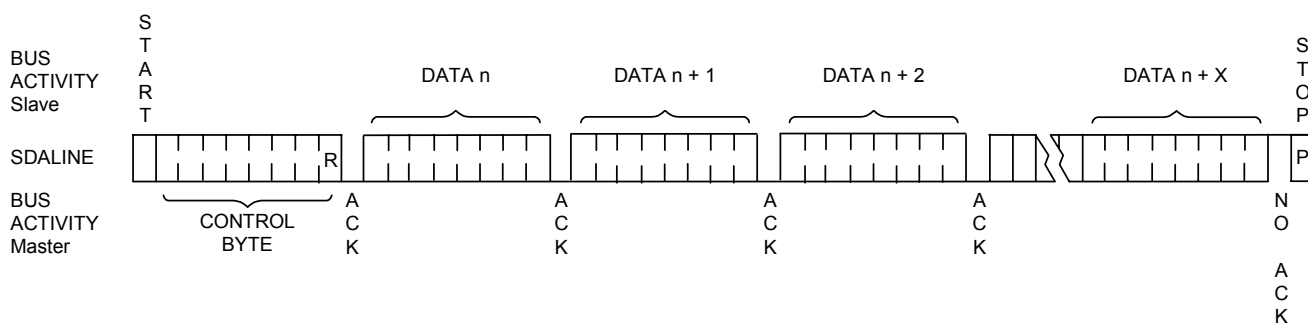
- 1) 初始 TWI
 - a. 通过设置，设置 TWI 波特率
 - b. 通过设置 $ETWI = 1$ ，使能 TWI 中断，且清除 $TWIF$
 - c. 通过设置 $TWIEN = 1$ ，使能 TWI
 - d. 设置 $STA = 1$ 来启动 TWI，产生一个 START 条件
- 2) 当发送一个 START 条件时， $TWIF$ 被设置且产生 TWI 中断，软件必须首先写控制字节到 $TWIDAT$ 中，然后清除 $TWIF$ 标记以允许下一次中断。在控制字节发送之后， STA 会被清除。
- 3) 从机接收到控制字节发出 ACK 信号，主机产生第 2 个中断。

如果 $ACK = 0$ ，表明通信出错，软件必须设置 $STA = 1$ 以开始第二次通信，或设置 $STO = 1$ 以中止通信。

如果 $ACK = 1$ ，表明通信成功，软件必须为再一次发送新的字节到 $TWIDAT$ 寄存器做准备。然后退出中断服务程序，同时 $TWIF$ 被清除。
- 4) 当接收到新的数据时，会产生中断

如果 $RVOK = 0$ ，表明没有接收到数据

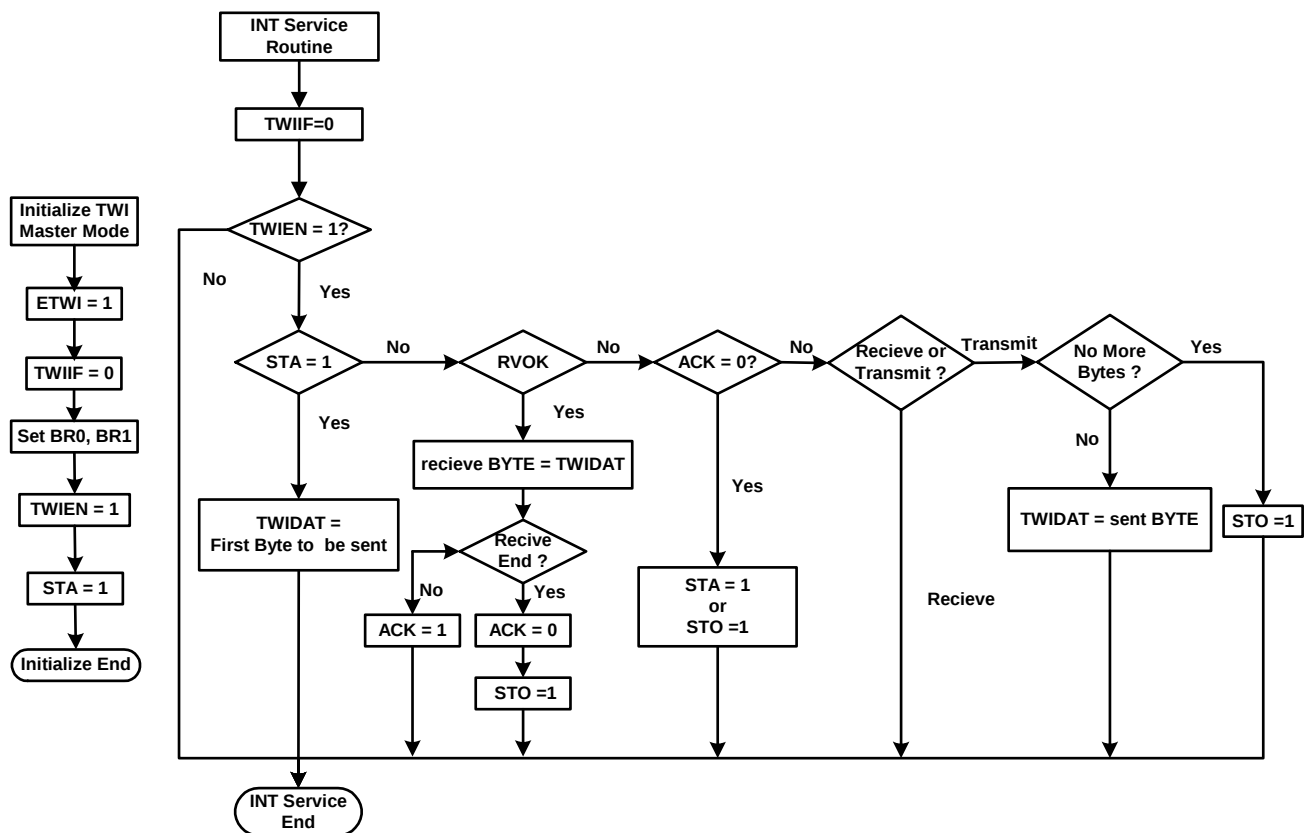
如果 $RVOK = 1$ ，表明接收到数据，软件从 $TWIDAT$ 寄存器中得到数据，然后设置 $ACK = 1$ 接收下一字节；设置 $ACK=0$ & $STO=1$ 中止通信。如果总线中止，硬件清除 STO 。



如果在通信期间发生错误，TWI 中断会被封闭，诸如 SCL 或 SDA 一直保持高或保持低。此时硬件会通过设置 $TWIEN = 0$ ， $STA = 0$ 关闭 TWI，然后产生中断。软件必须在中断服务程序中判断 $TWIEN$ 位以决定是否因为 TWI 错误而产生的中断。



软件流程图





8.3.3 寄存器

TWI 状态和控制寄存器

A6H	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	TWIEN	TWIIF	STA	STO	RXOK	ACK	BR1	BR0
读/写	读/写	读/写	读/写	读/写	只读	读/写	读/写	读/写
复位值 (POR/WDI/LVR)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	TWIEN	TWI 允许 0: 禁止 TWI 接口；如果有错误发生，由硬件清除这个位然后产生中断 1: 允许 TWI 接口
6	TWIIF	TWI 中断标记 0: 由软件清除 1: 由硬件设置表明 TWI 中断产生
5	STA	开始标记 0: 在传送第一个字节后由硬件清除，或错误发生。 1: 当总线闲置时，由软件设置产生一个 START 或 RESTART 条件。 如果设置 STA 和 STO，传送 START 条件后再传送 STOP 条件。
4	STO	停止标记 0: 在传送 STOP 条件后由硬件清除 1: 由软件设置产生一个 STOP 条件，TWI 引擎进入 idle 状态。
3	RXOK	接收 OK 标记 0: 未收到新字节，当如去 TWIDAT 时由硬件清除。 1: 收到新字节，准备读取新的字节，由硬件设置
2	ACK	确认标记 0: 在主接收模式下：由软件清除在 SDA 线路上产生 NAK 信号 在主传输模式下：由硬件清除在 SDA 线路上产生 TWI NAK 信号 1: 在主接收模式下：由软件清除在 SDA 线路上产生 ACK 信号 在主传输模式下：由硬件清除在 SDA 线路上产生 TWI ACK 信号
1-0	BR [1:0]	波特率 00: 波特率=PLL/160 bps 01: 波特率=PLL/80 bps 10: 波特率=PLL/40 bps 11: 波特率=PLL/20 bps

TWI 数据寄存器

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
TWIDAT	TWID7	TWID6	TWID5	TWID4	TWID3	TWID2	TWID1	TWID0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDI/LVR)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	TWID7-0	TWI 数据寄存器是移位寄存器的读/写缓冲器。 写数据到 TWIDAT 后，TWIDAT 中的数据直到移位寄存器为空时才移入移位寄存器。 成功接收一个数据时，移位寄存器中的值移入 TWIDAT 中，所以读 TWIDAT 时是读到的接收数据。如果在接收进行中读 TWIDAT，读出的数据为 TWIDAT 中的上一个数据，无论是从从机接收的还是软件写进去的。



8.4 增强型通用异步收发器 EUART

8.4.1 特性

- The SH79F32 内建两个增强型EUART，其和传统8052兼容。
- 波特率时钟源可选择来自系统时钟，定时器1/2的溢出率，但是EUART1的波特率时钟源不能选择来自定时器1。
- 增强功能包括错误检测和自动地址识别。
- EUART可工作在四种模式：模式0，模式1，模式2和模式3。
- 在执行发送前TxD 脚必须被设置为输出高。

8.4.2 EUART0

工作模式

EUART0 有 4 种工作模式。在通信之前用户必须先初始化 SCON。选择模式和波特率。如果使用模式 1 或模式 3 应先初始化定时器 1 或定时器 2。

在所有四种模式中，对 SBUF 写操作会启动发送。在模式 0 中由设置 RI = 0 和 REN = 1 启动接收，在 TxD0 引脚上输出移位控制脉冲，在 RxD0 引脚上输入数据。在其它模式中，如果 REN=1 则通过接收起始位来启动。

EUART 模式列表

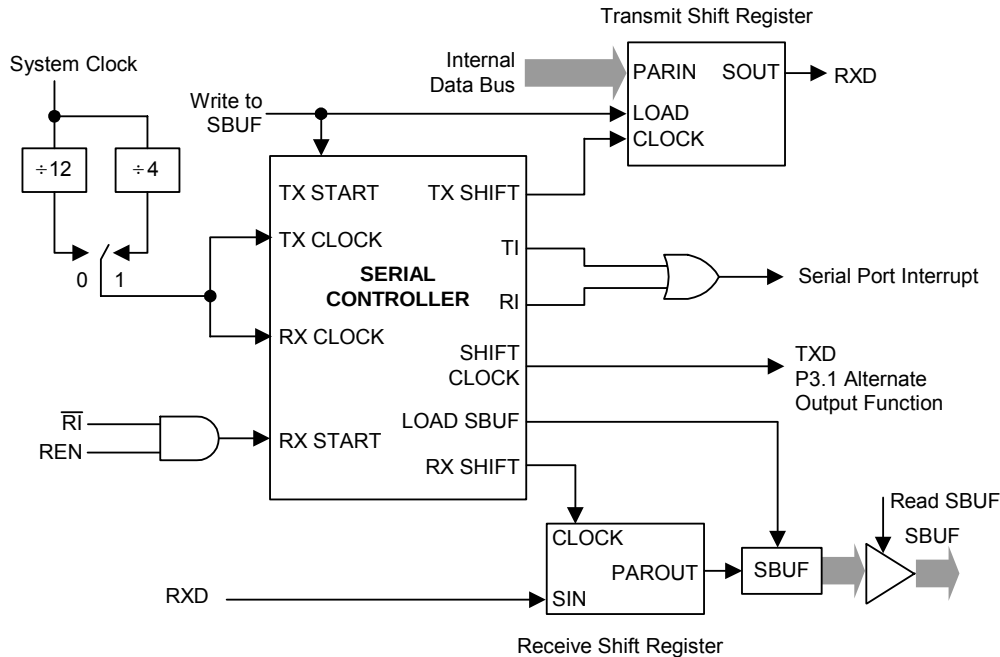
SM0	SM1	模式	类型	波特时钟	帧尺寸	开始位	中止位	第 9 位
0	0	0	同步	SysClk (4 或 12)	8 位	无	无	无
0	1	1	异步	Timer 1 或 2	10 位	1	1	无
1	0	2	异步	SysClk (32 或 64)	11 位	1	1	0, 1
1	1	3	异步	Timer 1 或 2	11 位	1	1	0, 1

**模式 0：同步，半双工通讯**

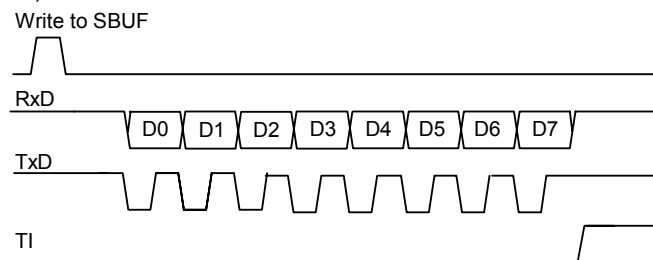
模式 0 为同步通信方式。在这个模式中，RXD0 脚上收发串行数据，TXD0 脚上发送移位时钟。因此这个模式是串行通信的半双工模式。在这个模式中，每帧收发 8 位，LSB 先收发。

通过设置 SM2 位(SCON.5)，波特率可被配置为系统时钟的 1/12 或 1/4。当设置 SM2 位为 0 时，串行端口以系统时钟的 1/12 运行。当设置为 1 时，串行端口以系统时钟的 1/4 运行。这是与标准 8051 唯一的不同点。

功能块框图如下图所示。

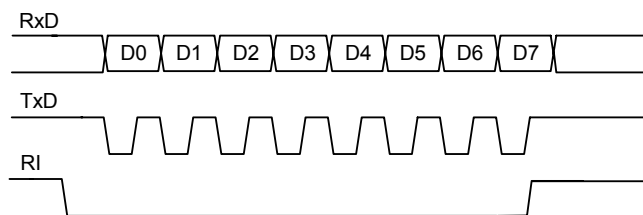


只要有数据写入 SBUF 都会启动发送，下一个机器周期发送控制模块开始发送。数据转换发生在移位时钟的下降沿，移位寄存器的内容逐次从左往右移位，0 从左边补位。当移位寄存器中的所有 8 位都移出后，发送控制模块停止发送操作，然后在下一个机器周期的上升沿 TI 置 1 (SCON.1)，并且 RxD0 保持高电平状态。



Send Timing of Mode 0

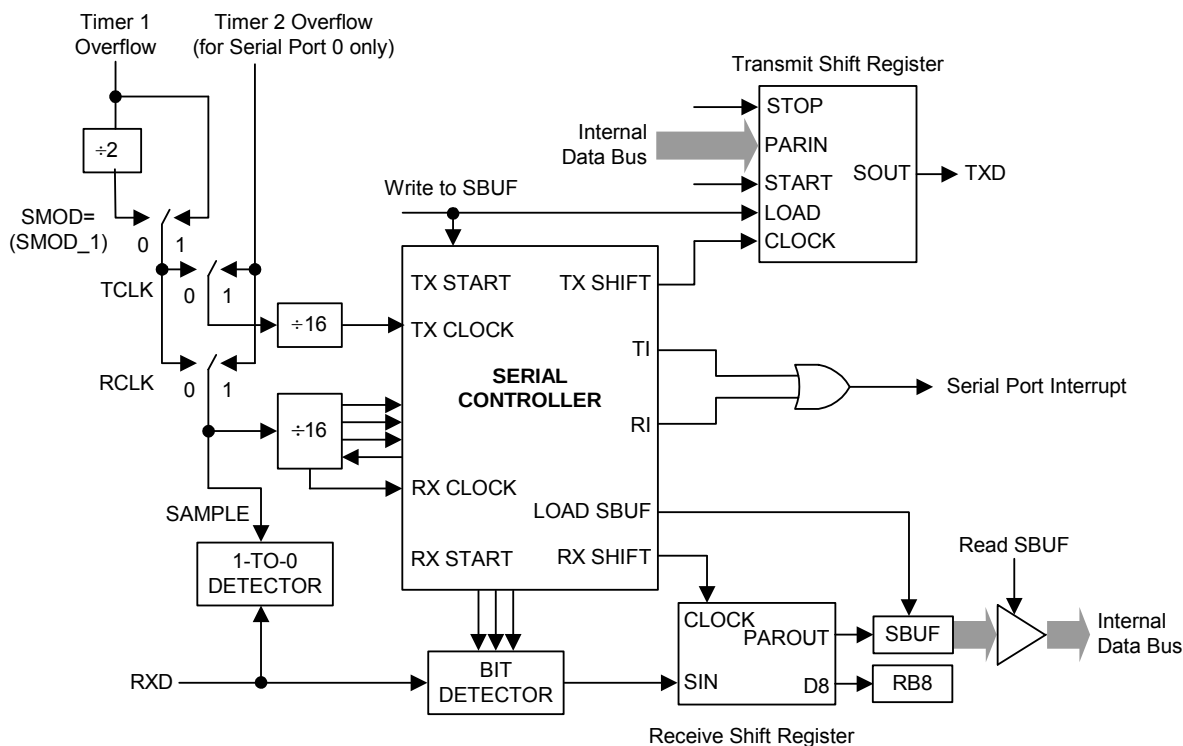
设置 REN (SCON.4)= 1 和 RI (SCON.0) = 0 使能接收，下一个机器周期启动接收，在移位时钟脉冲的上升沿锁存数据，接收转换寄存器的内容逐次向左移位。当所有 8 位都移入接收移位寄存器中后，接收控制模块停止接收，然后在下一个机器周期的上升沿 RI 置 1，直到被软件清 0，否则不允许接收。



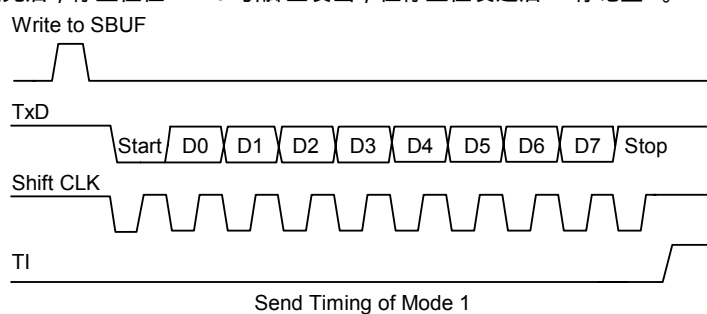
Receive Timing of Mode 0

**模式 1：8 位 EUART，可变波特率，异步全双工**

此模式为全双工异步模式，TxD0 发送数据，RxD0 接收数据。一个 10 位数据帧由一个开始位（逻辑 0），8 个数据位和一个停止位（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。串行收发波特率可编程为定时器 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。



只要有数据写入 SBUF 就会启动发送，实际上发送是从 16 分频计数器中的下一个翻转之后的机器周期后开始的，因此位时间与 16 分频计数器是同步的，与“写入 SBUF”信号不同步。开始位首先在 TxD0 引脚上移出，然后是 8 位数据。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 TxD0 引脚上发出，在停止位发送后 TI 标记置 1。

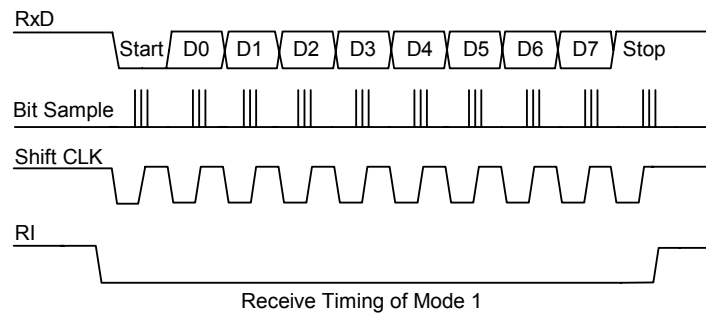


Send Timing of Mode 1

只要 REN 位置高，接收就被使能。接收过程始于 RxD0 端出现下降沿。为此，CPU 对 RxD0 不断采样，采样速率为波特率的 16 倍。当检测到下降沿时，16 分频计数器立即复位，这有助于计数器满刻度翻转时刻恰好与输入位的边沿对准。计数器的 16 个状态把每一位的时间分为 16 份，在第 7、8、9 状态时，位检测器对 RxD0 端的值进行采样。为抑制噪声，在这 3 个采样中至少有 2 次采样值一致，才认为采样到数据位。如果所接收的第一位不是 0，说明它不是一帧数据的起始位，该位被摒弃，接收电路复位，等待 RxD0 端另一个下降沿到来。若起始位有效，则移入移位寄存器，接着移入其它位到移位寄存器。当 8 个数据位和 1 个停止位移入之后，把移位寄存器的内容分别装入 SBUF 和 RB8 中，RI 置 1，但必须满足下列条件：

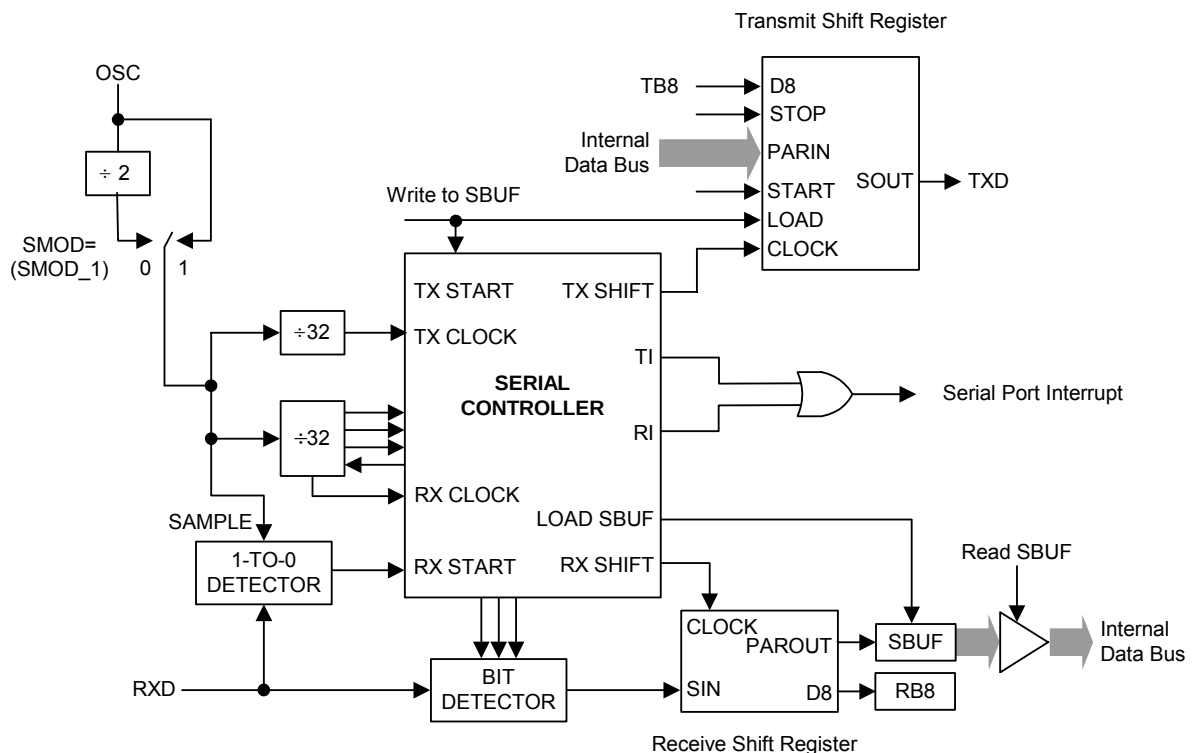
- 1) RI=0；
- 2) SM2 = 0 或者接收的停止位等于 1。

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 而 RI 被设置。否则接收的帧会丢失。这时，接收器将重新去探测 RxD0 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。

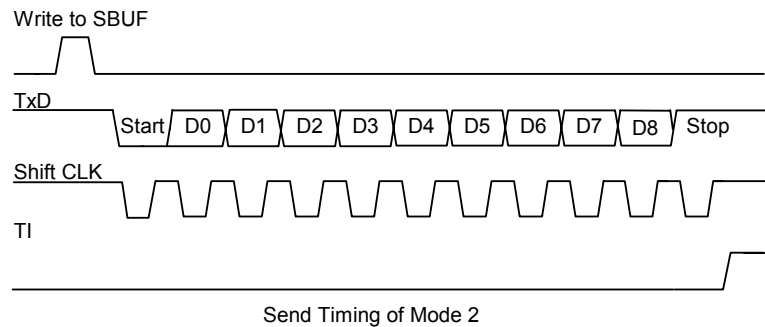


模式 2：9 位 EUART，固定波特率，异步全双工

此模式为异步全双工通信模式，功能块框图如下所示。一个 11 位数据帧由一个开始位（逻辑 0），8 个数据位，一个可编程的第 9 数据位和一个停止位（逻辑 1）组成。模式 2 支持多机通信和硬件地址识别功能（详见多机通信章节）。在数据传送时，第 9 数据位（SCON 中的 TB8）可以写 0 或 1，例如，在 PSW 中奇偶位 P，或用作多处理器通信中的数据/地址标记。当接收到数据时，第 9 数据位进入 RB8 而停止位不保留。PCON 中的 SMOD 位选择波特率为系统工作频率的 1/32 或 1/64。



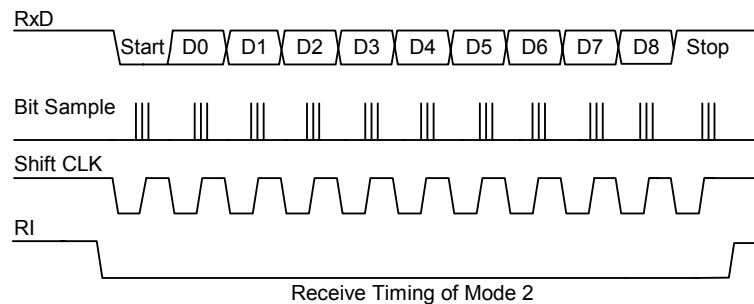
只要有数据写入 SBUF 就会启动发送，同时 TB8 中的内容也被载入到发送移位寄存器中的第 9 位。实际上发送是从 16 分频计数器中的下一个翻转之后的机器周期后开始的，因此位时间与 16 分频计数器是同步的，与“写入 SBUF”信号不同步。开始位首先在 TxD0 引脚上发出，然后是 9 数据位。在发送移位寄存器中的所有 9 位数据都发送完后，停止位在 TxD0 引脚上发出，在停止位发送后置位 TI 标志。



只要 REN 位置高，接收就被使能。接收过程始于 RxD0 端出现下降沿。为此，CPU 对 RxD0 不断采样，采样速率为波特率的 16 倍。当检测到下降沿时，16 分频计数器立即复位，这有助于计数器满刻度翻转时刻恰好与输入位的边沿对准。计数器的 16 个状态把每一位的时间分为 16 份，在第 7、8、9 状态时，位检测器对 RxD0 端的值进行采样。为抑制噪声，在这 3 个采样中至少有 2 次采样值一致，才认为采样到数据位。如果所接收的第一位不是 0，说明它不是一帧数据的起始位，该位被摒弃，接收电路复位，等待 RxD0 端另一个下降沿到来。若起始位有效，则移入移位寄存器，接着移入其它位到移位寄存器。当 9 个数据位和 1 个停止位移入之后，把移位寄存器的内容分别装入 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，且接收位符合本机地址。

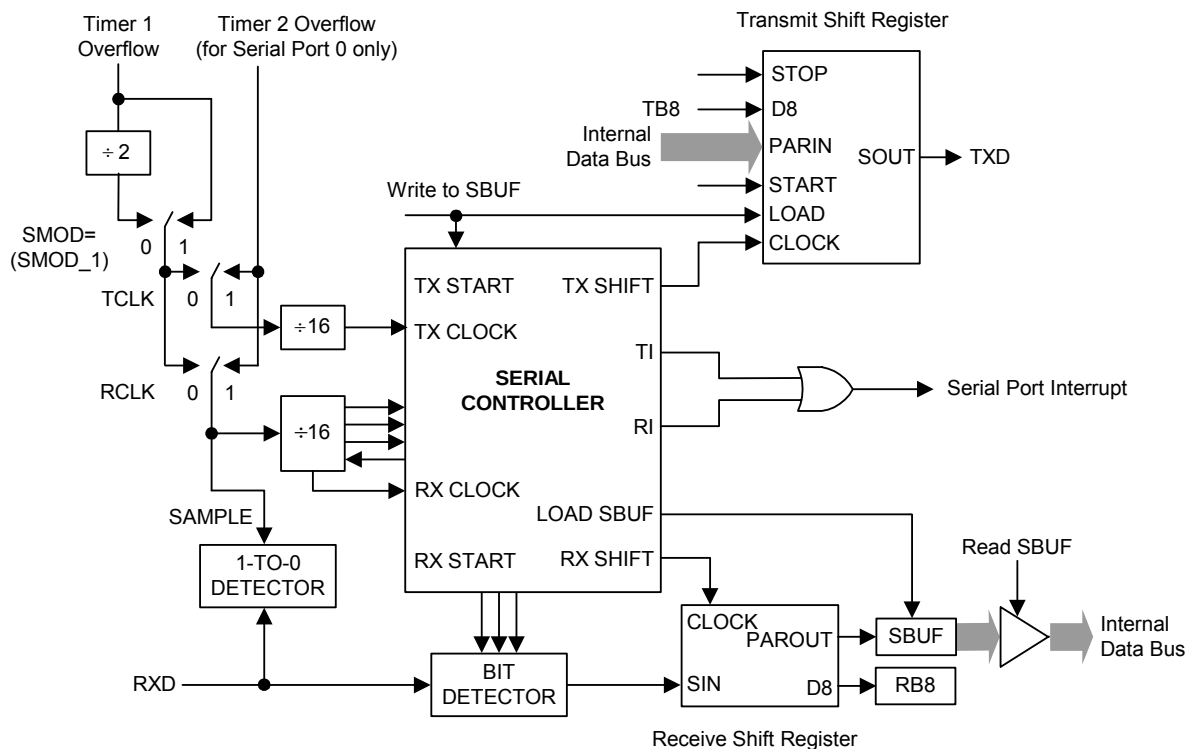
如果这些条件被满足，那么停止位移到 RB8，9 个数据位移到 SBUF 而 RI 被设置。否则接收的帧会流失。同时，接收器继续检测 RxD0 引脚上的另一个下降沿。用户必须用软件清除 RI，然后才能再次接收。





模式 3: 9 位 EUART, 可变波特率, 异步全双工

模式 3 使用模式 2 的传输协议以及模式 1 的波特率产生方式, 功能图如下。



在模式 0 中, 波特率是系统频率的 1/12 或 1/4, 由 SM2 位决定。当 SM2 设置为 0 时, 串行波特率为系统频率的 1/12; 当 SM2 设置为 1 时, 串行波特率为系统频率的 1/4。

在模式 1 和模式 3 中, EUART0 的波特率可由定时器 1 (自动重载模式) 或定时器 2 (波特率发生器模式) 的溢出率产生。定时器 1 模式 0 和模式 1 都能用作波特率发生器, 但使用起来颇为不便, 因此不推荐使用。分别设置 TCLK (T2CON.4) 和 RCLK (T2CON.5) 位来选择定时器 2 作为 TX0 和 RX0 的波特时钟源 (详见定时器)。无论设置 TCLK 或 RCLK 为逻辑 1, 定时器 2 都工作在波特率发生器模式。如果 TCLK 和 RCLK 都为逻辑 0, 只有定时器 1 作为 Tx0 和 Rx0 电路的波特时钟源。模式 1 和模式 3 波特率公式如下所示 8 位自动重载寄存器适用于定时器 1, SMOD(PCON.7) 位是 EUART0 波特率倍频器 [RCAP2H, RCAP2L] 是 16 位自动重载寄存器适用于定时器 2。T1CLK 是定时器 1 的时钟源, 而 T2CLK 是定时器 2 的时钟源。

$$BaudRate = \frac{2^{SMOD}}{32} \times \frac{T1CLK}{256 - TH1} \quad \text{波特率使用定时器 1, 自动重载模式}$$

$$BaudRate = \frac{1}{2 \times 16} \times \frac{T2CLK}{65536 - [RCAP2H, RCAP2L]} \quad \text{波特率使用定时器 2, T2CLK 为系统时钟}$$

$$BaudRate = \frac{1}{16} \times \frac{T2CLK}{65536 - [RCAP2H, RCAP2L]} \quad \text{波特率使用定时器 2, T2CLK 为 T2 脚时钟输入}$$

在模式 2 中, 波特率固定为系统时钟的 1/32 或 1/64。波特率在 SM2 位 (PCON.7) 中决定。当设置 SM2 位为 0 时, 串行端口以系统时钟的 1/64 运行。当设置为 1 时, 串行端口以系统时钟的 1/32 运行。

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



多机通信

软件地址识别

模式 2 和模式 3 有一个专门的应用领域，即多处理器通讯。在这两个模式下，接收的是 9 位数据，第 9 位进到 RB8 中，然后再来一位停止位。串口可以这样来编程：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）。要使串行口具有这一功能，可以将 SCON 寄存器的 SM2 位置 1。

在多处理机系统中，应用这一功能的方法如下所述。当主机需要发送一组数据给多个从机中的一个时，它先发出一地址字节，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，前者的第 9 位为 1，后者的第 9 位为 0。

SM2=1，从机不会被一个数据字节中断。然而，地址字节可以中断所有从机，这样，使每一台从机都检查一下所接收到的字节，以判别自己是不是目标从机。被寻址的从机清除它的 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=1，这些从机将不理睬进入到串行口的数据字节，继续做它们自己的事，直到一个新的地址字节的来临。

注意，模式 0 中，SM2 用来选择波特率。模式 1 中，SM2 用来检测停止位是否有效，如果 SM2=1，接收中断不会响应除非接收到一个有效的停止位。

自动(硬件)地址识别

在模式 2 和模式 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，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。紧接着接收中断后，从机将 SM2 清零，使能接收后续数据字节的中断。

在 9 位模式下第 9 个信息位为 1 以表明该信息内容是地址而非数据。当主机要发送一系列数据给一个从机时，它必须首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 SM2 置 1，忽视所有传送，直到接收到下一个地址字节。

使用自动地址识别特性时，主机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 位的值存于 SADDR 寄存器中。

SADEN 用于定义 SADDR 内哪几位需要使用而哪几位是无关位，如果 SADEN 中某一位为 0，则 SADDR 中相应位的被忽略，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另外的从机。

记忆码	从机 1	从机 2
SADDR	10100100	10100111
SADEN (0 掩码)	11111010	11111001
给定地址	10100x0x	10100xx1
广播地址 (或)	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）。类似的，从机 1 的位 1 为 0，从机 2 的位 1 被忽略。因此，只与从机 2 通讯时，主机必须发送位 1 为 1 的地址（10100011）。如果主机希望同时与两从机通讯，则位 0 为 1，位 1 为 0，位 3 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 00h，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理机通讯的特性，禁止了自动寻址模式。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。

**帧出错检测**

当寄存器 PCON 中的 SSTAT 位设置为逻辑 1 时，错检测功能是有有效的。

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 才可以访问状态位 (FE0, RXOVR0 和 TXCOL0)。SSTAT 位为逻辑 0 时可以访问模式选择位 (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（在 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（在 SCON 寄存器中的 RXOVR 位）置“1”。如果发生了接收溢出限，接收缓冲器中原来的数据将丢失。

帧错误

如果检测到一个无效（低）停止位，那么帧出错位（在寄存器 SCON 中的 FE）显示为“1”。

通讯中断检测

当连续检测到 11 个位都为低电平位时，则认为检测到一个通讯中断。由于通讯中断条件同样满足帧错误条件，因此检测到间隔时也会报告帧错误。一旦检测到间隔条件，UART 将进入空闲状态并一直保持，直至接收到有效停止位（RxD0 线上出现上升沿）。

8.4.3 EUART1

EUART1 的控制和工作模式与 EUART0 相同，但只使用定时器 2 作为波特率发生器，不管在 T2CON 寄存器中的 TCLK 和 RCLK 位是“1”还是“0”。TCK 和 RCK 只控制 EUART0 的波特率源。

如果定时器 2 的 TCLK=0, RCLK=0, 那么 EUART1 的波特率计算公式 $BaudRate = \frac{1}{16} \times \frac{SYSCLK}{65536 - [RCAP2H, RCAP2L]}$ 。

如果定时器 2 的 TCLK=1, RCLK=1, 那么 EUART1 的波特率计算公式 $BaudRate = \frac{1}{32} \times \frac{SYSCLK}{65536 - [RCAP2H, RCAP2L]}$ ，

与 EUART0 相同。

EUART1 还包含 IR 接口（请见 IR 章节）。

8.4.4 寄存器**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	SSTAT1	-	GF1	GF0	PD	IDL
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

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，且在模式 1 和 3 中被置 1，EUART0 的波特率会加倍 如果在模式 2 中设置，EUARTx 的波特率会加倍 is doubled; x = 0, 1
6	SSTAT	SCON [7:5] 功能选择 0 = SCON [7:5] 工作模式如同 SM0, SM1, SM2 1 = SCON [7:5] 工作模式如同 FE, RXOV, TXCOL
5	SSTAT1	SCON1 [7:5] 功能选择 0 = SCON1 [7:5] 工作模式如同 SM10, SM11, SM12 1 = SCON1 [7:5] 工作模式如同 FE1, RXOV1, TXCOL1
3-0	-	其他位请见“电源章节”



EUART0 相关 SFR

EUART0 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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-6	SM [0:1]	SSTAT = 0 时, UART0 串行模式控制 00 = 模式 0, 同步模式, 固定波特率 01 = 模式 1, 8 位异步模式, 可变波特率 10 = 模式 2, 9 位异步模式, 固定波特率 11 = 模式 3, 9 位异步模式, 可变波特率
7	FE	SSTA = 1 时, EUART0 帧错误标志 0 = 无帧出错, 由软件清除 1 = 帧出错, 由硬件设置
6	RXOV	SSTAT = 1 时, 接收完毕标记 0 = 无接收溢出, 由软件清除 1 = 接收溢出, 由硬件设置
5	SM2	SSTAT = 0 时, EUART0 多处理机通讯允许位 (第 9 位 “1” 校验器) 0 = 在模式 0 下, 波特率是系统时钟的 1/12 在模式 1 下, 禁止停止位检验, 停止位才能设置 RI, 产生中断 在模式 2 和 3 下, 任何字节都会设置 RI 产生中断 1 = 在模式 0 下, 波特率是系统时钟的 1/4 在模式 1 下, 允许停止位检验, 只有有效的停止位 (1) 才能设置 RI, 产生中断 在模式 2 和 3 下, 只有地址字节 (第 9 位=1) 能设置 RI 产生中断
5	TXCOL	SSTAT = 1 时, EUART0 发送冲突标记 0 = 无发送冲突, 由软件清除 1 = 有发送冲突, 由硬件设置
4	REN	EUART0 接收器允许 0 = 接收禁止 1 = 接收允许
3	TB8	EUART0 在模式 2 和 3 下发送的第 9 位数据位, 由软件设置或清除
2	RB8	EUART0 接收到的第 9 位数据位 在模式 0 下, RB8 没有用到。 在模式 1 下, 如果接收中断发生, RB8 是接收到的停止位。 在模式 2 和 3 下, 是接收到的第 9 位。
1	TI	EUART0 的传送中断标记 0 = 由软件清除。 1 = 由硬件设置, 在模式 0 下的 8 位数据后, 或在其他模式下的停止位到来时。
0	RI	EUART0 的接收中断标记 0 = 由软件清除 1 = 由硬件设置, 在模式 0 下的 8 位数据后, 或在其他模式下的停止位到来时。



EUART0 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
SBUF	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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	SBUF.7-0	SFR 访问两个寄存器：一个发送缓冲寄存器和一个接收锁存器。 SBUF 的写入将发送字节到发送缓冲寄存器中，然后开始传输。 SBUF 的读取返回到接收锁存器中的内容。

EUART0 从属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
SADDR	SADDR.7	SADDR.6	SADDR.5	SADDR.4	SADDR.3	SADDR.2	SADDR.1	SADDR.0
SADEN	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)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-0	SADDR.7-0	SFR SADDR 定义 EUART0 的从机地址。
7-0	SADEN.7-0	SFR SADEN 是一个位屏蔽寄存器，决定检验 SADDR 的哪个位地址： 0 = 不检验 SADDR 中的相应位。 1 = 检验在 SADDR 中的相应位。

注意：由于 TXD0/TXD1 与 P3.1/P2.7 复用，因此当有数据写入 SBUF/SBUF1 时，UART0/UART1 将通过 TXD0/TXD1 发送数据。当发送结束后，TXD0/TXD1 将恢复成作为 I/O 使用。所以当需要使用 UART/UART1 时，用户必须在 UART/UART1 初始化程序中将 P3.1/P2.7 设置成输出高的状态。



EUART1 相关 SFR

EUART1 控制和状态寄存器

9CH	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
SCON1	SM10/FE1	SM11/RXOV1	SM12/TXCOL1	REN1	TB81	RB81	TI1	RI1
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDI/LVR)	0	0	0	0	0	0	0	0

位编号	位符号	说明
7-6	SM10, SM11	SSTAT1 = 0 时, UART1 串行模式控制 00 = 模式 0, 同步模式, 固定波特率 01 = 模式 1, 8 位异步模式, 可变波特率 10 = 模式 2, 9 位异步模式, 固定波特率 11 = 模式 3, 9 位异步模式, 可变波特率
7	FE1	SSTAT1 = 1 时, EUART1 帧错误标志 0 = 无帧出错, 由软件清除 1 = 帧出错, 由硬件设置
6	RXOV1	SSTAT1 = 1 时, 接收完毕标记 0 = 无接收溢出, 由软件清除 1 = 接收溢出, 由硬件设置
5	SM12	SSTAT1 = 0 时, EUART1 多处理机通讯允许位 (第 9 位 “1” 校验器) 0 = 在模式 0 下, 波特率是系统时钟的 1/12 在模式 1 下, 禁止停止位检验, 停止位才能设置 RI1, 产生中断 在模式 2 和 3 下, 任何字节都会设置 RI1 产生中断 1 = 在模式 0 下, 波特率是系统时钟的 1/4 在模式 1 下, 允许停止位检验, 只有有效的停止位 (1) 才能设置 RI1, 产生中断 在模式 2 和 3 下, 只有地址字节 (第 9 位=1) 能设置 RI 产生中断
5	TXCOL1	SSTAT1 = 1 时, EUART1 发送冲突标记 0 = 无发送冲突, 由软件清除 1 = 有发送冲突, 由硬件设置
4	REN1	EUART1 接收器允许 0 = 接收禁止 1 = 接收允许
3	TB81	EUART1 在模式 2 和 3 下发送的第 9 位数据位, 由软件设置或清除
2	RB81	EUART1 接收到的第 9 位数据位 在模式 0 下, RB81 没有用到。 在模式 1 下, 如果接收中断发生, RB81 是接收到的停止位。 在模式 2 和 3 下, 是接收到的第 9 位。
1	TI1	EUART1 的传送中断标记 0 = 由软件清除。 1 = 由硬件设置, 在模式 0 下的 8 位数据后, 或在其他模式下的停止位到来时。
0	RI1	EUART1 的接收中断标记 0 = 由软件清除 1 = 由硬件设置, 在模式 0 下的 8 位数据后, 或在其他模式下的停止位到来时。



EUART1 数据缓冲器寄存器

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
SBUF1	SBUF1.7	SBUF1.6	SBUF1.5	SBUF1.4	SBUF1.3	SBUF1.2	SBUF1.1	SBUF1.0
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	SBUF1.7-0	SFR 访问两个寄存器：一个发送缓冲寄存器和一个接收锁存器。 SBUF 的写入将发送字节到发送缓冲寄存器中，然后开始传输。 SBUF 的读取返回到接收锁存器中的内容。

EUART1 从属地址和地址掩码寄存器

9EH-9FH	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
SADDR1	SADDR1.7	SADDR1.6	SADDR1.5	SADDR1.4	SADDR1.3	SADDR1.2	SADDR1.1	SADDR1.0
SADEN1	SADEN1.7	SADEN1.6	SADEN1.5	SADEN1.4	SADEN1.3	SADEN1.2	SADEN1.1	SADEN1.0
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

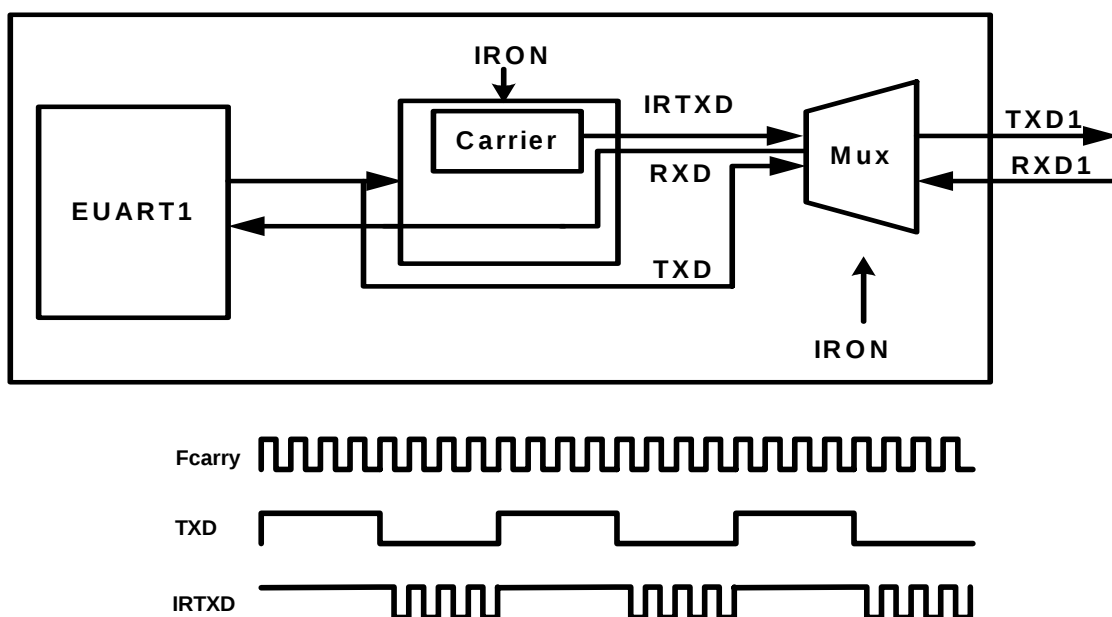
位编号	位符号	说明
7-0	SADDR1.7-0	SFR SADDR1 定义 EUART1 的从机地址。
7-0	SADEN1.7-0	SFR SADEN1 是一个位屏蔽寄存器，决定检验 SADDR1 的哪个位地址： 0 = 不检验 SADDR1 中的相应位。 1 = 检验在 SADDR1 中的相应位。



8.5 红外接口

SH79F32 的红外接口在 EUART1 内部。除了 EUART1 波形内的高频载波信号外，其他与 EUART1 一致。EUART1 和红外接口不能同时工作。IRF 寄存器决定载波频率。每个载波都与带 TXD1 信号的下沿同步进行。

当加载红外载波时，为了将 EUART1 误差减少到最小，要求波特率低于 9600bps。



IR 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
IRCON	IRON	IRF6	IRF5	IRF4	IRF3	IRF2	IRF1	IRF0
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	IRON	IR 打开/关闭控制 0 = 禁止红外接口，EUART1 作为普通串行端口 1 = 允许红外，红外频率载入 EUART1 的 Tx 信号
6-0	IRF[6:0]	IR 载波频率选择器 $F_{carrier} = \frac{SYSCLK}{2 \times (IRF + 1)}$ 如果要产生 38kHz 的载波(8.192MHz)，设置 IRF = 0x6C



8.6 实时时间时钟 (RTC)

8.6.1 特性

- 32768Hz 时钟输入，内部校准电流，精度 $\pm 5\text{ppm}$ ($\pm 0.5\text{s/d}$)。
- 半秒/秒、分钟、时、日、周、月、年寄存器。
- 自动跨月，闰年调整的日寄存器。
- 对 RTC 模块所涉及的寄存器的写操作，要等待 2 个 32K 时钟(大于 61us)才能真正写入寄存器。

注意：这个模块仅在选择使用 32768Hz 晶体振荡器时适用。

8.6.2 I/O 引脚

RTC 功能使用 2 个 I/O 引脚：CALIN 和 CALOUT。在校准期间，CALIN 作为输入引脚用于精度为 1ppm 的 1Hz 时钟输入。在校准之后，CALOUT 作为输出引脚用于补偿时钟输出，CALOUT 可以输出补偿后的 60 秒时钟或初始 32KHz 时钟。

8.6.3 功能说明

时间和日历功能：

RTC 模块以秒、分钟和小时提供时钟指示；以星期、日、月和年提供日历指示；并能对月和闰年进行自动调节。读取相关日历的各寄存器返回当前时间和日期。写入这些寄存器可设置时间和日期，而计数器会从新的设置开始重新计数。

日寄存器数据的循环会根据月和闰年自动调整，4 进制年寄存器的数据设置范围为 0-3。

寄存器	计数范围	计数溢出及自动数据重置	备注
SEC	00-59	59→00	-
MIN	00-59	59→00	-
HR	00-23	23→00	-
DAY	01-31	31→01	MTH=1,3,5,7,8,10,12
	01-30	30→01	MTH=4,6,9,11
	01-29	29→01	MTH=2, YR=0(闰年)
	01-28	28→01	MTH=2, YR=1,2,3(平年)
MTH	01-12	12→01	-
YR	0-3	3→0	YR=0 代表闰年
DOW	0-6	6→0	-

补偿

内建的频率补偿机制允许 RTC 模块可以使用非高精度 32768Hz 晶振为其提供 RTC 计数时钟。

补偿值可以由 RTC 模块校准功能完成，也可以由应用软件设置。

实时时钟功能

半秒/秒，分钟和小时寄存器提供了系统的实时时钟功能。半秒/秒寄存器可以每 0.5s 产生一个半秒中断请求信号，如果半秒中断允许(EHSEC=1)，CPU 将响应半秒中断。

日历功能

日期、星期、月和年寄存器提供日历功能。日计数器的计数范围会根据月和年(闰年)自动调节。年计数器计数范围为 0-3，其中 0 代表闰年。



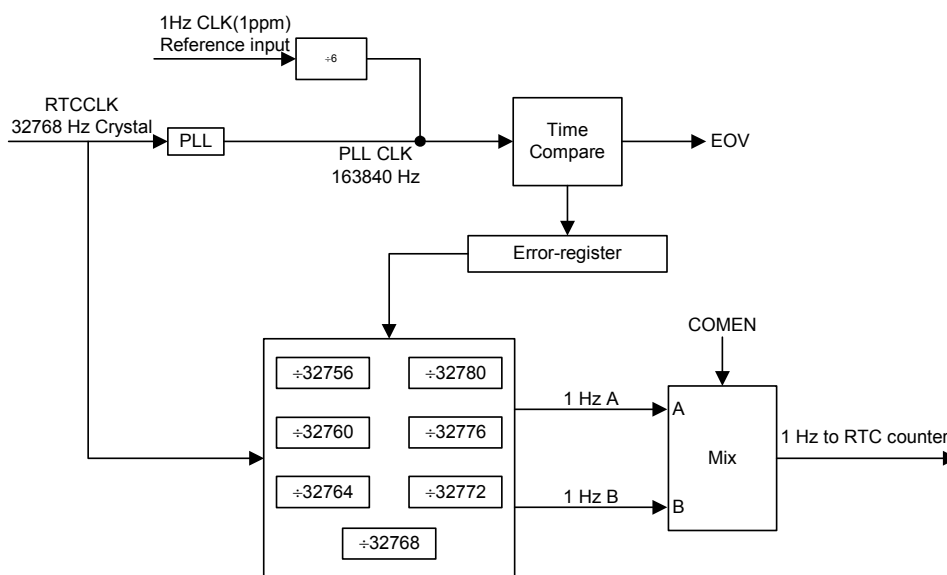
8.6.4 RTC 时钟校准和补偿 (方式 A)

概述

RTC 模块内建针对 RTCCLK 的校准和补偿电路。电路采用补偿 32768Hz 晶体谐振器频率偏差的方法以得到更加精准的 RTC 时钟。利用这个方法可以降低晶振的成本。

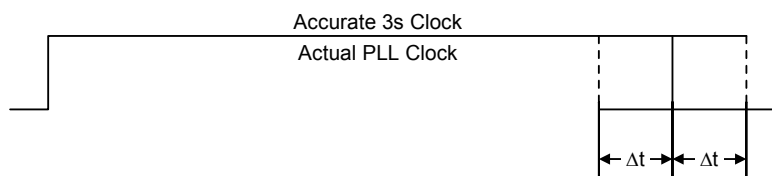
如果知道晶振的温度曲线特性及当前的温度，校准电路可以根据周围温度对晶振的频率偏差作出补偿。

电路提供最大±258ppm(-127~127)的补偿范围，适用于普通 30ppm 的晶体，也可以降低 PCB 成本。使用这种运算法则，RTC 模块能达到精确度在±5ppm 内的补偿 RTC 时钟。



RTC 时钟校准和补偿(模式 A)

自动校准电路通过 32768Hz 晶体谐振器驱动的 RTCCLK 信号与从 CALIN 引脚输入的 1Hz (1ppm)信号进行比较,通过设置在 RTC 校准寄存器 (RTCCR) 之后的 CAL 位来启动校准。在 3 秒的校准期间,“时间比较”模块比较两个时钟源的时间差异。



3 秒内 1HzCLK 和 PLLCLK 之间的时差

校准误差

在 1HzCLK 开始之后,RTCCLK 开始计数,在 1HzCLK 停止之后,PLLCLK 停止计数。在 3 秒之内 1HzCLK 和 RTCCLK(见上图)之间的差异可以用 PLLCLK 时钟表示如下:

$$E = PLLCLK \text{ cycles in } 3s - 32768 \times 5 \times 3$$

同步误差

因为 PLLCLK 和 1HzCLK 不能同步进行,导致的最大误差为±1×PLLCLK。

相应的误差为:

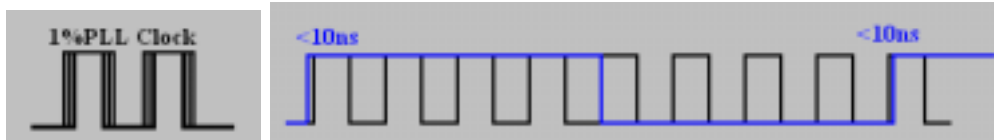
$$\frac{\pm 1}{32768 \times 5 \times 3} \times 10^6 \approx \pm 2 \text{ ppm}$$



同步误差

其次为 PLL 电路产生的误差。由于 PLL 电路的限制，存在 PLL 误差，每个 PLL 时钟有 1% 的误差。累计到每个基础时钟，只导致在每个基本时钟(32KHz)的上升沿中的大约 10ns 的误差。因此最大 PLL 误差可如下计算出：

$$\pm \left(\frac{2 \times 0.07}{32768 \times 5 \times 3} + \frac{10 \times 10^{-9}}{32768 \times 5 \times 3} \right) \times 10^6 < \pm 1 \text{ ppm}$$

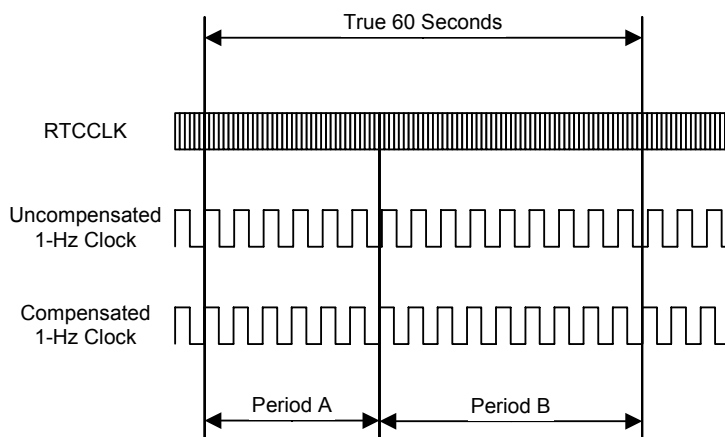


PLL 出错

因此总误差在 $\pm 3\text{ppm}$ 左右。

校准出错 E 储存在 8 位误差寄存器中(RTC 校准数据寄存器 RTCDAT)。负 E 值表示每一个调整周期中要减去 E 值，因为 RTCCLK 比理想值 32768Hz 慢；32768 个 RTCCLK 时钟周期比 1 秒更长。正 E 值表示每一个调整周期中要加上 E 值，因为 RTCCLK 比理想值 32768Hz 快；32768 个 RTCCLK 时钟周期比 1 秒更短。

如果时间差超过 127 个 RTCCLK 周期，E 寄存器溢出，EVOL 标记被置 1。超出 127 的余数被记录在 E 寄存器中。



1-Hz Clock Compensation

校准运算法则：

在校准之后，E 值存储在校准数据寄存器 RTCDAT 中，只有当 RTCCON 寄存器中的 COMEN 位被置'1'时，系统补偿功能才能启动，补偿时钟以每 60 秒为周期调整一次。

用户首先要在 RTC 可运行的每个温度点进行校准操作，然后将相应的校准值 E 存储在“校准表”(软件)中。当温度改变时，用户应该查询这个“校准表”，并将相应 E 值写入寄存器 RTCDAT 中。

	周期 A		周期 B	
	计数	分配器	计数	分配器
$-128 \leq E \leq -121$	$ E - 120$	32756	$180 - E $	32760
$-120 < E \leq -61$	$ E - 60$	32760	$120 - E $	32764
$-60 \leq E \leq -1$	$ E $	32764	$60 - E $	32768
$E = 0$	无补偿，分配器是 32768			
$1 \leq E \leq 60$	$ E $	32772	$60 - E $	32768
$120 \leq E \leq 61$	$ E - 60$	32776	$120 - E $	32772
$128 \leq E \leq 121$	$ E - 120$	32780	$180 - E $	32776
补偿时钟：每 60 秒调整一次				
补偿范围： $\pm 258\text{ppm}$				



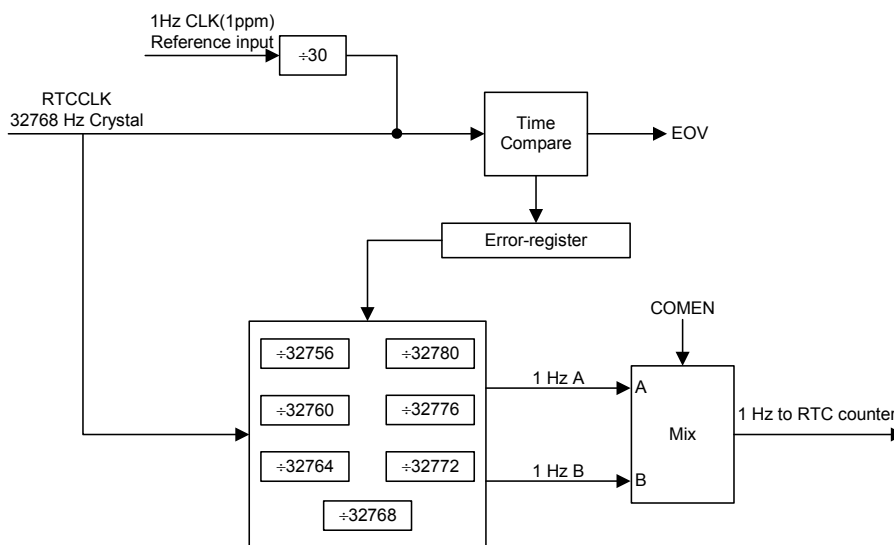
8.6.5 RTC 时钟校准和补偿 (方式 B)

概述：

RTC 模块内建针对 RTCCLK 的校准和补偿电路。电路采用补偿 32768Hz 晶体谐振器频率偏差的方法以得到更加精准的 RTC 时钟。利用这个方法可以降低晶振的成本。

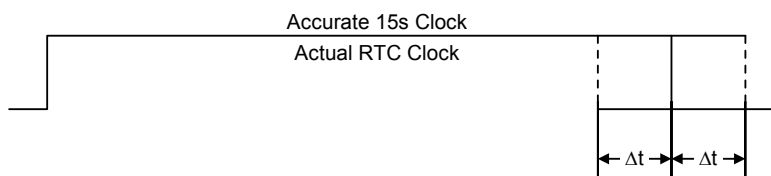
如果知道晶振的温度曲线特性及当前的温度，校准电路可以根据周围温度对晶振的频率偏差作出补偿。

电路提供最大±258ppm(-127~127)的补偿范围，适用于普通 30ppm 的晶体，也可以降低 PCB 成本。使用这种运算法则，RTC 模块能达到精确度在±5ppm 内的补偿 RTC 时钟。



RTC 时钟校准和补偿(模式 B)

自动校准电路通过 32768Hz 晶体谐振器驱动的 RTCCLK 信号与从 CALIN 引脚输入的 1Hz (1ppm)信号进行比较,通过设置在 RTC 校准寄存器 (RTCCR) 之后的 CAL 位来启动校准。在 15 秒的校准期间,“时间比较”模块比较两个时钟源的时间差异。



15 秒内 1HzCLK 和 RTCCLK 时差

校准误差

在 1HzCLK 开始之后,RTCCLK 开始计数,在 1HzCLK 停止之后,PLLCLK 停止计数。在 3 秒之内 1HzCLK 和 RTCCLK(见上图)之间的差异可以用 PLLCLK 时钟表示如下:

$$E = \text{RTCCLK cycles in 15s} - 32768 \times 15$$

同步误差

因为 PLLCLK 和 1HzCLK 不能同步进行,导致的最大误差为±1×PLLCLK。

相应的误差为：

$$\frac{\pm 0.5}{32768 \times 15} \times 10^6 \approx \pm 2 \text{ ppm}.$$

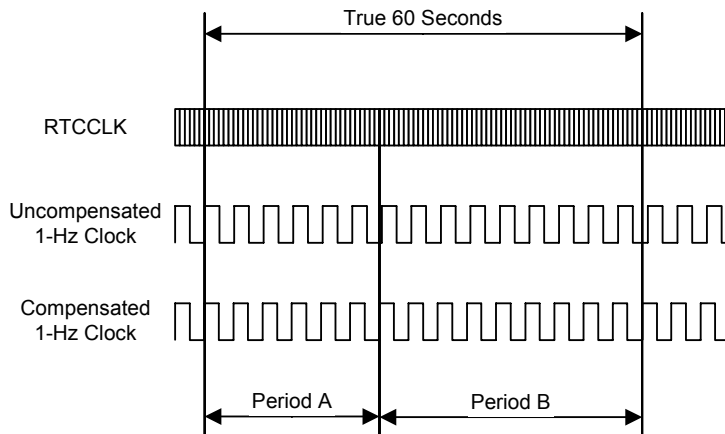




同步误差

校准出错 E 储存在 8 位误差寄存器中(RTC 校准数据寄存器 RTCDAT)。负 E 值表示每一个调整周期中要减去 E 值, 因为 RTCCLK 比理想值 32768Hz 慢; 32768 个 RTCCLK 时钟周期比 1 秒更长。正 E 值表示每一个调整周期中要加上 E 值, 因为 RTCCLK 比理想值 32768Hz 快; 32768 个 RTCCLK 时钟周期比 1 秒更短。

如果时间差超过 127 个 RTCCLK 周期, E 寄存器溢出, EVOL 标记被置 1。超出 127 的余数被记录在 E 寄存器中。



1-Hz Clock Compensation

校准运算法则：

在校准之后, E 值存储在校准数据寄存器 RTCDAT 中, 只有当 RTCCON 寄存器中的 COMEN 位被置'1'时, 系统补偿功能才能启动, 补偿时钟以每 60 秒为周期调整一次。

用户首先要在 RTC 可运行的每个温度点进行校准操作, 然后将相应的校准值 E 存储在“校准表”(软件)中。当温度改变时, 用户应该查询这个“校准表”, 并将相应 E 值写入寄存器 RTCDAT 中。

	周期 A		周期 B	
	计数	分配器	计数	分配器
$-128 \leq E \leq -121$	$ E - 120$	32756	$180 - E $	32760
$-120 < E \leq -61$	$ E - 60$	32760	$120 - E $	32764
$-60 \leq E \leq -1$	$ E $	32764	$60 - E $	32768
$E = 0$	无补偿, 分配器是 32768			
$1 \leq E \leq 60$	$ E $	32772	$60 - E $	32768
$120 \leq E \leq 61$	$ E - 60$	32776	$120 - E $	32772
$128 \leq E \leq 121$	$ E - 120$	32780	$180 - E $	32776
补偿时钟: 每 60 秒调整一次				
补偿范围: $\pm 258\text{ppm}$				



8.6.6 寄存器

RTC 校准控制寄存器

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
RTCCON	RTCEN	HSECIF	COMEN	CAL	AUTOCAL	OUTF1	OUTF0	EOVL
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	只读
复位值 (POR/WDT/LVR)	0	0	u	0	0	0	0	0

位编号	位符号	说明
7	RTCEN	实时时钟允许 0 = 禁止 RTC 模块 1 = 允许 RTC 模块
6	HSECIF	半秒中断标记 0 = 无半秒计数器增加 1 = 半秒计数器增加
5	COMEN	RTC 补偿允许 0 = 禁止补偿机制 1 = 允许补偿机制 注意：复位不改变 COMEN 位值
4	CAL	RTC 校准模式允许 0 = 禁止 RTC 校准模式，在自动校准完成后自动清除 1 = 允许 RTC 校准模式
3	AUTOCAL	RTC 自动校准允许 0 = 禁止 RTC 自动校准 1 = 允许 RTC 自动校准，P3.4 用作 CALIN 引脚来接收外灌的 1Hz 信号由 RTC 电路自动进行校准，校准完成后，CAL 和 AUTOCAL 位被自动清除。
2-1	OUTF [1:0]	校准模式下的 CALOUT 引脚输出频率选择 00 = 断开 CALOUT 引脚 01 = CALOUT 引脚输出周期为 60s 的时钟 当 COMMEN=1 时，输出经补偿过的周期为 60s 的时钟； 当 COMMEN=0 时，输出未经补偿过的周期为 60s 的时钟； 10 = CALOUT 引脚输出原始的(外接的)32768Hz 晶振时钟。 11 = CALOUT 引脚输出周期为 1s 的时钟 当 COMMEN=1 时，输出经补偿过的周期为 1s 的时钟； 当 COMMEN=0 时，输出未经补偿过的周期为 1s 的时钟。
0	EOVL	误差溢出标记 0 = 最后一次自动校准无误差溢出 1 = 最后一次自动校准有误差溢出，即频率偏差超出范围无法校准

注意：CAL 和 AUTOCAL 的相互关系如下：

CAL, AUTOCAL:

0x -- E 寄存器 RTCDAT 禁止写入。

10 -- E 寄存器 RTCDAT 仅可由软件写入，与 RTCEN 位状态无关。

11 -- E 寄存器仅可由 RTC 校准电路写入，即：仅当 RTCEN 位为'1'，且 CAL、AUTOCAL 两位均设置为'1'时，E 寄存器 RTCDAT 可由 RTC 校准电路将补偿数据写入。之后，CAL、AUTOCAL 两位由硬件自动清'0'。



RTC 补偿值 (E)寄存器

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
RTCDAT	E7	E6	E5	E4	E3	E2	E1	E0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	u	u	u	u	u	u	u	u

位编号	位符号	说明
7-0	E[7:0]	RTC 补偿值 (E 寄存器) E[7:0]补偿数值用来表示 RTC 工作时需要进行补偿的时钟个数。如果补偿值超过 ± 128 , E[7:0]保持溢出后剩余的数值, 然后设置 EOVL 标记。 当允许补偿时(COMEN = 1):如果 E 是负, 表示每一个调整周期中要减去 E 值; 如果 E 是正, 表示每一个调整周期中要加上 E 值。 在自动校准之后, RTC 补偿电路将补偿值写入 E[7:0]。 当 CAL = 1, AUTO CAL=0 时, 只能由用户写入 E[7:0]。 注意: 复位不改变 E[7:0]值

注意: 当允许校准时, RTC 时钟和日历功能不受影响。

秒和半秒寄存器

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
SEC	HSEC	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	u	u	u	u	u	u	u	u

位编号	位符号	说明
7	HSEC	半秒指示位, 按半秒的频率交替置 0 和 1。 在任何时候可读取这位而不影响秒计数器计数。
6-0	SEC6-0	寄存器(第 0 位 ~ 第 6 位)保存秒计数器的当前值(BCD)。可以在任何时候读取寄存器(第 0 位 ~ 第 6 位)而不影响计数器计数。写入寄存器会载入值到秒计数器, 计数器继续从新值开始计数。秒计数器的值在到达 59 之后滚动至 0。 0-59 之外的数据无法写入。 注意不要写入 0x、1x、2x、3x、4x (x=A~FH)非法数据。

分钟寄存器

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
MIN	-	MIN6	MIN5	MIN4	MIN3	MIN2	MIN1	MIN0
读/写	-	读/写	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	u	u	u	u	u	u	u

位编号	位符号	说明
6-0	MIN6-0	寄存器保存分钟计数器的当前值(BCD)。可以在任何时候读取寄存器而不影响计数器计数的。写入寄存器会载入值到分钟计数器, 计数器继续从新值开始计数。分钟计数器的值在到达 59 之后滚动至 0。 0-59 之外的数据无法写入。 注意不要写入 0x、1x、2x、3x、4x (x=A~FH)非法数据。



小时寄存器

C3H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
HR	-	-	HR5	HR4	HR3	HR2	HR1	HR0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	u	u	u	u	u	u

位编号	位符号	说明
5-0	HR5-0	寄存器保存小时计数器的当前值(BCD)。可以在任何时候读取寄存器而不影响计数器计数的。写入寄存器会载入值到小时计数器，计数器继续从新值开始计数。小时计数器的值在到达 23 之后滚动至 0。 0-23 之外的数据无法写入。 注意不要写入 0x、1x(x=A~FH)非法数据。

日寄存器

C4H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
DAY	-	-	DAY5	DAY4	DAY3	DAY2	DAY1	DAY0
读/写	-	-	读/写	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	u	u	u	u	u	u

位编号	位符号	说明
5-0	DAY5-0	寄存器保存日计数器的当前值(BCD)。可以在任何时候读取寄存器而不影响计数器计数的。写入寄存器会载入值到日计数器，计数器继续从新值开始计数。日计数器的值根据月和年寄存器的值在到达 28、29、30 或 31 之后滚动至 1。 1-31 之外的数据无法写入。 注意不要写入 1x、2x(x=A~FH)非法数据。

月寄存器

C5H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
MTH	-	-	-	MTH4	MTH3	MTH2	MTH1	MTH0
读/写	-	-	-	读/写	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	u	u	u	u	u

位编号	位符号	说明
4-0	MTH4-0	寄存器保存月计数器的当前值(BCD)。可以在任何时候读取寄存器而不影响计数器计数的。写入寄存器会载入值到月计数器，计数器继续从新值开始计数。月计数器的值在到达 12 之后滚动至 1。 1-12 之外的数据无法写入。 注意不要写入 0x(x=A~FH)非法数据。



年寄存器

C6H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
YR	-	-	-	-	-	-	YR1	YR0
读/写	-	-	-	-	-	-	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	-	-	u	u

位编号	位符号	说明
7-0	YR2-0	寄存器保存年计数器的当前值(BCD)。可以在任何时候读取寄存器而不影响计数器计数的。写入寄存器会载入值到年计数器，计数器继续从新值开始计数。 年计数器的值在到达 3 之后滚动至 0。 0 表示闰年，1-3 表示平年(硬件自动识别)，0-3 之外的数据无法写入。

星期寄存器

C7H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
DOW	-	-	-	-	-	DOW2	DOW1	DOW0
读/写	-	-	-	-	-	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	-	u	u	u

位编号	位符号	说明
2-0	DOW2-0	寄存器保存星期计数器的当前值(BCD)。可以在任何时候读取寄存器而不影响计数器计数的。写入寄存器会载入值到星期计数器，计数器继续从新值开始计数。星期计数器的值在到达 6 之后滚动至 0。 0-6 之外的数据无法写入。



8.7 模/数转换器(ADC)

8.7.1 特性

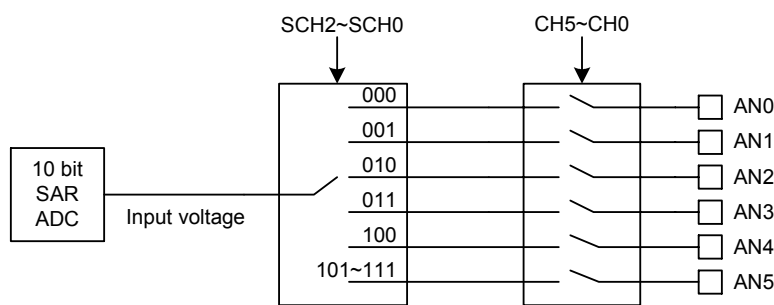
- 10 位 ADC
- $\pm 1\text{LSB}$ 微分非线性误差
- $\pm 2\text{LSB}$ 积分非线性误差
- 总转换时间: 采样保持时间+数据转换时间($12t_{AD}$)
- 6 通道输入
- ADC转换结束中断
- 数字比较器功能

SH79F32包括一个单端型、10位逐次逼近型数/模转换器，基准电压V_{ref}和V_{DD}相连，6个ADC通道共享一个ADC模块，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，置位ADCON寄存器中的ADCIF位并产生一个中断（如果允许ADC中断）。

ADC集成了数字比较功能，当ADC模块功能使能(ADON = 1 在 ADCON 寄存器)，比较功能使能(EC = 1 在 ADCON 寄存器中)时，可以对模拟输入进行AD转换与寄存器中的数字值比较。当相应的模拟输入数字值大于寄存器中的设定值(ADDH/L)时，会产生ADC中断。当GO/DONE位置1时，数字比较器可以连续工作，但不同于ADC转换功能。

带数字比较模块的ADC模块能在空闲模式下工作，并且由ADC中断唤醒，但不能工作在掉电模式下。

8.7.2 ADC 模块图



ADC 模块框图

寄存器

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)	0	0	0	-	0	0	0	0

位编号	位符号	说明
7	ADCON	ADC 允许 0 = 禁止 ADC 模块 1 = 允许 ADC 模块
6	ADCIF	ADC 中断标记 0 = 无 ADC 中断，由软件清除 1 = 由硬件置 1，表示已完成 AD 转换，或者模拟输入大于 ADDATH/ADDATL（如果允许比较）
5	EC	比较功能允许 0 = 禁止比较功能（默认） 1 = 允许比较功能
3-1	SCH [2:0]	ADC 信道选择 000 = ADC 信道 AN0



SH79F32

		001 = ADC 信道 AN1 010 = ADC 信道 AN2 011 = ADC 信道 AN3 100 = ADC 信道 AN4 其它 = ADC 信道 AN5
0	GO/DONE	ADC 状态标记 0 = 当完成 AD 转换时, 该位由硬件自动清除。如在转换期间清除该位会中止当前的转换。如果比较功能被允许, 该位只能软件清零, 不能硬件清零。 1 = 置 1 开始 AD 转换。

ADC 转换时间寄存器

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)	0	0	0	-	0	0	0	0

位编号	位符号	说明
7-5	TADC [2:0]	TADC [2:0] – ADC 时钟周期选择 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]	采样时间选择 采样时间 = $(TS [3:0]+1) * t_{AD} < 16 t_{AD}$

注意:

- 1) 请确保 $t_{AD} = 1\mu s$
- 2) 采样时间最小为 $2t_{AD}$, 即使 $TS[3:0] = 000$; 最大为 $15t_{AD}$, 即使 $TS[3:0] = 111$ 。
- 3) 在设置 $TS[3:0]$ 前, 请估算连接到 ADC 输入引脚的串联电阻。
- 4) 选择采样时间为 $2*t_{AD}$, V_{DD} 为 4.5V-5.5V 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=转换时间 ($12t_{AD}$) + 采样时间。

ADC 信道选择寄存器

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	CH5	CH4	CH3	CH2	CH1	CH0	-	-
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	-	-

位编号	位符号	说明
7-2	CH [5:0]	信道配置 0 = 将 P2.x (x=2~7)作为标准 I/O 端口 1 = 将 P2.x (x=2~7)作为 ADC 输入, 如果 P2.x 没有其它功能在使用中。



SH79F32

AD 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）	-	-	-	-	-	-	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）	0	0	0	0	0	0	0	0

位编号	位符号	说明
1-0 7-0	A9-A0	当完成转换后，数据更新为采样模拟电压的数字值。 如果 ADC 比较功能被允许(EC = 1)，这个值将与模拟输入进行比较。



8.8 脉冲宽度调制 (PWM)

8.8.1 特性

SH79P32 集成了一个 12 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。而 PWM 输出波形的占空比由 PWMD 寄存器来设置。

如果允许 EPWMN，PWM 模块还提供带固定相位关系的 PWMN 输出。这种固定相位关系还能提供停滞时间控制，停止时间控制在大部分上拉控制系统中的作用是十分显著的。

如果设置 EFLT，PWM 定时器能由 FLT 引脚的输入端口自动转换为无效。

PWM 定时器也能提供中断功能，用户可以在每个 PWM 周期中更改周期和占空比。

8.8.2 PWM 允许寄存器

PWM 允许控制寄存器

CFH	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
PWMEN	-	EFLT	-	-	EPWMN	-	-	EPWM
读/写	-	读/写	-	-	读/写	-	-	读/写
复位值 (POR/WDI/LVR)	-	0	-	-	0	-	-	0

位编号	位符号	说明
6	EFLT	FLT 引脚配置： 0= 用作 P0.5 或 SEG30 引脚（默认） 1= 用作 PWM 故障检测输入引脚
3	EPWMN	PWMN 输出允许位，当 PWM 模块被允许时： 0 = PWMN 引脚输出禁止 1 = PWMN 引脚输出允许 置 1 时，PWMN 引脚会延迟 1 个系统时钟和一个 PWM 时钟后才产生 PWM 波形
0	EPWM	PWMP 输出控制位 0 = PWMP 引脚输出禁止 1 = PWMP 引脚输出允许

无论是设置 EPWMN 还是设置 EPWM 时，PWM 模块都被允许。

如果一个有效的事件在故障引脚中发生，(由 FLTC 选择低电平或高电平)，PWM 输出将立即进入无效状态。FLT 引脚没有内部上拉电阻。

如果设置 EFLT 为“0”，这就意味着在 FLT 引脚上的电平对 PWM 没影响。



8.8.3 PWM 定时器锁定寄存器

这个寄存器用来控制 PWM 控制寄存器 1、PWM 控制寄存器 2、PWM 周期寄存器、PWM 占空比寄存器和 PWM 停滞时间控制寄存器的更改。只有当数据在地址位为#55h 的寄存器中，才能更改这些寄存器，否则不能更改。

在同一时间，只有禁止 PWM 模块时，才能更改 PWM 停滞控制寄存器，(EPWMN 和 EPWM 都复位)。

这个寄存器能增强 SH79F32 的干扰能力。

PWM 定时控制寄存器

E7H	第 7 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位	第 0 位
PWMLO	PWMLO. 7	PWMLO. 6	PWMLO. 5	PWMLO. 4	PWMLO. 3	PWMLO. 2	PWMLO. 1	PWMLO. 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	PWML	PWM 锁定 只有当 PWMLO=#55h，才能允许修改其他 PWM 寄存器

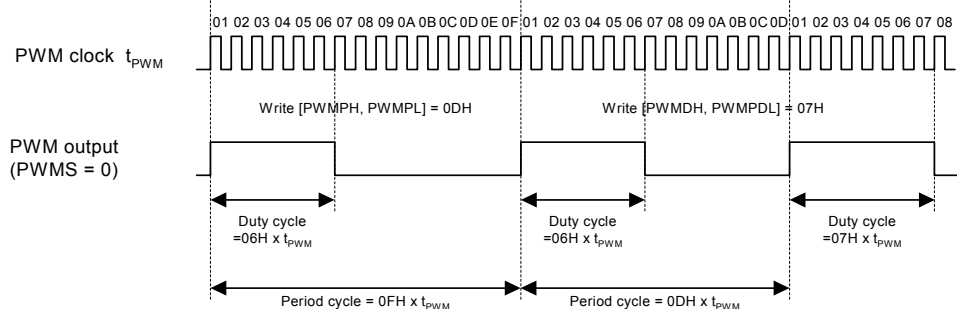


8.8.4 PWM 定时器

下列寄存器控制 PWM 波形 :PWMC, PWMPH/L 和 PWMDH/L。PWMC 用来控制 PWM 时钟源 ;PWMPH/L 用来控制 PWM 周期, PWMDH/L 用来控制 PWM 占空比。

在 PWM 输出允许期间能更改这三个寄存器。所有更改都会影响到下一个 PWM 周期。

注意：对于 PWMPH/L, PWMDH/L, 只有在写入高字节寄存器后才能更改 PWM 定时器, 写入低位字节不起作用。



PWM 输出周期或占空比更改示例

PWM 定时控制寄存器

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
PWMC	PWMIE	PWMIF	-	FLTS	FLTC	PWMS	PWMCK1	PWMCK0
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

位编号	位符号	说明
7	PWMIE	PWM 中断允许位 (当设置在 IEN1 中的 EPWM 位时) 0 = 禁止 PWM 中断 1 = 允许 PWM 中断
6	PWMIF	PWM 中断标记 0 = 由软件清除 1 = 由硬件设置表明 PWM 周期计数器溢出
4	FLTS	FLT 状态位: 0: PWM 在正常状态下, 由软件清除。 1: PWM 在无效状态下, 自动由硬件设置。
3	FLTC	FLT 引脚配置: 0: 当 FLT 为低电平时, 阻止 PWM 输出 (默认) 1: 当 FLT 为高电平时, 阻止 PWM 输出
2	PWMS	PWMS - PWM 输出优先级选择器: 0 = 高有效 (默认) 1 = 低有效
1-0	PWMCK [1:0]	PWM 时钟选择器: 00 = 振荡器或 PLL 时钟/2 (默认) 01 = 振荡器或 PLL 时钟/4 10 = 振荡器或 PLL 时钟/8 11 = 振荡器或 PLL 时钟/16 如果选择频率为 32KHz 并且允许 PLL, 时钟源为 PLL 时钟, 否则, 时钟源是振荡器时钟。

注意：

- 1) PWMC 寄存器中的 FLTS 和 FLTC 位能影响所有 PWM 定时器, 而 PWMC 寄存器中的 PWMS, PWMCK1 [1:0]只能影响 PWM。
- 2) 阻止 PWM 即是阻止 PWM 输出保持低电平(PWMS=0)或高电平(PWMS=1)。
- 3) 一旦检测到 FLT 引脚的高/低电平, PWM 输出会保持无效状态。
- 4) 当一个 FAULT 输入到达时, 无法清除 FLTS 位。
- 5) 当用户程序清除 FLT 状态位(FLTS)后, 在以下 PWM 周期的开头 PWM 输出立即被允许。



PWM 周期控制寄存器

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
PWMP.L	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)	0	0	0	0	0	0	0	0
D4H	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.H	-	-	-	-	PWMP.11	PWMP.10	PWMP.9	PWMP.8
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	0	0	0	0

位编号	位符号	说明
7-0 3-0	PWMP.11-0	PWM 输出周期 = [PWMP.H, PWMP.L] * PWM 时钟 当[PWMP.H, PWMP.L] = 000H 时, PWMP 引脚输出 GND 如果 PWMS “0”。 当[PWMP.H, PWMP.L] = 000H 时, PWMP 引脚输出 HIGH 如果 PWMS “1”。

PWM 周期控制寄存器

D5H	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.L	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)	0	0	0	0	0	0	0	0
D6H	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.H	-	-	-	-	PWMD.11	PWMD.10	PWMD.9	PWMD.8
读/写	-	-	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	-	-	-	-	0	0	0	0

位编号	位符号	说明
	PWMD.11-0	a. PWM 占空比控制, 能控制 PWM 波形的前半段 b. PWMS = 0, PWMP 引脚的前半段波形是 HIGH PWMS = 1, PWMP 引脚的前半段波形是 LOW c. 如果[PWMP.H, PWMP.L] ≤ [PWMD.H, PWMD.L], 设置 PWMS 为 0, PWMP 引脚输出高电平。 如果[PWMP.H, PWMP.L] > [PWMD.H, PWMD.L], 设置 PWMS 为 1, PWMP 引脚输出低电平。 d. 当[PWMD.H, PWMD.L] = 000H, 如果 PWMS “0”, PWMP 引脚输出 GND, 当[PWMD.H, PWMD.L] = 000H, 如果 PWMS “1”, PWMP 引脚输出 HIGH。

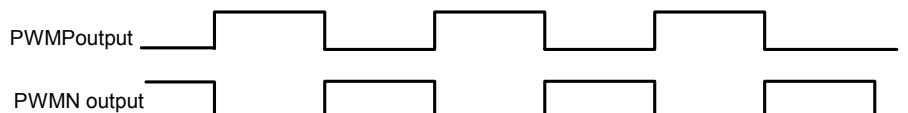
编程注意事项：

- 1) 设置 PWMLO 寄存器地址位为 55H, 选择 PWM 模块系统时钟。
- 2) 通过写适当的值到 PWM 周期控制寄存器 (PWMP) 或 PWM 占空比寄存器 (PWMD) 设置 PWM 周期: 首先设置低四位, 然后设置高四位。
- 3) 通过写适当的值到 PWM 控制寄存器 (PWMC) 的 PWMS 位选择占空比换的 PWM 输出模式。
- 4) 为了输出适当的 PWM 波形, 通过写 PWM 控制寄存器 (PWMC) 中的 EPWM 或 EPWM 位为“1”来允许 PWM 模块工作。
- 5) 如果 PWM 周期或者占空比需要改变, 操作流程如同步骤 b 或者步骤 c 说明。修改后的重载计数器的值在下一个周期开始有效。
- 6) PWMLO 寄存器中的数据修改方式与 55h 中的不同是为了增强免干扰功能。



8.8.5 PWM 负极输出(PWMN)

SH79F32 的 PWM 模块有其他形式的带 PWMP 引脚输出的固定相位关系：PWMN 引脚输出。这种引脚输出由 PWMEN 寄存器中的 EPWMN 位控制。在一个 PWM 时钟过后，PWMN 引脚将会产生一个 PWMP 引脚输出波形，这个波形有固定相位关系。



PWMP 和 PWMN 引脚输出波形

注意：

在某些特殊条件下，PWMN 与 PWM 有同样的相位，诸如基于相应 PWMS 位上的高电平或 GND。这些功能由 FLT 引脚实现。

8.8.6 PWM 停滞时间控制

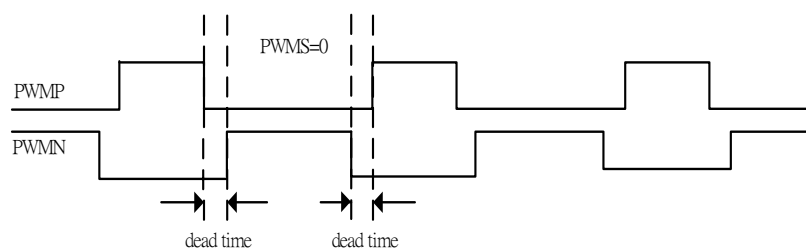
在 SH79F32 中，PWMP 和 PWMN 引脚中的输出波形中的 PWM 模块与停滞时间控制结合，主要用于电机控制应用。

由 PWM 停滞时间控制寄存器控制停滞时间。

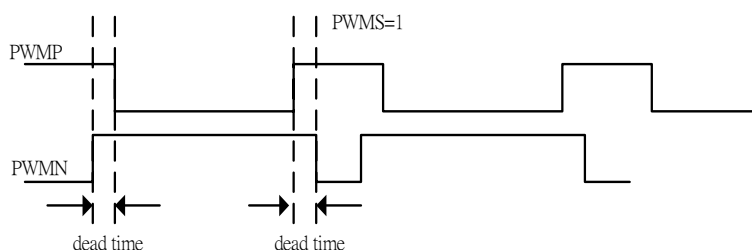
PWM 停滞时间控制寄存器

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
PWMDT	DT.7	DT.6	DT.5	DT.4	DT.3	DT.2	DT.1	DT.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	DT.7-0	以 $PWMDT * T_{osc}$ 计算停滞时间周期 停滞时间不能超过 PWM 占空比，用户必须确保 $T_{dead} < \text{占空比}/2$ 周期，否则 PWMP 引脚的输出为 HIGH（当 $PWMS=0$ 或 GND 当 $PWMS=1$ ）。

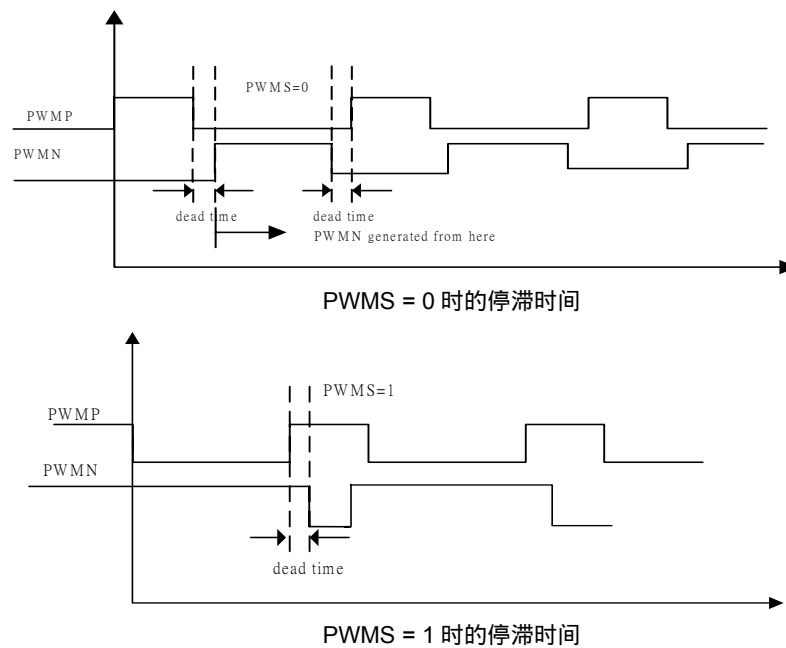


PWMS = 0 时的停滞时间



PWMS = 1 时的停滞时间

设置 EPWMN 时，PWMN 的负输出会延迟一个 PWM 周期来产生停滞时间控制。

**注意：**

必须在 PWM 输出允许前设置停滞时间，否则停滞时间不会更改。如需更改停滞时间，请您先禁止 PWM 输出（当 PWMLO 为#55h 时），然后再更改停滞时间，允许 PWM。如果您想确定与 PWM 相关的寄存器时候由于干扰更改，请最后更改 PWMLO 中与#55h 不同的数据。

要产生停滞时间，请确保 $(\text{PWM Period} - \text{PWM Duty}) > 2 * \text{PWM 停滞时间控制}$ 。否则 PWMN 的输出为高电平(PWMS=1 或 GND，PWMS=0)

PWMDT 用来控制停滞时间，固定步幅值在振荡器时钟时间上，但周期和占空比步幅值与 PWMCK1~0 有关。



8.9 低电压检测(LPD)

LPD 功能用来监测供电电压，如果供电电压低于指定值时产生内部标记。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前，软件可以采取一些保护措施。

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	-	-	LPDS3	LPDS2	LPDS1	LPDS0
读/写	读/写	R	-	-	读/写	读/写	读/写	读/写
复位值 (POR/WDT/LVR)	0	0	-	-	0	0	0	0

位编号	位符号	说明
7	LPDEN	LPD 允许 0 = 禁止低电压检测 1 = 允许低电压检测
6	LPDF	LPD 标记 0 = 无 LPD 发生，由硬件清除，即当前电压高于在 LPDS [3:0] 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 [3:0] 中设置的 LPD 电压
3-0	LPDS [3:0]	LPD 电压设置 0000 = 2.4V 0001 = 2.4V 0010 = 2.4V 0011 = 2.6V 0100 = 2.8V 0101 = 3.0V 0110 = 3.2V 0111 = 3.4V 1000 = 3.6V 1001 = 3.8V 1010 = 4.0V 1011 = 4.2V 1100 = 4.4V 1101 = 4.6V 1110 = 4.8V 1111 = 5.0V

8.10 低电压复位(LVR)

低电压复位（简称 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，系统将产生内部复位。LVR 去抖动时间 T_{LVR} 大约为 30us~60us。

通过代码选项，可以控制 LVR 的使能与禁止。

LVR 电路有以下功能，当允许 LVR 功能时：(t 表示电压低于设定电压 V_{LVR} 的时间)

- 当 $V_{DD} \leq V_{LVR}$ 且低电平周期时间 $t \geq T_{LVR}$ 时产生系统复位。
- 当 $V_{DD} > V_{LVR}$ 或 $V_{DD} < V_{LVR}$ ，但低电平周期时间 $t < T_{LVR}$ 时释放系统复位。

在交流电或大容量电池应用中，接通大负载后导致 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，保护系统不受低于标准电源环境的影响。



8.11 看门狗定时器(WDT)及复位状态

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，因此在掉电模式下仍会持续运行。当 WDT 溢出时，系统将自动复位。通过代码选项可以允许或禁止该功能。

WDT 控制位(WDT.2-0)用来选择不同的溢出时间。WDT 溢出后，WDT 溢出标志 (WDOF) 将由硬件自动设置为“1”。读或者写 RSTSTAT 寄存器，WDT 会重新开始计数。

复位状态寄存器包含有一些复位标志位，如下表所示：

RSTSTAT 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/WDT/LVR)	*	-	*	*	*	0	0	0

位编号	位符号	说明
7	WDOF	WDT 溢出标志位 0 = 没有看门狗复位发生，由上电复位或软件清 0 1 = 看门狗复位发生，由看门狗复位位置 1 在掉电状态(Power-Down)看门狗复位发生时，看门狗溢出标志不会被置 1
5	PORF	上电复位标志位 由上电复位置 1，只能由软件清除
4	LVRF	LVR 复位标志位 由 LVR 复位置 1，由软件或上电复位清除
3	CLRF	引脚复位标志位 由引脚复位置 1，由软件或上电复位清除
2-0	WDT [2:0]	WDT 溢出周期控制 000 = RC Clock/2 ¹⁷ (典型值=4096ms) 001 = RC Clock/2 ¹⁵ (典型值=1024ms) 010 = RC Clock/2 ¹³ (典型值=256ms) 011 = RC Clock/2 ¹² (典型值=128ms) 100 = RC Clock/2 ¹¹ (典型值=64ms) 101 = RC Clock/2 ⁹ (典型值=16ms) 110 = RC Clock/2 ⁷ (典型值=4ms) 111 = RC Clock/2 ⁵ (典型值=1ms)

*注: RSTSTAT 初始值根据不同类型的复位而不同。

注: 内部 RC 会有漂移, 详见相关电气特性



8.12 电源管理

8.12.1 特性

- 两种低功耗模式: 空闲模式(IDLE mode)和掉电模式(Power-Down mode)
- 两种退出低功耗模式的方法: 中断退出和复位(RESET)退出.

为减少功耗, SH79F32 提供两种低功耗模式: 空闲模式模式和省电模式。这两种模式都由 PCON 和 SUSLO 寄存器控制。

8.12.2 空闲模式

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 55H, 随即将 PCON 寄存器中的 IDL 位置 1, 使 SH79F32 进入空闲模式。如果按顺序要求的连续指令不被满足, 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) 产生一个中断。在预热定时结束之后, 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(如果被允许), LVR 复位(如果被允许))。在预热定时结束之后, 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 SH79F32 复位。然后程序从地址位 0000H 开始执行。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

掉电模式可以使 SH79F32 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 55H, 随即将 PCON 寄存器中的 PD 位置 1, 使 SH79F32 进入掉电模式。如果按顺序要求的连续指令不被满足, 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, SH79F32 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会清除 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)能使 SH79F32 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 SH79F32 会被复位。然后程序会从 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。

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



8.12.4 寄存器

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	SSTAT1	-	GF1	GF0	PD	IDL
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

位编号	位符号	说明
7-5	-	其他位请见“EUART 章节”
3-2	GF[1:0]	用于软件的通用标记
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 = 由软件设置进入掉电模式
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 = 由软件设置进入空闲模式

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	SUSL7	SUSL6	SUSL5	SUSL4	SUSL3	SUSL2	SUSL1	SUSL0
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	SUSL7-0	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 位将被硬件清除 进入空闲模式 MOV SUSLO, #55H ORL PCON, #01H 进入掉电模式 MOV SUSLO, #55H ORL PCON, #02H

注意：在空闲模式，只有 CPU 的系统时钟被中止，外围模块如定时器，ADC 等的时钟不受影响（尽管它们的时钟源也是系统时钟），所以可以应用定时器，ADC 等的中断唤醒空闲模式。但是在掉电模式下，所以时钟都将被切断。只有外部中断可以唤醒高级空闲模式。



8.13 预热计数器

SH79F32 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。

预热计数器周期

- 1) 在 32768Hz 晶体谐振器模式下，预热计数器预分频比为 2^{12} (4096)。
- 2) 在 400kHz - 8MHz 晶体/陶瓷谐振器模式下，预热计数器预分频比为 2^{12} (4096)。
- 3) 在内部 RC 模式下，预热计数器预分频比为 2^7 (128)。

注意： 在系统上电时，首先需要 6 毫秒左右的时间让内部调节器稳定，然后预热计数器才能消除起振的不稳定状态。

振荡器类型	上电复位 / 低电压复位		引脚复位		看门狗复位		从掉电模式唤醒	
	TPWRT *	Warm up	TPWRT *	Warm up	TPWRT *	Warm up	TPWRT *	Warm up
晶体谐振器	6ms	4096	6ms	4096	2ms	0	2ms	4096
陶瓷谐振器	6ms	4096	6ms	4096	2ms	0	2ms	4096
内部 RC	6ms	128	6ms	128	2ms	0	2ms	128

8.14 代码选项

OP_OSC [2:0]:

- 000: 内部 RC (8MHz)
- 010: 外部时钟(400kHz - 8MHz)
- 011: 32768Hz 晶体谐振器 (PLL 只在此模式下有效) (默认)
- 101: 晶体谐振器/陶瓷谐振器 (400kHz - 8MHz)

OP_WDT [7]:

- 0: 禁止 WDT 功能 (默认)
- 1: 允许 WDT 功能

OP_RTC [4]:

- 0: 方式 A (默认)
- 1: 方式 B.

OP_LCD [3]:

- 0: 传统电容型 LCD 模式 (默认)
- 1: SLP (Super Low Power) LCD 模式

OP_LVREN [6] 和 OP_REG [5]

OP_LVREN[6] /OP_LVRLE[5]	内部稳压源	VP3 - V _{DD}	LVR (2.3V)	LVR (3.1V)	LCD 驱动电压	应用场合
00	开	断开	开	关	2.9V	V _{DD} =3.6V~5.5V
01	开	断开	开	关	2.9V	V _{DD} =3.6V~5.5V
10	关(当 V _{DD} <3.6V) 关(当 V _{DD} >3.6V)	短接 断开	开	关	V _{DD} 2.9V	V _{DD} < 3.6V
11	开	断开	开	开	2.9V	V _{DD} =3.6V~5.5V



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	1	3
DEC @Ri	内部 RAM 减 1	0x16-0x17	2	3
INC DPTR	数据指针加 1	0xA3	1	4
MUL AB 8 x 8 16 x 8	累加器乘寄存器 B	0xA4	1	11 20
DIV AB 8 / 8 16 / 8	累加器除以寄存器 B	0x84	1	11 20
DAA	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

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



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 (not taken) (taken)	累加器为零转移	0x60	2	3 5
JNZ rel (not taken) (taken)	累加器为非零转移	0x70	2	3 5
JC rel (not taken) (taken)	C 置位转移	0x40	2	2 4
JNC rel (not taken) (taken)	C 清零转移	0x50	2	2 4
JB bit,rel (not taken) (taken)	直接寻址位置位转移	0x20	3	4 6
JNB bit,rel (not taken) (taken)	直接寻址位清零转移	0x30	3	4 6
JBC bit, rel (not taken) (taken)	直接寻址位置位转移并清该位	0x10	3	4 6
CJNE A,direct,rel (not taken) (taken)	累加器与直接寻址字节不等转移	0xB5	3	4 6
CJNE A,#data,rel (not taken) (taken)	累加器与立即数不等转移	0xB4	3	4 6
CJNE Rn,#data,rel (not taken) (taken)	寄存器与立即数不等转移	0xB8-0xBF	3	4 6
CJNE @Ri,#data,rel (not taken) (taken)	内部 RAM 与立即数不等转移	0xB6-0xB7	3	4 6
DJNZ Rn,rel (not aken) (taken)	寄存器减 1 不为零转移	0xD8-0xDF	2	3 5
DJNZ direct,rel (not taken) (taken)	直接寻址字节减 1 不为零转移	0xD5	3	4 6
NOP	空操作	0	1	1



10 电气特性

极限参数*

直流供电电压 -0.3V ~ +6.0V

输入/输出电压..... GND-0.3V ~ $V_{DD}+0.3V$

工作环境温度 -40°C ~ +85°C

存储温度 -55°C ~ +125°C

FLASH 存储器写/擦除操作..... 0°C ~ +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 1 ($V_{DD}=3.0 - 5.5V$, $GND = 0V$, $T_A = -40^{\circ}C$ 到 $+85^{\circ}C$, 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}	3.0	5.0	5.5	V	$32768Hz \leq f_{osc} \leq 8.192MHz$,
工作电流	I_{OP}	-	5	10	mA	$f_{osc} = 8.192MHz$ 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) LPD 打开, WDT 打开, PLL 打开 关闭其他所有功能
待机电流 (空闲模式: IDLE)	I_{SB1}	-	15	20	μA	$f_{osc} = 32768Hz$, PLL 关, 系统时钟= $f_{osc}/12$ 所有输出引脚无负载 所有数字输入引脚不浮动 LCD 打开 (不包括 LCD 面板) RTC 打开, WDT 关闭, LVR 打开 关闭其他所有功能
待机电流 (空闲模式: IDLE)	I_{SB2}	-	2	3	mA	$f_{osc} = 8.192MHz$ 所有输出引脚无负载 所有数字输入引脚不浮动 LCD 打开 (不包括 LCD 面板) RTC 打开, WDT 关闭, LVR 打开 关闭其他所有功能
待机电流 (掉电模式: Power-Down)	I_{SB3}	-	5	10	μA	$f_{osc} = OFF$, PLL 关 所有输出引脚无负载 所有数字输入引脚不浮动 LCD 关闭 RTC 关闭, WDT 关闭, LVR 打开 关闭其他所有功能
WDT 电流	I_{WDT}	-	1	3	μA	所有输出引脚无负载; 看门狗打开 $V_{DD}=5.0V$
LPD 电流	I_{LPD}	-	5	8	μA	
输入低电压 1	V_{IL1}	GND	-	$0.3 \times V_{DD}$	V	I/O 端口
输入高电压 1	V_{IH1}	$0.7 \times V_{DD}$	-	V_{DD}	V	I/O 端口
输入低电压 2	V_{IL2}	GND	-	$0.2 \times V_{DD}$	V	$\overline{RST}$, T0, T1, T2, T2EX, INT0, INT1, INT2, SCK, CALIN (施密特触发器)
输入高电压 2	V_{IH2}	$0.8 \times V_{DD}$	-	V_{DD}	V	$\overline{RST}$, T0, T1, T2, T2EX, INT0, INT1, INT2, SCK, CALIN (施密特触发器)
输入漏电流	I_{IL}	-1	-	1	μA	输入无上拉, $V_{IN} = V_{DD}$ or GND,
上拉电阻	R_{PH}	-	20	-	$k\Omega$	$V_{DD} = 5.0V$, $V_{IN} = GND$
输出高电压	V_{OH}	$V_{DD} - 0.7$	-	-	V	I/O 端口, $I_{OH} = -10mA$, $V_{DD} = 5.0V$
输出低电压	V_{OL1}	-	-	$GND + 0.6$	V	I/O 端口, $I_{OL} = 15mA$, $V_{DD} = 5.0V$
LCD 输出内阻	R_{ON}	-	5	-	$k\Omega$	SEG1~32, COM1~4 V1, V2, V3 的电压变化小于 0.2V. $V_{DD} = 3.0V \sim 5.0V$

*: “典型值”下的数据是在 5.0V, 25°C 下测得的, 除非另有说明



直流电气特性 1 ($V_{DD} = 2.4\text{--}3.6\text{V}$, $GND = 0\text{V}$, $T_A = -40^\circ\text{C}$ 到 $+85^\circ\text{C}$, 除非另有说明)

参数	符号	最小值	典型值	最大值	单位	条件
工作电压	V_{DD}	2.4	3.0	3.6	V	$32768\text{Hz} \leq f_{osc} \leq 4.096\text{MHz}$,
工作电流	I_{OP}	-	5	10	mA	$f_{osc} = 4.096\text{MHz}$ 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) LPD 打开, WDT 打开, PLL 打开 关闭其他所有功能
待机电流 (空闲模式: IDLE)	I_{SB1}	-	11	15	μA	$f_{osc} = 32768\text{Hz}$, PLL 关, 系统时钟= $f_{osc}/12$ 所有输出引脚无负载 所有数字输入引脚不浮动 LCD 打开 (不包括 LCD 面板) RTC 打开, WDT 关闭, LVR 打开 关闭其他所有功能
待机电流 (空闲模式: IDLE)	I_{SB2}	-	2	3	mA	$f_{osc} = 4.096\text{MHz}$ 所有输出引脚无负载 所有数字输入引脚不浮动 LCD 打开 (不包括 LCD 面板) RTC 打开, WDT 关闭, LVR 打开 关闭其他所有功能
待机电流 (掉电模式: Power-Down)	I_{SB3}	-	3	4	μA	$f_{osc} = \text{OFF}$, PLL 关 所有输出引脚无负载 所有数字输入引脚不浮动 LCD 关闭 RTC 关闭, WDT 关闭, LVR 打开 关闭其他所有功能
WDT 电流	I_{WDT}	-	1	3	μA	所有输出引脚无负载; 看门狗打开 $V_{DD}=3.0\text{V}$
LPD 电流	I_{LPD}		5	8	μA	
输入低电压 1	V_{IL1}	GND	-	$0.3 \times V_{DD}$	V	I/O 端口
输入高电压 1	V_{IH1}	$0.7 \times V_{DD}$	-	V_{DD}	V	I/O 端口
输入低电压 2	V_{IL2}	GND	-	$0.2 \times V_{DD}$	V	RST, T0, T1, T2, T2EX, INT0, INT1, INT 2, SCK, CALIN (施密特触发器)
输入高电压 2	V_{IH2}	$0.8 \times V_{DD}$	-	V_{DD}	V	RST, T0, T1, T2, T2EX, INT0, INT1, INT 2, SCK, CALIN (施密特触发器)
输出高电压	V_{OH}	$V_{DD} - 0.4$	-	-	V	I/O Ports, $I_{OH} = -3\text{mA}$, $V_{DD} = 3.0\text{V}$
输出低电压	V_{OL1}	-	-	$GND + 0.6$	V	I/O Ports, $I_{OL} = 8\text{mA}$, $V_{DD} = 3.0\text{V}$

*: “典型值”下的数据是在 3.0V, 25°C 下测得的, 除非另有说明



5V 模/数转换器电气特性

参数	符号	最小值	典型值	最大值	单位	条件
供电电压	V_{AD}	4.5	5.0	5.5	V	
精度	N_R	-	10	-	bit	$GND \leq V_{AIN} \leq V_{REF}$
A/D 输入电压*	V_{AIN}	GND	-	V_{REF}	V	
A/D 输入电阻	R_{AIN}	2	-	-	$M\Omega$	$V_{IN}=5.0V$
A/D 转换电流	I_{AD}	-	1	3	mA	ADC 模块工作, $V_{DD}=5.0V$
A/D 输入电流	I_{ADIN}			10	mA	$V_{DD} = 5.0V$
模拟电压源推荐阻抗	Z_{AIN}			10.0	$k\Omega$	
微分非线性误差	D_{LE}	-	-	± 1	LSB	$F_{OSC}=8.192MHz$, $V_{DD} = 5.0V$
积分非线性误差	I_{LE}			± 2	LSB	$F_{OSC} = 8.192MHz$, $V_{DD} = 5.0V$
满刻度误差	E_F	-	± 1	± 3	LSB	$F_{OSC} = 8.192MHz$, $V_{DD} = 5.0V$
偏移误差	E_Z	-	± 0.5	± 2	LSB	$F_{OSC} = 8.192MHz$, $V_{DD} = 5.0V$
总绝对误差	E_{AD}	-	-	± 3	LSB	$F_{OSC} = 8.192MHz$, $V_{DD} = 5.0V$
总转换时间**	T_{CON}	14	-	-	us	10 位精度和 $F_{OSC} = 8.192MHz$, $V_{DD} = 5.0V$

*: “典型值”下的数据是在 5.0V, 25°C 下测得的, 除非另有说明

**：与 AD 连接的设备电阻须小于 10k。

3V 模/数转换器电气特性

参数	符号	最小值	典型值	最大值	单位	条件
供电电压	V_{AD}	2.4	3.0	3.6	V	
精度	N_R	-	10	-	bit	$GND \leq V_{AIN} \leq V_{REF}$
A/D 输入电压*	V_{AIN}	GND	-	V_{REF}	V	
A/D 输入电阻	R_{AIN}	2	-	-	$M\Omega$	$V_{IN}=3.0V$
A/D 转换电流	I_{AD}	-	1	3	mA	ADC 模块工作, $V_{DD}=3.0V$
A/D 输入电流	I_{ADIN}			10	uA	$V_{DD} = 3.0V$
模拟电压源推荐阻抗	Z_{AIN}			10.0	$k\Omega$	
微分非线性误差	D_{LE}	-	-	± 1	LSB	$F_{OSC}=4.096MHz$, $V_{DD} = 3.0V$
积分非线性误差	I_{LE}			± 2	LSB	$F_{OSC}=4.096MHz$, $V_{DD} = 3.0V$
满刻度误差	E_F	-	± 1	± 3	LSB	$F_{OSC}=4.096MHz$, $V_{DD} = 3.0V$
偏移误差	E_Z	-	± 0.5	± 2	LSB	$F_{OSC}=4.096MHz$, $V_{DD} = 3.0V$
总绝对误差	E_{AD}	-	-	± 3	LSB	$F_{OSC}=4.096MHz$, $V_{DD} = 3.0V$
总转换时间	T_{CON}	28	-	-	us	10 位精度, $F_{OSC} = 4.096MHz$, $V_{DD} = 3.0V$

*: “典型值”下的数据是在 3.0V, 25°C 下测得的, 除非另有说明



交流电气特性

$V_{DD} = 3.0V \sim 5.5V$, $GND = 0V$, $T_A = -40^{\circ}C$ 到 $+85^{\circ}C$, $F_{OSC} = 32768Hz \sim 8.192MHz$, 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_{OSC}		1	2	s	振荡器 = 32768Hz
PLL 开始时间	T_{PLL}		2		ms	不包括振荡器起振时间
PLL 频率变化	$ \Delta F /F$	-	-	0.5	%	不间断 256 个时钟的平均频率
复位脉冲宽度	t_{RESET}	10	-	-	us	低电平有效
复位引脚上拉电阻	R_{RPH}	-	100	-	k Ω	$V_{DD} = 5.0V$, $V_{IN} = GND$
WDT RC 频率	T_{WDT}	16	32	48	kHz	

低电压复位电气特性

$V_{DD} = 3.0V \sim 5.5V$ 或 $2.4V \sim 3.6V$, $GND = 0V$, $T_A = -40^{\circ}C$ 到 $+85^{\circ}C$, 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 1	V_{LVRH}	3.0	3.1	3.2	V	LVR 允许 $V_{DD} = 3.0V \sim 5.5V$
LVR 电压 2	V_{LVRL}	2.2	2.3	2.4	V	LVR 允许 $V_{DD} = 2.4V \sim 3.6V$

32.768Hz 晶体谐振器电气特性

参数	符号	最小值	典型值	最大值	单位	条件
频率	F_{32K}	-	32768	-	Hz	
负载电容	C_L	-	12.5	-	pF	



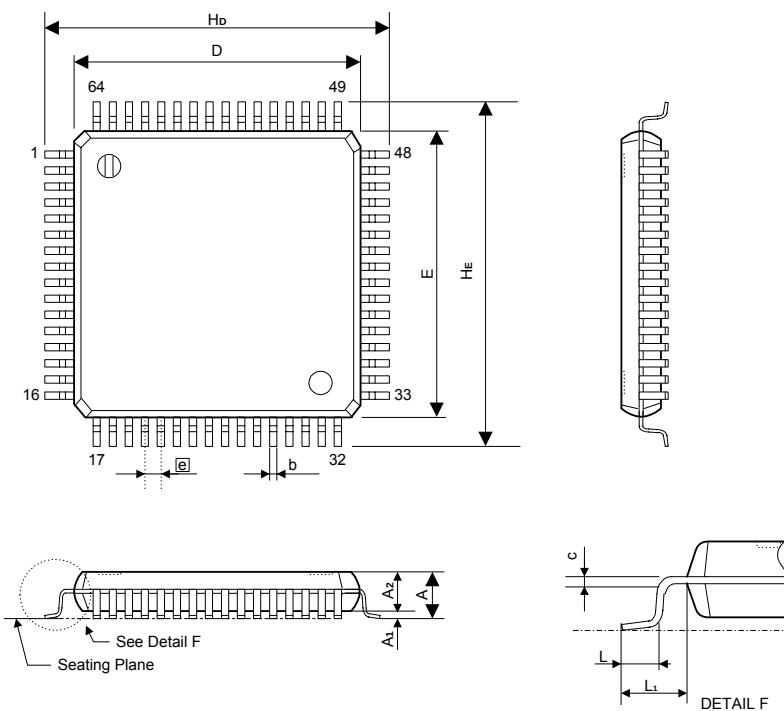
11 订购信息

产品编号	封装
SH79F32P/064PR	LQFP64



12 封装信息

LQFP64 外形尺寸
BODY SIZE: 10*10
单位: 英寸/毫米



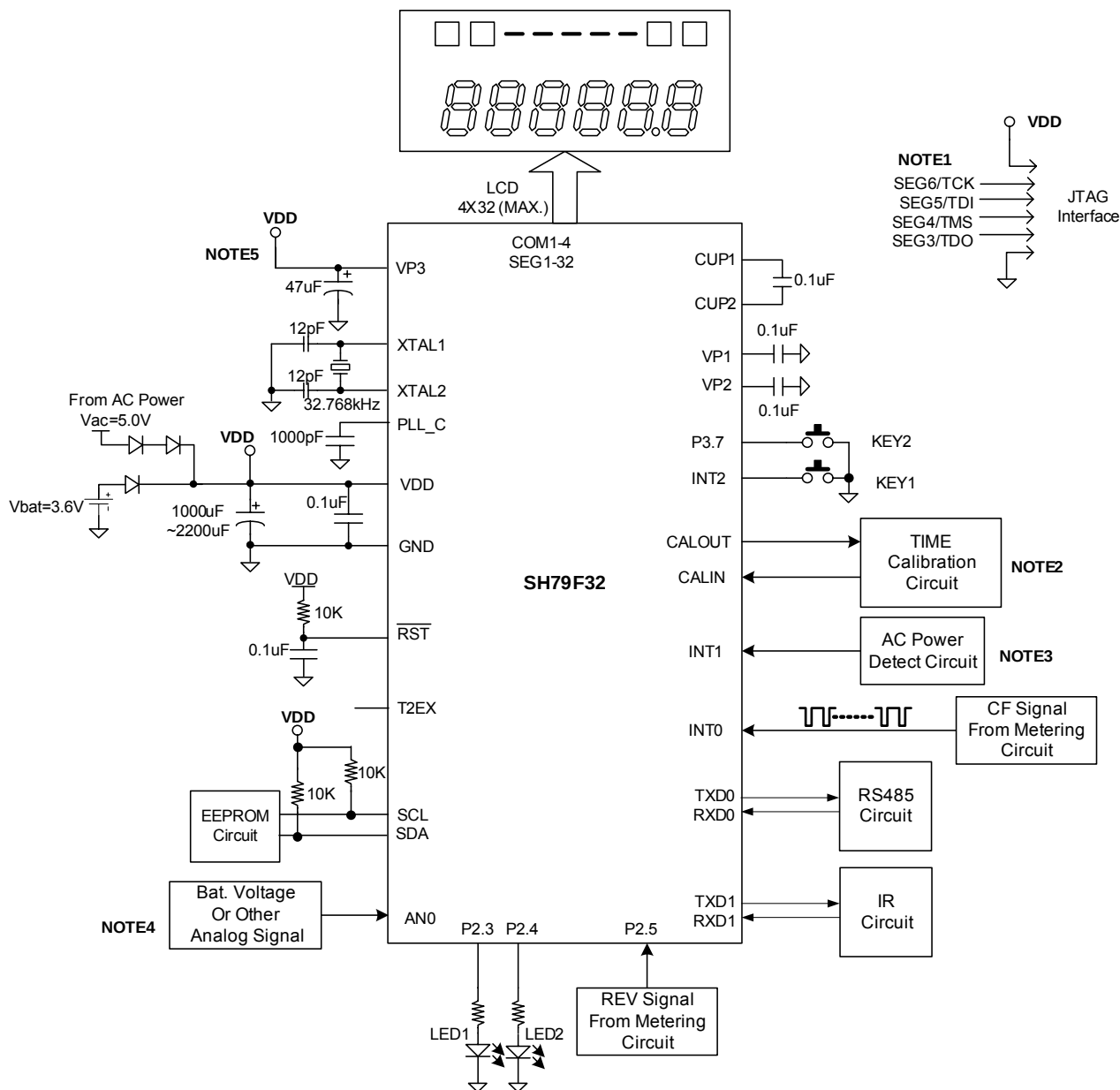
符号	英寸单位尺寸	毫米单位尺寸
A	0.063 (MAX)	1.60 (MAX)
A1	0.002 (MIN.), 0.006(MAX.)	0.05 (MIN), 0.15 (MAX)
A2	0.055 ± 0.002	1.40 ± 0.05
b	0.009 ± 0.002	0.22 ± 0.05
c	0.004 (MIN), 0.008 (MAX)	0.09 (MIN), 0.20 (MAX)
D	0.394 BASIC	10.00 BASIC
E	0.394 BASIC	10.00 BASIC
e	0.020 BASIC	0.50 BASIC
Hd	0.472 BASIC	12.00 BASIC
HE	0.472 BASIC	12.00 BASIC
L	0.024 ± 0.006	0.60 ± 0.15
L1	0.039 REF	1.00 REF



13 应用电路

单相电表.

- 4X32 LCD 显示 ;
- RS232/485, IR, E²PROM(TWI), 内建 RTC。



注：

- 1) 请参考第 17 页 JTAG 管脚连接图。
- 2) 可选电路，用于高精度时钟应用。
- 3) 也可用 LPD 功能实现 AC 电源检测。
- 4) 可选电路，用于电池电压检测。
- 5) 在应用时如果电源电压在 3.6V~5.5V 范围内，断开 VP3 和 VDD 管脚，同时 LCD 驱动电压固定为 2.9V。



14 在线支持

中颖电子为您提供在线技术支持：

www.sinowealth.com 和 bbs.sinowealth.com

我们的网站为您提供以下服务：

- 您可以在我们的网站上方便的搜索关于 IC 的资料；
- 关于公司及资料的新闻更新；
- 最新产品信息。



规格更改记录

SH79F32 规格书版本记录		
版本	记录	日期
0.7	依据 SH79F32 SPEC V0.7 (英文版) 翻译.	May. 2008
0.6	依据 SH79F32 SPEC V0.6 (英文版) 翻译.	Feb. 2008



目录

1 特性	1
2 概述	2
3 方框图	3
4 引脚配置	4
5 引脚描述	6
6 SFR 映像	8
7 标准功能	15
7.1 指令扩展	15
7.2 RAM	16
7.3 Flash 程序存储器	17
7.3.1 特性	17
7.3.2 ICP 模式下的 Flash 操作	17
7.3.3 SSP 功能	19
7.4 低电压运行	24
7.5 系统时钟和振荡器	25
7.5.1 特性	25
7.5.2 振荡器类型	26
7.5.3 谐振器负载电容选择	27
7.6 I/O 端口	28
7.6.1 特性	28
7.6.2 寄存器	28
7.6.3 端口模块图	29
7.6.4 端口共用	30
7.7 定时器	34
7.7.1 特性	34
7.7.2 Timer 0 & Timer 1	34
7.7.3 Timer 2	38
7.7.4 关于定时器的补充说明 :	43
7.8 中断	44
7.8.1 特性	44
7.8.2 程序超范围中断(OVL)	44
7.8.3 中断允许控制	45
7.8.4 中断标志	47
7.8.5 中断向量	48
7.8.6 中断优先级	48
7.8.7 中断处理	49
7.8.8 中断响应时间	50
7.8.9 外部中断输入	50
7.8.10 中断汇总	51
8 增强功能	52
8.1 LCD 驱动器	52



8.1.1 特性.....	52
8.1.2 寄存器.....	52
8.1.3 LCD 电源.....	55
8.1.4 LCD RAM 配置.....	56
8.2 SPI (串行外部设备接口) 控制器.....	59
8.2.1 特征:	59
8.2.2 信号描述.....	59
8.2.3 波特率.....	60
8.2.4 功能描述.....	60
8.2.5 工作模式.....	60
8.2.6 出错检测.....	63
8.2.7 中断.....	63
8.2.8 寄存器.....	64
8.3 两线串行接口 (TWI)	66
8.3.1 特性.....	66
8.3.2 TWI 工作模式.....	66
8.3.3 寄存器.....	69
8.4 增强型通用异步收发器 EUART	70
8.4.1 特性.....	70
8.4.2 EUART0.....	70
8.4.3 EUART1.....	77
8.4.4 寄存器.....	77
8.5 红外接口.....	82
8.6 实时时间时钟 (RTC)	83
8.6.1 特性.....	83
8.6.2 I/O 引脚.....	83
8.6.3 功能说明.....	83
8.6.4 RTC 时钟校准和补偿 (方式 A)	84
8.6.5 RTC 时钟校准和补偿 (方式 B)	86
8.6.6 寄存器.....	88
8.7 模/数转换器(ADC)	92
8.7.1 特性.....	92
8.7.2 ADC 模块图.....	92
8.8 脉冲宽度调制 (PWM)	95
8.8.1 特性.....	95
8.8.2 PWM 允许寄存器.....	95
8.8.3 PWM 定时器锁定寄存器.....	96
8.8.4 PWM 定时器.....	97
8.8.5 PWM 负极输出(PWMN).....	99
8.8.6 PWM 停滞时间控制.....	99
8.9 低电压检测(LPD).....	101
8.10 低电压复位(LVR).....	101
8.11 看门狗定时器(WDT)及复位状态.....	102
8.12 电源管理.....	103
8.12.1 特性.....	103



8.12.2 空闲模式	103
8.12.3 掉电模式	103
8.12.4 寄存器	104
8.13 预热计数器	105
振荡器类型	105
上电复位 /低电压复位	105
看门狗复位	105
从掉电模式唤醒	105
8.14 代码选项	105
9 指令集	106
10 电气特性	110
11 订购信息	115
12 封装信息	116
13 应用电路	117
14 在线支持	118
规格更改记录	119