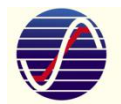
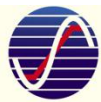


# 8 位微控制器 ASM87L164 用户手册

(Version 0.5)

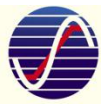


Anhui *Sine Microelectronics Co., Ltd*  
安徽赛腾微电子有限公司



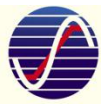
## 目录

1. 产品概述.....	6
2. 主要特色.....	6
3. 管脚描述.....	9
3-1. 管脚排列.....	9
3-2. 管脚定义.....	10
4. 功能方框图.....	12
5. CPU 及指令系统.....	13
5-1. CPU.....	13
5-2. 寻址方式.....	13
5-3. 指令系统.....	14
5-4. 特殊功能寄存器映像(SFR).....	14
5-5. SFR 说明.....	15
6. 存储器结构.....	20
6-1. Flash 存储器.....	20
6-2. 随机数据存储(SRAM).....	21
7. I/O 端口.....	23
7-1. I/O 端口相关寄存器.....	23
7-2. I/O 端口结构.....	27
7-3. I/O 端口复用.....	28
8. 时钟和复位系统.....	30
8-1. 时钟系统.....	30
8-2. 上电复位过程.....	32
8-3. 复位系统.....	33
8-4. 不同复位方式对寄存器、SRAM 与 PC 指针的影响.....	37



---

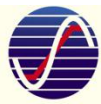
9. 省电模式.....	38
9-1. 空闲模式(Idle).....	38
9-2. 掉电模式 (Power-Down) .....	38
9-3. 深度睡眠方式.....	39
9-4. 唤醒时间.....	40
10. 中断系统.....	41
10-1. 中断源和中断向量.....	41
10-2. 中断优先级.....	42
10-3. 中断处理流程.....	43
10-4. 中断相关寄存器.....	43
11. 定时器 Timer0/Timer1.....	47
11-1. 功能框图.....	47
11-2. 控制寄存器.....	47
11-3. 工作模式.....	50
12. 定时器 2.....	53
12-1. 概述.....	53
12-2. 功能框图.....	53
12-3. 控制寄存器.....	54
13. 定时器 3.....	57
13-1. 控制寄存器.....	57
14. UART0 单元.....	59
14-1. 概述.....	59
14-2. 功能描述.....	59
14-3. 控制寄存器.....	60
15. UART1 单元.....	62
15-1. 概述.....	62



---

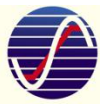
15-2. 控制寄存器.....	62
16. I2C 单元.....	64
16-1. 概述.....	64
16-2. 功能描述.....	64
16-3. 工作模式.....	65
16-4. I2C 总线上数据传输.....	74
16-5. 控制寄存器.....	76
17. SPI 单元.....	80
17-1. 概述.....	80
17-2. 特性.....	80
17-3. 功能描述.....	81
17-4. SPI 寄存器.....	83
18. 16 位 PWMC.....	86
18-1. PWM 结构框图.....	86
18-2. PWM 模块控制寄存器.....	87
19. 8+2 通道 12 位 ADC.....	92
19-1. 概述.....	92
19-2. 控制寄存器.....	93
19-3. ADC 转换步骤.....	97
20. 低电压检测(LVD).....	99
21. 可编程运算放大器(PGA).....	100
21-1. PGA 控制寄存器.....	100
22. IAP 控制单元 (Data EEPROM).....	103
22-1. IAP 相关控制寄存器.....	103
22-2. IAP 操作流程.....	105
22-3. IAP 读写范例.....	105





---

23. 电气特性.....	107
23-1. 极限参数.....	107
23-2. 推荐工作条件.....	107
23-3. ESD 特性.....	107
23-4. 直流电气特性.....	107
23-5. IRC 电气特性.....	109
23-6. ADC 电气特性.....	109
23-7. PGA 电气特性.....	110
23-8. LVD 电气特性.....	111
23-9. LVR 电气特性.....	112
23-10. 各项曲线图.....	113
24. 封装外形.....	119
25. 修订记录.....	124



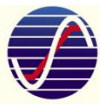
## 1. 产品概述

ASM87L164 是赛腾微最新推出的极具性价比的增强型单周期 8051 兼容内核闪存型单片机。ASM87L164 内部集成 16KB Flash ROM、256Byte 内部 SRAM、768Byte 外部 SRAM、1KBData EEPROM 以及 12 位 1MSPS 采样率的 ADC、16MHz 高频 RC 振荡器和 32KHz 低频 RC 振荡器, 内部 POR、LVR/LVD 以及 WDT 保证系统可靠运行, 另包含 PWM/LVD/UART/I2C/SPI/Timer 等外设功能模块, TWD 两线调试模块可方便用户进行程序设计。

ASM87L164 具有合理的资源配置、低功耗与超强抗干扰性能, 使之非常适合应用于各种小家电、电机控制以及无线充电等消费类电子领域。

## 2. 主要特色

- 1) 高速增强型单周期 8051 兼容内核、高度继承传统 8051 系列 MCU 开发与软件资源;
- 2) 业界领先的 0.11um 1.5V/5V 嵌入式闪存 (eFlash) 工艺制造;
- 3) 超宽工作电压 1.8~5.5V, 宽工作温度 -40~85°C, 宽工作频率 0~24MHz;
- 4) 超强抗电磁干扰性能 EFT $\geq$ 4.8KV, HBM $\geq$ 8KV, CDM $\geq$ 2KV;
- 5) 16KB Flash ROM (擦写次数>2 万次)、支持 ISP (在系统编程);
- 6) 256 字节的内部 SRAM 和 768 字节的外部 SRAM;
- 7) 1KB 的 Data EEPROM, 支持在应用编程 (IAP);
- 8) 先进的两线调试 (TWD) 功能方便用户在系统中调试和烧写程序;
- 9) 三种时钟系统
  - 外部晶体振荡器 (1~24MHz) 或者外部有源时钟灌入 (0~24MHz);
  - 内部高精度 16MHz RC 振荡器, 常温精度为 $\pm$ 1%, -40~85°C 范围内精度为 $\pm$ 2%;
  - 内部低频 32KHz RC 振荡器;
  - 支持内部高频+内部低频模式;
  - 支持外部高频+内部低频模式;
- 10) 上电复位(POR)与精准低电压复位(LVR)
  - 上电复位电路(POR);
  - 内嵌外部电源掉电复位电路;
  - 精准低电压复位 (LVR) 1.6V~2.8V 范围内 8 级可选;
- 11) LVD (低压检测)
  - 8 级可选 (4.5V/4.0V/3.6V/3.0V/2.6V/2.4V/2.0V/1.8V);
  - 产生中断信号;
- 12) PGA (可编程运算放大器)
  - 内部集成 1 路可编程运算放大器, 增益可调范围 X1~X70;
  - PGA 输出与 ADC 通道 8、P20 引脚相连;
- 13) 21 个中断源
  - 11 个内部中断源: Timer0、Timer1、Timer2、Timer3、PWM、LVD、SPI、I2C、UART0、UART1 和 ADC;
  - 共计 10 个外部中断 IO 口, 分别是 INT0/P14/P30、INT1/P15/P31、EX0/P00、EX1/P01、EX2/P05、



EX3/P06、EX4/P07、EX5/P10、EX6/P11、EX7/P12，对应两个外部中断程序入口；

- 两级中断优先级可设；

#### 14) 外围数字模块

- 2 个 16 位定时/计数器(Timer0/Timer1)，可编程预设分频系数；
- 1 个 16 位定时/计数器 (Timer2)，具有 PWM 输出、输入捕获功能；
- 16 位看门狗定时器(WDT)，可用于产生系统复位信号，也可以作为长时间定时器使用；
- 2 路独立的 UART 串行通讯口，Tx0 与 Rx0 分别与 P15 与 P16 管脚复用，Tx1 与 Rx1 分别与 P13 与 P12 管脚复用；
- 1 路 I2C 总线，速率可通过 Timer1 配置，标准模式最高支持 100kbit/s，高速模式最高支持 400kbit/s。
- SPI 总线支持 Master & Slave 模式，相位、极性、速率可选，最高速率为 8Mbps；
- 6\*16 位 PWM，可单独分开使用，也可配对成 3 对互补带死区控制 PWM 来使用，内置 8 位预分频器。PWM0、PWM1、PWM2、PWM3、PWM4、PWM5 输出分别与 P02/P22、P04/P23、P07/P24、P10/P25、P12/P26、P13/P27 复用管脚；

#### 15) (8+2)通道高达 1MSPS 12 bit ADC

- 8 路外部模拟信号输入通道，2 个内部通道(内部基准电压 (0.8V)、外部电源电压/4)；
- 支持 12 位数字转换精度，有效位>10bit；
- 5 种参考电压选择，1 个外部参考电压为 P1.6 管脚，4 个内部参考电压且可设定，分别是 VDD、4V、3V 与 2V；
- 支持内部参考电压校准修正，校正后精度在-40~85°C 范围内为 ±1%；
- 支持内部比较器，当转换值大于比较器阈值时才产生 AD 中断；
- 内嵌硬件取平均电路可选择对 4、8、16、32 次转换数据进行取平均；

#### 16) 四种省电 (Power Saving) 模式

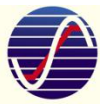
- 低速模式 (Slow)；
- 空闲模式 (Idle)；
- 睡眠模式 (Sleep)；
- 掉电模式 (Power Down)，其中深度掉电模式下电流 1uA；
- 三种省电模式均可通过外部中断唤醒；

#### 17) I/O 引脚配置 (26 个)

- 双向可配置输入输出端口：P0、P1、P2、P3，内置弱上拉电阻，可配置上下拉电阻；
- 10 个外部中断 IO，分别是 INT0/P14/P30、INT1/P15/P31、EX0/P00、EX1/P01、EX2/P05、EX3/P06、EX4/P07、EX5/P10、EX6/P11、EX7/P12，触发类型由寄存器设置；
- P0.3 与复位脚 RST 共用，RST 为低电平复位，上电复位后默认为复位脚；
- P0.0/P0.1 和晶振管脚 XIN/XOUT 复用，上电复位后默认为普通 IO 管脚；
- 外部中断 I/O 口低电平触发可唤醒处于空闲与掉电状态下的 MCU，并继续工作；
- 四种可配置工作模式：Quasi-bidirectional (准双向)、Push-Pull (推挽)、High-impedance (高阻) 与 Open-drain (开漏)；

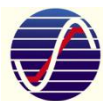
#### 18) 驱动能力

- P02/ P04/ P07/ P10/ P12/ P13 对应的 PWM 输出口支持 20mA 灌电流驱动,其余 IO 为 10mA



19) 封装形式: SOP28/TSSOP28/ TSSOP20/SOP20

型号	ROM	RAM	Data EEPROM	I/O Pin	中断 IO 口	UART	I2C	SPI	Timer (bit)	PWM (bit)	ADC (bit)	ISP	封装
ASM87L164 (28pin)	16KB	256B+ 768B	1KB	26	10	2	1	1	4x16	6x16	8x12	√	SOP20/TS SOP20/TS SOP28/ SOP28



### 3. 管脚描述

#### 3-1. 管脚排列

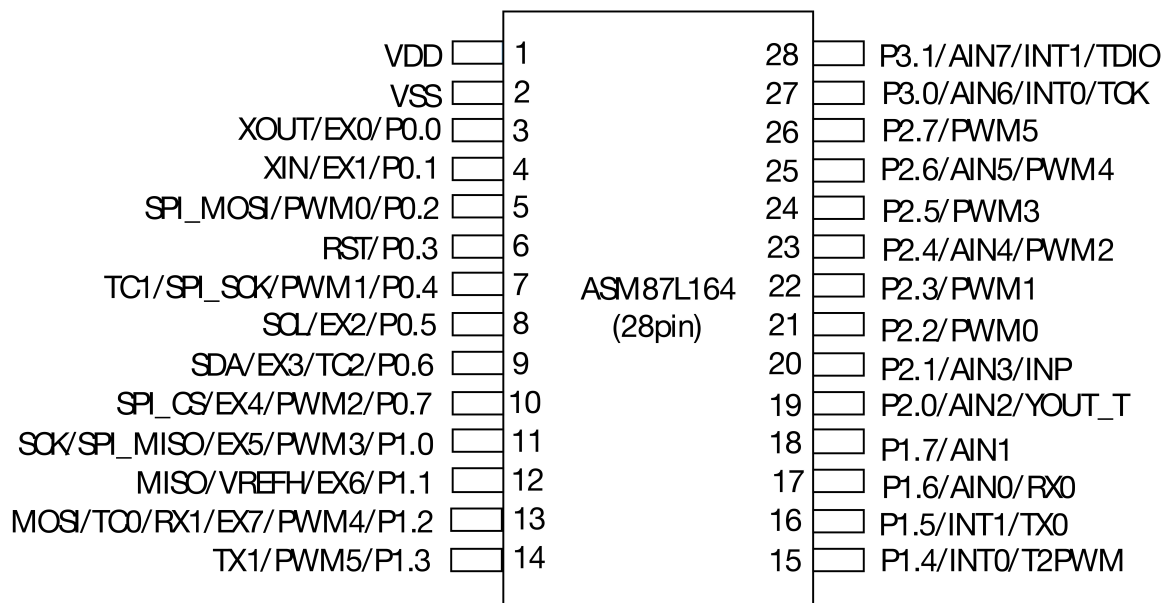


图 3-1 ASM87L164 管脚排列图 (TSSOP28 Pin)

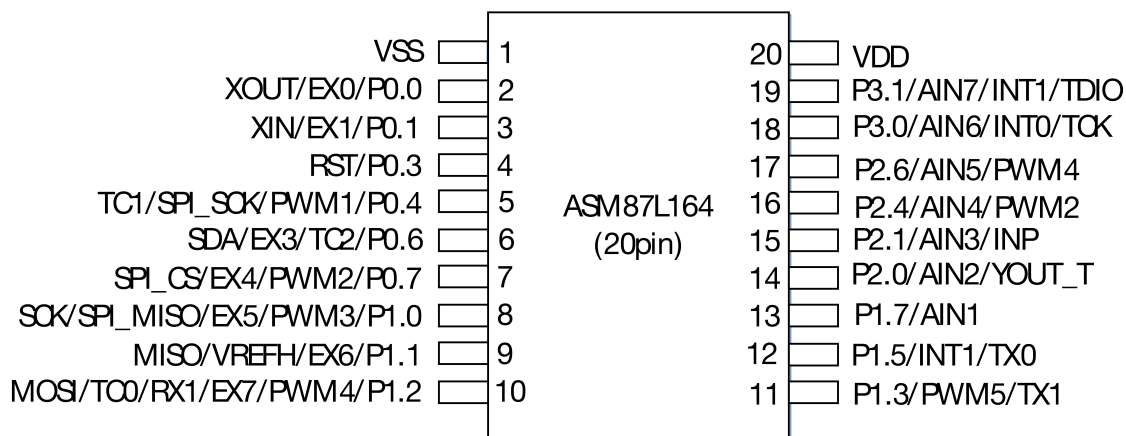
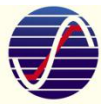
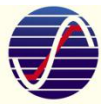


图 3-2 ASM87L164 管脚排列图 (TSSOP20 Pin)

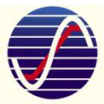


### 3-2. 管脚定义

管脚名称	管脚类型	默认类型	功能描述
VDD/VSS	Power		电源/地
P0.0/ EX0/ XOUT	I/O	准双向	P0.0: 普通双向五种模式可配置、内置弱上拉 IO, 亦为上电复位后默认设置之管脚; EX0: 外部中断 EX0 输入, 具有唤醒功能; XOUT: 外接晶振的输出管脚。
P0.1/ EX1/ XIN	I/O	准双向	P0.1: 普通双向五种模式可配置、内置弱上拉 IO, 亦为上电复位后默认设置之管脚; EX1: 外部中断 EX1 输入, 具有唤醒功能; XIN: 外接晶振的输入管脚。
P0.2/ PWM0/ SPI_MOSI	I/O	高阻输入 上下拉关闭	P0.2: 普通双向五种模式可配置、内置弱上拉 IO; PWM0: PWM 输出通道 0; SPI_MOSI: SPI 数据信号, Master 输出 Slave 输入;
RST/ P0.3/ (RST)	I/O	复位输入	RST: 系统复位输入引脚, 施密特触发器输入架构, 低电平有效, 亦为上电复位默认设置之管脚; P0.3: 普通双向五种模式可配置、内置弱上拉 IO; (RST): 兼容 164 的烧写管脚。
P0.4/ PWM1/ SPI_SCK/ TC1	I/O	高阻输入 上下拉关闭	P0.4: 普通双向五种模式可配置、内置弱上拉 IO; PWM1: PWM 输出通道 1; SPI_SCK: SPI 时钟信号。 TC1: 定时/计数器 1 的信号输入引脚。
P0.5/ EX2/ SCL	I/O	准双向 上下拉关闭	P0.5: 普通双向四种模式可配置、内置弱上拉 IO; EX2: 外部中断 EX2 输入, 具有唤醒功能; SCL: I2C 时钟信号。
P0.6/ TC2/ EX3/ SDA	I/O	准双向 上下拉关闭	P0.6: 普通双向五种模式可配置、内置弱上拉 IO; TC2: 定时器 2 的捕获输入引脚; EX3: 外部中断 EX3 输入, 具有唤醒功能; SDA: I2C 地址数据信号。
P0.7/ PWM2/ EX4/ SPI_CS	I/O	高阻输入 上下拉关闭	P0.7: 普通双向五种模式可配置、内置弱上拉 IO; PWM2: PWM 输出通道 2; EX4: 外部中断 EX4 输入, 具有唤醒功能; SPI_CS: SPI 使能信号。
P1.0/ PWM3/ EX5/ SPI_MISO/ (SCK)	I/O	高阻输入 上下拉关闭	P1.0: 普通双向五种模式可配置、内置弱上拉 IO; PWM3: PWM 输出通道 3; EX5: 外部中断 EX5 输入, 具有唤醒功能; ; SPI_MISO: SPI 数据信号, Master 输入 Slave 输出 (SCK): 兼容 081 的烧写管脚。
P1.1/ EX6/ VREFH/ (MISO)	I/O	准双向 上下拉关闭	P1.1: 普通双向五种模式可配置、内置弱上拉 IO; EX6: 外部中断 EX6 输入, 具有唤醒功能; VREFH: ADC 外部参考电压源; (MISO): 兼容 081 的烧写管脚。
P1.2/ PWM4/ EX7/ RX1/ TC0/ (MOSI)	I/O	高阻输入 上下拉关闭	P1.2: 普通双向五种模式可配置、内置弱上拉 IO; PWM4: PWM 输出通道 4; 出厂前可配置*; EX7: 外部中断 EX7 输入, 具有唤醒功能; RX1: UART1 串口输入。 TC0: 定时/计数器 0 的信号输入引脚; (MOSI): 兼容 081 的烧写管脚;



P1.3/ PWM5/ TX1	I/O	高阻输入 上下拉关闭	P1.3: 普通双向五种模式可配置、内置弱上拉IO; PWM5: PWM输出通道5; TX1: UART1串口输出。
P1.4/ INT0/ T2PWM	I/O	准双向 上下拉关闭	P1.4: 普通双向五种模式可配置、内置弱上拉IO; INT0: 外部中断 0 输入, 具有唤醒功能; T2PWM: Timer2的PWM输出。
P1.5/ INT1/ TX0	I/O	准双向 上下拉关闭	P1.5: 普通双向五种模式可配置、内置弱上拉IO; INT1: 外部中断 1 输入, 具有唤醒功能; TX0: UART0串口输出。
P1.6/ AIN0/ RX0	I/O	准双向 上下拉关闭	P1.6: 普通双向五种模式可配置、内置弱上拉IO; AIN6: ADC 输入通道6; RX0: UART0串口输入。
P1.7/ AIN1	I/O	准双向 上下拉关闭	P1.7: 普通双向五种模式可配置、内置弱上拉IO; AIN1: ADC 输入通道1。
P2.0/ AIN2/ YOUT_T	I/O	准双向 上下拉关闭	P2.0: 普通双向五种模式可配置、内置弱上拉IO; AIN2: ADC 输入通道2; YOUT_T: PGA输出端口;
P2.1/ AIN3/ INP	I/O	准双向 上下拉关闭	P2.1: 普通双向五种模式可配置、内置弱上拉IO; AIN3: ADC 输入通道3; INP: PGA信号输入端口;
P2.2/ PWM0	I/O	准双向 上下拉关闭	P2.2: 普通双向四种模式可配置、内置弱上拉IO; PWM0: PWM输出通道0;
P2.3/ PWM1	I/O	准双向 上下拉关闭	P2.3: 普通双向四种模式可配置、内置弱上拉IO; PWM1: PWM输出通道1;
P2.4/ AIN4/ PWM2	I/O	准双向 上下拉关闭	P2.4: 普通双向四种模式可配置、内置弱上拉IO; AIN4: ADC 输入通道4; PWM2: PWM 输出通道 2;
P2.5/ PWM3	I/O	准双向 上下拉关闭	P2.5: 普通双向四种模式可配置、内置弱上拉IO; PWM3: PWM输出通道3;
P2.6/ AIN5/ PWM4	I/O	准双向 上下拉关闭	P2.6: 普通双向四种模式可配置、内置弱上拉IO; AIN5: ADC 输入通道5; PWM4: PWM 输出通道 4;
P2.7/ PWM5	I/O	准双向 上下拉关闭	P2.7: 普通双向四种模式可配置、内置弱上拉IO; PWM5: PWM输出通道5;
P3.0/ AIN6/ INT0/ TCK	I/O	默认TCK, 上 下拉关闭, 如 要将其配置为 GPIO, 需将 dbc1[7:6]配置 为2'b01;  上下拉关闭	P3.0: 普通双向四种模式可配置、内置弱上拉IO; AIN6: ADC 输入通道4; INT0: 外部中断 0 输入, 具有唤醒功能; [TCK]: 2 线烧写时钟脚;
P3.1/ AIN7/ INT1/ TDIO	I/O	默认TDIO, 上 下拉关闭	P3.0: 普通双向四种模式可配置、内置弱上拉IO; AIN7: ADC 输入通道7; INT1: 外部中断 1 输入, 具有唤醒功能; [TDIO]: 2 线烧写数据脚;



## 4. 功能方框图

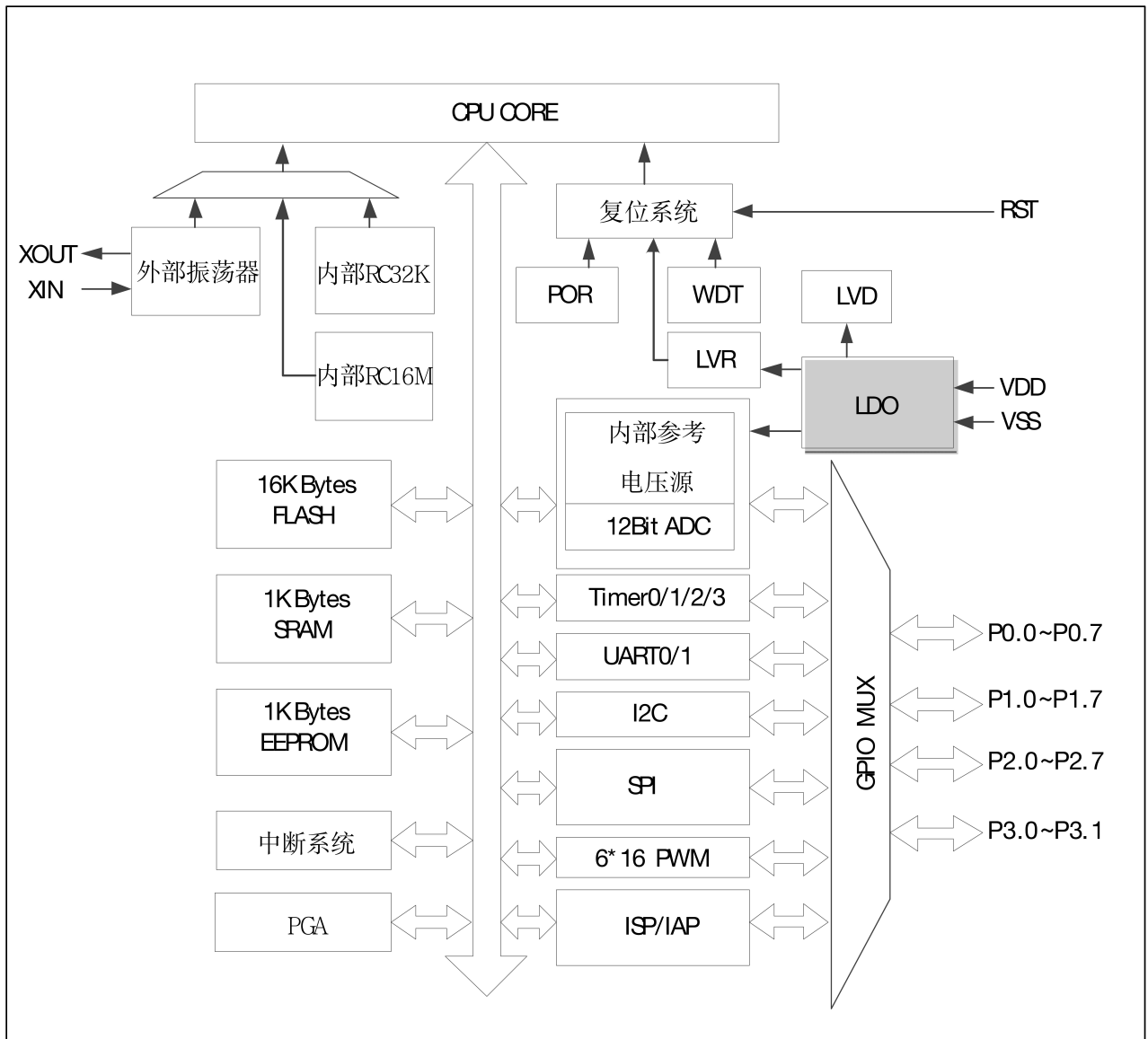
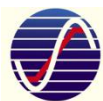


图 4-1 ASM87L164 内部功能框图





## 5. CPU 及指令系统

### 5-1. CPU

ASM87L164 的 CPU 核是一个超快速的单指令周期 8051 兼容内核，其指令兼容传统 8051 内核单片机指令，但在同样振荡频率下，较之传统的 8051 内核运行速度更快。

### 5-2. 寻址方式

ASM87L164 的 CPU 指令寻址方式包括：①立即寻址②直接寻址③间接寻址④寄存器寻址⑤相对寻址⑥变址寻址⑦位寻址。

#### 5-2-1. 立即寻址

立即寻址也称为立即数寻址，它是在指令操作数中直接给出参加运算的操作数。

#### 5-2-2. 直接寻址

在直接寻址方式中，指令中给出的不是操作数本身，而是操作数所在的单元地址。直接寻址方式只能用来表示特殊功能寄存器、内部数据寄存器和位地址空间。其中特殊功能寄存器和位地址空间只能用直接寻址方式访问。

#### 5-2-3. 间接寻址

间接寻址采用 R0 或 R1 前加“@”符号来表示。

#### 5-2-4. 寄存器寻址

寄存器寻址，寻址时对选定的工作寄存器 R7~R0、累加器 A、通用寄存器 B、地址寄存器和进位 C 中的数进行操作。其中寄存器 R7~R0 由指令码的低 3 位表示，A、B、DPTR 及进位位 C 隐含在指令码中。因此，寄存器寻址也包含一种隐含寻址方式。寄存器工作区的选择由程序状态寄存器 PSW 中的 RS1、RS0 来决定。指令操作数指定的寄存器均指当前工作区的寄存器。

#### 5-2-5. 相对寻址

相对寻址是将程序计数器 PC 中的当前值与指令第二字节给出的数相加，其结果作为转移指令的转移地址。转移地址也成为转移目的地址，PC 中的当前值成为基地址，指令第二字节给出的数成为偏移量。由于目的地址是相对于 PC 中的基地址而言，所以这种寻址方式成为相对寻址。偏移量为带符号的数，所能表示的范围为 -128~+127。这种寻址方式主要用于转移指令。

#### 5-2-6. 变址寻址

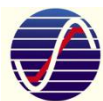
在变址寻址方式中，指令操作数指定一个存放变址基址的变址寄存器。变址寻址时，偏移量与变址基值相加，其结果作为操作数的地址。变址寄存器有程序计数器 PC 和地址寄存器 DPTR。

```
MOVC A, @A+DPTR
```

表示累加器 A 为偏移量寄存器，其内容与地址寄存器 DPTR 中的内容相加，其结果作为操作数的地址，取出该单元中的数送入累加器 A 中。

#### 5-2-7. 位寻址

位寻址是指对一些可进行位操作的内部数据存储器和特殊功能寄存器进行位操作时的寻址方式。在进行位操作时，借助于进位位 C 作为位操作累加器，指令操作数直接给出该位的地址，然后根据操作码的性质对该位进行位操作。位地址与字节直接寻址中的字节地址编码方式完全一样，主要由操作指令的性质加以区分，使用时应特别注意。



### 5-3. 指令系统

(略)

### 5-4. 特殊功能寄存器映像(SFR)

与其他 8051 内核兼容 MCU 一样，ASM87L164 单片机有着自己的 SFR。这些 SFR 寄存器的地址位于 80H~FFH，有些可以位寻址，有些不能位寻址。能够进行位寻址操作的寄存器地址末位数都是“0”或“8”，这些寄存器在需要改变单个位的数值时非常方便。所有的 SFR 特殊功能寄存器都必须使用直接寻址方式寻址。ASM87L164 的特殊功能寄存器名称及地址如下表。

表 5-1 ASM87L164 SFR 特殊功能寄存器一览表

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
F8	IP1	-	-	-	-	-	-	-
F0	B	-	-	-	-	PDP1	PUP1	SRST
E8	IEB	T3CON	T3CNTL	T3CNTH	I2CCTRL	I2CADR	I2CSTA	I2CDAT
E0	ACC	ADCCMP	PWMCNTH	PWMMR4	ADCFG0	ADCFG1	DBC1	DBC2
D8	AUXCON	PWMMR2	PWMMR3	PWMCNTL	ADCCSR	ADCCDL	ADCCDH	ADCIN_EN
D0	PSW	PWMCR0	PWMCR1	PWMCR2	PWMCR3	PWMCR4	PWMPR	PWMMR1
C8	-	-	-	-	-	-	-	LVDCTRL
C0	-	-	IAPDATA	IAPADDRH	IAPADDRL	IAPCTRL	IAPLOCK	-
B8	IP0	SPICTRL	SEND	RCVD	SPISTA	SPICLKDIV	DBC1	DBC2
B0	P3	-	P0CFGA	P0CFGB	P1CFGA	P1CFGB	P2CFGA	P2CFGB
A8	IEA	ADCREF	ADCAVE	T2_CLKDIV	T2_CAPCON	-	-	-
A0	P2	T2_PWMPL	T2_PWMPH	T2_PWMCL	T2_PWMCH	TL2	TH2	OSCM
98	S0CON	S0BUF	-	S1CON	S1BUF	PUP0P2	INTEXT_F	PDP0P2
90	P1	T2CON	DPS	-	-	-	WDTRELL	WDTRELH
88	TCON	TMOD	TL0	TL1	TH0	TH1	TMCON	INTEXT_CTRL
80	P0	SP	DPL0	DPH0	DPL1	DPH1	WDTCN	PCON



### 5-5. SFR 说明

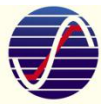
ASM87L164 特殊功能寄存器 SFR 的详细解释说明参看表 5-2。

表 5-2 ASM87L164 SFR 特殊功能寄存器详解

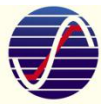
符号	地址	说明	7	6	5	4	3	2	1	0	复位值
P0	80h	P0 口数据寄存器	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	FFH
SP	81h	堆栈指针	SP[7:0]								00H
DPL0	82h	DPTR 数据指针 0 低位	DPL0[7:0]								00H
DPH0	83h	DPTR 数据指针 0 高位	DPH0[7:0]								00H
DPL1	84h	DPTR 数据指针 1 低位	DPL1[7:0]								00H
DPH1	85h	DPTR 数据指针 1 高位	DPH1[7:0]								00H
WDTCON	86h	看门狗控制寄存器	WDTCLR	WDTSTA	WDTRL	WDTRSTEN	WDTEN	FRSEL			00H
PCON	87h	电源管理控制寄存器	SMOD	-	-	-	-	-	STOP	IDLE	00H
TCON	88h	定时器控制寄存器	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H
TMOD	89h	定时器工作模式寄存器	GATE1	C/T1	M11	M01	GATE0	C/T0	M10	M00	00H
TL0	8Ah	定时器 0 低 8 位	TL0[7:0]								00H
TL1	8Bh	定时器 1 低 8 位	TH0[7:0]								00H
TH0	8Ch	定时器 0 高 8 位	TH0[7:0]								00H
TH1	8Dh	定时器 1 高 8 位	TH1[7:0]								00H
TMCON	8Eh	定时分频器	CKDIV1				CKDIV0				BBH
INTEXT_C TRL	8Fh	外部中断控制寄存器	P12EX	P11EX	P10EX	P07EX	P06EX	P05EX	P01EX	P00EX	00H
P1	90h	P1 口数据寄存器	P1.2	P1.1	P1.0	P0.7	P0.6	P0.5	P0.1	P0.0	FFH
T2CON	91h	定时器 2 控制寄存器	T3TIM_EN	pwmoen	pwmoint_en	pwmpint_en	t2_tf	t2_tr	t2_mode		00H
DPS	92h	DP 数据指针选择器	-	-	-	-	-	-	-	SPS	00H
WDTRELL	96h	看门狗计数器重载低 8 位	WDTRELL[7:0]								00H
WDTRELH	97h	看门狗计数器重载高 8 位	WDTRELH[7:0]								00H
SCON0	98h	UART0 控制寄存器	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SBUF0	99h	UART0 数据存储寄存器	SBUF[7: 0]								00H
SCON1	9Bh	UART1 控制寄存器	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H
SBUF1	9Ch	UART1 数据存储寄存器	SBUF[7: 0]								00H
PUP0P2	9Dh	P0/P2 端口上拉电阻控制	P07PU	P06PU	P05PU	P24PU	P23PU	-	P01PU	P00PU	00H



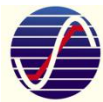
		制寄存器									
PDP0P2	9Fh	P0/P2 端口下拉电阻控制寄存器	P07PD	P06PD	P05PD	P24PD	P23PD	-	P01PD	P00PD	00H
INTEXT_F	9EH	外部中断标志寄存器	P12EF	P11EF	P10EF	P07EF	P06EF	P05EF	P01EF	P00EF	00H
P2	A0h	P2 口数据寄存器	-	-	-	P2.4	P2.3	-	-	-	FFH
T2_PWMPL	A1h	定时器 2PWM 周期寄存器低 8 位	T2_PWMPL[7:0]								00H
T2_PWMPH	A2h	定时器 2PWM 周期寄存器高 8 位	T2_PWMPH[7:0]								00H
T2_PWMCL	A3h	定时器 2PWM 比较寄存器低 8 位	T2_PWMCL[7:0]								00H
T2_PWMCH	A4h	定时器 2PWM 比较寄存器高 8 位	T2_PWMCH[7:0]								00H
TL2	A5h	定时器 2 低 8 位	TL2[7:0]								00H
TH2	A6h	定时器 2 高 8 位	TH2[7:0]								00H
OSCM	A7h	时钟选择寄存器	FSCCLK_SEL	RC32KEN	LDOEN	CLKSEL	OSCEN	RCEN	RCFR		03H
IEA	A8h	中断使能控制寄存器 0	EA	EADC	EPWM	ES	ET1	EX1	ET0	EX0	00H
ADCREFL	A9h	ADC/LVR 选择控制寄存器	LVREN	ADCPSEL[3]	ADSEL[7]	REFSEL			OFFSETX	OFFSET	0Ch
ADCAVE	AAh	ADC 取平均寄存器	VREFSEL	INT1EN	INT0EN		AVE_EN	AVES[2]	AVES[1]	AVES[0]	60H
T2_CLKDIV	ABh	定时器 2 时钟分频寄存器	-				T2_CLKDIV[3:0]				00H
T2_CAPCON	ACH	定时器 2 捕获控制寄存器	-	-	-	capcr	cap21s[1:0]		Capen0	Capf0	00H
P3CFGAL	ADh	P3 口模式设置寄存器 A	-	-	-	-			P31_CFGA	P30_CFGA	FFH
P3CFGBL	Aeh	P3 口模式设置寄存器 B	-	-	-	-			P31_CFGB	P30_CFGB	00H
PUP2	Afh	P2 口上拉电阻控制寄存器	PUP2[7]	PUP2[6]	PUP2[5]	PUP2[4]	PUP2[3]	PUP2[2]	PUP2[1]	PUP2[0]	00H
P3	B0h	P3 口数据寄存器	-	-	-	-			P3.1	P3.0	FFH
PDP2	B1h	P2 口下拉电阻控制寄存器	PDP2[7]	PDP2[6]	PDP2[5]	PDP2[4]	PDP2[3]	PDP2[2]	PDP2[1]	PDP2[0]	00H



P0CFGA	B2h	P0 口模式设置寄存器 A	P07_ CFGA	P06_ CFGA	P05_ CFGA	P04_ CFGA	P03_ CFGA	P02_ CFGA	P01_ CFGA	P00_ CFGA	1FH
P0CFGB	B3h	P0 口模式设置寄存器 B	P07_ CFGB	P06_ CFGB	P05_ CFGB	P04_ CFGB	P03_ CFGB	P02_ CFGB	P01_ CFGB	P00_ CFGB	E0H
P1CFGA	B4h	P1 口模式设置寄存器 A	P17_ CFGA	P16_ CFGA	P15_ CFGA	P14_ CFGA	P13_ CFGA	P12_ CFGA	P11_ CFGA	P10_ CFGA	F7H
P1CFGB	B5h	P1 口模式设置寄存器 B	P17_ CFGB	P16_ CFGB	P15_ CFGB	P14_ CFGB	P13_ CFGB	P12_ CFGB	P11_ CFGB	P10_ CFGB	08H
P2CFGA	B6h	P2 口模式设置寄存器 A	-	-	-	P24_ CFGA	P23_ CFGA	-	-	-	E7H
P2CFGB	B7h	P2 口模式设置寄存器 B	-	-	-	P24_ CFGB	P23_ CFGB	-	-	-	18H
IP0	B8h	中断优先级寄存器	IP0[7:0]								00H
SPICTRL	B9h	SPI 控制寄存器	SPICTRL[7:0]								60H
SEND	Bah	SPI 发送数据寄存器	SEND[7:0]								00H
RCVD	BBh	SPI 接收数据寄存器	RCVD[7:0]								00H
SPISTA	BCh	SPI 状态寄存器	SPISTA[7:0]								00H
SPICLKDI V	BDh	SPI 主模式时钟分频寄存器	SPICLKDIV[7:0]								05H
DBC1	BEh	PWM 死区控制寄存器 1	-				PWM0/1	PWM2/3	PWM4/5		00H
DBC2	BFh	PWM 死区控制寄存器 2	RFU	PERIOD				DBSEL2	DBSEL1	DBSEL0	00H
IAPDATA	C2h	IAP 数据寄存器	IAPDATA[7: 0]								00H
IAPADDRH	C3h	IAP 地址寄存器高位	IAPADDRH[7: 0]								00H
IAPADDRL	C4h	IAP 地址寄存器低位	IAPADDRL[7: 0]								00H
IAPCTRL	C5h	IAP 模式控制寄存器	IAPLOCK EN	-	IAPLOCK	IAPER	IAPWR	IAPINF	IAPEN	80H	
IAPLOCK	C6h	IAPLOCK 寄存器	IAPLOCK[7: 0]								00H
IOMUX	C7h	IO 功能复用控制寄存器	见后文章节描述								00H
LVDCTRL	CFh	LVD 控制寄存器	见后文章节描述								00H
PSW	D0h	程序状态寄存器	CY	AC	F0	RS1	RS0	OV	F1	P	00H
PWMCR0	D1h	PWM0 比较寄存器	PWMCR0[7:0]								00H
PWMCR1	D2h	PWM1 比较寄存器	PWMCR1[7:0]								00H
PWMCR2	D3h	PWM2 比较寄存器	PWMCR2[7:0]								00H
PWMCR3	D4h	PWM3 比较寄存器	PWMCR3[7:0]								00H

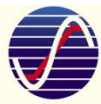


PWMCR4	D5h	PWM4 比较寄存器	PWMCR4[7:0]								00H
PWMPR	D6h	PWM 周期寄存器	PWMPR[7:0]								00H
PWMMR1	D7h	PWM 管理寄存器 1	-	INTF	OVF	POLAR0~4				00H	
AUXCON	D8h	辅助功能寄存器	-	PDFL	-	AD_Thresh	PINOSC	PRST	INT1SEL	INT0SEL	00H
PWMMR2	D9h	PWM 管理寄存器 2	PWME	-	PWMRL	-	PWMSCALE				00H
PWMMR3	DAh	PWM 管理寄存器 3	CF1_FL	CF1_EN	CF0_FL	CF0_EN	-	CAPE	EG1	EG0	00H
PWMCNTL	DBh	PWM 计数器寄存器低 8 位	PWMCNTL[7:0]								00H
ADCCSR	DCh	ADC 控制寄存器	ADC_EN	ADCCLKS	EOCO	ADC_START	ADCPSEL[2:0]				60H
ADCDRL	DDh	ADC 数据寄存器低位	LVRSEL				ADCCDL[3:0]*				00H
ADCDRH	DEh	ADC 数据寄存器高位	ADCCDH[7:0]								00H
ADCIN_EN	DFh	ADC 输入口使能寄存器	VREF_EN	P16	P15	P14	P13	P12	P11	P10	00H
ACC (A)	E0h	累加器	ACC[7:0]								00H
ADCCMP	E1h	ADC 阈值比较寄存器	ADCCMP[7:0]								00H
PWMCNTH	E2h	PWM 计数器寄存器高 8 位	PWMCNTH[7:0]								00H
PWMMR4	E3h	PWM 管理寄存器 4	-	-	-	PWMOE0~4				00H	
ADCFG0	E4h	ADC 配置寄存器 0	MODE	SH_CTR		OTR	R_CAL				00H
ADCFG1	E5h	ADC 配置寄存器 1	Reserved			TRIM	ADC Trim				00H
P0ANA	E6h	P0 口模拟输入配置寄存器									FFH
IEB	E8h	中断使能控制寄存器 1	见后文章节描述								00H
T3CON	E9h	定时器 3 控制寄存器	见后文章节描述								00H
T3CNTL	EAh	定时器 3 低 8 位计数器	见后文章节描述								00H
T3CNTH	EBh	定时器 3 高 8 位计数器	见后文章节描述								00H
I2CCTRL	ECh	I2C 控制寄存器	见后文章节描述								00H
I2CADR	EDh	I2C 地址寄存器	见后文章节描述								00H
I2CSTA	EEh	I2C 状态寄存器	见后文章节描述								F8H
I2CDAT	EFh	I2C 数据寄存器	见后文章节描述								00H
B	F0h	B 寄存器	B[7:0]								00H
CKCON	F1h	外部 RAM 访问时钟寄	访问外部 RAM 时, 应将该寄存器配置为 0xAA								00H



		寄存器									
PDP1	F5h	P1 口上拉电阻控制寄存器	见后文章节描述							00H	
PUP1	F6h	P1 口下拉电阻控制寄存器	见后文章节描述							00H	
SRST	F7h	软件复位寄存器	-	-	-	-	-	-	-	SRSTEN	00H

注：ADCCDL 是多用途寄存器，除了用来存储 ADC 转换结果低 4 位，还用于设置 LVR 复位电压点。



## 6. 存储器结构

ASM87L164 具有 16KBFlash ROM、256 字节片内 SRAM、768 字节片外 RAM 与 1K 字节数据 EEPROM。这些存储器的地址空间与结构示意图如下：

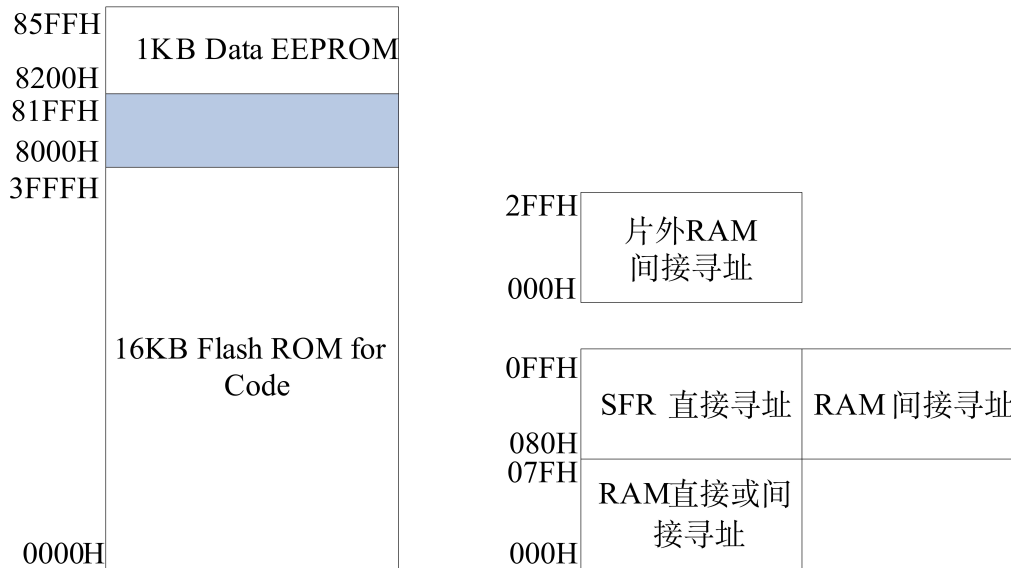


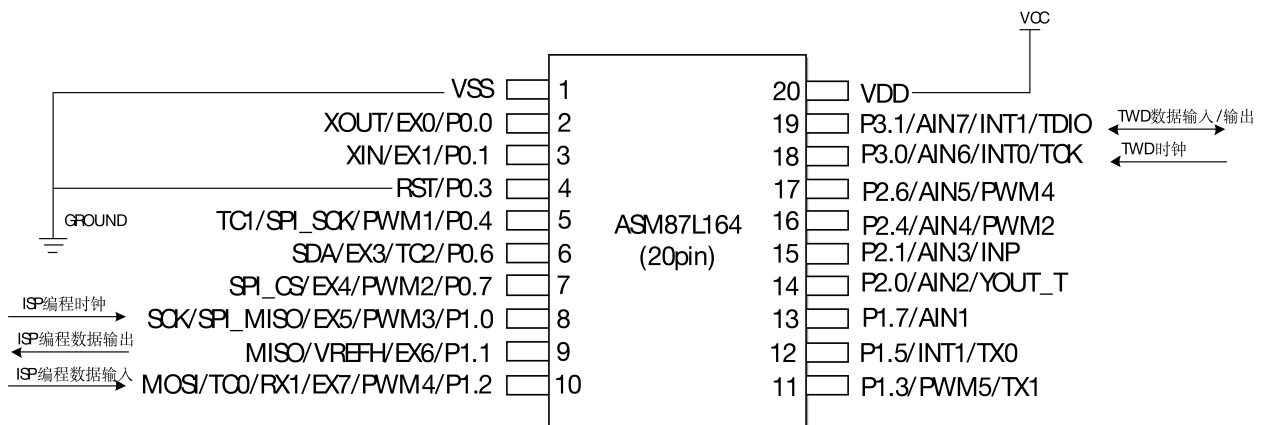
图 6-1 Flash ROM 和 SRAM 结构框图

### 6-1. Flash 存储器

ASM87L164 有 16KB 的 Flash ROM，地址是 0000H~3FFFH。此 16KB 存储器可以反复擦写读，擦写次数可达 2 万次，数据保持能力为 10 年。ASM87L164 通过专用烧录器对 Flash ROM 进行擦除、写入、读出以及加密等操作。ASM87L164 采用严格的加密技术，加密后 Flash ROM 不可读，也不可写，必须整片擦除后才能再次进行用户编程写入操作。

地址为 8200H~85FFH 区域的 Flash ROM 可以作为 Data EEPROM 使用(即支持用户在应用程序中对此区域进行擦除、写入以及读取操作，详细操作请看 IAP 章节)，此 1K 字节 Flash 可反复擦写 10 万次，也可以通过 MOVC 指令读取数据。

ASM87L164 的 Flash ROM 通过 P0.3 (RST)、P1.0 (SCK)、P1.1 (MISO)、P1.2 (MOSI) 以及 VDD、VSS 来进行编程，也可以通过 P3.0 (TCK)、P3.1 (TDIO) 以及 VDD、VSS 进行编程。下图以 20 脚封装片为例给出了编程状态下的管脚连接。注：FLASH 编程时，需要将 P13 接至 VDD。





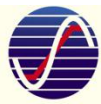


图 6-2 ASM87L164 Flash ROM 编程状态下管脚连接图

## 6-2. 随机数据存储(SRAM)

ASM87L164 为数据存储提供了内部 256 字节和外部 768 字节的 RAM，下列为存储器空间分配。

- 低位 128 字节的内部 RAM (地址从 00H 到 7FH) 可直接或间接寻址
- 高位 128 字节的内部 RAM (地址从 80H 到 FFH) 只能间接寻址
- 特殊功能寄存器 (SFR, 地址从 80H 到 FFH) 只能直接寻址
- 片外 768 字节的外部 RAM (地址从 000H 到 2FFH) 只能通过 DPTR 间接访问

高位 128 字节内部 RAM 占用的地址空间和 SFR 的地址空间相同，但在物理上与 SFR 的空间是分离的。当一个指令访问地址高于 7FH 的内部位置时，CPU 可以根据指令的寻址方式来区分是访问高位 128 字节 RAM 数据还是访问 SFR。

对于低 128 字节数据存储，完全兼容标准 8051 寻址方式。

地址 00H ~ 7FH RAM 既可以采用直接寻址，也可以采用间接寻址；

地址 00H ~ 1FH RAM 为寄存器存储器；

地址 20H ~ 2FH RAM 为可位寻址区；

地址 30H ~ 7FH RAM 为通用 RAM。

对于高 128 字节数据存储，地址 80H ~ FFH 的 RAM 仅可采取间接寻址的方式访问，特殊功能寄存器 (SFR) 80 ~ FFH 只可直接寻址。

因为高 128 字节的 RAM 和 SFR 占用相同的地址，所以高 128 字节 RAM 只能采用间接寻址的方式进行访问。尽管 RAM 和 SFR 的地址相同，但它们在物理上是独立的。

在程序访问该段空间时，由直接寻址和间接寻址的方式来区分是访问 SFR 还是高 128 字节的 RAM。使用时请注意区别。

以如下两条语句为例，再比较一下访问 SRF 与 RAM 的区别。

```

////////////////////////////////////
MOV 0A0H, #55H;
////////////////////////////////////

```

该条指令为一条直接寻址的指令，根据上面的描述我们知道，直接寻址要访问的是对应地址的特殊功能寄存器。0A0H 对应的为 P2 口寄存器，执行的结果是在 P2 口上输出 55H。

```

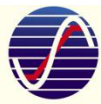
////////////////////////////////////
MOV R0, #0A0H;
MOV @R0, #55H;
////////////////////////////////////

```

这是间接寻址的方式，根据上面的描述，可以得到的结论是该条指令要访问的高 128 字节的 RAM，所以执行该条指令结果是对 0a0H 地址处 RAM 写上 55H 值。

对于片外的 768 字节 RAM，它的地址空间是从 000H 到 2FFH，和片内 RAM 地址上有重叠部分，但物理上是独立的。**要特别注意，在访问片外 RAM 时，需将外部 RAM 访问时钟寄存器 CKCON (地址 F1H) 初始化为 0xAA。**片外 768 字节的 RAM 只能借助 DPTR 寄存器来间接访问，举例如下：

要将 34H 写入片外 RAM 的 022H 地址



////////////////////////////////////

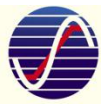
```
MOV CKCON, #0AAH
MOV  DPL, #22H
MOV  DPH, #00H
MOVA, #34H
MOVX @DPTR, A
```

////////////////////////////////////

片外 RAM 的 022H 地址读出该地址保存的值

```
MOV  DPL, #22H
MOV  DPH, #00H
MOVX A, @DPTR
MOVR1, A
```

////////////////////////////////////



## 7. I/O 端口

ASM87L164 提供最多 26 个通用输入输出 IO 端口，此 26 个 IO 端口既可以用作普通 IO 口，也可以与其他功能复用。IO 可以通过模式配置寄存器配置成以下五种 IO 模式中的一种：准双向 IO 模式（传统 8051 的 IO 模式）、推挽输出模式、高阻输入、开漏输出模式、模拟输入模式（仅 ADC 通道）。

准双向 IO 结构：也是传统 8051 单片机 IO 模式，当对一个 I/O 口写“0”时，它有很强 (>20mA) 的拉低(Sink)能力，此时把该 I/O 口视为输出“Output”；如果对该 I/O 口写“1”后，该 I/O 口会有弱上拉的能力保持该 I/O 口为高输出，而此时允许使用者从外部输入信号，盖过该弱上拉。简单地说，写“0”代表强的“Output”，写“1”则代表“Input”。

推挽结构：如果有特别的强输出需求，用户可以设定对应的寄存器 PxCFGA/PxCFGB，让 I/O 口输出很强的“1”，此时 I/O 口具有较强的电流驱动能力，而非准双向 IO 中的允许被外部输入盖过的弱输出“1”。

高阻输入结构：高阻态，仅作为 IO 口输入使用。

开漏输出结构：IC 内部的上拉电阻断开，需要外接上拉电阻。

模拟输入结构：P0 可配置为模拟 IO 口。

### 7-1. I/O 端口相关寄存器

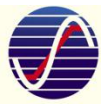
#### 7-1-1. 端口配置寄存器

ASM87L164 的 I/O 端口配置寄存器 PxCFGn 与数据寄存器分别如表 7-1 与 7-2 所示。

表 7-1 I/O 端口配置寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
B2H	P0CFGA	P07_ CFGA	P06_ CFGA	P05_ CFGA	P04_ CFGA	P03_ CFGA	P02_ CFGA	P01_ CFGA	P00_ CFGA	6BH
B3H	P0CFGB	P07_ CFGB	P06_ CFGB	P05_ CFGB	P04_ CFGB	P03_ CFGB	P02_ CFGB	P01_ CFGB	P00_ CFGB	94H
B4H	P1CFGA	P17_ CFGA	P16_ CFGA	P15_ CFGA	P14_ CFGA	P13_ CFGA	P12_ CFGA	P11_ CFGA	P10_ CFGA	F2H
B5H	P1CFGB	P17_ CFGB	P16_ CFGB	P15_ CFGB	P14_ CFGB	P13_ CFGB	P12_ CFGB	P11_ CFGB	P10_ CFGB	0DH
B6H	P2CFGA	P27_ CFGA	P26_ CFGA	P25_ CFGA	P24_ CFGA	P23_ CFGA	P22_ CFGA	P21_ CFGA	P20_ CFGA	FFH
B7H	P2CFGB	P27_ CFGB	P26_ CFGB	P25_ CFGB	P24_ CFGB	P23_ CFGB	P22_ CFGB	P21_ CFGB	P20_ CFGB	00H
ADH	P3CFGA	P37_ CFGA	P36_ CFGA	P35_ CFGA	P34_ CFGA	P33_ CFGA	P32_ CFGA	P31_ CFGA	P30_ CFGA	FFH
AEH	P3CFGB	P37_ CFGB	P36_ CFGB	P35_ CFGB	P34_ CFGB	P33_ CFGB	P32_ CFGB	P31_ CFGB	P30_ CFGB	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

IO 端口控制方式说明：



{Pxx\_CFGA, Pxx\_CFGB} = {0,0} : 开漏输出; {Pxx\_CFGA, Pxx\_CFGB} = {1,0} : 准双向;  
{Pxx\_CFGA, Pxx\_CFGB} = {0,1} : 高阻态; {Pxx\_CFGA, Pxx\_CFGB} = {1,1} : 推挽输出。

注: 具有 PWM 输出功能的 P0.2, P0.4, P0.7, P1.0, P1.2, P1.3 I/O 端口上电复位默认状态为高阻态, 用作输入口, P0.3 口上电复位默认状态为高阻输入 (复位功能), 其他 I/O 口上电复位后默认模式为双向。

表 7-2 I/O 端口数据寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
80H	P0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	FFH
90H	P1	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	FFH
A0H	P2	P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	FFH
B0H	P3	--	--	--	--	--	--	P3.1	P3.0	FFH
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

7-1-2. PUP0P2(9DH) : P0/P2 端口上拉电阻控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
9DH	PUP0P2	P07PU	P06PU	P05PU	P24PU	P23PU	-	P01PU	P00PU	00H
读/写		W/R	W/R	W/R	W/R	W/R	-	W/R	W/R	

备注: 此寄存器的 bit4 为 P24 的下拉使能, 非上拉使能。

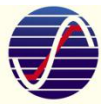
位编号	位符号	说明		
7	P07PU	P0.7 上拉控制,	1: 上拉使能;	0: 上拉关闭
6	P06PU	P0.6 上拉控制,	1: 上拉使能;	0: 上拉关闭
5	P05PU	P0.5 上拉控制,	1: 上拉使能;	0: 上拉关闭
4	<b>P24PD</b>	P2.4 下拉控制,	1: 下拉使能;	0: 下拉关闭
3	<b>P23PU</b>	P2.3 上拉控制,	1: 上拉使能;	0: 上拉关闭
2	-	保留		
1	P04PU	P0.4 上拉控制,	1: 上拉使能;	0: 上拉关闭
0	P02PU	P0.2 上拉控制,	1: 上拉使能;	0: 上拉关闭

7-1-3. PDP0P2 (9FH) : P0/P2 端口下拉电阻控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
9FH	PDP0P2	P07PD	P06PD	P05PD	<b>P24PU</b>	P23PD	-	P01PD	P00PD	00H
读/写		W/R	W/R	W/R	<b>W/R</b>	W/R	-	W/R	W/R	

备注: 此寄存器的 bit4 为 P24 的上拉使能, 非下拉使能。

位编号	位符号	说明		
7	P07PD	P0.7 下拉控制,	1: 下拉使能;	0: 下拉关闭
6	P06PD	P0.6 下拉控制,	1: 下拉使能;	0: 下拉关闭
5	P05PD	P0.5 下拉控制,	1: 下拉使能;	0: 下拉关闭
4	<b>P24PU</b>	P2.4 上拉控制,	1: 上拉使能;	0: 上拉关闭
3	<b>P23PD</b>	P2.3 下拉控制,	1: 下拉使能;	0: 下拉关闭



2	-	保留
1	P04PD	P0.4 下拉控制, 1: 下拉使能; 0: 下拉关闭
0	P02PD	P0.2 下拉控制, 1: 下拉使能; 0: 下拉关闭

7-1-4. PUP1 (F6H) : P1 端口上拉电阻控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
F6H	PUP1	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明		
7	P17PU	P1.7 上拉控制,	1: 上拉使能;	0: 上拉关闭
6	P16PU	P1.6 上拉控制,	1: 上拉使能;	0: 上拉关闭
5	P15PU	P1.5 上拉控制,	1: 上拉使能;	0: 上拉关闭
4	P14PU	P1.4 上拉控制,	1: 上拉使能;	0: 上拉关闭
3	P13PU	P1.3 上拉控制,	1: 上拉使能;	0: 上拉关闭
2	P12PU	P1.2 上拉控制,	1: 上拉使能;	0: 上拉关闭
1	P11PU	P1.1 上拉控制,	1: 上拉使能;	0: 上拉关闭
0	P10PU	P1.0 上拉控制,	1: 上拉使能;	0: 上拉关闭

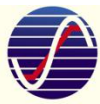
7-1-5. PDP1 (F5H) : P1 端口下拉电阻控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
F5H	PDP1	P17PD	P16PD	P15PD	P14PD	P13PD	P12PD	P11PD	P10PD	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R

位编号	位符号	说明		
7	P17PD	P1.7 下拉控制,	1: 下拉使能;	0: 下拉关闭
6	P16PD	P1.6 下拉控制,	1: 下拉使能;	0: 下拉关闭
5	P15PD	P1.5 下拉控制,	1: 下拉使能;	0: 下拉关闭
4	P14PD	P1.4 下拉控制,	1: 下拉使能;	0: 下拉关闭
3	P13PD	P1.3 下拉控制,	1: 下拉使能;	0: 下拉关闭
2	P12PD	P1.2 下拉控制,	1: 下拉使能;	0: 下拉关闭
1	P11PD	P1.1 下拉控制,	1: 下拉使能;	0: 下拉关闭
0	P10PD	P1.0 下拉控制,	1: 下拉使能;	0: 下拉关闭

7-1-6. PUP2 (AFH) : P2 端口上拉电阻控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
AFH	PUP2	P27PU	P26PU	P25PU	P31PU	P30PU	P22PU	P21PU	P20PU	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	



位编号	位符号	说明		
7	P27PU	P2.7 上拉控制,	1: 上拉使能;	0: 上拉关闭
6	P26PU	P2.6 上拉控制,	1: 上拉使能;	0: 上拉关闭
5	P25PU	P2.5 上拉控制,	1: 上拉使能;	0: 上拉关闭
4	P31PU	P3.1 上拉控制,	1: 上拉使能;	0: 上拉关闭
3	P30PU	P3.0 上拉控制,	1: 上拉使能;	0: 上拉关闭
2	P22PU	P2.2 上拉控制,	1: 上拉使能;	0: 上拉关闭
1	P21PU	P2.1 上拉控制,	1: 上拉使能;	0: 上拉关闭
0	P20PU	P2.0 上拉控制,	1: 上拉使能;	0: 上拉关闭

7-1-7. PDP2 (B1H) : P2 端口下拉电阻控制寄存器

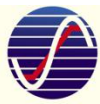
地址	名称	7	6	5	4	3	2	1	0	上电复位值
B1H	PDP2	P27PD	P26PD	P25PD	P31PD	P30PD	P22PD	P21PD	P20PD	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R

位编号	位符号	说明		
7	P27PD	P2.7 下拉控制,	1: 下拉使能;	0: 下拉关闭
6	P26PD	P2.6 下拉控制,	1: 下拉使能;	0: 下拉关闭
5	P25PD	P2.5 下拉控制,	1: 下拉使能;	0: 下拉关闭
4	P31PD	P3.1 下拉控制,	1: 下拉使能;	0: 下拉关闭
3	P30PD	P3.0 下拉控制,	1: 下拉使能;	0: 下拉关闭
2	P22PD	P2.2 下拉控制,	1: 下拉使能;	0: 下拉关闭
1	P21PD	P2.1 下拉控制,	1: 下拉使能;	0: 下拉关闭
0	P20PD	P2.0 下拉控制,	1: 下拉使能;	0: 下拉关闭

7-1-8. IOMUX (C7H) : IO 端口功能复用控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
C7H	IOMUX	PWM5SEL	PWM4SEL	PWM3SEL	PWM2SEL	PWM1SEL	PWM0SEL	INT1SEL	INT0SEL	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R

位编号	位符号	说明		
7	PWM5SEL	PWM5 输出 Pin 脚选择,	1: P27;	0: P13
6	PWM4SEL	PWM4 输出 Pin 脚选择,	1: P26;	0: P12
5	PWM3SEL	PWM3 输出 Pin 脚选择,	1: P25;	0: P10



4	PWM2SEL	PWM2 输出 Pin 脚选择, 1: P24; 0: P07
3	PWM1SEL	PWM1 输出 Pin 脚选择, 1: P23; 0: P04
2	PWM0SEL	PWM0 输出 Pin 脚选择, 1: P22; 0: P02
1	INT1SEL	INT1 输入 Pin 脚选择, 1: P31; 0: P15
0	INT0SEL	INT0 输入 Pin 脚选择, 1: P30; 0: P14

## 7-2. I/O 端口结构

### 7-2-1. 模拟 IO 模式

为实现模拟信号采集，P1/P2 口，可通过配置寄存器配置为模拟 IO。

P1/P2 IO 口配置寄存器 **ADCIN\_EN**

P1/P2 配置为模拟 IO 口，通过寄存器 **ADCIN\_EN** 控制 (见 ADC 章节. **ADCIN\_EN(DFH)**: ADC 通道属性寄存器)。

### 7-2-2. 准双向模式 (Quasi-Bi)

准双向口有 2 个上拉的 MOS 管以适应不同的需要，分别称为“弱 (Weak) 上拉”、“极弱 (Very Weak) 上拉”。在 2 个上拉 MOS 管中，第 1 个上拉 MOS 管称为“弱上拉”，当输出口线寄存器为 1 且引脚本身也为 1 时打开。此上拉提供基本驱动电流使准双向口输出为 1。如果引脚输出为 1 而由外部装置下拉到低时，“弱上拉”上拉 MOS 管关闭而“极弱上拉”MOS 管打开并维持开通状态。如需把这个引脚强制拉为低，外部装置必须有足够的灌电流能力使引脚上的电压降到门槛电压以下。

第 2 个上拉 MOS 管称为“极弱上拉”，当口线锁存为 1 时打开。当引脚悬空时，这个极弱的上拉源产生很弱的电流将引脚上拉为高电平。

当口线锁存为 0 时，它有很强 (>20mA) 的拉低(Sink)能力，此时把该 I/O 口视为输出“Output”。

准双向模式的端口结构示意图如下：

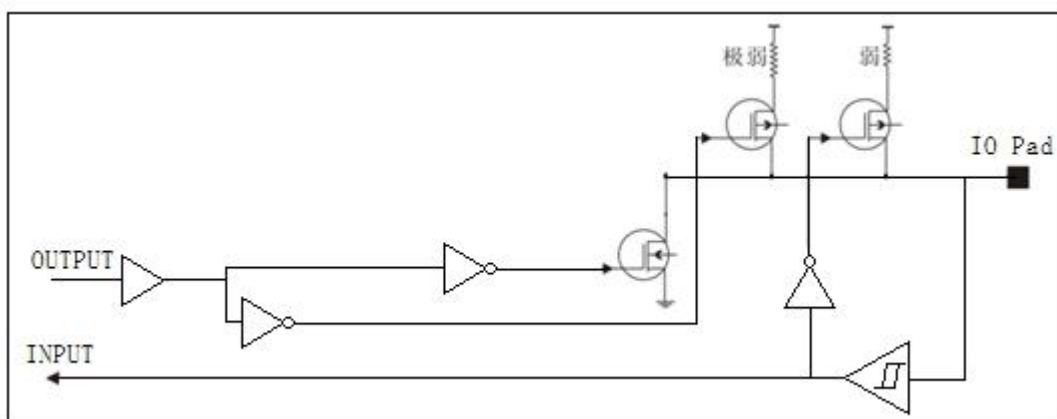
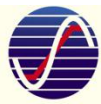


图 7-1 准双向 I/O 端口结构图

### 7-2-3. 强推挽输出模式(Push-Pull)

强推挽输出配置的下拉结构与开漏输出与准双向口的下拉结构相同，但当锁存器为 1 时能提供持续的强上拉，即能够提供持续的大电流驱动 (P1 口提供 20mA 驱动能力，P0、P2 口提供 10mA 驱动能力)





输出。强推挽输出模式的端口结构示意图如下：

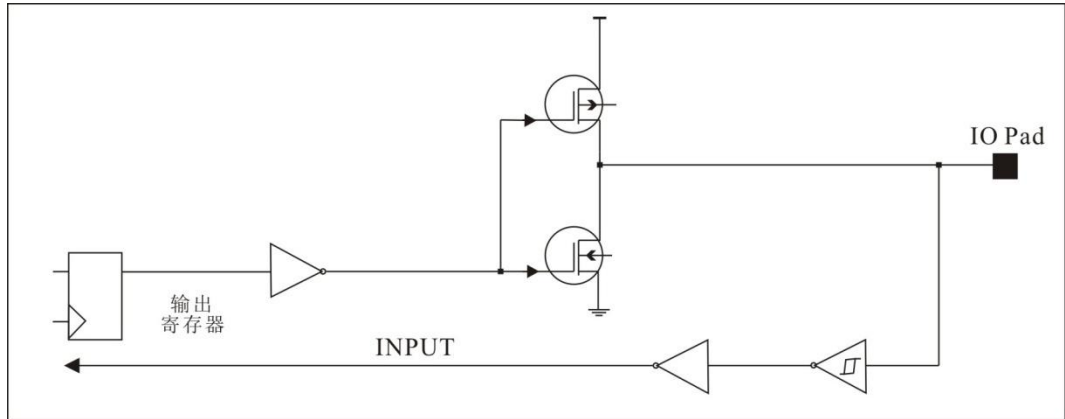


图 7-2 推挽输出 I/O 端口结构图

#### 7-2-4. 高阻输入模式 (Input Only)

此种模式仅有输入，没有输出能力。高阻输入模式的端口结构示意图如下图所示：

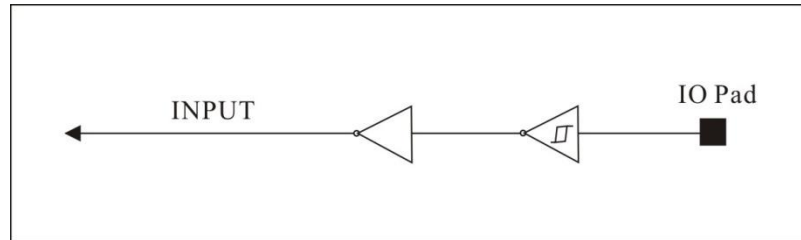


图 7-3 高阻输入 I/O 端口结构图

#### 7-2-5. 开漏输出模式 (Open Drain)

此种模式没有输出高电平的能力。如果需要输出高电平，用户必须外接上拉电阻。此时外加引脚的电压不能超过  $VDD+0.3V$ 。开漏输出模式的端口结构示意图如下：

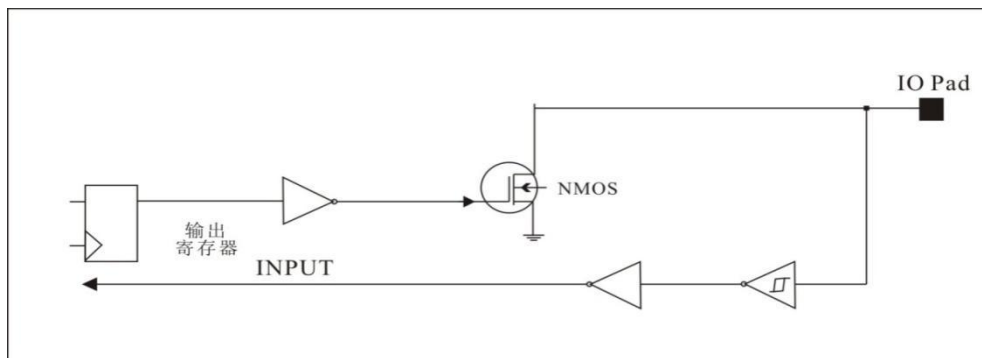
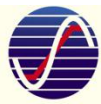


图 7-4 开漏模式 I/O 端口结构图

### 7-3. I/O 端口复用

ASM87L164 的 26 个双向 I/O 端口大部分均能共享作为第二或第三种特殊功能，比如 P0.3、P0.1、P0.0 就分别与外部复位管脚 RST 以及外部晶振管脚 XIN/ XOUT 复用。P0.3 在上电复位后默认作为外部复位管脚，而 P0.1/P0.0 则在上电复位后默认为普通 I/O 管脚。用户若使用 P0.3 作为普通 I/O 端口，需用户程序通过对辅助控制寄存器 AUXCON (D8H) 进行设置后才可以使使用。

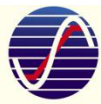




下面再简要罗列一下 I/O 端口复用情况，共享优先级遵循 GPIO 优先级最低的规则。

Pin name	Pin Type	默认类型
P00	P00/XOUT/EX0	10mA IO, 准双向, 无上下拉
P01	P01/XIN/EX1	10mA IO, 准双向, 无上下拉
P02	P02/PWM0/SPI_MOSI	20mA IO, 高阻输入, 上下拉关闭
P03	P03/RST	10mA IO, 复位输入, 无上下拉
P04	P04/PWM1/SPI_SCK/TC1	20mA IO, 高阻输入, 上下拉关闭
P05	P05/SCL/EX2	10mA IO, 准双向, 上下拉关闭
P06	P06/SDA/EX3/TC2	10mA IO, 准双向, 上下拉关闭
P07	P07/PWM2/SPI_CS/EX4	20mA IO, 高阻输入, 上下拉关闭
P10	P10/PWM3/SPI_MISO/EX5/SCK	20mA IO, 高阻输入, 上下拉关闭
P11	P11/EX6/MISO	10mA IO, 准双向, 上下拉关闭
P12	P12/PWM4/EX7/TC0/MOSI/RX1	20mA IO, 高阻输入, 上下拉关闭
P13	P13/PWM5/TX1/TCAP0	20mA IO, 高阻输入, 上下拉关闭
P14	P14/T2PWM/INT0/TCAP1	10mA IO, 准双向, 上下拉关闭
P15	P15/TX0/INT1	10mA IO, 准双向, 上下拉关闭
P16	P16/AIN0/RX0	10mA IO, 准双向, 上下拉关闭
P17	P17/AIN1	10mA IO, 准双向, 上下拉关闭
P20	P20/AIN2/YOUT_T	10mA IO, 准双向, 上下拉关闭
P21	P21/AIN3/INP	10mA IO, 准双向, 上下拉关闭
P22	P22/PWM0	10mA IO, 准双向, 上下拉关闭
P23	P23/PWM1	10mA IO, 准双向, 上下拉关闭
P24	P24/AIN4/PWM2	10mA IO, 准双向, 上下拉关闭
P25	P25/PWM3	10mA IO, 准双向, 上下拉关闭
P26	P26/AIN5/PWM4/VREFH	10mA IO, 准双向, 上下拉关闭
P27	P27/PWM5	10mA IO, 准双向, 上下拉关闭
P30	P30/AIN6/INT0/TCK	10mA IO, 默认 TCK, 上下拉关闭, 如要将其配置为 GPIO, 需将 dbc1[7:6]配置为 2'b01;
P31	P31/AIN7/INT1/TDIO	10mA IO, 默认 TDIO, 上下拉关闭

注意：在芯片正常使用时，请将将 dbc1[7:6]配置为 2'b01，关闭 TWD 调试功能。



## 8. 时钟和复位系统

### 8-1. 时钟系统

#### 8-1-1. 三时钟模式

ASM87L164 采用三时钟系统, 内置 16MHz RC 时钟振荡器(HIRC)、内置 32KHzRC 时钟振荡器(LIRC) 与外接晶体振荡器时钟。这三个时钟都可用作系统时钟源, 并可通过时钟管理寄存器实现切换, 得到系统时钟 Fsys, 默认 HIRC 振荡时钟作为系统时钟。HIRC 在芯片出厂前精确被校准到  $\pm 1\%$  (@5V/25°C), 此 HIRC 受工作温度和工作电压影响会有一定的漂移, 在 2.0~5.5V 的电压范围、-40°C~85°C 温度范围内, 频率漂移小于  $\pm 2\%$ 。

Fsys 为系统时钟频率即 CPU 内核工作频率, 该时钟频率可以通过时钟管理模块实现 HIRC 与外部晶振时钟之间切换。HIRC 振荡器的频率还可以通过时钟管理寄存器 OSCM 之[1,0]两位实现 1、2、4、16 分频, 从而得到频率分别为 16MHz、8MHz、4MHz 和 1MHz 的 Fsys 时钟信号。

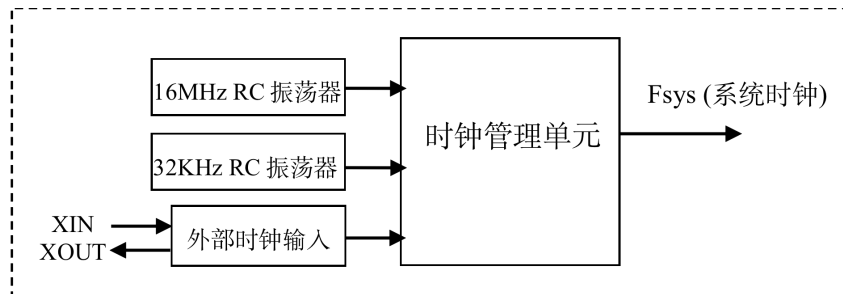
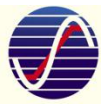


图 8-1 ASM87L164 的双时钟模式结构框图

OSCM (A7H): 时钟管理寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
A7H	OSCM*	FSCLK_SEL	RC32KEN	LDOEN	CLKSEL	OSCEN	RCEN	RCFR		42H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R		

编号位	位符号	说明
7	FSCLK_SEL	内部高低频时钟选择, 1: 内部低频时钟; 0: 内部高频时钟
6	RC32KEN	内部低频时钟使能, 1: 内部 32K 时钟使能; 0: 内部 32K 时钟关闭
5	LDOEN	内置 LDO 使能, 1: LDO 关闭; 0: LDO 打开
4	CLKSEL	工作时钟选择, 1: 选择外部时钟; 0: 选择内部时钟
3	OSCEN	外部晶振时钟输入使能, 1: 使能外部 OSC 时钟; 0: 关闭外部 OSC 时钟
2	RCEN	内置 HIRC 振荡器使能, 1: 关闭内置 HIRC 振荡器; 0: 使能内置 HIRC 振荡器
1~0	RCFR	HIRC 振荡器频率选择: <b>01: 8MHz;</b> <b>10: 4MHz</b> <b>00: 1 MHz</b> <b>11: 16MHz</b>



注:

- 1) OSCM 是多用途寄存器，除了用来设置时钟模块各项参数，还用于设置芯片内置 LDO 开关状态。
- 2) 内外时钟信号切换时，需先将内外时钟都打开，切换时钟后再根据需要关闭没选中的时钟源。
- 3) 内部高低频时钟信号切换时，需先将高低频时钟开启，切换时钟后再根据需要关闭没选中的时钟源。
- 4) 在使用外部晶振，同时芯片内部使能 ADC 模块时，内部 RC 振荡器要同时使能。
- 5) 外部晶振/陶振 1MHz 时，起振电容为 20pF，其它频率时起振电容为 10pF

### 8-1-2. 内置 RC 时钟振荡器

ASM87L164 内置一个标称值 16MHz、可校准 HIRC 振荡器，校准后 HIRC 振荡器输出精度 $\pm 2\%$  (VDD=2.0~5.5V, -40~85°C) 的时钟信号。ASM87L164 上电启动后缺省选择 HIRC 振荡器时钟作为系统时钟。

ASM87L164 正常上电复位后，HIRC 振荡器开始输出稳定的时钟信号。该时钟信号可以通过内部校准控制寄存器或者 Flash 闪存信息区中的校准字来达到频率值校准，以得到精准的 16MHz 时钟信号。

### 8-1-3. 外部时钟

ASM87L164 支持外接晶振时钟、陶瓷谐振器以及有源时钟源等外部时钟信号。

当选用外部时钟信号时，用户首先通过辅助控制寄存器 AUXCON (D8H) 将 P0.1、P0.0 两个 IO 管脚设置为晶振管脚 XIN 与 XOUT，然后设置时钟管理寄存器 OSCM 的 Bit3 (OSCEN) 打开外部时钟，之后再设置 OSCM 的 Bit4 (CLKSEL) 确定 CPU 时钟 Fsys 选择外部时钟。

当选用有源时钟直接灌时钟信号时，则此有源时钟信号需要直接接到 XIN，XOUT 则必须浮空。

外部时钟和 HIRC 振荡器可以同时工作，此时 CPU 时钟 Fsys 由控制寄存器 OSCM 的 Bit 4 (CLKSEL) 来设置时钟源。

OSCM[4] = 1: 使用外部时钟

OSCM[4] = 0: 使用 HIRC 振荡器

### 8-1-4. 上电后必须进行内部时钟初始化操作

推荐操作步骤如下:

- 1) 上电后系统时钟 Fsys 默认使用内部时钟源。
- 2) 清零 OSCM 之 OSCM[1], OSCM[0]位, 设置系统时钟 Fsys 时钟频率为 1MHz。
- 3) 延时约 1ms 后, 系统时钟切换成内部 16MHz、8MHz 或 4MHz(根据需要), 也可切换成外部时钟源。
- 4) 执行 5 个 NOP 指令。

### 8-1-5. 系统时钟使用内部时钟源 (上电后 Fsys 默认用内部时钟源)

推荐操作步骤如下:

- 1) 系统时钟 Fsys 设置内部 1MHz 频率
- 2) 延时约 1ms
- 3) 系统时钟 Fsys 切换到内部其它频率 (如 8MHz 或 4MHz)



4) 执行 5 个 NOP 指令。

建议参考代码如下：

```
OSCM &= 0xFC;    //系统时钟 Fsys 切换到内部 1MHz
for (i=0; i<250; i++) //延时约 1ms
{
    _nop_(); _nop_(); _nop_(); _nop_();    //_nop_()需添加头文件#include <intrins.h>
}
OSCM |= 0x01;    //Fsys 切换到内部 8MHz
//OSCM |= 0x01;    //Fsys 切换到内部 4MHz
_nop_(); _nop_(); _nop_(); _nop_(); _nop_();
```

#### 8-1-6. Fsys 由内部时钟 HIRC 切换到外部时钟

操作步骤如下：

- 1) 系统时钟 Fsys 先设置成内部 1MHz(方法参考上述示例代码)
- 2) 使能晶振管脚，P0.1/P0.0 为晶振脚：AUXCON |=0x08
- 3) 使能外部时钟：OSCM |= 0x08;
- 4) 延时等待外部时钟稳定，延时时间建议大于 25ms
- 5) Fsys 由内部时钟切换到外部时钟：OSCM |= 0x10;
- 6) 执行 5 个 NOP 指令
- 7) 时钟切换成功后根据需要决定是否需要关闭内部时钟 HIRC

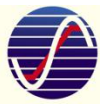
#### 8-1-7. Fsys 由外部时钟切换到内部时钟 HIRC

操作步骤如下：

- 1) 使能内部时钟 HIRC，延时约 1ms(如内部时钟未关闭可省此步)
- 2) 执行系统时钟切换指令，系统时钟切换至内部时钟
- 3) 执行 5 个 NOP 指令
- 4) 是否关闭外部 OSC 根据需要而定。

## 8-2. 上电复位过程

ASM87L164 在上电以后，用户软件执行之前，会经过如下三个阶段：



- (1) 上电复位阶段
- (2) 系统复位，加载 FLASH 信息字阶段
- (3) 正常操作阶段

➤ 上电复位阶段：

在这个阶段，ASM87L164 一直处于上电复位的状态。当 ASM87L164 内核电源电压高过上电复位门限电压，内部 RC 振荡器开始输出有效稳定的时钟信号，同时内部计数器开始计数，产生上电复位信号并持续大约 256us，上电复位才算结束。

➤ 系统复位，调入信息字阶段：

上电复位结束，ASM87L164 开始系统复位，系统复位相当于复位延时。ASM87L164 内部有一个计数器在上电复位结束后开始计数。当计数器计数到相当于 1.5 毫秒对应的数值后，系统复位才会结束。系统复位期间，系统会加载 Flash 信息区信息字对系统自身的一些参数进行配置。

➤ 正常操作阶段：

结束系统复位，加载信息字阶段后，MCU 开始从 Flash 中读取指令代码即进入正常操作阶段。

### 8-3. 复位系统

ASM87L164 共有六种复位方式：①上电复位 (POR, Power On Reset)②外部复位脚 (RST) 复位③低电压复位 LVR (Low Voltage Reset) ④软件复位 (Software Reset) ⑤看门狗 (WDT) 复位

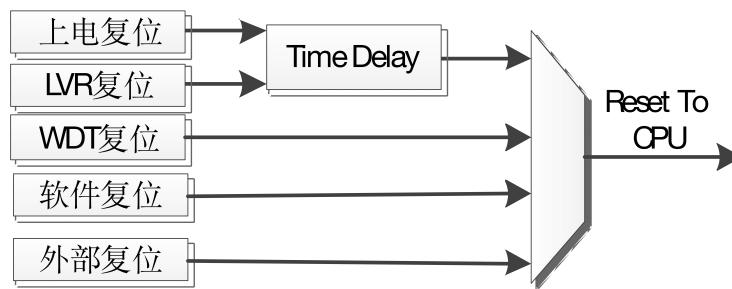


图 8-2 ASM87L164 复位系统功能框图

#### 8-3-1. 上电复位 POR

ASM87L164 内部集成了专门设计的上电复位 POR 电路模块，以确保可靠的上电复位功能，该 POR 电路同时具有掉电复位的功能，POR 检测的是 LDO 输出电压（内核电源电压）。如下图，当芯片内核电源电压 VDDC 上升到  $V_{sp0}$  (典型值为 0.6V) 时，内部复位脉冲 POR 信号输出包含了延时滤波( $T_d$ ,典型值为 100us)，是真正用于系统复位的复位脉冲信号，POR 复位的电压释放点 ( $V_{sp1}$ ) 为 1.3V。

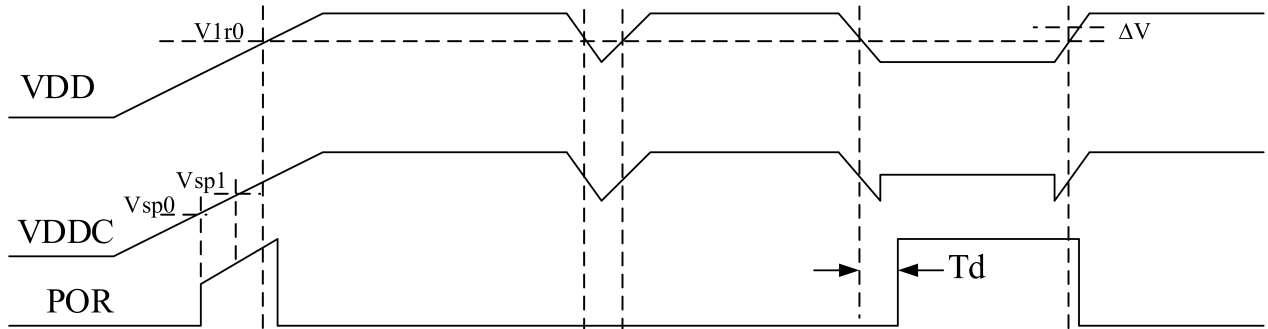
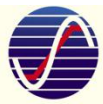


图 8-3 ASM87L164 上电复位时序示意图

### 8-3-2. 外部 RST 复位

外部 RST 复位就是在复位管脚上施加一定宽度的复位脉冲信号 (RST 为低电平有效的复位信号, 即施加一定宽度的低电平信号, 可以作为外部 RST 复位), 从而使芯片复位。ASM87L164 的复位管脚 RST 与 P0.3 复用, 上电时作为复位管脚使用, 复位结束后可由软件将其设置为普通 IO 口。

### 8-3-3. 低电压复位 LVR

ASM87L164 内建低电压复位电路模块, 缺省为使能状态。当配置为使能时, 芯片电源电压低于设置的复位门限电压, LVR 模块工作并产生复位信号, 整个芯片进入复位状态。而当电源电压回升到复位电压门限以上, 复位信号释放, 芯片重新进入正常工作状态。LVR 复位模块图如下图所示。

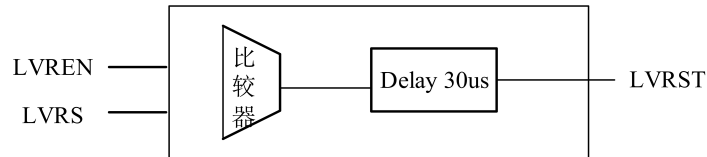


图 8-4 ASM87L164 低电压复位模块图

LVRS: LVR 监测电压点选择控制信号(4Bit), 可以设定 8 种检测电压点的任意一种。

LVRST: LVR 输出复位信号, 当电源电压低于设置的 LVR 复位电压点, LVR 内部复位脉冲信号输出 (高电平), 再经过 30us (典型值) 延时滤波输出 LVRST 信号。

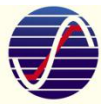
LVREN: LVR 使能控制信号, 默认为 0, LVR 处于使能状态。

LVR 的使能控制、复位电压点选择设置的寄存器分别与 ADC 模块的参考源选择寄存器 ADCREF、ADC 模块转换结果低 4 位寄存器 ADCDRL 共用, 具体设置如下。

#### LVR 使能控制 (ADC 参考源选择寄存器 ADCREF Bit7)

地址	名称	7	6	5	4	3	2	1	0	上电复位值
A9H	ADCREF	LVREN	ADCPSEL[3]	ADSEL[7]	REFSEL			OFFSETX	OFFSET	0CH
读/写		W/R	W/R	W/R	W/R			W/R	W/R	

位编号	位符号	说明
7	LVREN	LVR 复位电路使能: 0: 使能 LVR, 上电默认值



		1: 关闭 LVR
--	--	-----------

**LVR 复位电压点选择 (ADC 转换结果高 4 位 ADCDRL Bit 7~4)**

地址	名称	7	6	5	4	3	2	1	0	上电复位值
DDH	ADCDRL	LVRSEL[3:0]				ADCDRL[3:0]				00H
读/写		W				R				

位编号	位符号	说明	
7~4	LVRSEL[3:0]	LVR 复位电压点选择	
		LVRS	LVR Point
		0000	1.6V
		0001	1.7V
		0010	1.8V
		0011	2.0V
		0100	2.2V
		0101	2.4V
		0110	2.6V
		0111	2.8V

关于 LVR 模块及其使用，进一步说明如下：

- 1) LVR 在系统上电复位后处于默认使能状态，且默认复位电压检测点为 1.6V；
- 2) 使用 LVR，须先通过 ADCDRL 选择设定的 LVR 复位电压点，然后再通过 ADCREF 打开 LVR。

**8-3-4. 软件复位**

ASM87L164 也可以由软件通过设置软件复位寄存器 SRST 来实现复位。

**SRST (F7H) 软件复位控制寄存器**

地址	名称	7	6	5	4	3	2	1	0	上电复位值
F7H	SRST	-	-	-	-	-	-	-	SRSTEN	00H
读/写		-	-	-	-	-	-	-	W/R	

位编号	位符号	说明
7~1	-	保留
0	SRSTEN	软件复位使能信号 1: 软件复位;      0: 正常工作;      必须连写 2 次, 才能有效

**8-3-5. 看门狗(WDT)复位**

ASM87L164 内置 16 位看门狗电路 WDT，时钟源为系统时钟 Fsys。其系统结构如下图所示。



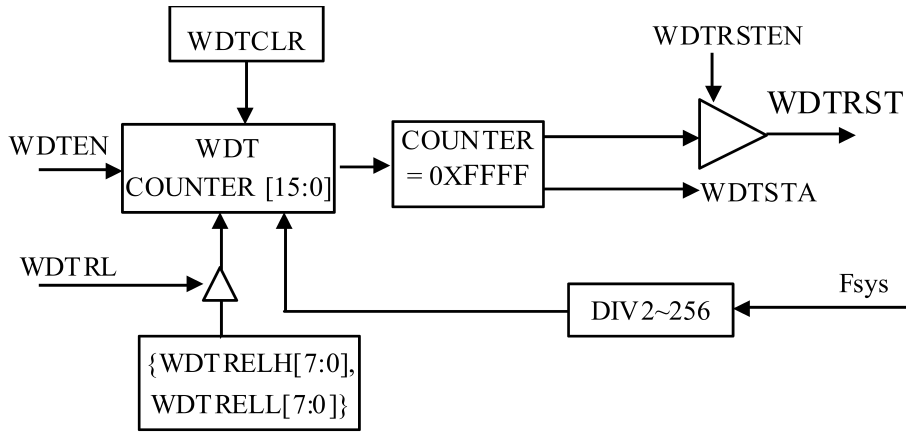
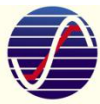


图 8-4 ASM87L164 看门狗电路模块图

下面对上图简要说明如下:

- 1) 看门狗计数器WDT Counter为16位计数器，可重载初始值WDTRELLH与WDTRELL；
- 2) 看门狗计数器输入时钟为系统CPU时钟(Fsys)分频得到，分频系数为2,4,8,16,32,64,128,256；
- 3) 看门狗计数器WDT Counter计数到0XFFFF产生溢出，标志位WDTCON[6]置位；
- 4) 看门狗计数器溢出时，若看门狗复位使能打开，则输出看门狗复位信号WDRST，从而复位整个系统。

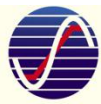
ASM87L164 的 WDT 相关寄存器包括看门狗控制器 (WDTCON)、看门狗重载寄存器 (WDTREL)等, 具体描述如下。

### WDTCON (86H) : 看门狗控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
86H	WDTCON	WDTCLR	WDTSTA	WDTRL	WDRSTEN	WDTEN	FRSEL[2:0]			00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R			

位编号	位符号	说明
7	WDTCLR	看门狗刷新使能，1：刷新看门狗，写完后自动清零
6	WDTSTA	看门狗溢出标志，1：看门狗溢出，0：没有溢出
5	WDTRL	重载允许控制： 1：允许重载 WDTRELL，WDTRELH，从而调整看门狗计数时间； 0：不能重载
4	WDRSTEN	看门狗复位使能控制 1：看门狗系统复位使能打开 0：看门狗复位使能关闭
3	WDTEN	看门狗工作使能 1：看门狗工作使能打开 0：看门狗关闭
2~0	FRSEL[2:0]	分频系数选择，如下表所示。





FRSEL[2:0]	分频系数	最大看门狗延时时间 (以内置 8M Hz 时钟为例)
000	2	16x1024us≈ 16ms
001	4	32x1024us≈ 32ms
010	8	64x1024us≈ 64ms
011	16	128x1024us≈ 128ms
100	32	256x1024us≈ 256ms
101	64	512x1024us≈ 512ms
110	128	1024x1024us≈ 1s
111	256	2048x1024us≈ 2s

### WDTRELL (96H) : 看门狗重载寄存器低 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位值
96H	WDTRELL	WDTRELL[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

说明: 看门狗计数器重载低 8 位, 调整看门狗计数时间

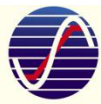
### WDTRELH (97H) : 看门狗重载寄存器高 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位值
97H	WDTRELH	WDTRELH[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

说明: 看门狗计数器重载高 8 位, 调整看门狗计数时间

## 8-4. 不同复位方式对寄存器、SRAM 与 PC 指针的影响

复位源	配置寄存器	SFR 寄存器	SRAM	PC 指针
上电复位	默认值	默认值	xxh	00h
LVR 复位	默认值	默认值	xxh	00h
软件复位	不变	默认值	不变	00h
看门狗复位	不变	默认值	不变	00h
外部复位	不变	默认值	不变	00h



## 9. 省电模式

为减少功耗，ASM87L164 提供三种低功耗省电模式：空闲 (Idle) 模式、掉电 (Power-Down) 模式和深度睡眠模式，其中空闲模式与掉电模式都由电源管理控制寄存器 PCON 来控制，PCON 寄存器说明如下。

### PCON 寄存器：电源管理控制寄存器\*

地址	名称	7	6	5	4	3	2	1	0	上电复位值
87H	PCON	SMOD	-	-	-	-	-	STOP	IDLE	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明	
7	SMOD0	UART 波特率控制模式	1: 加倍; 0: 不加倍
6	SMOD1	UART 波特率控制模式	1: 加倍; 0: 不加倍
6~2	-	保留	
1	STOP	掉电模式进入使能	1: 进入掉电模式 0: 正常模式
0	IDLE	空闲模式进入使能	1: 进入省电模式 0: 正常模式

注：PCON 寄存器通过 ORL 运算指令进行赋值。

### 9-1. 空闲模式 (Idle)

空闲模式能够降低系统功耗，在此模式下，程序中止运行，CPU 时钟停止，但外部设备时钟继续运行。ASM87L164 进入空闲工作状态以后，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。在空闲模式状态下，ASM87L164 各组成模块的工作状态如下：

- CPU 停止工作
- 时钟模块工作
- 其它外设 (TIMER0、TIMER1、PWM、ADC、UART 等) 工作

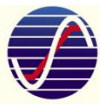
ASM87L164 进入 Idle 工作状态的方法是，设置 PCON[0]=1，则下一指令周期进入 Idle 省电模式。将 ASM87L164 从 Idle 状态唤醒，则有以下两种方式：

- (1) 外部中断唤醒，继续执行程序；
- (2) 外部复位，程序重新载入，程序代码从地址 00H 开始读取。

### 9-2. 掉电模式 (Power-Down)

掉电模式也称睡眠模式，掉电模式可以使 ASM87L164 进入功耗非常低的状态。掉电模式将停止 CPU 和外围设备的所有时钟信号，在进入掉电模式前所有 CPU 的状态都被保存，如 PC，PSW，SFR，RAM 等。ASM87L164 进入掉电模式工作状态以后，各组成模块的工作状态如下：

- CPU 停止工作
- 时钟模块停止工作
- 所有外设停止工作



ASM87L164 进入掉电模式工作状态的方法是，设置 PCON[1]=1，则下一指令周期 CPU 即进入掉电省电模式。将 ASM87L164 从掉电模式状态唤醒，则有以下两种方式：

- (1) 外部中断（低电平）触发唤醒，继续执行程序；
- (2) 外部复位，程序重新载入，程序代码从地址 00H 开始读取。

### 9-3. 深度睡眠方式

ASM87L164 另外提供了一种极低功耗的深度睡眠模式，该模式仅提供极小的供电能力，以保持 SRAM 存储内容，其他所有模块均处于关闭状态。

深度睡眠模式的进入流程如下：

- (1) 关闭低频时钟 RC32K（上电默认打开）：配置寄存器 OSCM &= 0xBF；
- (2) 关闭 LVR（上电默认打开）：配置寄存器 ADCREF |= 0x80；
- (3) 关闭 ADC 与 ADC VREF 电路：配置寄存器 ADCIN\_EN &= 0x7F，ADCCSR &= 0x7F；
- (4) 将所有 IO 口设置为双向口，配置寄存器 {Pxx\_CFGA, Pxx\_CFGB} = {1,0}，配置 AUXCON |= 0x04；
- (5) LDO 进入 STADNBY 状态：配置寄存器 OSCM |= 0x20，该配置会在 STOP 模式进入后生效；
- (6) 睡眠模式配置成深度睡眠模式：AUXCON |= 0x40；
- (7) STOP 模式进入：配置寄存器 PCON |= 0x02；

将 ASM87L164 从掉电模式状态唤醒，可通过外部中断触发方式唤醒芯片，程序继续执行。

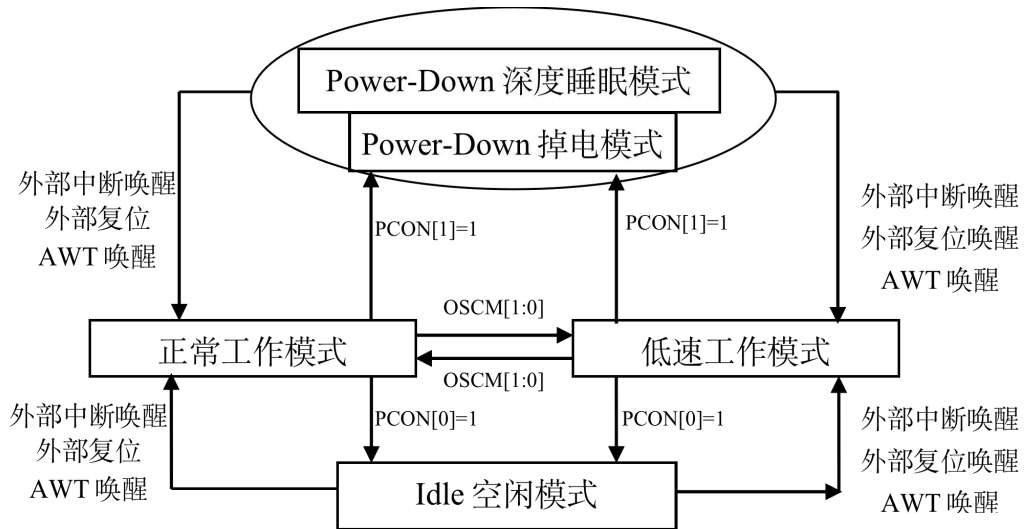


图 9-1 ASM87L164 工作模式转换示意图



### AUXCON(D8H)辅助功能寄存器

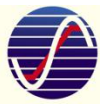
地址	名称	7	6	5	4	3	2	1	0	上电复位值
D8H	AUXCON	-	PDFL	-	AD_Thresh	PINOSC	PRST	INT1SEL	INT0SEL	00H
读/写			W/R			W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	-	保留
6	PDFL	深度睡眠模式使能， 1: 深度睡眠模式 0: 睡眠模式
5	-	保留，其值须保证为“0”，若配置成“1”，会出现未知状况。
4	AD_Thresh	AD 阈值使能控制位，1: 使能 ADC 阈值比较功能， 0 禁止 ADC 阈值比较功能
3	PINOSC	晶振管脚(P0.0/P0.1)属性设置， 0: P0.0/ P0.1 为普通 IO， 1: P0.0/P0.1 为晶振 Pad
2	PRST	外部复位管脚(P0.3)属性设置， 1: P0.3 为普通 IO， 0: P0.3 为外部复位管脚
1	INT1SEL	外部中断 1 边沿选择， 0: 下降沿； 1: 上升沿
0	INT0SEL	外部中断 0 边沿选择， 0: 下降沿； 1: 上升边

#### 9-4. 唤醒时间

不管 CPU 选用内部 RC 振荡时钟还是外部时钟，ASM87L164 进入掉电模式以后的时钟模块都是处于停止状态的。ASM87L164 从睡眠模式被唤醒时，需要等待一定时间以使时钟振荡电路进入稳定工作状态，从而整个芯片系统再进入正常工作状态。这一等待时间称之为唤醒时间，ASM87L164 掉电模式的唤醒时间是 1024 个时钟周期。

由于 ASM87L164 的空闲模式不关闭时钟模块，所以从空闲模式中唤醒不需要唤醒时间。



## 10. 中断系统

ASM87L164 系列单片机提供了 12 个中断源以及多达 10 个的中断 IO 口。这 12 个中断源分为 4 个中断优先级，并可以单独分别设置为高优先级或者低优先级。每个中断分别有独立的优先级设置位、中断标志、中断向量和使能位，总的使能位 EA 可以实现所有中断的打开或者关闭。

- 12 个中断源
- 10 个内中断：Timer0、Timer1、Timer2、I2C、Timer3、PWM、UART0、SPI/UART1、ADC 与 LVD 中断
- 共计 10 个外部中断 IO 口，分别是 INT0/P1.4/P3.0、INT1/P1.5/P3.1、EX0/P0.0、EX1/P0.1、EX2/P0.5、EX3/P0.6、EX4/P0.7、EX5/P1.0、EX6/P1.1、EX7/P1.2，对应两个外部中断程序入口。其中 INT0、INT1 中断可选择低电平触发或者边沿触发，EX0、EX7 仅支持高电平触发，其他 IO 中断仅支持低电平触发。
- 双沿唤醒：EX0 和 EX4、EX5、EX6 中的一个搭配使用，EX7 和 EX1、EX2、EX3 中的一个搭配使用。
- 四级中断优先级可设。

注：

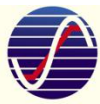
要使用双沿唤醒时，必须上升沿唤醒 IO 和下降沿唤醒 IO 在芯片外短接并接上外部触发信号，且必须在唤醒后中断处理程序中关闭当前中断口的使能，打开另一个电平的中断口使能；在中断初始化中必须先检测当前两个口的电平状态再打开下次电平变化唤醒对应的扩展中断口。通过将寄存器 `ADCAVE&=0x90`，关闭 INT0 和 INT1 输入，如果它们不关闭，会反复进中断；

另外，在使用 EX0 与 EX7 时，IO 需要配置为高阻输入模式。

### 10-1. 中断源和中断向量

ASM87L164 的中断源、中断向量以及相关控制位如下表所示。

中断源	中断发生时间	向量地址	允许位	优先级控制	默认优先级	中断号 (C51)	能否唤醒掉电模式
INT0/EX0/EX1/EX2/EX3	外部中断 0 发生	0003H	EX0	IP1[0] IP0[0]	1(高)	0	能
Timer0	定时器 0 溢出	000BH	ET0	IP1[1] IP0[1]	2	1	不能
INT1/EX4/EX5/EX6/EX7	外部中断 1 发生	0013H	EX1	IP1[2] IP0[2]	3	2	能
Timer1	定时器 1 溢出	001BH	ET1	IP1[3] IP0[3]	4	3	不能
UART0	UART 发送/接收结束	0023H	UART	IP1[4] IP0[4]	5	4	不能



PWM	PWM 溢出	002BH	PWM	IP1[5] IP0[5]	6	5	不能
ADC	ADC 转换结束	0033H	ADC	IP1[0] IP0[0]	7	6	不能
I2C	I2C 发送接收结束	003BH	I2C	IP1[1] IP0[1]	8	7	不能
SPI/Uart1	SPI/Uart1 发送接收结束	0043H	SPI/Uart1	IP1[0] IP0[0]	9	8	不能
Timer3	定时器 3 溢出	004BH	ETIM3	IP1[1] IP0[1]	10	9	能
LVD	LVD 中断标准产生	0053H	LVD	IP1[2] IP0[2]	11	10	不能
Timer2	定时器 2 溢出	005BH	ET2	IP1[3] IP0[3]	12(低)	11	不能

注:

- 1) INT0/EX0/EX1/EX2/EX3 共用中断矢量 03H, INT1/EX4//EX5/EX6/EX7 共用中断矢量 13H。
- 2) EX0/EX1/EX2/EX3; EX4/EX5/EX6/EX7 需要通过控制寄存器 INTEXT\_CTRL 使能, 也需要通过查询 (查询 INTEXT\_F) 的方法确定中断源。

在 EA=1 及各中断使能控制为 1 的情况下, 各中断发生情况如下:

定时器中断: Timer0、Timer1 和 Timer2 溢出时会产生中断并将中断标志 TF0、TF1 和 TF2 置为“1”, 当单片机响应对应定时器中断时, 中断标志 TF0、TF1 和 TF2 会被用户软件来清零。

PWM 中断: 当 PWM 计数器溢出时 (PWM 计数器计数到 0xFFFF), OVF 位(PWM 中断标志位)会被硬件自动置“1”, PWM 中断产生。在响应 PWM 中断后, 硬件并不会自动清除 OVF 位, PWM 中断标记位必须由用户程序负责清除。

ADC 中断: ADC 中断的发生时间为 ADC 转换完成时, 其中断标志就是 ADC 转换结束标志 EOCO (ADCCSR[4])。当使用者设定 ADC\_START 开始转换前, EOCO 必须由用户程序负责清除为“0”; 当转换完成后, EOCO 会被硬件自动置为“1”。使用者在 ADC 中断发生之后, 进入中断服务程序后, 硬件不会自动清除 EOCO, 必须由用户程序负责清除。

外部 IO 中断 INTx(x=0~1): 外部 IO 中断共用两个中断向量, 当外部 IO 中断口有中断条件发生时, 外部中断就发生了。这 2 个外部中断标志是系统隐藏式的, 不需要用户做处理, 硬件会自动清除。INT0/INT1 对应的外部中断, 可通过设置 SFR 将其设置成边沿触发, 也可设置成低电平触发; 用户还可通过 IP 寄存器来设置每个中断的优先级级别。

## 10-2. 中断优先级

ASM87L164 单片机的中断具有四个中断优先级, 这些中断源的请求可通过配置寄存器 IP0 与 IP1 编程为高优先级中断或者低优先级中断, 即可实现两级中断服务程序的嵌套。一个正在执行的低优先级中断



能被高优先级中断请求所中断，但不能被另一个同一优先级的中断请求所中断，一直执行到结束，遇到返回指令 RETI，返回主程序后再执行一条指令才能响应新的中断请求。也就是说

- 1) 低优先级中断可被高优先级中断请求所中断，反之不能；
- 2) 任何一种中断，在响应过程中，不能被同一优先级的中断请求所中断。

中断查询顺序：ASM87L164 单片机的同一优先级中断，如果同时来几个中断，则中断响应的优先顺序同标准 8051 中的中断查询号相同，即查询号小的会优先响应，查询号大的会慢响应。

Priority	IP1[x]	IP0[x]
最高	1	1
次高	1	0
次低	0	1
最低	0	0

### 10-3. 中断处理流程

当一个中断产生并且被 CPU 响应，则主程序运行被中断，将执行下述操作：

1. 当前正在执行的指令执行完；
2. PC 值被压入堆栈，保护现场；
3. 中断向量地址载入程序计数器 PC；
4. 执行相应的中断服务程序；
5. 中断服务程序结束并 RETI；
6. 将 PC 值退栈，并返回执行中断前的程序。

在此过程中，系统不会立即执行其它同一优先级的中断，但会保留所发生的中断请求，在当前中断处理结束后，再执行一条指令，之后转去执行新的中断请求。

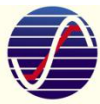
### 10-4. 中断相关寄存器

#### 10-4-1. IEA(A8H)中断使能控制寄存器 0

地址	名称	7	6	5	4	3	2	1	0	上电复位值
A8H		EA	EADC	EPWM	ES	ET1	EX1	ET0	EX0	00H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明		
7	EA	中断使能总开关	1: 使能	0: 关闭
6	EADC	ADC 中断使能	1: 使能	0: 关闭
5	EPWM	PWM 中断使能	1: 使能	0: 关闭
4	ES	串口 UART0 中断使能	1: 使能	0: 关闭
3	ET1	定时器 1 中断使能	1: 使能	0: 关闭
2	EX1	外部中断 1 使能	1: 使能	0: 关闭
1	ET0	定时器 0 中断使能	1: 使能	0: 关闭
0	EX0	外部中断 0 使能	1: 使能	0: 关闭





10-4-2. IEB(E8H)中断使能控制寄存器 1

地址	名称	7	6	5	4	3	2	1	0	上电复位值
E8H	IEB	-	-	-	-	ET2	ELVD	ESPI/EUART1/ETI M3	EI2C	00H
读/写		-	-	-	-	W/R	W/R	W/R	W/R	

位编号	位符号	说明	
7	-	保留	
6	-	保留	
5	-	保留	
4	-	保留	
3	ET2	定时器 2 中断使能	1: 使能 0: 关闭
2	ELVD	LVD 中断使能	1: 使能 0: 关闭
1	ESPI/EUART1/ETI M3	SPI\UART1\TIM3 中断使能	1: 使能 0: 关闭
0	EI2C	I2C 中断使能	1: 使能 0: 关闭

10-4-3. IP0 (B8H) 中断优先级寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
B8H	IP0	-	-	IP0[5:0]						00H
读/写		-	-	W/R						

10-4-4. IP1 (F8H) 中断优先级寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
B8H	IP1	-	-	IP1[5:0]						00H
读/写		-	-	W/R						

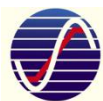
Priority	IP1[x]	IP0[x]
最高	1	1
次高	1	0
次低	0	1
最低	0	0

IP1[0]/IP0[0]控制 INT0/EX0/EX1/EX2/EX3/ADC/SPI/Uart1

IP1[1]/IP0[1]控制 Timer0/I2C/Timer3

IP1[2]/IP0[2]控制 INT1/EX4/EX5/EX6/EX7/LVD





IP1[3]/IP0[3]控制 Timer1/Timer2

IP1[4]/IP0[4]控制 UART0

IP1[5]/IP0[5]控制 PWM

#### 10-4-5. ADCAVE (AAH) : 外部中断控制寄存器 (与 ADC 寄存器复用)

地址	名称	7	6	5	4	3	2	1	0	上电复位值
AAH	ADCAVE	VREF_SEL	INT1EN	INT0EN	-	AVE_EN	AVES[2]	AVES[1]	AVES[0]	60H
读写		-	-	-	-	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	VREF_SEL	ADC 参考电压 Trim 选择(用户模式下配置为 0)
6	INT1EN	外部中断 INT1 输入使能 1: 输入使能打开 0: 输入使能关闭
5	INT0EN	外部中断 INT0 输入使能 1: 输入使能打开 0: 输入使能关闭
4	-	保留
3	AVE_EN	ADC 硬件取平均使能位, 1: 使能硬件取平均; 0: 关闭硬件取平均
2	AVES[2: 0]	AVES[2: 1]为取平均选择位 000: 不取平均 001: 累计 4 个转换值取一个平均; 010: 累计 8 个转换值取一个平均; 011: 累计 16 个转换值取一个平均; 100: 累计 32 个转换值取一个平均; 其他: 不取平均
1		
0		

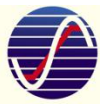
#### 10-4-6. INTEXT\_CTRL(8FH) : 外部中断控制寄存器

除 INT0、INT1 外, 另 8 个外部中断 IO 口, 分别对应 EX0/P0.0、EX1/P0.1、EX2/P0.5、EX3/P0.6、EX4/P0.7、EX5/P1.0、EX6/P1.1、EX7/P1.2 共享 03H, 13H 中断入口。

**注意:** 在使用 EX0 与 EX7 时, IO 需要配置为高阻输入模式。

地址	名称	7	6	5	4	3	2	1	0	上电复位值
8FH	INTEXT_CTRL	P12EX	P11EX	P10EX	P07EX	P06EX	P05EX	P01EX	P00EX	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	P12EX	1: P1.2 外部中断 EX7 使能 0: P1.2 外部中断禁止
6	P11EX	1: P1.1 外部中断 EX6 使能 0: P1.1 外部中断禁止

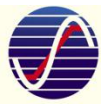


5	P10EX	1: P1.0 外部中断 EX5 使能 0: P1.0 外部中断禁止
4	P07EX	1: P0.7 外部中断 EX4 使能 0: P0.7 外部中断禁止
3	P06EX	1: P0.6 外部中断 EX3 使能 0: P0.6 外部中断禁止
2	P05EX	1: P0.5 外部中断 EX2 使能 0: P0.5 外部中断禁止
1	P01EX	1: P0.1 外部中断 EX1 使能 0: P0.1 外部中断禁止
0	P00EX	1: P0.0 外部中断 EX0 使能 0: P0.0 外部中断禁止

10-4-7. INTEXT\_F(9EH) : 外部中断标志寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
9EH	INTEXT_F	P12EF	P11EF	P10EF	P07EF	P06E	P05EF	P01EF	P00EF	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	P12EF	1: P1.2 外部中断 EX7 标志, 对应 13H 中断向量 0: P1.2 无中断申请
6	P11EF	1: P1.1 外部中断 EX6 标志, 对应 13H 中断向量 0: P1.1 无中断申请
5	P10EF	1: P1.0 外部中断 EX5 标志, 对应 13H 中断向量 0: P1.0 无中断申请
4	P07EF	1: P0.7 外部中断 EX4 标志, 对应 13H 中断向量 0: P0.7 无中断申请
3	P06EF	1: P0.6 外部中断 EX3 标志, 对应 03H 中断向量 0: P0.6 无中断申请
2	P05EF	1: P0.5 外部中断 EX2 标志, 对应 03H 中断向量 0: P0.5 无中断申请
1	P01EF	1: P0.1 外部中断 EX1 标志, 对应 03H 中断向量 0: P0.1 无中断申请
0	P00EF	1: P0.0 外部中断 EX0 标志, 对应 03H 中断向量 0: P0.0 无中断申请



## 11. 定时器 Timer0/Timer1

ASM87L164 单片机内部有三个 16 位定时器/计数器分别称为 T0、T1 和 T2，T0 和 T1 具有计数方式和定时方式两种工作模式。特殊功能寄存器 TMOD 中有一个控制位 C/Tx 来选择 T0 和 T1 是定时器还是计数器。它们本质上都是一个加法计数器，只是计数的来源不同。定时器的来源为系统时钟或者其分频时钟，但计数器的来源为外部管脚的输入脉冲。GATE<sub>x</sub> 和 TR<sub>x</sub> 是 T0 和 T1 在定时器/计数器模式计数的开关控制。

计数器模式下，P1.2/TC0 和 P0.4/TC1 管脚上的每出现一个脉冲，T0 和 T1 的计数值分别增加 1。

定时器模式下，可通过特殊功能寄存器 TMCON 来选择 T0 和 T1 的计数来源是 F<sub>sys</sub> 的 1~16 分频，缺省情况下，是 F<sub>sys</sub>/12。

定时器/计数器 T0 有如下 4 种工作模式，定时器/计数器 T1 有如下 4 种模式的前 3 种工作模式（模式 3 不支持）：

模式 0：13 位定时器/计数器模式

模式 1：16 位定时器/计数器模式

模式 2：8 位自动重载模式

模式 3：两个 8 位定时器/计数器模式

在上述 4 种模式中，T0 和 T1 都支持的模式有模式 0、模式 1、模式 2，T0 支持模式 3，T1 不支持模式 3。

### 11-1. 功能框图

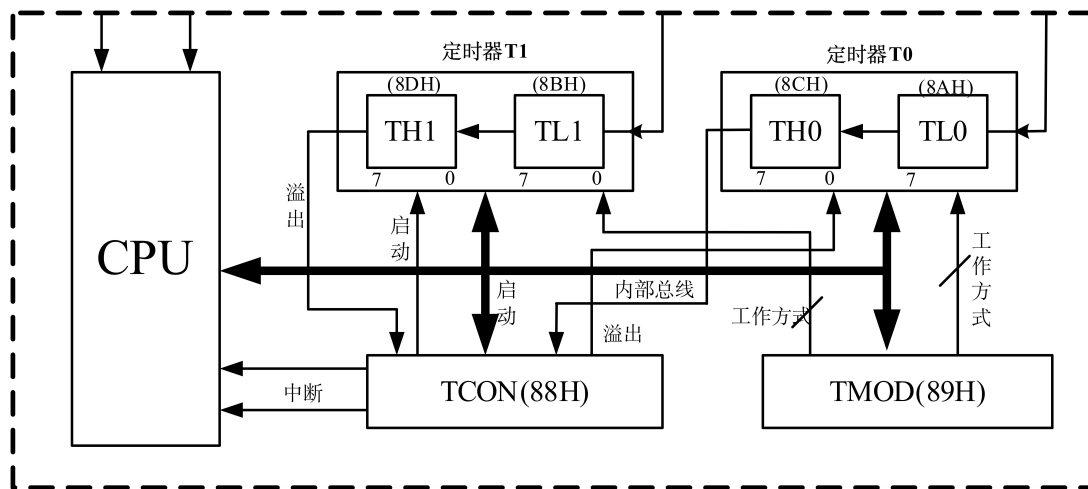
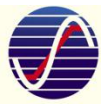


图 11-1 ASM87L164 定时器/计数器功能框图

### 11-2. 控制寄存器

Timer0/1 相关控制寄存器如下：

符号	地址	说明	7	6	5	4	3	2	1	0	上电复位值	
TCON	88H	定时器控制寄存器	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H	
TMOD	89H	定时器模式寄存器	GATE	C/T1	M11	M01	GATE0	C/T0	M10	M00	00H	
TL0	8AH	定时器 0 低 8 位										00H
TL1	8BH	定时器 1 低 8 位										00H



TH0	8CH	定时器 0 高 8 位			00H
TH1	8DH	定时器 1 高 8 位			00H
TMCON	8EH	定时器频率控制寄存器	CKDIV1	CKDIV0	BBH

### 11-2-1. TMOD (89H) :定时器工作模式寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
89H	TMOD	GATE1	C/T1	M11	M10	GATE0	C/T0	M01	M00	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

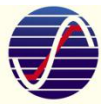
位编号	位符号	说明
7	GATE1	T1 门控制位。1: C/T1 的启动受双重控制, 即要求 TR1 和 INT1 同时为高。 0: C/T1 的启动仅受 TR1 控制。
6	C/T1	T1 定时器/计数器选择位。C/T1 = 1, 为计数器方式; C/T1 = 0, 为定时器方式。
5	M11	T1 模式选择, 定时器/计数器的 4 种工作方式由 M11M01 设定。
4	M10	00: 13 位定时器/计数器, TL1 高 3 位无效 01: 16 位定时器/计数器, TL1 和 TH1 全有效 10: 8 位自动重载定时器, 溢出时将 TH1 存放的值自动重装载入 TL1 11: 定时器/计数器 1 无效 (停止计数)
3	GATE0	T0 门控制位。1: C/T0 的启动受双重控制, 即要求 TR0 和 INT0 同时为高。 0: C/T0 的启动受 TR0 控制。
2	C/T0	T0 定时器/计数器选择位。C/T0 = 1, 为计数器方式; C/T0 = 0, 为定时器方式。
1	M01	T0 模式选择, 定时器/计数器的 4 种工作方式由 M10M00 设定。
0	M00	00: 13 位定时器/计数器, TL0 高 3 位无效 01: 16 位定时器/计数器, TL0 和 TH0 全有效 10: 8 位自动重载定时器, 溢出时将 TH0 存放的值自动重装载入 TL0 11: 定时器 0 此时作为双 8 位定时器/计数器。TL0 作为一个 8 位定时器/计数器, 由标准定时器 0 的控制位控制 (TCON 寄存器); TH0 仅作为一个 8 位定时器, 由定时器 1 的控制位控制 (TCON 寄存器)。

注:

TMOD 寄存器中 TMOD[0]~TMOD[3]是设置 T0 工作模式; TMOD[4]~TMOD[7]是设置 T1 的工作模式。定时器和计数器 Tx 功能由特殊功能寄存器 TMOD 的控制位 C/Tx 来选择, Mx1 和 Mx0 都是用来选择 C/Tx 的工作模式。

### 11-2-2. TCON(88H):定时器控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位
88H	TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	



位编号	位符号	说明
7	TF1	定时器 1 溢出标志位。当定时器 1 计满溢出时，由硬件使 TF1 置“1”，并且申请中断， <b>在中断服务中必须通过软件清零此标志位。</b>
6	TR1	定时器 1 运行控制位。由软件清“0”关闭定时器 1。当 GATE=1，且 INT1 为高电平时，TR1 置“1”启动定时器 1；当 GATE=0，TR1 置“1”启动定时器 1
5	TF0	定时器 0 溢出标志。其功能及操作情况同 TF1
4	TR0	定时器 0 运行控制位。其功能及操作情况同 TR1
3	IE1	外部中断 1 请求标志。1：外部中断 0：请求发生，请求发生后自动清零
2	IT1	外部中断 1 触发方式选择位，1：边沿触发；0：低电平触发
1	IE0	外部中断 0 请求标志。1：外部中断 0：请求发生，请求发生后自动清零
0	IT0	外部中断 0 触发方式选择位，1：边沿触发；0：低电平触发

### 11-2-3. TH1/TH0: 定时器 0/1 高 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位值
8CH	TH0	TH0[7:0]								00H
8DH	TH1	TH1[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

### 11-2-4. TL1/TL0: 定时器 0/1 低 8 位

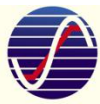
地址	名称	7	6	5	4	3	2	1	0	上电复位值
8AH	TL0	TL0[7:0]								00H
8BH	TL1	TL1[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

### 11-2-5. TMCON: 定时器频率控制寄存器

ASM87L164 的指令为单周期指令，这一点和传统的 12 个时钟周期为 1 个指令周期的 8051 单片机有所不同，Timer0/1 的计数时钟可以通过 TMCON 寄存器配置获得。

地址	名称	7	6	5	4	3	2	1	0	上电复位
8EH	TMCON	CKDIV1				CKDIV0				BBH
读/写		W/R				W/R				

位编号	位符号	说明
7~4	CKDIV1	定时器 1 时钟输入，即作为 TIMER1 计数的时钟输入：



		<p>4'b0000: 系统频率作为 TIMER1 计数时钟输入</p> <p>4'b0001: 2 分频, 系统频率 2 分频后送给 TIMER1 做时钟输入</p> <p>4'b0010: 3 分频, ...</p> <p>...</p> <p>4'b1111: 16 分频...</p>
3~0	CKDIV0	<p>定时器 0 时钟输入, 即作为 TIMER0 计数的时钟输入:</p> <p>4'b0000: 系统频率作为 TIMER0 计数时钟输入</p> <p>4'b0001: 2 分频, 系统频率 2 分频后送给 TIMER0 做时钟输入</p> <p>4'b0010: 3 分频, ...</p> <p>...</p> <p>4'b1111: 16 分频...</p>

### 11-3. 工作模式

当定时器/计数器为定时工作方式时, 计数器的加 1 信号由振荡器的 N 分频信号产生, 即每过 N 个系统时钟, 计数器加 1, 直至计满溢出为止。计数频率最快可以设置为  $F_{sys}$ ; 而最慢可以设置为  $F_{sys}/16$ 。

如前所述, ASM87L164 的 Timer0/1 各有四种工作模式, 具体说明如下表所示。

模式	工作方式	功能描述
00	工作方式 0	13 位计数器
01	工作方式 1	16 位计数器
10	工作方式 2	自动再装入 8 位计数器
11	工作方式 3	定时器 0: 分成两个 8 位计数器 定时器 1: 停止计数

下面就 TIMER0 (T0) 的工作模式为例, 进一步说明如下。

ASM87L164 的定时器/计数器 T0 和 T1 可选择四种工作模式, 即模式 0、模式 1、模式 2、模式 3。

在模式 0、1 和 2 时, T0 和 T1 的工作模式相同; 在模式 3 时, 两个定时器的的工作模式不同。

模式 0 是一个 13 位定时器/计数器。

在这种模式下, 16 位寄存器 TH0 和 TL0, 只用 13 位, 其中 TL0 的高 3 位未用, 其余位占整个 13 位的低 5 位, TH0 占高 8 位。

当 TL0 的低 5 位溢出时, 向 TH0 进位, 而 TH0 溢出时向中断标志 TF0 进位, 即硬件置位 TF0, 并申请中断 (ET0)。

模式 1 是一个 16 位定时器/计数器。

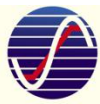
模式 2 是一个 8 位定时器/计数器, TH0/TH1 作为重载寄存器。TL1/TL0 作为 8 位计数器

当 C/T=0 时:

T0 对系统时钟的分频时钟进行计数, 即工作于定时方式。

当 C/T=1 时:

控制开关使引脚 TC0 (INT0/P0.0) 与 13 位计数器相连, 外部计数脉冲由引脚 TC0 (INT0/P0.0) 输入。



当外部信号电平发生“1”到“0”跳变时，计数器加1，这时，T0成为外部事件计数器。这就是计数工作方式。

GATE=0时，或门输出点电位为常“1”，或门被封锁，于是，引脚INT0(TC0/P0.0)输入信号无效。这时或门输出“1”打开与门，与门输出端电位取决于TR0状态，于是由TR0一位就可控制计数器开关开启或关断。

若软件使TR0置1，便接通计数开关，启动T0在原值上加1计数，直至溢出。溢出时，13位寄存器清0，TF0置位，并申请中断，T0仍从0重新开始计数。若TR=0，则关断计数开关，停止计数。

当GATE=1时，或门输出取决于INT0(TC0/P0.0)引脚的输入电平。仅当INT0输入高电平且TR0=1时，与门输出高电平，使计数开关闭合，T0开始计数，当INT0由1变0时，T0停止计数。这一特性可以用来测量在INT0端出现的正脉冲的宽度。

各种工作方式结构如下图所示：

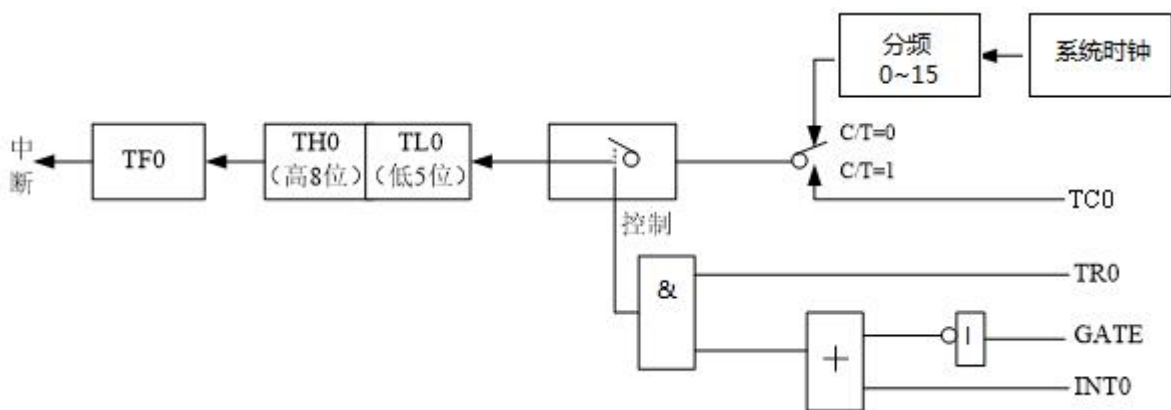


图11-2 T0/T1 方式0结构

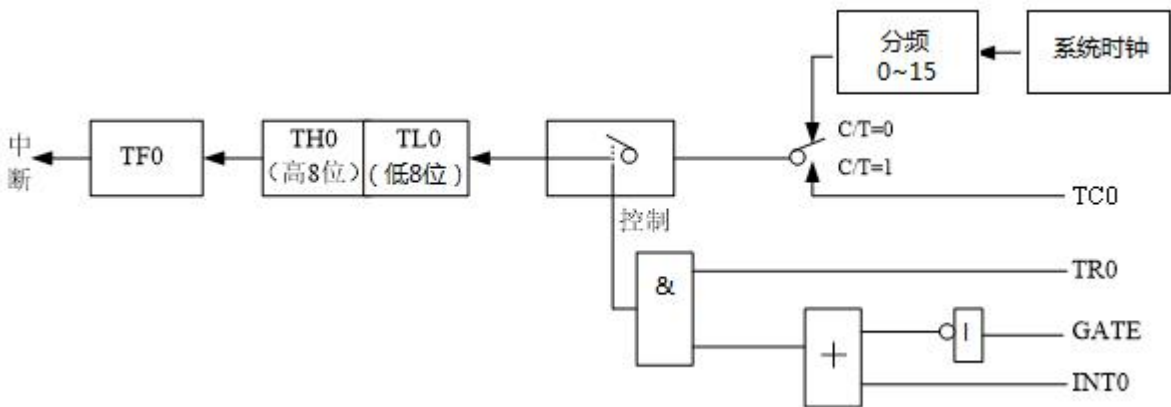


图11-3 T0/T1 方式1结构

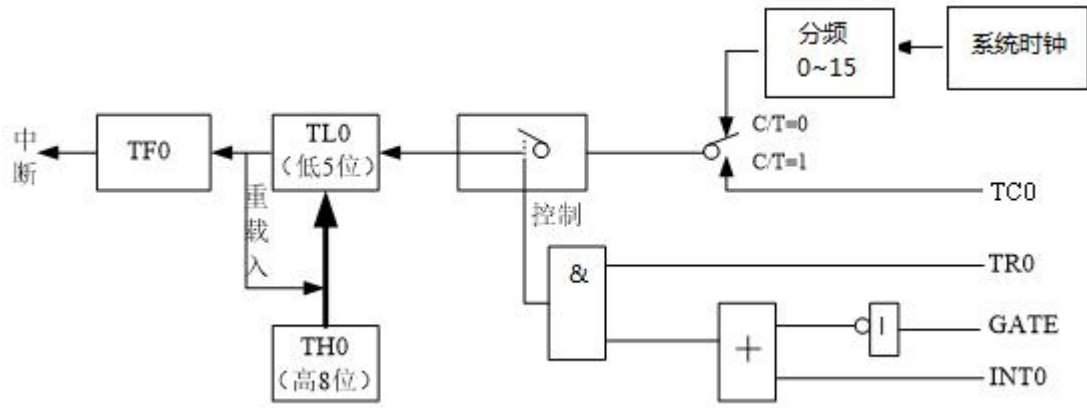
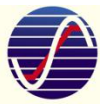


图11-4 T0/T1 方式2结构

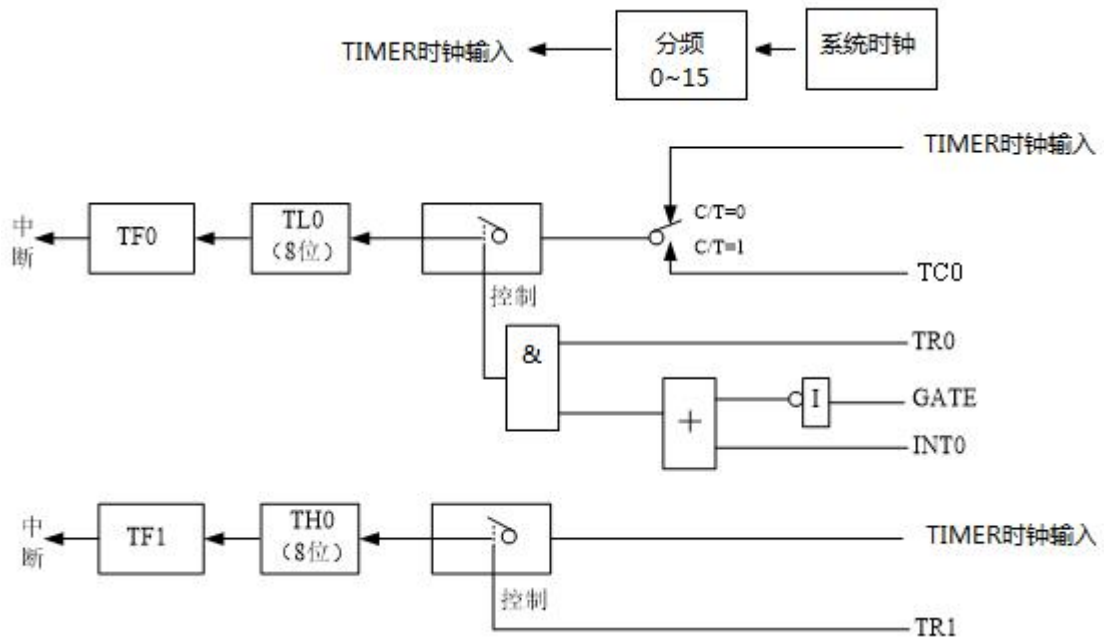
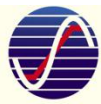


图11-5 T0 方式3结构





## 12. 定时器 2

### 12-1. 概述

定时器 2 是 16 位自动重载定时器，通过两个数据寄存器 TH2 和 TL2 访问。由 T2CON 寄存器控制。IEB 寄存器的 ET2 位 (bit3) 置 1 使能定时器 2 中断（详见中断章节）。

定时器 2 有四种工作模式：16 位自动重载定时器、16 位自动重载计数器、16 位 PWM 发生器、16 位定时器单通道捕获功能，可以设置分频比。

定时器 2 有一个 16 位计数器/定时器寄存器 (TH2, TL2)。当 TH2 与 TL2 被写时，用作定时器重载寄存器，当被读时，被用作计数寄存器。TR2 位置 1 使定时器 2 开始工作，在定时或计数模式下，定时器在 0xFFFF 发生溢出。

在 TR2 为 1 时，对 TH2/TL2 的写操作，不会影响计数器的值，只会改变重载寄存器的值，这个改变后的值在下次溢出时会被重载进计数寄存器。在 TR2 为 0 时，对 TH2/TL2 的写操作首先会改变重载寄存器的值，然后在 TR2 为 1 时，重载寄存器的值自动加载至计数寄存器。

#### 定时器 2 的四种工作模式介绍:

定时器 2 工作在计数模式时，计数器从设置的初值开始对外部引脚 TC2 的下降沿脉冲进行计数，当计数器计数至 0xFFFF 时发生溢出，若使能定时器 2 中断 (IEB[3]=1)，则可产生溢出中断。

定时器 2 工作在定时模式时，定时时钟源为系统时钟且可根据 T2\_CLKDIV 设置分频系数，定时器初值由寄存器 TH2/TL2 设置，定时器在 0xFFFF 发生溢出，若使能定时器 2 中断 (IEB[3]=1)，则可产生溢出中断。

定时器 2 工作在 PWM 输出模式时，可根据周期寄存器 T2\_PWMPH/T2\_PWMPL 设置 PWM 周期，可根据寄存器 T2\_PWMCH/T2\_PWMCL 设置 PWM 的占空比，示意图如下图 1 所示。

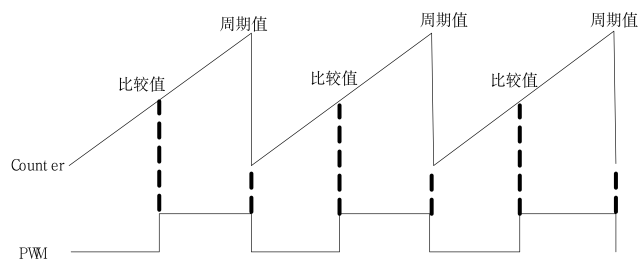


图 1 定时器 2 PWM 模式功能框图

定时器 2 工作在输入捕获模式时，可捕获外部引脚信号 TC2 的上升沿、下降沿对应的计数值，从而计算出 TC2 信号的周期、占空比。捕获模式的配置可通过 T2\_CAPCON 进行设置。

**注意：**在读 TH2 和 TL2 时，需要确保 TR2=0。（当 TR2=1 时，由于正在计数，读出的 TH2 与 TL2 不准）

### 12-2. 功能框图

定时器 2 在定时和计数模式下的结构功能图如下：

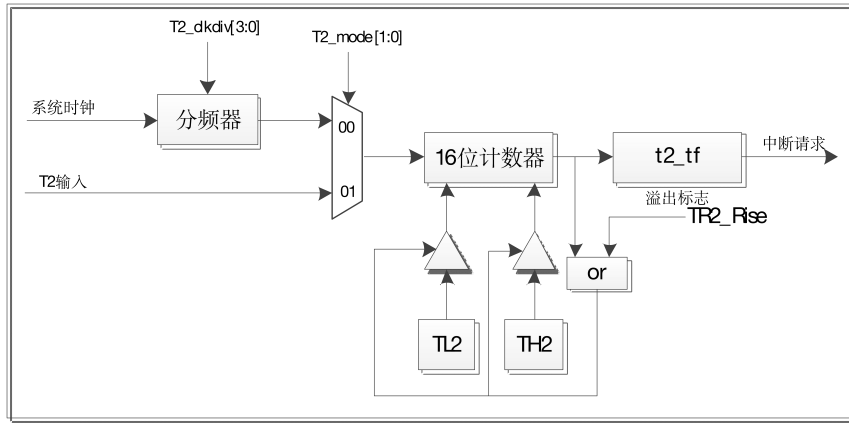
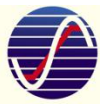


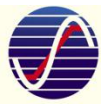
图 2 定时器 2 的定时/计数模式功能框图

### 12-3. 控制寄存器

#### 12-3-1. T2CON (91H) : 定时器 2 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位
91H	T2CON	T3TIM_EN	pwmoen	pwmcint_en	pwmpint_en	t2_tf	t2_tr	t2_mode	00H	
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R		

位编号	位符号	说明
7	T3TIM_EN	定时器 3 定时器模式使能 1: 定时器 3 工作在 16 位定时器模式 0: 定时器 3 关闭 16 位定时器模式
6	pwmoen	Timer2 的 PWM 输出使能。 1: 输出使能打开 0: 输出使能关闭
5	pwmcint_en	Timer2 的 PWM 比较中断使能。 1: 比较中断使能打开 0: 比较中断使能关闭
4	pwmpint_en	Timer2 的 PWM 周期中断使能。 1: 周期中断使能打开 0: 周期中断使能关闭
3	t2_tf	Timer2 的中断标志，硬件自清。
2	t2_tr	Timer2 的运行控制位。 1: Timer2 运行 0: Timer2 关闭
1-0	t2_mode	Timer2 的工作模式控制位。 00: 定时模式



		01: 计数模式 10: PWM 输出模式 11: 捕获模式
--	--	--------------------------------------

12-3-2. T2\_CAPCON (ACH) : 定时器 2 捕获控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位
91H	T2_CAPCON	-	-	-	capcr	cap2ls[1:0]		capen0	capf0	00H
读/写		-	-	-	W/R	W/R		W/R	W/R	

位编号	位符号	说明
4	capcr	捕获模式自动清除 该位使能,当捕获事件发生,在 TH2 与 TL2 内的数据移入 T2_PWMCH 与 T2_PWMCL 后, 应将自动清除 TH2 与 TL2 计数寄存器的值。 1: 捕获事件发生后定时器 2 的计数值自动清 0 0: 捕获事件发生后定时器 2 计数值按之前计数值继续累加
3-2	cap2ls[1:0]	输入捕获通道条件选择 00: 下降沿 01: 上升沿 10: 上升沿或下降沿 11: 保留
1	capen0	输入捕获通道使能位 1: 打开输入捕获通道 0: T 关闭输入捕获通道
0	capf0	输入捕获通道标志位 如果输入捕获通道检测到边沿触发事件发生, 该位由硬件置位, 由软件清零。

12-3-3. T2\_CLKDIV: 定时器 2 时钟分频寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位
ABH	T2_CLKDIV	-				T2_CLKDIV[3:0]				00H
读/写		-				W/R				

12-3-4. TH2/TL2: 定时器 2 的高 8 位/低 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位
A6H	TH2	TH2[7:0]								00H
A5H	TL2	TL2[7:0]								00H
读/写		W/R								

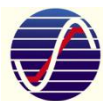


12-3-5. T2\_PWMPH/T2\_PWMPL: 定时器 2 PWM 周期寄存器的高 8 位/低 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位
A2H	T2_PWMPH	T2_PWMPH[1:0]								00H
A1H	T2_PWMPL	T2_PWMPL[1:0]								00H
读/写		W/R								

12-3-6. T2\_PWMCH/T2\_PWMCL: 定时器 2 PWM 比较寄存器或捕获寄存器的高 8 位/低 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位
A4H	T2_PWMCH	T2_PWMCH[1:0]								00H
A3H	T2_PWMCL	T2_PWMCL[1:0]								00H
读/写		W/R								



## 13. 定时器 3

定时器 3 是一个 16 位定时器，时钟源为系统时钟源，可以实现 16 位定时器的功能，如果定时器 3 中断使能打开，计数器溢出时可产生溢出中断。

定时器 3 带有定时器初值重载功能，可通过寄存器 T3CNTL、T3CNTH 设置定时器 3 的计数初值，初值的重载可通过设置寄存器 T3CON 的 RLDEN 位来实现，当 RLDEN 有效且定时器计数溢出时，T3CNTL 与 T3CNTH 中的值被自动加载进定时器的实际工作寄存器中。在定时器 3 初次工作时，RLDEN 位由 0 置 1 时也会使定时器 3 的初值被重载。用户根据寄存器 T3CNTL、T3CNTH 的值来选择芯片的定时时间。

在中断使能有效的前提下，当定时器 3 计数溢出，同时触发中断，中断标志位 T3OVF 需要软件清零。

### 13-1. 控制寄存器

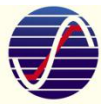
#### 13-1-1. T3CON (E9H) : Timer3 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位
E9H	T3CON	T3IE	T3OVF	Reload_EN	-	T3EN	-			00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R			

位编号	位符号	说明
7	T3IE	Timer3 中断使能 1: 中断使能打开 0: 中断使能关闭
6	T3OVF	定时器 3 中断标志 1: 定时器 3 计数溢出时，硬件自动置 1，需软件清 0 0: 定时器 3 没有产生溢出
5	RLDEN	定时器 3 初值重载使能 1: 计数器初值重载使能打开 0: 计数器初值重载使能关闭
4	-	保留
3	T3EN	定时器 3 工作使能 1: 定时器 3 使能 0: 定时器 3 禁用
2-0	-	保留

#### 13-1-2. T3CNTH (EBH) / T3CNTL (EAH) : 定时器 3 初值寄存器高 8 位/低 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位
EAH	T3CNTL	T3CNTL [7:0]								00H
EBH	T3CNTH	T3CNTLH[7:0]								00H



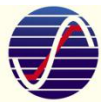
读/写	W/R	
-----	-----	--

13-1-3. T2CON (91H) : Timer2 控制寄存器

备注: 定时器 3 的工作模式使能借用定时器 2 控制寄存器的 Bit7.

地址	名称	7	6	5	4	3	2	1	0	上电复位
91H	T2CON	T3TIM_EN	pwmoen	pwmcint_en	pwmpint_en	t2_tf	t2_tr	t2_mode		00H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R		

位编号	位符号	说明
7	T3TIM_EN	定时器 3 定时器模式使能 1: 定时器 3 工作在 16 位定时器模式 0: 定时器 3 关闭 16 位定时器模式
6	pwmoen	Timer2 的 PWM 输出使能。 1: 输出使能打开 0: 输出使能关闭
5	pwmcint_en	Timer2 的 PWM 比较中断使能。 1: 比较中断使能打开 0: 比较中断使能关闭
4	pwmpint_en	Timer2 的 PWM 周期中断使能。 1: 周期中断使能打开 0: 周期中断使能关闭
3	t2_tf	Timer2 的中断标志, 硬件自清。
2	t2_tr	Timer2 的运行控制位。 1: Timer2 运行 0: Timer2 关闭
1-0	t2_mode	Timer2 的工作模式控制位。 00: 定时模式 01: 计数模式 10: PWM 输出模式 11: 捕获模式



## 14. UART0 单元

### 14-1. 概述

ASM87L164 的 UART0 (串行端口) 是全双工的, 即它可同时发送及接收, 同时带有接收缓冲, 所以前一接收字节从 S0BUF 寄存器内读出之前可开始接收下一字节。但是如果下一字节接收完毕而前一字节仍未读出, 则前一字节将被丢失。串行端口接收及发送寄存器均通过 S0BUF 访问。对 S0BUF 写操作则装入发送寄存器, 读 S0BUF 则访问一个物理上分开的接收寄存器。

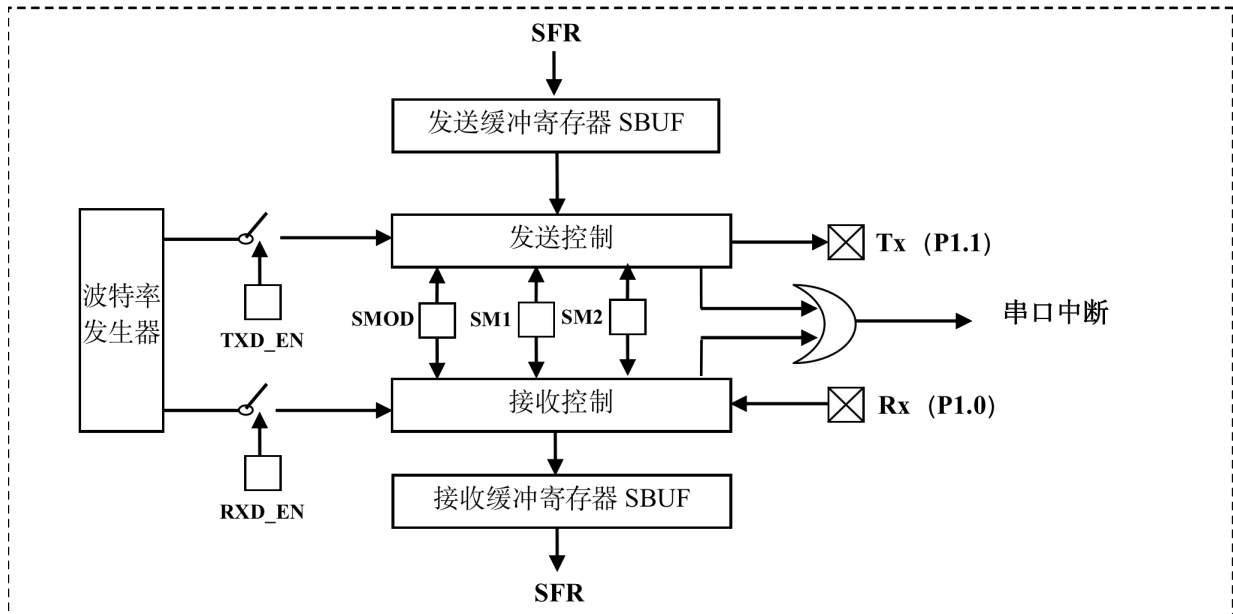


图14-1 UART0功能方框图

### 14-2. 功能描述

ASM87L164 的 UART0 有如下 4 种可选工作模式:

#### 1) 模式 0

串行数据通过 Rx 进出。Tx 输出时钟。每次发送或接收以 LSB (低位) 作首位, 每次 8 位。波特率固定为 MCU 时钟频率的  $1/2(F_{sys}/2)$ 。

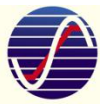
#### 2) 模式 1

Tx 脚发送, Rx 脚接收, 每次数据为 10 位, 一个起始位 (逻辑 0), 8 个数据位 (LSB 作首位) 及一个停止位 (逻辑 1)。当接收数据时, 停止位存于 SCON 的 RB8 内, 波特率可变, 由定时器 1 溢出速率决定。

#### 3) 模式 2

Tx 脚发送, Rx 脚接收, 每次数据为 11 位, 一个起始位 (逻辑 0), 8 个数据位 (LSB 作首位), 一个可编程第 9 位数据及一个停止位 (逻辑 1)。

发送时, 第 9 个数据位 (SCON 内 TB8 位) 可置为 0 或 1。例如奇偶位 (PSW 内 P 位) 移至 TB8。接收时, 第 9 位数据存入 SCON 的 RB8 位, 停止位忽略。波特率可编程为 MCU 时钟频率的  $1/32$  或  $1/64$ , 由 PCON 内 SMOD 位决定。



4) 模式 3

Tx 脚发送, Rx 脚接收, 每次数据为 11 位, 一个起始位 (逻辑 0), 8 个数据位 (LSB 为首位), 一可编程的第 9 位数据及一个停止位。事实上模式 3 除了波特率外均与模式 2 相同。其波特率可变并由定时器 1 溢出率决定。

在上述 4 种模式中, 发送过程是以一条写 SBUF 作为目标寄存器的指令开始的, 模式 0 时接收过程开始应设置 RI=0 及 REN=1, 其它模式下如若 REN=1 则通过起始位初始化。

### 14-3. 控制寄存器

#### 14-3-1. SCON0(98H): UART0 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
98H	SCON0	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	SM0	模式控制位 0
6	SM1	模式控制位 1
5	SM2	模式控制位 2, 多机通讯控制位
4	REN	UART0 接收使能
3	TB8	模式 2,3 的 第 9 位发送控制位
2	RB8	模式 2,3 的 第 9 位接收控制位
1	TI	UART0 发送中断标志
0	RI	UART0 接收中断标志

SM0、SM1: 模式控制, 见下表:

模式	SM0	SM1	UART0 工作模式	
模式 0	0	0	同步移位寄存器, 波特率 $F_{sys}/2$ 。	
模式 1	0	1	8 位 UART, 波特率可变	
模式 2	1	0	9 位 UART, 波特率 $F_{sys}/64$ 或 $F_{sys}/32$	
			SMOD	Baud Rate
			0	$F_{sys}/64$
		1	$F_{sys}/32$	
模式 3	1	1	9 位 UART, 波特率可变	

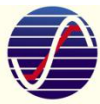
注: 可变波特率设计时参照公式(定时器 1 需工作于 8 位自动装载方式):

- $Baud\ Rate = 2^{smod} \times F_{sys} / (32 \times TDIV \times (256 - TH1))$
- $TH1 = 256 - (F_{sys} \times 2^{smod}) / (TDIV \times 32 \times Baud)$

说明:

- 1) TH1 是定时器 Timer1 的高 8 位。



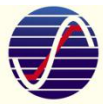


2) TDIV 是 Timer1 时钟对系统时钟 fsys 的分频比, Timer1 时钟默认为系统时钟 12 分频, 此时 TDIV=12, 建议 Timer1 作 UART0 波特率用时, Timer 时钟设置成系统时钟 1 分频, 此时 TDIV=1

SM2: 多机通信控制位, 多机通信是工作于方式 2 和方式 3, SM2 位主要用于方式 2 和方式 3。接收状态, 当串行口工作于方式 2 或 3, 以及 SM2=1 时, 只有当接收到第 9 位数据 (RB8) 为 1 时, 才把接收到的前 8 位数据送入 SBUF0, 且置位 RI 发出中断申请, 否则会将接收到的数据放弃。当 SM2=0 时, 就不管第 9 位数据是 0 还是 1, 都会将数据送入 SBUF0, 并发出中断申请。

14-3-2. SBUF0(99H): UART0 数据寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
99H	SBUF0	SBUF0[7:0]								00H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	



## 15. UART1 单元

### 15-1. 概述

UART1 和 UART0 在功能上是完全一样的，所以可参考 UART0 的功能描述和结构框图，不同的只是复用管脚和寄存器地址。

### 15-2. 控制寄存器

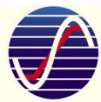
#### 15-2-1. SCON1(9BH): UART1 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
9BH	SCON1	SM10	SM11	SM12	REN1	TB18	RB18	TI1	RI1	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	SM10	模式控制位 0
6	SM11	模式控制位 1
5	SM12	模式控制位 2, 多机通讯控制位
4	REN1	UART1 接收使能
3	TB18	模式 2,3 的 9 <sup>th</sup> 发送控制位
2	RB18	模式 2,3 的 9 <sup>th</sup> 接收控制位
1	TI1	UART 发送中断标志
0	RI1	UART 接收中断标志

SM10、SM11: 模式控制, 见下表:

模式	SM10	SM11	UART1 工作模式	
模式 0	0	0	同步移位寄存器, 波特率 $F_{sys}/2$ 。	
模式 1	0	1	8 位 UART, 波特率可变	
模式 2	1	0	9 位 UART, 波特率 $F_{sys}/64$ 或 $F_{sys}/32$	
			SMOD	Baud Rate
			0	$F_{sys}/64$



			1	Fsys/32
模式 3	1	1	9 位 UART, 波特率可变	

注：可变波特率设计时参照公式(定时器自动装载方式工作):

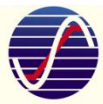
- $Baud\ Rate = 2^{smod} \times F_{sys} / (32 \times TDIV \times (256 - TH1))$
- $TH1 = 256 - (F_{sys} \times 2^{smod}) / (TDIV \times 32 \times Baud)$

这里的 TH1 是定时器 Timer1 的高 8 位。

SM12: 多机通信控制位, 多机通信是工作于方式 2 和方式 3, SM12 位主要用于方式 2 和方式 3。接收状态, 当串行口工作于方式 2 或 3, 以及 SM12=1 时, 只有当接收到第 9 位数据 (RB18) 为 1 时, 才把接收到的前 8 位数据送入 S1BUF, 且置位 RI1 发出中断申请, 否则会将接收到的数据放弃。当 SM12=0 时, 就不管第 9 位数据是 0 还是 1, 都会将数据送入 S1BUF, 并发出中断申请。

#### 15-2-2. SBUF1(9CH): UART1 数据寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
9CH	SBUF1	SBUF1[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	



## 16. I2C 单元

### 16-1. 概述

ASM87L164 内置 I2C 模块，I2C 为双线，双向串行总线，为设备之间的数据通讯提供了简单有效的方法。标准 I2C 是多主机总线，包括总线冲突检测和仲裁机制以防止在两个或多个主机试图同时控制总线时发生的数据冲突。

数据在主机与从机通过 SCL 时钟线控制在 SDA 数据线上实现一字节一字节的同步传输，一个 SCL 时钟脉冲传输 1 个数据位，数据由最高位 MSB 首先传输，每个传输字节后跟随一个应答位，每个位在 SCL 为高时采样；因此，SDA 线只有在 SCL 为低时才可以改变，在 SCL 为高时 SDA 保持稳定。当 SCL 为高时，SDA 线上的跳变视为命令中断（START 或 STOP），参考下图 I2C 总线时序。。

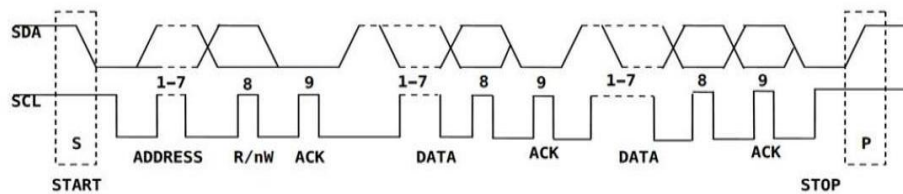
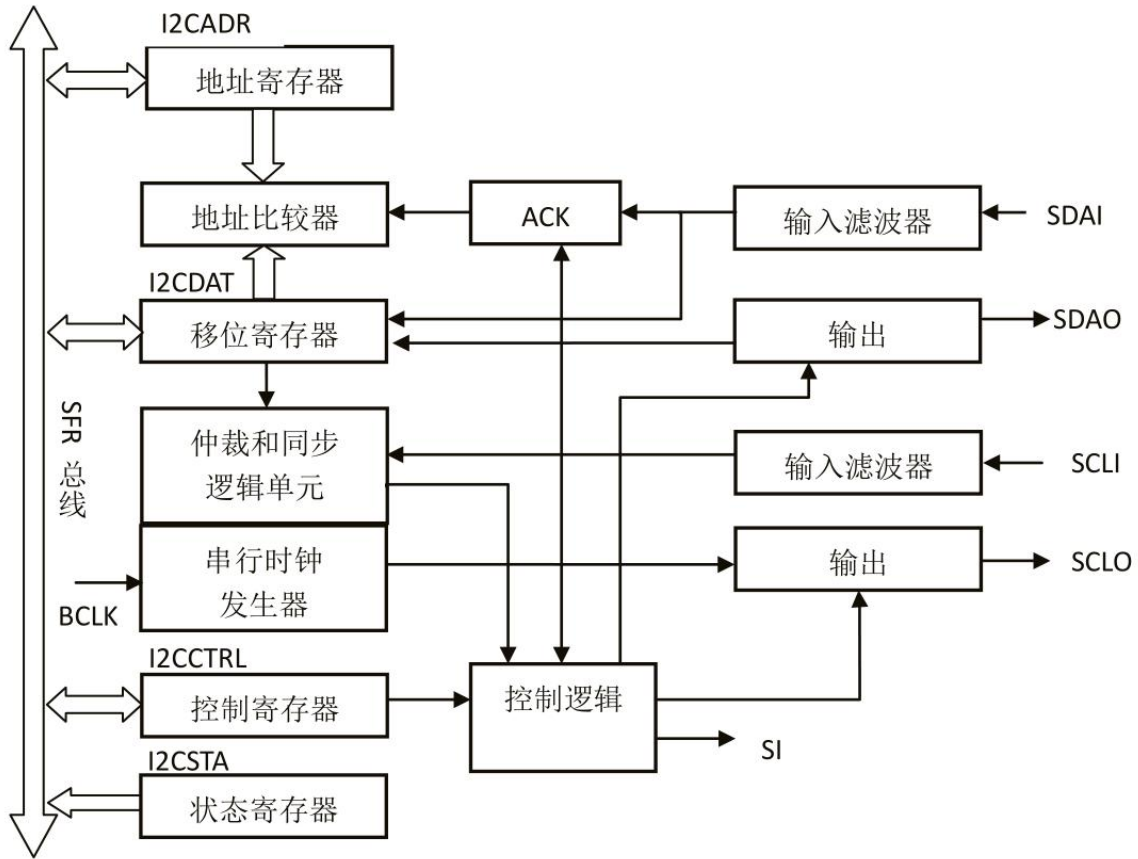
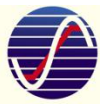


图16-1 I2C总线时序

### 16-2. 功能描述

ASM87L164 I2C 总线使用双线连接到总线“SCL”（串行时钟线）和“SDA”（串行数据线）的设备间传送信息。由于只有无方向端口，I2C 组件需要使用到引脚的漏端开路缓冲器。每个连接到总线的设备都能使用软件通过特定地址寻址。I2C 标准是一个具有冲突检测机制和仲裁机制的真正意义上的多主机总线。它能防止两个或者多个主机在同时开始传输数据时发生数据冲突。滤波逻辑可以过滤数据总线上的毛刺来保护数据的完整性。I2C 总线控制器执行 8bit 的双向数据传输，标准模式最高支持 100kbit/s，高速模式最高支持 400kbit/s。



ASM87L164 的 I2C 支持以下 4 种操作模式:

1) 主发送模式:

在 SCL 输出串行时钟时, 通过 SDA 输出串行数据。

2) 主接收模式:

在 SCL 输出串行时钟时, 串行数据由 SDA 接收。

3) 从接收模式:

串行数据和串行时钟由 SDA 和 SCL 接收。

4) 从发送模式:

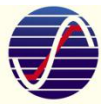
在 SCL 输入串行时钟时, 通过 SDA 输出串行数据。

设置 ENS1 寄存器后, 硬件处于非寻址从模式。在从模式, I2C 寻找它自己的从地址和一般呼叫地址。如果其中一个地址被发现, I2C 切换到地址从模式, 并产生中断。之后 I2C 作为从发射器或从接收器操作。当 MCU 希望把 I2C 总线设置为主时, I2C 模块要等到 I2C 总线自由释放。当 I2C 总线自由释放, I2C 模块产生一个 Start, 发送从地址和读写位。根据传输方向位, I2C 可作为主发送器或主接收器操作来操作。MCU 可以用 4 个 SFR 寄存器, 控制 I2C 的 4 种模式。

## 16-3. 工作模式

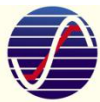
### 16-3-1. I2C 主机发送模式

必须将 ENS 置“1”来使能 I2C 模块。如果 aa 位复位, 当另一个器件正变成总线主机时, I2C 模块将不会应答其自身的从机地址或通用调用地址。换句话说, 如果 AA 位复位, I2C 接口就不能进入从机模式。STA, STO 和 SI 必须复位。此时, 可通过置位 STA 位进入主发送模式。一旦总线空闲, I2C 逻辑会马上



测试 I2C 总线并产生一个起始条件。当发送起始条件时，串行中断标志 (SI) 置位，状态寄存器 (I2CSTA) 中的状态代码为 0x08。中断服务程序利用该状态代码进入相应的状态服务程序，将从机地址和数据方向位 (SLA+W) 装入 I2CDAT。I2CCTRL 的 SI 位必须在串行传输继续之前复位。当发送完从机地址和方向位且接收到一个应答位时，串行中断标志 (SI) 再次置位，I2CSTA 中可能是一系列不同的状态代码。主机模式下为 0x18, 0x20 或 0x38，从机模式 (AA=逻辑 1) 为 0x68, 0x78 或 0xB0。每个状态代码对应的操作在下表中详细介绍。在发送完重复起始条件 (状态 0x10) 后，I2C 模块通过将 SLA+R 装入 I2CDAT 切换到主接收模式。

状态代码	I2C 总线和硬件状态	应用软件响应					I2C硬件执行的下一个动作
		读/写 I2CDAT	写 I2CCTRL				
			ST A	ST O	SI	AA	
08 H	已发送起始条件	装入SLA+W	X	0	0	X	将发送 SLA+W ， 接收 ACK
10 H	已发送重复起始条件	装入SLA+W	X	0	0	X	同上
		装入 SLA+R	X	0	0	X	将发送 SLA +R ， I2C 自动切换到主接收模式
18 H	已发送 SLA+W 已接收 ACK	装入数据字节	0	0	0	X	将发送数据字节， 将接收 ACK
		无I2CDAT动作	1	0	0	X	将发送重复起始条件
		无 I2CDAT 动作	0	1	0	X	将发送停止条件， STO 标志位复位
		无 I2CDAT 动作	1	1	0	X	将先发送停止条件， 随后发送起始条件， STO 标志位复位
20 H	已发送 SLA +W 已接收非 ACK	装入数据字节	0	0	0	X	将发送数据字节， 将接收 ACK
		无I2CDAT动作	1	0	0	X	将发送重复起始条件
		无 I2CDAT 动作	0	1	0	X	将发送停止条件， STO 标志位复位
		无 I2CDAT 动作	1	1	0	X	将先发送停止条件， 随后发送起始条件， STO 标志位复位
28 H	已发送 I2CDAT 中的数据， 已接	装入数据字节	0	0	0	X	将发送数据字节， 将接收 ACK
		无I2CDAT动作	1	0	0	X	将发送重复起始条件



	收ACK	无 I2CDAT 动作	0	1	0	X	将发送停止条件, STO 标志位复位
		无 I2CDAT 动作	1	1	0	X	将先发送停止条件, 随后发送起始条件, STO 标志位复位
30 H	已发送 I2CDAT 中的数据	装入数据字节	0	0	0	X	将发送数据字节, 将接收 ACK
		无 I2CDAT 动作	1	0	0	X	将发送重复起始条件
		无 I2CDAT 动作	0	1	0	X	将发送停止条件, STO 标志位复位
		无 I2CDAT 动作	1	1	0	X	将先发送停止条件, 随后发送起始条件, STO 标志位复位
38 H	在 SLA+R/W 或写数据字节时丢失仲裁	无 I2CDAT 动作	0	0	0	X	I2C 总线被释放, 进入不可寻址从模式
		无 I2CDAT 动作	1	0	0	X	当 I2C 总线空闲时发送起始条件

表 12 I2C 主机发送模式状态表

举例如下:

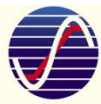
```

I2CCTRL = 0x63;           //使能 I2C、清除标志位、发送 Start
for(i=0; i<1; i++)
{ _nop_(); _nop_(); _nop_(); _nop_(); _nop_();
  _nop_(); _nop_(); _nop_(); _nop_(); _nop_();}
I2CDAT = 0xA0;           //发送数据 0xA0
I2CCTRL = 0x43;         //清除标志位
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
I2CDAT = 0x00;         //发送数据 0x00
I2CCTRL = 0x43;         //清除标志位
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
I2CDAT = 0x96;         //发送数据 0x96
I2CCTRL = 0x43;         //清除标志位
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
I2CCTRL = 0x53;         //使能 Stop、清中断
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志

```

### 16-3-2. I2C 主机接收模式

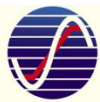
在主接收模式中, 主机所接收的数据字节来自从发送器。按主发送模式中的方法初始化传输。当发送



完起始条件后，中断服务程序必须把 7 位从机地址和数据方向位 (SLA+R) 装入 I2CDAT。必须先清除 I2CCTRL 中的 SI 位，再继续执行串行传输。当发送完从机地址和数据方向位且接收到一个应答位时，串行中断标志 SI 再次置位，这时，I2CSTA 中可能是一系列不同的状态代码。主机模式下为 0x40, 0x48 或 0x38，从机模式 (AA=1) 为 0x68, 0x78 或 0xB0。每个状态代码对应的操作详见下表。在发送完重复起始条件 (状态 0x10) 后，I2C 模块通过将 SLA+W 装入 I2CDAT 切换到主发送模式。

状态代码	I2C 总线和硬件状态	应用软件响应				I2C硬件执行的下一个动作	
		读/写 I2CDAT	写 I2CCTRL				
			ST A	ST O	SI		AA
08H	已发送起始条件	装入SLA+R	X	0	0	X	将发送 SLA+R，接收 ACK
10H	已发送重复起始条件	装入SLA+R	X	0	0	X	同上
		装入 SLA+W	X	0	0	X	将发送 SLA+W，I2C 自动切换到主接收模式
38H	在非ACK中丢失仲裁	无I2CDAT动作	0	0	0	X	I2C 总线被释放，进入从模式
		无 I2CDAT 动作	1	0	0	X	当 I2C 总线空闲时发送起始条件
40H	已发送 SLA+R 已接收ACK	无I2CDAT动作	0	0	0	0	将接收数据字节，将返回非 ACK
		无 I2CDAT 动作	0	0	0	1	将接收数据字节，将返回 ACK
48H	已发送 SLA+R 已接收非 ACK	无 I2CDAT 动作	1	0	0	X	将发送重复起始条件
		无 I2CDAT 动作	0	1	0	X	将发送停止条件，STO 标志位复位
		无 I2CDAT 动作	1	1	0	X	将先发送停止条件，随后发送起始条件，STO 标志位复位
50H	已接收数据字节，ACK已返回	读取数据字节	0	0	0	0	将接收数据字节，将返回非 ACK
		读取数据字节	0	0	0	1	将接收数据字节，将返回 ACK
58H	已接收数据字	读取数据字节	1	0	0	X	将发送重复起始条件





H	节, 非ACK已返回	读取数据字节	0	1	0	X	将发送停止条件, STO 标志位复位
---	------------	--------	---	---	---	---	--------------------

表18-3 I2C 主机接收模式状态表

举例如下:

```

I2CCTRL = 0x67;           //使能 I2C、使能 Strat、清中断
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
I2CDAT  = 0xA0;           //发送数据 0xA0
I2CCTRL = 0x47;           //清标志位
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
I2CDAT  = 0x00;           //发送数据 0x00
I2CCTRL = 0x47;           //清标志位
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
//Slave 自动回复 ACK

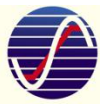
I2CCTRL = 0x67;           //使能 Strat、清中断
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
I2CDAT  = 0xA1;           //发送数据 0xA1
I2CCTRL = 0x47;           //清标志位
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志
I2CCTRL = 0x43;           //清标志位、无需接收 ACK
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志

I2CCTRL = 0x57;           //使能 Stop、清中断
while((I2CCTRL & 0x08)==0x00) {} //等待中断标志

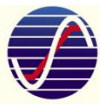
```

### 16-3-3. I2C 从机接收模式

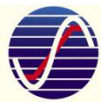
在从接收模式中，从机接收的数据字节来自发送器。高 7 位是主机寻址时 I2C 模块响应的地址。如果 LSB (GC) 被置位，I2C 模块将响应通用调用地址 (0x00)；否则忽略通用调用地址。I2C 总线速率的设置不影响从机模式中的 I2C 模块。必须置位 I2CEN 来使能 I2C 模块。AA 位必须置位以使能 I2C 模块来应答其自身从机地址或通用调用地址。STA, STO 和 SI 必须复位。当 I2CADR 和 I2CCTRL 完成初始化后，I2C 模块一直等待，直至被从机地址寻址，之后是数据方向位寻址，为了工作在从接收模式中，数据方向位必须为“0” (W)。接收完其自身的从机地址和 W 位后，串行中断标志 (SI) 置位，可从 I2CSTA 中读出一个有效的状态代码。该状态代码用作状态服务程序的向量。每个状态代码的对应操作见下表。如果 I2C 模块在主机模式中仲裁丢失，也可进入从接收模式 (请参考状态 0x68 和 0x78 的描述)。如果 AA 位在传输过程中复位，则在接收完下一个数据字节后 I2C 模块将向 SDA 返回一个非应答 (逻辑 1)。当 AA 复位时，I2C 模块不响应其自身的从机地址或通用调用地址。但是，I2C 总线仍被监控，而且，地址识别可随时通过置位 AA 来恢复。这就意味着 AA 位可临时将 I2C 模块从 I2C 总线上分离出来。



状态代码	I2C 总线和硬件状态	应用软件响应					I2C硬件执行的下一个动作
		读/写 I2CDAT	写 I2CCTRL				
			ST A	ST O	SI	AA	
60 H	已接收自身的 SLA+W; 已接收 ACK	无 I2CDAT 动作	X	0	0	0	将接收数据字节, 将返回非 ACK
		无 I2CDAT 动作	X	0	0	1	将接收数据字节, 将返回 ACK
68 H	主控时在 SLA+R/W 丢失仲裁; 已接收自身的 SLA+W; 已返回 ACK;	无 I2CDAT 动作	X	0	0	0	将接收数据字节, 将返回非 ACK
		无 I2CDAT 动作	X	0	0	1	将接收数据字节, 将返回 ACK
70 H	已接收通用调用地址 (0x00); 已返回 ACK;	无 I2CDAT 动作	X	0	0	0	将接收数据字节, 将返回非 ACK
		无 I2CDAT 动作	X	0	0	1	将接收数据字节, 将返回 ACK
78 H	主控时在 SLA+R/W 中丢失仲裁; 已接收通用调用地址; 已返回 ACK;	无 I2CDAT 动作	X	0	0	0	将接收数据字节, 将返回非 ACK
		无 I2CDAT 动作	X	0	0	1	将接收数据字节, 将返回 ACK
80 H	前一次寻址使用自身从地址; 已接收数据字节; 已返回 ACK;	无 I2CDAT 动作	X	0	0	0	将接收数据字节, 将返回非 ACK
		无 I2CDAT 动作	X	1	0	1	将接收数据字节, 将返回 ACK
88 H	前一次寻址使用自身从地址; 已接收数据字节; 已返回非 ACK;	读取数据字节	0	0	0	0	切换到不可寻址从模式; 不识别自身从地址或通用地址;
		读取数据字节	0	0	0	1	切换到不可寻址从模式; 不识别自身从地址或通用地址;



		读取数据字节	1	0	0	0	切换到不可寻址从模式； 不识别自身从地址或通用地址；
		读取数据字节	1	0	0	1	切换到不可寻址从模式； 不识别自身从地址或通用地址； 当总线空闲后发送起始条件；
90 H	前一次寻址使用通用调用地址；已接收数据；已返回ACK；	读取数据字节	1	0	0	X	将接收数据字节，将返回非 ACK
		读取数据字节	0	1	0	X	将接收数据字节，将返回 ACK
98 H	前一次寻址使用通用调用地址；已接收数据；已返回非ACK；	读取数据字节	0	0	0	0	切换到不可寻址从模式； 不识别自身从地址或通用地址；
		读取数据字节	0	0	0	1	切换到不可寻址从模式； 不识别自身从地址或通用地址；
		读取数据字节	1	0	0	0	切换到不可寻址从模式； 不识别自身从地址或通用地址； 当总线空闲后发送起始条件；
		读取数据字节	1	0	0	1	切换到不可寻址从模式； 不识别自身从地址或通用地址； 当总线空闲后发送起始条件；
A0 H	当使用从接收/从发送模式中静态寻址时，接收到停止条件或重复起始条件	无 I2CDAT 动作	0	0	0	0	切换到不可寻址从模式； 不识别自身从地址或通用地址；
		无 I2CDAT 动作	0	0	0	1	切换到不可寻址从模式； 不识别自身从地址或通用地址；
		无 I2CDAT 动作	1	0	0	0	切换到不可寻址从模式； 不识别自身从地址或通用地址； 当总线空闲后发送起始条件；



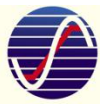
		无 I2CDAT 动作	1	0	0	1	切换到不可寻址从模式； 不识别自身从地址或通用地址； 当总线空闲后发送起始条件；
--	--	-------------	---	---	---	---	--

表18-3 I2C 从机接收模式状态表

#### 16-3-4. I2C 从机发送模式

在从发送模式中，向主接收器发送数据字节。数据传输按照从接收模式中的情况初始化。当初始化 I2CADR 和 I2CCTRL 后，I2C 模块一直等待，直至被自身的从机地址寻址，之后是数据方向位，该数据方向位必须为“1”（R），以便 I2C 模块工作在从发送模式下。接收完其自身的从机地址和 R 位后，串行中断标志（SI）置位，并且可从 I2CSTA 中读取一个有效的状态代码。该状态代码用作状态服务程序的向量，每个状态代码的对应操作见下表所示。如果 I2C 模块在主机模式下时仲裁丢失，则可进入从发送模式（见状态 0xB0）。如果 AA 位在传输过程中复位，则 I2C 模块将发送最后一个字节并进入状态 0xC0 或 0xC8。I2C 模块切换到非寻址的从机模式，如果继续传输，它将忽略主接收器。因此主接收器接收所有 1 作为串行数据。当 AA 复位时，I2C 模块不响应其自身的从机地址或通用调用地址。但是，I2C 总线仍被监控，而且，地址识别可随时通过置位 AA 来恢复。这就意味着 AA 位可用来暂时将 I2C 模块从 I2C 总线上分离出来。

状态代码	I2C 总线和硬件状态	应用软件响应					I2C硬件执行的下一个动作
		读/写 I2CDAT	写 I2CCTRL				
			ST A	ST O	SI	AA	
A8 H	已接收自身的 SLA+R; 已返回 ACK	装入数据字节	X	0	0	0	将发送最后一个数据字节; 将接收 ACK;
		装入数据字节	X	0	0	1	将发送一个数据字节; 将接收 ACK;
B0 H	主控时在 SLA+R/W 丢失仲裁; 已接收自身的 SLA+R; 已返回 ACK;	装入数据字节	X	0	0	0	将发送最后一个数据字节; 将接收 ACK;
		装入数据字节	X	0	0	1	将发送一个数据字节; 将接收 ACK;
B8 H	已发送数据; 已接收 ACK;	装入数据字节	X	0	0	0	将发送最后一个数据字节; 将接收 ACK;
		装入数据字节	X	0	0	1	将发送一个数据字节; 将接收 ACK;



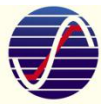
C0 H	已发送数据字节; 已接收非ACK;	无 I2CDAT 动作	0	0	0	0	切换到不可寻址从模式; 不识别自身从地址或通用地址;
		无 I2CDAT 动作	0	0	0	1	切换到不可寻址从模式; 不识别自身从地址或通用地址;
		无 I2CDAT 动作	1	0	0	0	切换到不可寻址从模式; 不识别自身从地址或通用地址; 当总线空闲后发送起始条件;
		无 I2CDAT 动作	1	0	0	1	切换到不可寻址从模式; 不识别自身从地址或通用地址; 当总线空闲后发送起始条件;
C8 H	装入的数据字节已被发送; 已接收ACK;	无 I2CDAT 动作	0	0	0	0	切换到不可寻址从模式; 不识别自身从地址或通用地址;
		无 I2CDAT 动作	0	0	0	1	切换到不可寻址从模式; 不识别自身从地址或通用地址;
		无 I2CDAT 动作	1	0	0	0	切换到不可寻址从模式; 不识别自身从地址或通用地址; 当总线空闲后发送起始条件;
		无 I2CDAT 动作	1	0	0	1	切换到不可寻址从模式; 不识别自身从地址或通用地址; 当总线空闲后发送起始条件;

表 18-5 从机发送模式状态表

### 16-3-5. I2C 其他状态

I2CSTA= 0xF8: 这个状态码表示没有任何可用的相关信息, 因为串行中断标志 SI 还没有置位。这种情况在其它状态和 I2C 模块还未开始执行串行传输之间出现。

I2CSTA= 0x00: 该状态代码表示在 I2C 串行传输过程中出现了总线错误。当格式帧的非法位置上出现了起始或停止条件时总线错误产生。这些非法位置是指在串行传输过程中的地址字节、数据字节或应答位。当外部干扰影响到内部 I2C 模块信号时也会产生总线错误。总线错误出现时 SI 置位。要从总线错误中恢复, STO 标志必须置位, SI 必须被清除。这使得 I2C 模块进入“非寻址的”从机模式 (已定义的状态)



并清除 STO 标志 (I2CCTRL 中的其它位不受影响)。SDA 和 SCL 线被释放 (不发送停止条件)。

状态代码	I2C 总线和硬件状态	应用软件响应				I2C硬件执行的下一个动作	
		读/写 I2CDAT	写 I2CCTRL				
			ST A	ST O	SI		AA
F8 H	无可用的相关状态信息; SI=0;	无 I2CDAT 动作	无 I2CDAT 动作				等待或执行当前传输
00 H	由于非法的起始或停止条件的出现, 在主机或被选中的从机将出现总线错误; 当外部干扰使 I2C 进入未定义的状态时也会出 0x00 状态	无 I2CDAT 动作	0	1	0	X	只有在主机或被寻址的从机模式中, 内部硬件受影响。一般情况下, 总线被释放, I2C 模块切换到非寻址的从机模式。STO 复位。

表 18-6 其他状态表

### 16-4. I2C 总线上数据传输

主机发出从机接收 7 位地址(一个字节), 传输方向未改变。

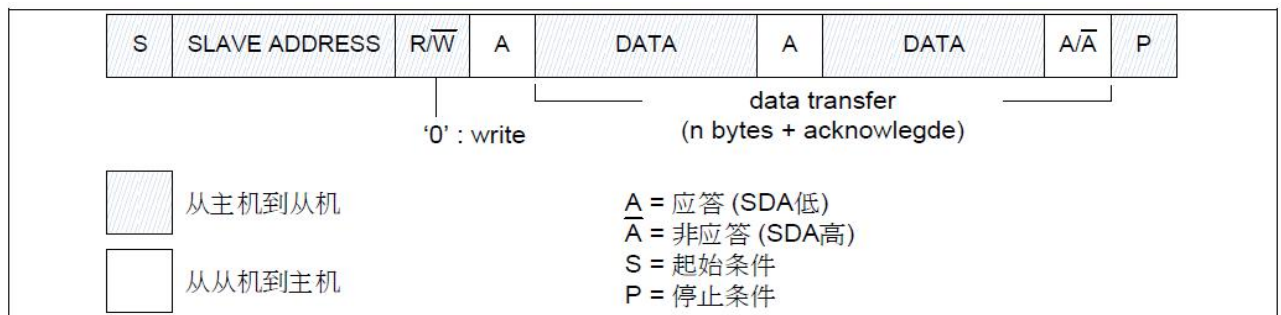


图 15-1 主机向从机传输数据

第一个字节后主机紧接着由从机读取数据(内容为从机地址), 传输方向改变。



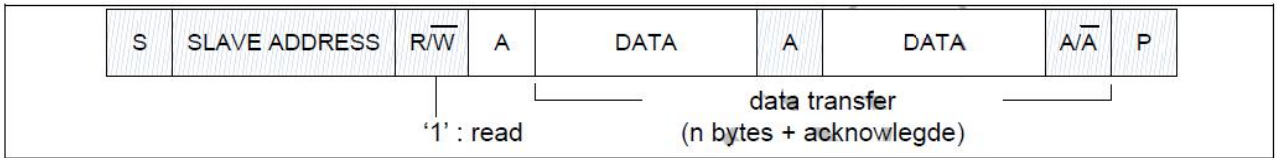
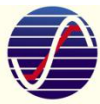


图 15-2 主机由从机读取地址

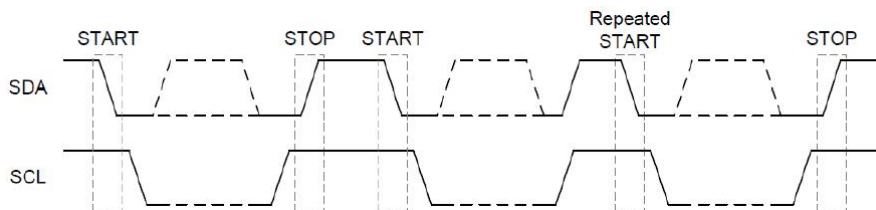
#### 16-4-1. 起始和重复起始信号

当总线处于空闲状态下，说明没有主机对总线发起传输请求(SCL 和 SDA 线同时为高)，主机可以通过发送一个 START 信号来发起传输请求。

起始信号：通常表示为 S-bit，当 SCL 线为高时，SDA 线上信号由高至低，标示总线上产生起始信号，新的传输开始。

重复起始信号(Sr)：即在两个 START 信号之间没有 STOP 信号。主机采用这种方法与另一个从机或相同的从机以不同传输方向进行通信（例如：从写入设备到从设备读出），而不释放总线。

STOP 信号：主机向总线发出停止信号结束数据传送。停止信号，通常用 P-bit 表示，当 SCL 线为高时，SDA 线上出现由低到高的信号，被定义为停止信号。



START 和 STOP 时序

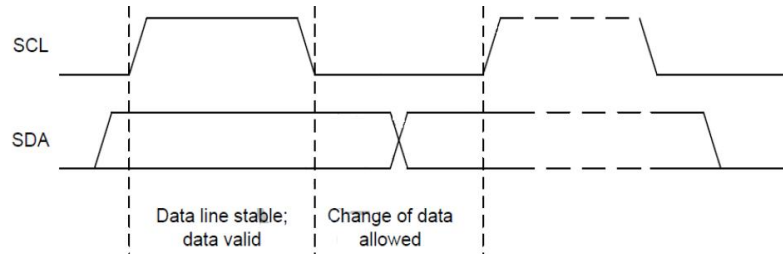
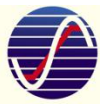
#### 16-4-2. 从机地址传输

START 信号是从机地址时，主机立即传输数据的第一位。这是一个跟随有一个 RW 位的 7 位调用地址，RW 位控制从机的信号传输方向。系统中没有两个从机有相同的地址，只有被主机寻址的从机会通过在第 9 个 SCL 时钟周期将 SDA 置为低电平作为应答。

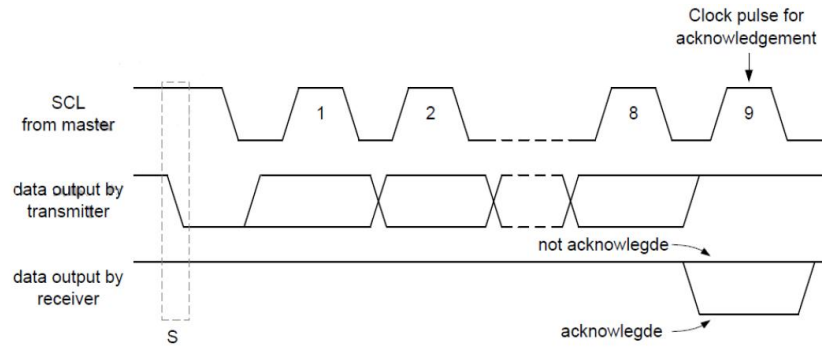
#### 16-4-3. 数据传输

当从机地址被成功识别，就可以根据 RW 所决定的方向，开始一字节一字节的数据传输，每个传输字节最后带一个第 9 时钟周期上的响应信号，如果从机上产生无响应信号(NACK)，主机可以产生停止信号来退出数据传输，或者产生重复起始信号开始新一轮的数据传输。

当主机作为接收器件时，发生无响应信号(NACK)，从机释放 SDA 线，使主机产生停止信号或重复起始信号。



I2C 总线上位传输



I2C 总线上应答信号

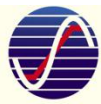
## 16-5. 控制寄存器

### 16-5-1. I2CCTRL: I2C 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
ECH	I2CCTRL	CR2	ENS1	STA	STO	SI	AA	CR1	CR0	00H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	CR2	时钟速率控制位 bit2
6	ENS1	<b>I2C 模块使能</b> 当 ENS1=0 时，SDA 和 SCL 输出高阻，SDA 和 SCL 输入信号被忽略； 当 ENS1=1 时，I2C 单元使能。
5	STA	<b>START 标志使能</b> 当 STA=1，I2C 模块检查 I2C 总线，当总线空闲时，产生 START 状态。
4	STO	<b>STOP 标志使能</b> 当 STOP=1 并且 I2C 模块为主模式，STOP 状态将会发送到 I2C 总线。

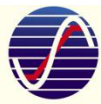




3	SI	<p>I2C 中断标志位</p> <p>在 26 个 I2C 状态中，除了状态 F8 以外，其他 25 个 I2C 状态都可以进入硬件设置中断标志电路。SI 被硬件设置为 1，SI 标志必须被软件清零。</p>
2	AA	<p>应答标志使能</p> <p>当 AA=1 且下面的事件发生时，将返回一个 ACK:</p> <ul style="list-style-type: none"> <li>- 从地址被接收到;</li> <li>- 当 GC 位被设置有效时，广播地址被接收到;</li> <li>- 当为主接收模式时，一个字节的数据被接收到;</li> <li>- 当为从接收模式时，一个字节的数据被接收到;</li> </ul> <p>当 AA=0 时且下面的事件发生时，无 ACK 返回:</p> <ul style="list-style-type: none"> <li>- 当为主接收模式时，一个字节的数据被接收到;</li> <li>- 当为从接收模式时，一个字节的数据被接收到。</li> </ul>
1	CR1	时钟速率控制位 bit1
0	CR0	时钟速率控制位 bit0

下表表示系统时钟为各种频率，分频配置寄存器 CR 配置如表内时，SCL 的频率值。

CR2	CR1	CR0	Bit frequency(kHz) at clk				Clk divided by
			6MHz	12MHz	16MHz	24MHz	
0	0	0	94	188	250	375	64
0	0	1	107	214	286	428	56
0	1	0	125	250	333	500	48
0	1	1	150	300	400	600	40
1	0	0	25	50	67	100	240
1	0	1	200	400	533	800	30
1	1	0	400	800	1067	1600	15
1	1	1	3000	6000	8000	12000	2



### 16-5-2. I2CADR: I2C 地址寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
EDH	I2CADR	ADR	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0	GC	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7:1	ADR	I2C 从机模式地址
0	GC	广播地址应答使能.  1-使能;  0-禁止;

### 16-5-3. I2CDAT: I2C 数据寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
EFH	I2CDAT	DAT	DAT6	DAT5	DAT4	DAT3	DAT2	DAT1	DAT0	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7:0	DAT	I2C 串行数据

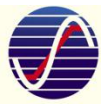
### 16-5-4. I2CSTA: I2C 状态寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
EEH	I2CSTA	STA	STA3	STA2	STA1	STA0	保留	保留	保留	F8H
只读		R	R	R	R	R	R	R	R	

位编号	位符号	说明
7:3	STA[4:0]	I2C 状态寄存器  详见 15-3 章节。
2:1	保留	



0	<i>I2CEN</i>	I2C 输出使能  1: 输出使能打开  0: 输出使能关闭
---	--------------	--



## 17. SPI 单元

### 17-1. 概述

SPI(Serial Peripheral Interface)总线是同步串行外设接口。SPI 接口使用 4 条线: 串行时钟线 (SCK), 主机输出/从机输入线 (MOSI), 主机输入/从机输出线 (MISO), 低电平有效从机选择线 (CS)。模块支持主/从两种工作模式, SPI 内部有独立的但有关联的 SPI\_MASTER 和 SPI\_SLAVE 两个模块, 当配置为主模式时, SPI\_MASTER 工作; 当配置为从模式时, SPI\_SLAVE 工作。SPI 端口是全双工, 即它可同时发送及接收, 同时带有接收缓冲, 所以前一接收字节从缓存寄存器内读出之前可开始接收下一字节。但是如果下一字节接收完毕而前一字节仍未读出, 则前一字节将被丢失。SPI\_MASTER 和 SPI\_SLAVE 的写操作则均装入同一个发送寄存器, 对 SPI 的读操作则均装入同一个接收寄存器。

SPI 工作在主模式时, 串行时钟分频器根据分频控制寄存器 SPICLKDIV[7:0], 产生串行时钟 SCK; 当工作在从模式时, 该分频控制寄存器不起作用。

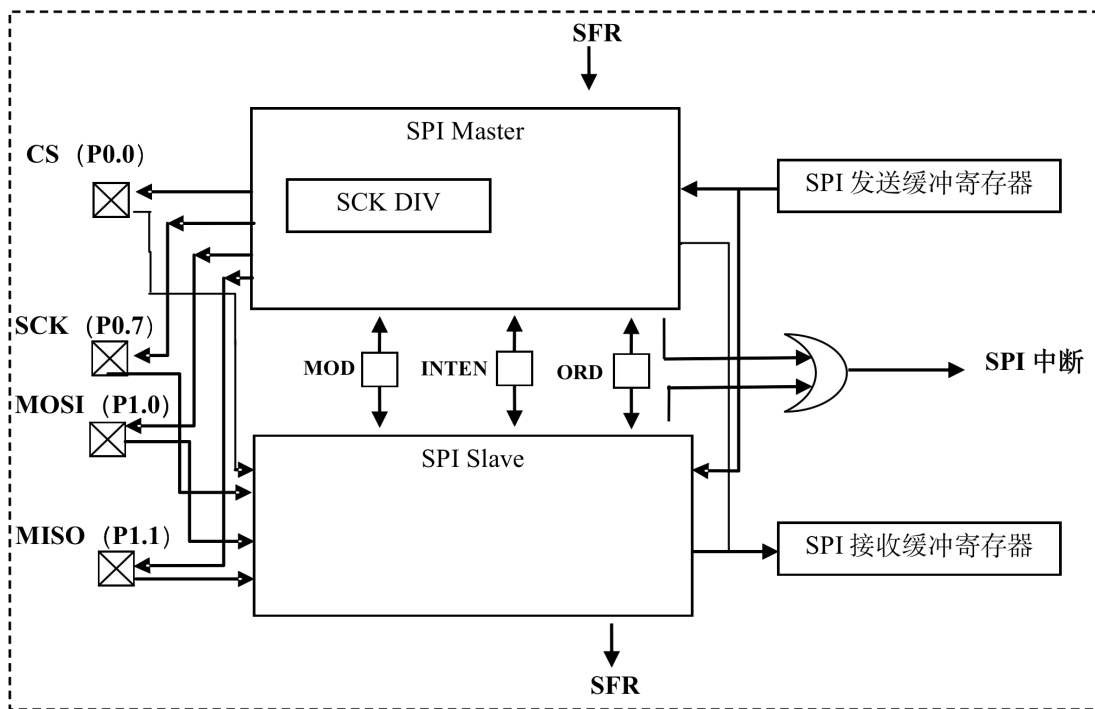
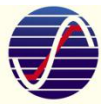


图1-1 SPI功能方框图

### 17-2. 特性

- 支持主/从模式
- 传输数据为 1 字节
- 支持 MSB 或 LSB 优先传输
- 主机模式下可输出多种串行时钟频率
- 可配置的串行时钟极性和相位
- 支持中断方式



### 17-3. 功能描述

#### 17-3-1. 主机工作模式

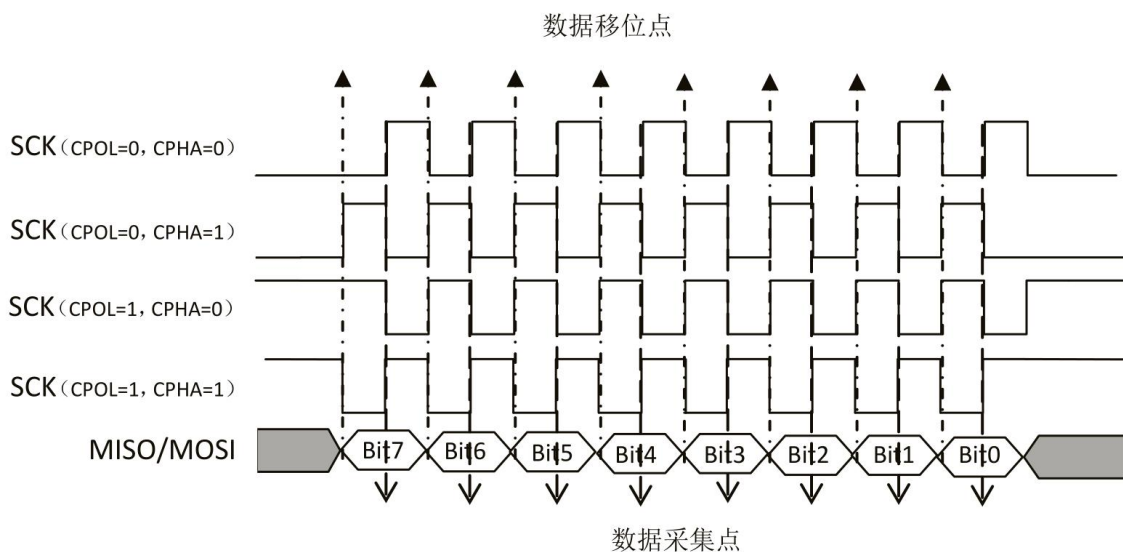
SPI 总线上的所有数据传输由 SPI 主设备启动，主机/从机控制位 SPICTRL.0 位置 0 将 SPI 配置为主机方式。当 SPI 为主机模式时，使能 SPI (将 SPICTRL.1 位置 1) 同时向 SPI 发送数据寄存器 SEND 写入一个字节时，数据传输开始。SPI 主器件立即在 MOSI 线上串行移出数据，同时在 SCK 上发送串行时钟，在传输结束后 SPISTA.0 位(MSPF)中断标志被硬件置为 1。如果 SPI 中断使能打开，将产生一个中断请求。在全双工应用中，当 SPI 主器件向从器件发送数据时，被寻址的 SPI 从器件可以同时在 MISO 线上向主器件发送数据。接收数据接收后 SPISTA.1 位(SSPF)中断标志被硬件置 1。SPISTA[1:0]位不能自动清零，需要对该状态位软件写零，将状态位清零。主控制器通过读 RCVD 寄存器得到接收到的数据。

主机模式数据时钟时序如下图，SPICTRL.6 位是 CPHA(Clock Phase)控制位，SPICTRL.5 位是 CPOL(Clock Polarity)控制位，SPICTRL.7 位为串行数据传输顺序控制位 (ORD)，ORD 置 0 时，SPI 串行数据是高位在前，低位在后的顺序传输。SPI 的主设备和从设备的 ORD，CPHA，CPOL 的设置要保持一致。

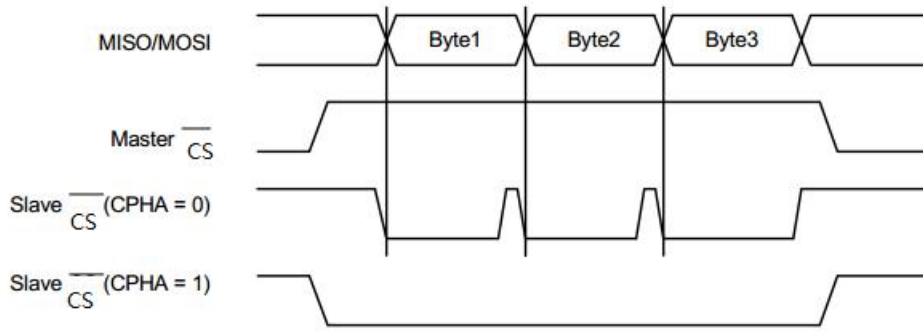
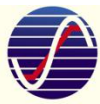
通过软件设置 SPICTRL 寄存器的 CPOL 位和 CPHA 位，用户可以选择 SPI 时钟极性和相位的四种组合方式。CPOL 位定义时钟的极性，即空闲时的电平状态，它对 SPI 传输格式影响不大。CPHA 位定义时钟的相位，即定义允许数据采样移位的时钟边沿。

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，CS 引脚的下降沿从设备开始发送数据。CS 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。

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 的第二个沿之前完成写 SEND 的操作。CS 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。



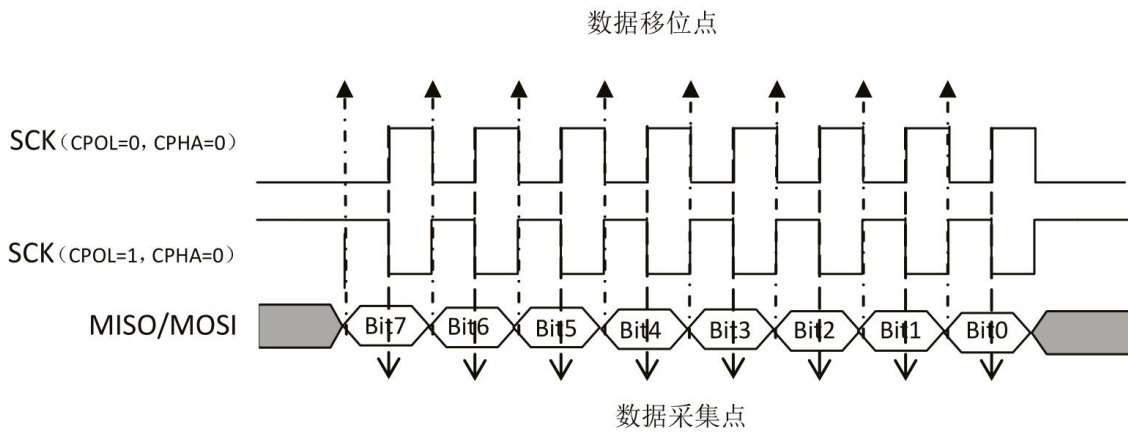
主机模式数据时钟时序



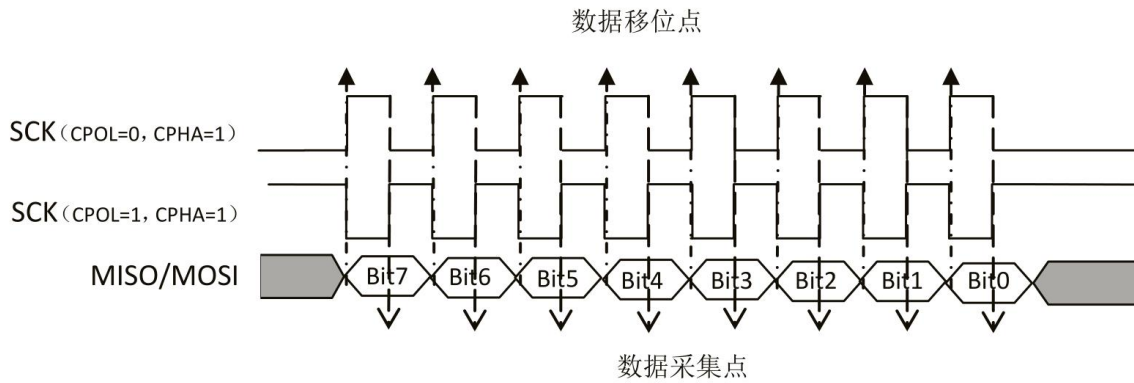
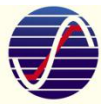
CPHA/CS 时序

### 17-3-2. 从机工作模式

当 SPI 控制位 SPICTRL.0 位置 1，将 SPI 配置为从机工作模式。当 SPI 为从机模式时，由主器件控制串行时钟 (SCK)，从 MOSI 移入数据。SPI 逻辑中的计数器对 SCK 边沿计数，当完成 8 位数据移位后，产生 SPI 中断，从机接收中断标志被置为“1”，即 SPISTA.3(SRCV)被置“1”。通过读 RCVD 得到接收到的数据。从器件发送数据功能，通过写 SEND，来预装要发送给主器件的数据，在主器件发送 SCK 时，将数据移位发送到 MISO 线上发动给主器件。数据发送结束，从机发送中断标志被置为“1”，即 SPISTA.2(SSND)被置“1”。SPISTA[3:2]不能自动置零，需要对该状态位写零，将状态位清零。从机模式数据时钟时序如下图。



从机模式数据时钟时序 (CPHA=0)



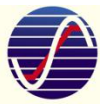
从机模式数据时钟时序 (CPHA=1)

## 17-4. SPI 寄存器

### 17-4-1. SPICTRL(B9H): SPI 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
B9H	SPICTR	ORD	CPHA	CPOL	-	SSENA	SPII	SPIEN	SM	60H
	读/写	W/R	W/R	W/R	-	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	ORD	数据传输方向选择位 0: MSB First, 先发最高位 1: LSB First, 先发最低位
6	CPHA	相位控制位 0: 主设备在 SCK 周期的第一个沿采集数据 1: 主设备在 SCK 周期的第二个沿采集数据
5	CPOL	极性控制位 0: 在 Idle 状态下 SCK 处于低电平 1: 在 Idle 状态下 SCK 处于高电平
4	-	保留
3	SSENA	SPI_CS 引脚控制位 0: 在主从模式下, 打开 SPI_CS 引脚, SPI_CS 信号由硬件拉低; 每接收/发送完成 8bit 数据, SPI_CS 信号硬件拉高, 下一次接收/发送开始时, SPI_CS 再次拉低; 1: 在主从模式下, 关闭 SPI_CS 引脚, SPI_CS 信号由用户通过软件配置 P07 口实现;
2	SPIEN	SPI 中断使能 0: 中断使能关闭



		1: 中断使能打开
1	SPIEN	SPI 使能控制位 0: 关闭 SPI 1: 打开 SPI
0	SM	SPI 主从设备选择位 0: MCU 为主设备 1: MCU 为从设备

17-4-2. SEND(BAH): SPI 发送数据寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
BAH	SEND	SEND[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

17-4-3. RCVD(BBH): SPI 接收数据寄存器

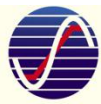
地址	名称	7	6	5	4	3	2	1	0	上电复位值
BBH	RCVD	RCVD[7:0]								00H
读/写		R	R	R	R	R	R	R	R	

17-4-4. SPISTA(BCH): SPI 状态寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
BCH	SPISTA	-	PGA_G AINSEL [2]	PGA_G AINSEL [1]	PGA_G AINSEL [0]	PGA_PD .	PGA_OE N	SSPIF	MSPIF	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	-	保留位
6~4	PGA_GAINS EL[2:0]	PGA 放大增益选择 000: 1 001: 10





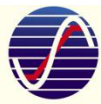
		010: 20 011: 30 100: 40 101: 50 110: 60 111: 70
3	PGA_PD	PGA 工作模式选择 0: 正常工作模式 1: Power Down 模式
2	PGA_OEN	PGA 输出使能 0: PGAOUT_T 关闭 1: PGAOUT_T 打开, 通过 P20 引脚输出
1	SSPIF	作为 SPI 从设备的数据传送标志位 0: 由软件清零 1: 表明已经完成数据传送, 由硬件置 1, 若使能 SPI 中断, 则申请中断;
0	MSPIF	作为 SPI 主设备的数据传送标志位 0: 由软件清零 1: 表明已经完成数据传送, 由硬件置 1, 若使能 SPI 中断, 则申请中断;

17-4-5. SPICKDIV(BDH): SPI 主模式时钟分频寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
BDH	SPICKDIV	SPICKDIV[7:0]								05H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

注: SPI 主模式下, 输出 SPI 时钟:

- SPICKDIV 不能设置为 0
- $F_{sck} = F_{sys} / 2 * SPICKDIV$



## 18. 16 位 PWMC

ASM87L164 内建 PWM 控制器 (PWMC)，内部带有一个 16 位独立计数器，可以支持 5 路独立的 PWM 输出，完成 5 路 PWM 控制功能。该 PWMC 主要功能如下：

- (1) 提供 16 位 PWM 精度；
- (2) PWM0~PWM4 周期相同，占空比可单独设置（使能互补功能时，PWM5 可输出波形）；
- (3) PWM 输出极性可由寄存器设置；
- (4) 提供 1 个 PWM 溢出中断；
- (5) PWM 带有 4bit 死区单元 (DB, Deadband)，死区单元使能后，在电机控制中可以防止电机短路导通。

该 PWMC 可使用周期寄存器 PWMPR 来控制 PWM 输出波形的周期，占空比的调整则由比较寄存器 PWMCRx 来实现，PWMMRx 管理寄存器控制 PWMC 工作模式。

### 18-1. PWM 结构框图

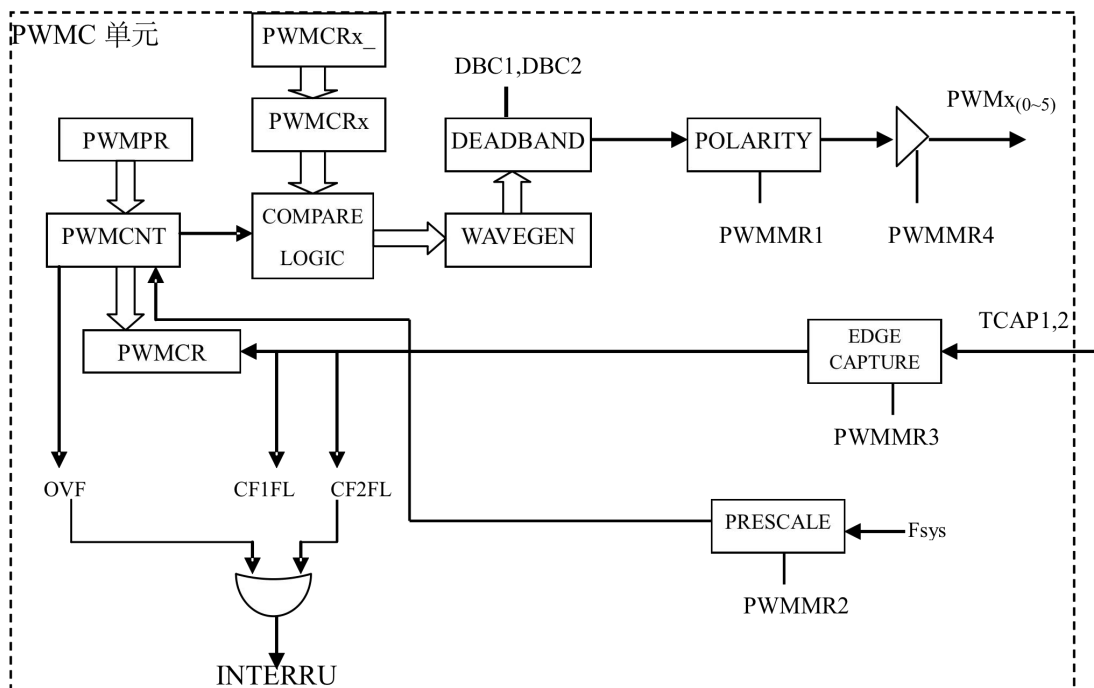
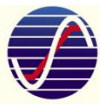


图 18-1 PWMC 结构框图

注：PWMCRx\_ 是 PWMCRx 的阴影寄存器，对应同一个地址。

ASM87L164 的 PWMC 模块功能框图如图 13-1 所示，其各组成部分及其功能简述如下：

- (1) 预分频器 PRESCALE 为 8 位计数器，可配置成分频系数  $F_{sys}$ 、 $F_{sys}/2$ 、 $F_{sys}/4$ 、... $F_{sys}/256$  (PWMMR2)；
- (2) 计数器 PWMCNT 是 16 位计数器，递增计数；



- (3) 计数器 PWMCNT 周期性的从 PWMPR 为初值开始计数(reload 使能), 到 0xFFFF 溢出, 产生溢出中断;
- (4) 计数器 PWMCNT 和比较器 PWMCR<sub>x</sub>(x=0,1,2,3,4)比较, 产生 PWM<sub>x</sub>(x=0~4)波形, 占空比为 0~65535/65536;
- (5) PWM<sub>x</sub> 输出到 IO 端口, 极性控制由 PWMMR4 控制;
- (6) 计数器可以通过 PWMPR 重载, 即 PWMPR 重载到 PWMCNT;
- (7) 比较器 PWMCR<sub>x</sub> (x=0~4) 可重载;
- (8) 控制寄存器为 PWMMR1~4(管理控制寄存器);
- (9) PWM 输出可以直接代替蜂鸣器(Buzzer)使用;
- (10) 具有捕获(Capture)功能, 用以捕获 P1.3/P1.4(TCAP1/2)端口边沿跳变。该功能可用于电机控制, 检查电机转速。

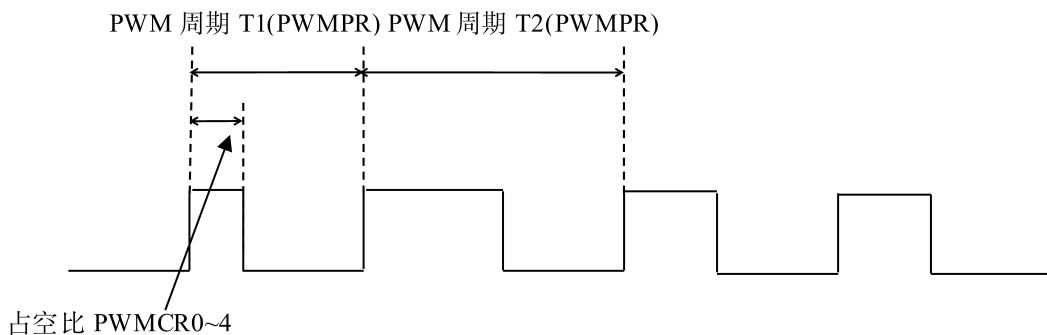


图 18-2 PWM 周期设置示意图

如上图所示, 第一个 PWM 周期 T1 由 PWMCNT 初值决定, 各通道占空比由 PWMCR0~4 设定; 此后 PWM 周期 T2, 通过重载寄存器计数溢出后重载 PWMPR 决定, 各通道占空比由 PWMCR0~4 设定。

## 18-2. PWM 模块控制寄存器

### 18-2-1. PWMCNTL (DBH) : PWM 低 8 位计数器 (LSB)

PWM 计数器: 该寄存器用于初始化 PWM 计数器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
DBH	PWMCNTL	PWMCNT[7:0]								00H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

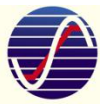
### 18-2-2. PWMCNTH (E2H) : PWM 高 8 位计数器 (MSB)

PWM 计数器: 该寄存器用于初始化 PWM 计数器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
E2H	PWMCNTH	PWMCNT[15:8]								00H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

### 18-2-3. PWMCR0~4 (D1H~D5H) : PWM 计数比较器

PWM 比较寄存器: 用于调整 PWM 占空比。该组每个寄存器控制内部 16 位的 PWMCR<sub>x</sub>\_BUF 寄存器, 对 PWMCR<sub>x</sub> 第一次写操作写入 PWMCR<sub>x</sub>\_BUF 高 8 位, 第二次写操作写入 PWMCR<sub>x</sub>\_BUF 低 8 位。



当 PWM CNT 计数大于 PWM CR<sub>x</sub> 时，PWM 输出低电平，

当 PWM CNT 计数小于 PWM CR<sub>x</sub> 时，PWM 输出高电平。

地址	名称	7	6	5	4	3	2	1	0	上电复位值
D1H	PWMCR0	PWMCR0[7:0]								00H
D2H	PWMCR1	PWMCR1[7:0]								00H
D3H	PWMCR2	PWMCR2[7:0]								00H
D4H	PWMCR3	PWMCR3[7:0]								00H
D5H	PWMCR4	PWMCR4[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

#### 18-2-4. PWMPR (D6H) : PWM 周期寄存器

PWM 周期寄存器: 用于设置 PWM 周期。该寄存器控制内部 16 位的 PWMPR\_BUF 寄存器, 对 PWMPR 第一次写操作写入 PWMPR\_BUF 高 8 位, 第二次写操作写入 PWMPR\_BUF 低 8 位。

当重载使能时, 计数器计数到 FFFFH 时, 就会重载该周期寄存器的值到计数器 PWM CNT, 用于调整 PWM 周期。

地址	名称	7	6	5	4	3	2	1	0	上电复位值
D6H	PWMPR	PWMPR[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

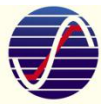
位编号	位符号	说明
7~0	PWMPR	周期寄存器: 当重载使能时, 计数器计数到 FFFFH 时, 就会重载该周期寄存器的值到计数器 PWM CNT, 用于调整 PWM 周期

#### 18-2-5. PWMMR1 (D7H) : PWM 管理控制寄存器 1

地址	名称	7	6	5	4	3	2	1	0	上电复位值
D7H	PWMMR1	POLAR5	INTF	OVF	POLAR4	POLAR3	POLAR2	POLAR1	POLAR0	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	POLAR5	PWM5 输出极性控制 1: 正向, 0: 反向
6	INTF	PWM 计数器溢出中断使能 1: 使能溢出中断 0: 禁止溢出中断
5	OVF	PWM 计数器溢出标志 1: 产生了计数溢出 0: 没有产生计数溢出
4	POLAR4	PWM4 输出极性控制 1: 正向, 0: 反向
3	POLAR3	PWM3 输出极性控制 1: 正向, 0: 反向
2	POLAR2	PWM2 输出极性控制 1: 正向, 0: 反向
1	POLAR1	PWM1 输出极性控制 1: 正向, 0: 反向
0	POLAR0	PWM0 输出极性控制 1: 正向, 0: 反向

注: 正向: PWM CNT 计数大于 PWM CR<sub>x</sub> 时, PWM 输出低电平; 反之, 输出高电平;



反向: PWMCNT 计数大于 PWMCRx 时, PWM 输出高电平; 反之, 输出低电平。

18-2-6. PWMMR2 (D9H) : PWM 管理控制寄存器 2

地址	名称	7	6	5	4	3	2	1	0	上电复位值
D9H	PWMMR2	PWME	-	PWMRL	-	PWMSCALE				00H
读/写		W/R	-	W/R		W/R				

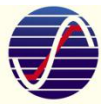
位编号	位符号	说明																				
7	PWME	PWM 使能控制: 0: PWM 关闭 ; 1: PWM 使能																				
6	PWMCPL	PWM 周期寄存器、比较寄存器重载使能 1: 使能重载, 当计数器溢出时, 周期寄存器与比较寄存器重载配置值 0: 不使能重载																				
5	PWMRL	PWMCNT 计数器溢出后重载使能, PWMCNT 重载 PWMPR 周期寄存器的值 1: 使能重载 0: 不使能重载																				
3~0	PWMSCALE	分频系数 <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Divider</th> <th>PWMSCALE</th> </tr> </thead> <tbody> <tr><td>2</td><td>0000</td></tr> <tr><td>4</td><td>0001</td></tr> <tr><td>8</td><td>0010</td></tr> <tr><td>16</td><td>0011</td></tr> <tr><td>32</td><td>0100</td></tr> <tr><td>64</td><td>0101</td></tr> <tr><td>128</td><td>0110</td></tr> <tr><td>256</td><td>0111</td></tr> <tr><td>1</td><td>1000</td></tr> </tbody> </table>	Divider	PWMSCALE	2	0000	4	0001	8	0010	16	0011	32	0100	64	0101	128	0110	256	0111	1	1000
Divider	PWMSCALE																					
2	0000																					
4	0001																					
8	0010																					
16	0011																					
32	0100																					
64	0101																					
128	0110																					
256	0111																					
1	1000																					

18-2-7. PWMMR3 (DAH) : PWM 管理控制寄存器 3

该寄存器主要用于根据 IO 口变化, 捕获 PWMCNT 计数器的值

地址	名称	7	6	5	4	3	2	1	0	上电复位值
DAH	PWMMR3	CF1_FL	CF1_EN	CF0_FL	CF0_EN	-	CAPE	EG1	EG0	00H
读/写		W/R	W/R	W/R	W/R		W/R	W/R	W/R	

位编号	位符号	说明
7	CF1_FL	边沿捕获 1 边沿跳变状态标志位
6	CF1_EN	边沿捕获 1 边沿跳变中断使能
5	CF0_FL	边沿捕获 0 边沿跳变状态标志位
4	CF0_EN	边沿捕获 0 边沿跳变中断使能



3	-	保留
2	CAPE	捕获工作使能, 此时{PWMCr1, PWMCr0}用于捕获 CPIN1 与 CPIN0 上边沿跳变时刻计数器数据, 此时 PWM 为 16 位计数器 (8 位计数器+8 位分频器)
1	EG1	边沿捕获 1 的边沿选择, 1: 上升沿 0: 下降沿
0	EG0	边沿捕获 0 的边沿选择, 1: 上升沿 0: 下降沿

注:

- 1) 捕获输入 CPIN0 来自 P1.3, CPIN1 来自 P1.4;
- 2) 捕获单元主要用于电机控制, 通过电机反馈来确定转速。

当 PWM 使用捕获功能时, 捕获单元捕获的是 PWMCNTL+PWMCNTH (16bit) 和预分频器 PreScale (8bit) (预分频值只能通过 TCA0 来捕获) 的值。

对应 TCA0, 分别保存到 PWMCr0 (PreScale 分频器 8bit), PWMCr1 (PWNCNTL), PWMCr2 (PWNCNTH); 对应 TCA1, 分别保存到 PWMCr3 (PWNCNTL), PWMCr4 (PWNCNTH), 比较寄存器用于保存计数器的值。

#### 18-2-8. PWMMR4 (E3H) : PWM 管理控制寄存器 4

该寄存器主要用于 PWM 使能输出

地址	名称	7	6	5	4	3	2	1	0	上电复位值
E3H	PWMMR4	-	-	PWMOE5~0						00H
读/写		-	-	W/R						

位编号	位符号	说明
7~6	-	保留
5~0	PWMOEx	PWMx 输出使能控制位, 1: 使能输出, 0: 不输出 PWM。

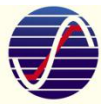
#### 18-2-9. DBC1(BEH) PWM 死区控制寄存器 1

地址	名称	7	6	5	4	3	2	1	0	上电复位值
BEH	DBC1	-	-	-	-	-	PWM0,1	PWM2,3	PWM4,5	00H
写		-	-	-	-	-	W/R	W/R	W/R	

位编号	位符号	说明
7~3	-	保留
2	PWM0,1	死区控制使能
1	PWM2,3	死区控制使能
0	PWM4,5	死区控制使能

注:

对 DBC1 控制, 建议采用 OR 指令来控制 bit2~bit0。



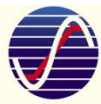
18-2-10. DBC2(BFH) PWM 死区控制寄存器 2

地址	名称	7	6	5	4	3	2	1	0	上电复位值
BFH	DBC2	RESERVE	PERIOD				DBSEL2	DBSEL1	DBSEL0	00H
	写	-	W/R				W/R	W/R	W/R	

位编号	位符号	说明
7	RESERVE	保留
6~3	PERIOD	死区周期
2~0	DBSELx	死区时钟分频选择 DBSEL2~0; Fclk <sub>in</sub> =F <sub>sys</sub> /DBSEL

注:

- 1) 死区单元内部有 4bit 计数器，以死区时钟分频选择的分频时钟，从周期 period 计数到 0，所以周期越大，意味着死区越大。
- 2) BSEL2~0 合并 3bit 控制分频器，分频器可以支持 000 : 2 分频；001: 四分频；010:8 分频 011:16 分频；100:32 分频；分频前时钟是系统时钟。
- 3) DBC2 寄存器仅支持 MOV 指令。



## 19. 8+2 通道 12 位 ADC

### 19-1. 概述

ASM87L164 包含了 8 个外部通道 (AIN0~AIN7) 和 3 个内部通道 (VBG、内部 VDD/4、PGA 输出) 的单端 12 位逐次逼近型模数转换器 (ADC)，可以将模拟信号转换成 12 位数字信号。8 个 ADC 外部通道都可以独立输入模拟信号，但是每次转换只能使用一个通道。用户进行 ADC 转换时，首先要选择输入通道 (AIN0~AIN7)，然后把 ADC\_EN 和 ADC\_START 位置“1”，启动 ADC 转换。转换结束后，系统自动将 ADC 转换结束标志位 EOCO 设置为“1”，并将转换结果存入寄存器 ADCDRH 和寄存器 ADCDRL 中。ADC 具有阈值比较功能，当 ADC 使能阈值比较功能后，只有 ADC 转换值高 8 位大于阈值比较寄存器 ADCCMP 时才产生转换完成标志和 ADC 中断。为了提高软件效率，ADC 内置硬件取平均模块，用户可根据需要对多次转换值进行取平均。

**当 ASM87L164 中选用内置 16MHz RC 振荡器，满足下述条件 ADC 转换可达 1MSPS:**

- 1).控制寄存器 ADCCSR[6:5]设置为 00，即 ADC 此时输入时钟频率是 16MHz;
- 2).在 ADC 为 12bit 模式时，控制寄存器 SH\_CTR<1:0>设置为 00，即 ADC 采样时间为 4 个时钟周期;

**ADC 的参考电压 (VREF) 有 5 种选择:**

- 1) 内部 2V、3V、4V 三种电压可任选其一，该内部电压在-40~85℃温度范围内误差精度可达  $\pm 1\%$ ;
- 2) VDD 管脚 (即电源 VDD，也是默认参考电压源)；
- 3) 选择外部管脚 P1.6(Vref)来选择参考电压。

ASM87L164 的 ADC 模块功能框图如下图所示。

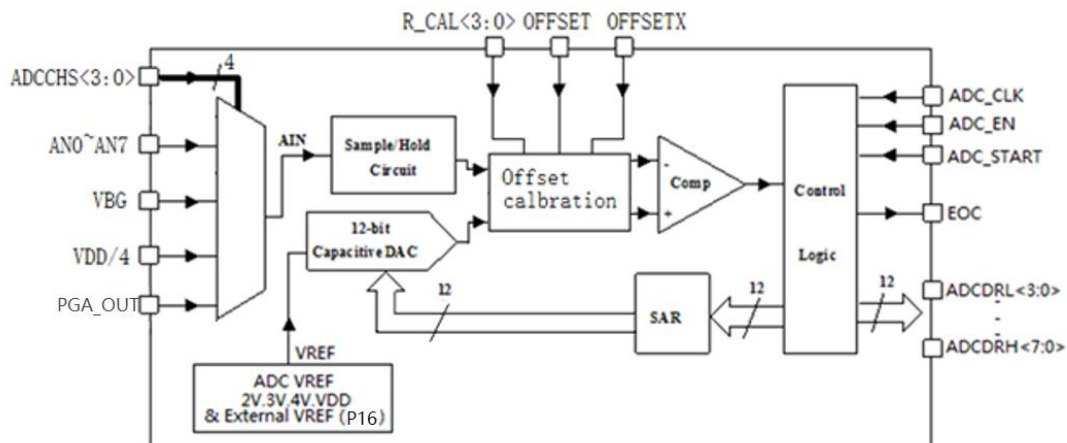
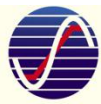


图 19-1 ASM87L164ADC 模块功能方框图

注: EOC 为 ADC 转换结束标志脉冲，经过数字逻辑产生 ADC 中断信号 EOCO。





## 19-2. 控制寄存器

### 19-2-1. ADCCSR (DCH) : ADC 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
DCH	ADCCSR	ADC_EN	ADCCLKS	EOCO	ADC_START	ADCPSEL[2:0]			60H	
读/写		W/R	W/R	W/R	W/R	W/R				

位编号	位符号	说明
7	ADC_EN	ADC 使能位, 1: ADC 使能; 0: ADC 关闭
6~5	ADCCLKS	ADC 时钟选择* 00: Fsys/1; 01:Fsys/2 ; 10: Fsys/4; 11:Fsys/8
4	EOCO	ADC 转换结束标志位, 1: 转换结束, 需要用户清除; 0: 转换没有结束
3	ADC_START	ADC 转换启动位, 1: ADC 转换开始; 0: ADC 停止转换
2~0	ADCPSEL[2:0]	寄存器 ADCREF(A9H)中的 bit6 和 ADCPSEL[2:0]一起组成 ADCPSEL[3:0], 控制 ADC 输入通道选择: 0000: P1.6 为 ADC (AIN0) 0001 : P1.7 为 ADC (AIN1) 0010 : P2.0 为 ADC (AIN2) 0011 : P2.1 为 ADC (AIN3) 0100 : P2.4 为 ADC (AIN4) 0101 : P2.6 为 ADC (AIN5) 0110: P3.0 为 ADC (AIN6) 0111 : P3.1 为 ADC (AIN7) 1000 : PGA_OUT(PGA 输出) 1001: VBG(内部产生) 1010: VDD/4(内部产生)

注: Fsys 为 CPU 系统时钟, 可以是内部 RC 时钟, 也可以是外部时钟。

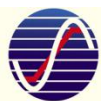
### 19-2-2. ADCDRH (DEH) : ADC 转换结果高 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位
DEH	ADCDRH	ADCDRH[7:0]								00H
读/写		R								

位编号	位符号	说明
7~0	ADCDRH	ADCDRH 为 ADC 转换结果之 12 位数据高 8 位

### 19-2-3. ADCDRL (DDH) : ADC 转换结果低 4 位

地址	名称	7	6	5	4	3	2	1	0	上电复位值
DDH	ADCDRL*	LVRSEL				ADCDRL[3:0]				00H
读/写		W				R				



位编号	位符号	说明			
7~4	LVRSEL	LVR 复位电压点选择:			
		LVRS	LVR Point	LVRS	LVR Point
		0000	1.6V	1000	3.0V
		0001	1.7V	1001	3.2V
		0010	1.8V	1010	3.4V
		0011	2.0V	1011	3.6V
		0100	2.2V	1100	3.8V
		0101	2.4V	1101	4.0V
		0110	2.6V	1110	4.2V
0111	2.8V	1111	4.5V		
3~0	ADCDDL	ADCDDL 为 ADC 转换结果之 12 位数据低 4 位			

注: ADCDDL 是多用途寄存器, 除用于保存 ADC 转换结果的低 4 位外, 还用于设置 LVR 复位电压点。  
ADCDDL 在读操作时, 3~0 位为 ADC 转换结果之 12 位数据的低 4 位;  
ADCDDL 在写操作时, 7~4 位则用来设置 LVR 复位电压点。

#### 19-2-4. ADCIN\_EN (DFH) : ADC 通道属性寄存器

ADCIN\_EN 为 ADC 模拟输入通道属性寄存器, 1: 设置该通道为模拟 IO; 0: 设置该通道为数字 IO。

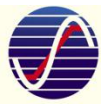
ADDR	NAME	7	6	5	4	3	2	1	0	上电复位值
DFH	ADCIN_EN	VREF_EN	ADSEL[6:0]							00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	VREF_EN*	ADC 的内部参考源(VREF)使能位, 1: VREF 使能工作, 可以使用 VDD、外部 P16 端口和内部精确电压源(2V、3V、4V)作为参考电压; 0: VREF 关闭
6	ADSEL[6]	端口 P3.0 数模控制位: 0: P3.0 为数字端口; 1: P3.0 为模拟端口
5	ADSEL[5]	端口 P2.6 数模控制位: 0: P2.6 为数字端口; 1: P2.6 为模拟端口
4	ADSEL[4]	端口 P2.4 数模控制位: 0: P2.4 为数字端口; 1: P2.4 为模拟端口
3	ADSEL[3]	端口 P2.1 数模控制位: 0: P2.1 为数字端口; 1: P2.1 为模拟端口
2	ADSEL[2]	端口 P2.0 数模控制位: 0: P2.0 为数字端口; 1: P2.0 为模拟端口
1	ADSEL[1]	端口 P1.7 数模控制位: 0: P1.7 为数字端口; 1: P1.7 为模拟端口
0	ADSEL[0]	端口 P1.6 数模控制位: 0: P1.6 为数字端口; 1: P1.6 为模拟端口

注: ASM87L164 进入深度睡眠状态时, 一定要把该控制位 VREF\_EN 设置为 0, 关掉 VREF 模块。

#### 19-2-5. ADCREF (A9H) : ADC 参考源选择寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
A9H	ADCREF*	LVREN	ADCPSEL[3]	ADSEL[7]	REFSEL		OFFSETX	OFFSET		0CH



读写	W/R	W/R	W/R	W/R	W/R	W/R	
----	-----	-----	-----	-----	-----	-----	--

位编号	位符号	说明
7	LVREN	LVR 复位电路使能, 0: 使能(默认) 1: 关闭 LVR
6	ADCPSEL[3]	和寄存器 ADCCSR(DCH)中的 ADCPSEL[2:0]一起组成 ADCPSEL[3:0], 控制 ADC 通道选择, 具体参见寄存器 ADCCSR(DCH)中 ADCPSEL[2:0]的描述。
5	ADSEL[7]	P31 模拟数字通道选择: 0: P31 为数字端口; 1: P31 为模拟端口
4~2	REFSEL	ADC 参考电压源选择 000: 内部 2V; 001: 内部 3V; 010: 内部 4V; 011: 电源 VDD; 100: 外部 VREF(P1.6); 其他: VDD
1	OFFSETX	比较器输入正负端交换控制位, 默认为 0 <b>此版本中该功能取消, 请设为 0</b>
0	OFFSET	ADC offset 控制: OFFSET 为 1:+OFFSET; 0: -OFFSET

注: ADCREF 是多用途控制器, 除了用作设置 ADC 参考源, 还用来设置 LVR 模块开关状态。LVR 复位电压点设置则在 ADCDRL 寄存器中完成。

#### 19-2-6. ADCCMP (E1H) : ADC 阈值比较寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
E1H	ADCCMP	ADCCMP [7:0]								00H
读/写		W/R								

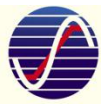
位编号	位符号	说明
7~0	ADCCMP	用于设置 ADC 阈值比较值

注: 使用ADC阈值比较功能时, 必须置位AUXCON寄存器之bit4, 转换

#### 19-2-7. ADCAVE (AAH) : ADC 取平均寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
AAH	ADCAVE	VREF_SEL	INT1EN	INT0EN	-	AVE_EN	AVES[2]	AVES[1]	AVES[0]	60H
读写		W/R	W/R	W/R	-	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	VREF_SEL	ADC 参考电压 Trim 选择(用户模式下配置为 0)
6	INT1EN	外部中断 INT1 输入使能



		1: 输入使能打开 0: 输入使能关闭
5	INTOEN	外部中断 INTO 输入使能 1: 输入使能打开 0: 输入使能关闭
4	-	保留
3	AVE_EN	ADC 硬件取平均使能位 1: 使能硬件取平均 0: 关闭硬件取平均
2	AVES[2: 0]	AVES[2: 1]为取平均选择位 000: 不取平均 001: 累计 4 个转换值取一个平均; 010: 累计 8 个转换值取一个平均; 011: 累计 16 个转换值取一个平均; 100: 累计 32 个转换值取一个平均; 其他: 不取平均
1		
0		

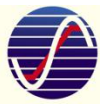
注：用户若要使用ADC硬件取平均，则阈值比较功能自动失效，同时ADC转换模式需要配置成连续转换模式

#### 19-2-8. ADCFG0 (E4H) : ADC 配置寄存器 0

地址	名称	7	6	5	4	3	2	1	0	上电复位值
E4H	ADCFG0	MODE	SH_CTR	OTR	R_CAL				00H	
读/写		W/R	W/R	W/R	W/R					

位编号	位符号	说明	
7	MODE	选择 12bit 或 10bit ADC; 1, 12bit ADC; 0, 10bit ADC, 软件读取 ADC 转换结果寄存器<11:0>的高 10 位	
6~5	SH_CTR	采样保持时间表:	
		SH_CTR	ADC S/H Time (T <sub>SH</sub> )
		00	4*ADC clock
		01 or 10	6*ADC clock
	11	8*ADC clock	
4	OTR	ADC 单次转换模式控制 1: 单次转换 0: 连续转换(使能硬件取平均值以及使能阈值比较功能时, 需设置连续转换)	
3~0	R_CAL	ADC offset 选择, 和寄存器 ADCREF 中的 OFFSET 配合使用: <b>该版本此功能取消, 请选择默认值 0x0;</b>	

注：如果不使用 OFFSET 和 R\_CAL 寄存器时, 请将 R\_CAL 置 0, OFFSET 置 0 或 1。



### 19-3. ADC 转换步骤

ASM87L164 的 ADC 可以选择电源电压 VDD，内部精准电压源 2V、3V、4V 以及外部参考电压输入 VREF 作为 ADC 参考电压。

注意事项：

1. 当电源 VDD<4V 时，选择内部 4V 作为参考电压，实际参考电压为 VDD。对于选择内部 2V、3V 为参考电压，情况亦类似；
2. 当选择内部参考电压时，电源电压要求至少为：内部参考电压+0.5V；
3. 外部参考电压源 VREF 是和模拟输入通道 0 (AIN0) 共用，故此选择使用外部参考源 VREF 时，模拟输入通道 0 (AIN0) 不能同时使用，此时 AIN0 需要配置成模拟输入通道，反之亦然。

用户使用 ASM87L164 的 ADC 进行模数转换的操作步骤如下，ADC 转换时序如图 14-2 所示。

- 1) 设定 ADC 中断使能，即中断寄存器 IE[6]设置为 1；
- 2) 使能 ADC (ADCCSR[7]=1)，设定 ADC 输入管脚和通道 (ADSEL[7:0]和 ADCCSR[3:0]控制)；
- 3) 设定 ADC 参考电压 VREFH (ADCREFL[4:2])；
- 4) 设定 ADC 转换所用的频率(ADCCSR[6:5])；
- 5) 设定 ADC 参考电压使能 (ADCIN\_EN[7]=1)与 ADC 模块参考电压(ADCCSR[3])，等待 100uS 以上时间再进行下一步动作；
- 6) 设定 ADC\_START (由低电平变为高电平)，即 ADCCS[3]，转换开始；
- 7) 等到 EOCO=1，先高位后低位分别从 ADCDRH、ADCDRL 获得 12 位数据，本次转换完成；
- 8) 用户软件清除 EOCO 标志，等待下一次转换结果；
- 9) 如不更改输入通道以及参考电压，则重复 6~8 的步骤，即实现 ADC 连续转换功能。

如果用户在上述步骤 7)ADC 转换结束时清除 ADC\_START 与 EOCO 信号，则实现单次转换。

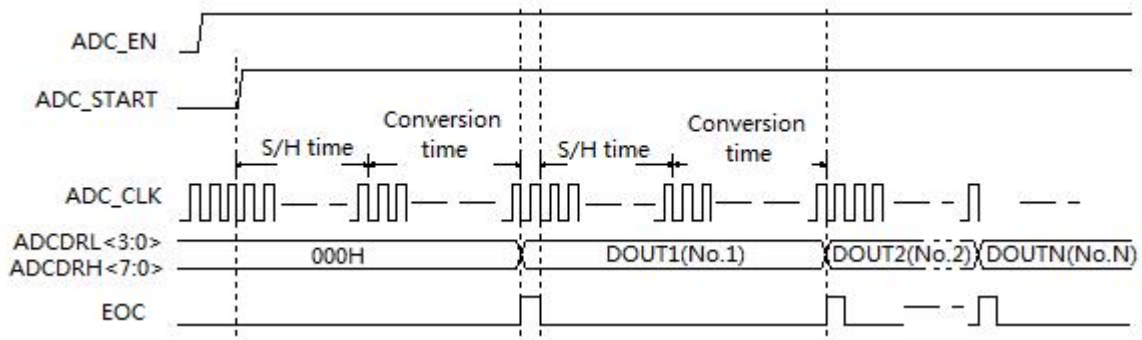
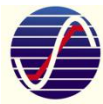


图 19-2ASM87L164ADC 转换时序图

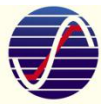
就以上 ADC 转换时序，进一步说明如下：

- ✓ ADC\_CLK: ADC 的工作时钟，软件可配置，详见寄存器 ADCCSR 第 6 位，第 5 位；
- ✓ S/H time: ADC 采样保持时间，为 4、6 或 8 个 ADC\_CLK；
- ✓ Conversion time: ADC 转换时间，12 个 ADC\_CLK；
- ✓ EOC: ADC 转换结束脉冲，宽度为 1 个 ADC\_CLK，后由数字逻辑产生 ADC 中断信号 EOCO；
- ✓ 完成一次转换所需 ADC\_CLK 的时钟数为 S/H time + Conversion time。



注:

在设定 ADC 对应中断寄存器 IE 的 Bit6 之前, 用户最好用软件先清除 EOCO, 并且在 ADC 中断服务程序执行完时, 也同时清除 EOCO, 以避免不断地产生 ADC 中断。



## 20. 低电压检测(LVD)

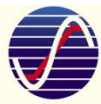
ASM87L164 内嵌低电压检测模块，通过设置 LVDEN 可以开启或屏蔽该功能，LVDS 选择检测电压值，可设置 1.8V-4.5V 等不同电压，当 VDD 电压低于设置电压时 LVD\_F 置 1 并产生中断。

LVDCTRL (CFH) : LVD 控制寄存器

地址	名称	7	6	5	4	3	2	1	0	上电复位值
CFH	LVDCTRL	LVD_F	LVD_INTEN	LVD_TRIMEN	-	LVDS			LVDEN	00H
读/写		W/R	W/R	W/R	-	W/R			W/R	

位编号	位符号	说明
7	LVD_F	LVD 检测标志位: 1 表示 LVD 检测到电源电压低于设定电压, 必须软件清 0;
6	LVD_INTEN	LVD 中断使能位: 1 表示使能 LVD 中断, 0 表示关闭 LVD 中断;
5	LVD_TRIMEN	LVD 校准使能位: 1 表示使能 LVD 校准, <b>该位用户使用时必须写 0;</b>
4	-	保留;
3~1	LVDS[2:0]	LVD 检测电压选择位, 当检测到电源电压低于该位设置值时产生标志和中断 000: 1.8V; 001: 2.0V; 010: 2.4V; 011: 2.6V; 100: 3.0V; 101: 3.6V; 110: 4.0V; 111: 4.5V;
0	LVD_EN	LVD 使能位: 1 表示使能 LVD, 0 表示关闭 LVD;

注: LVD\_TRIMEN为出厂前的校准使用控制位, 用户使用时必须关闭校准, 否则可能会出现误校准。



## 21. 可编程运算放大器(PGA)

ASM87L164 内部集成 1 路可编程运算放大器，增益可调范围 X1~X70，可有效放大交流信号与直流信号，PGA 输出与 ADC 通道 8、P20 引脚相连。

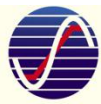
### 21-1. PGA 控制寄存器

#### 21-1-1. SPISTA(BCH): 与 SPI 状态寄存器复用

地址	名称	7	6	5	4	3	2	1	0	上电复位值
BCH	SPISTA	-	PGA_G AINSEL [2]	PGA_G AINSEL [1]	PGA_G AINSEL [0]	PGA_PD .	PGA_OE N	SSPIF	MSPIF	00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	-	保留位
6~4	PGA_GAINS EL[2:0]	PGA 放大增益选择 000: 1 001: 10 010: 20 011: 30 100: 40 101: 50 110: 60 111: 70
3	PGA_PD	PGA 工作模式选择 0: 正常工作模式 1: Power Down 模式
2	PGA_OEN	PGA 输出使能 0: PGAOUT_T 关闭 1: PGAOUT_T 打开，通过 P20 引脚输出
1	SSPIF	作为 SPI 从设备的数据传送标志位 0: 由软件清零 1: 表明已经完成数据传送，由硬件置 1，若使能 SPI 中断，则申请中断；
0	MSPIF	作为 SPI 主设备的数据传送标志位 0: 由软件清零 1: 表明已经完成数据传送，由硬件置 1，若使能 SPI 中断，则申请中断；





OPA offset 测试方法如下:

### 测试方法一

#### 1) OPA offset testbench

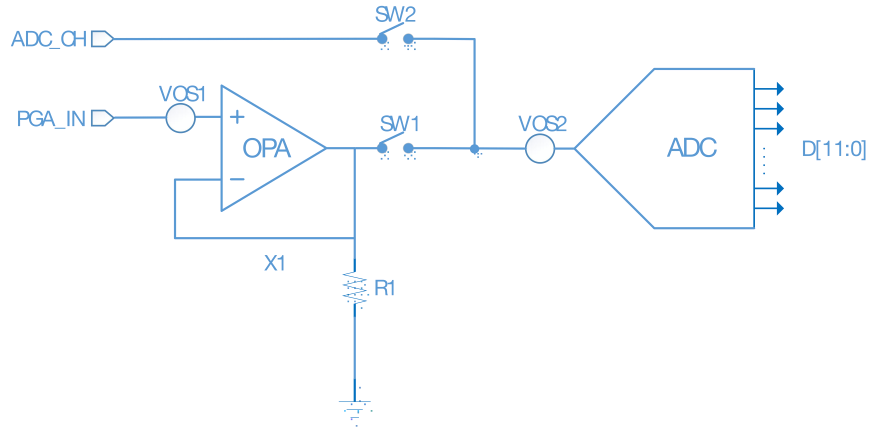


图 2 OPA offset testbench

#### 2) OPA offset test flow

Step 1: SW1 off, SW2 on

ADC\_CH=1V, ADC work, read ADC output D[11:0]. For example, Data1=D[11:0] = 12'b1000,0100,0000;

Step 2: SW1 on, SW2 off

a) SW2 off, SW1 on;

b) PGA set to X1 mode(unit gain buffer);

c) PGA\_IN=1V, ADC work, read ADC output D[11:0]. For example, Data2=D[11:0] = 12'b1000,0100,0100;

Step 3: Write back to PGA offset register

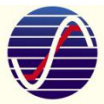
#### OPA offset register

7	6	5	4	3	2	1	0
Pol	OPA_offset						

Pol : OPA offset polarity. 1: positive; 0: negative.

OPA\_offset: OPA offset value.

OPA\_Offset = Data2 – Data1 (e.g. 8'b0000, 0100)



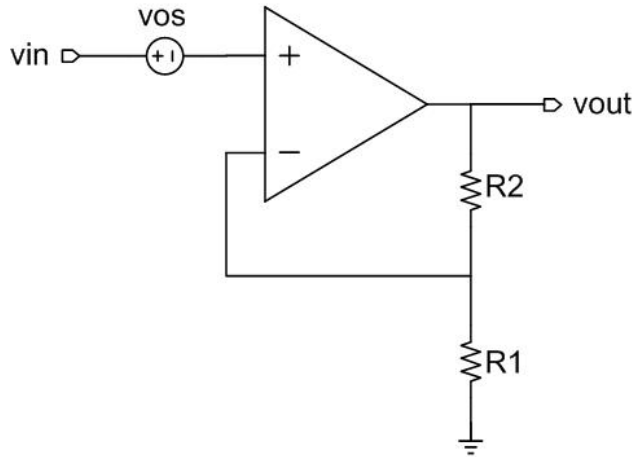
If Data2 > Data1

Pol=1

If Data2 < Data1

Pol=0

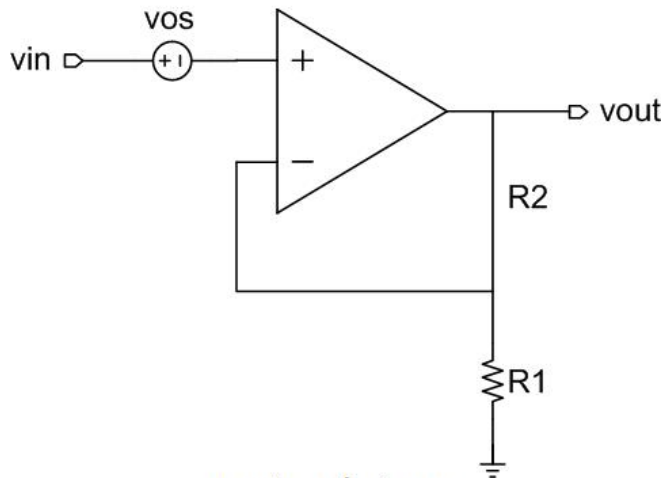
## 测试方法二



$$v_{out} = \frac{R1 + R2}{R1} (v_{in} + v_{os}) = Gain * (v_{in} + v_{os})$$

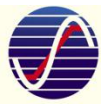
测试条件:

- 1) VDD=5V;
- 2) 输入加 DC 电压 20mv;
- 3) PGA 设置为 gain = 1;



$$v_{out} = v_{in} + v_{os}$$

$$v_{os} = v_{out} - v_{in}$$



## 22. IAP 控制单元 (Data EEPROM)

ASM87L164 在 16KB 程序存储器空间之外, 另有 1K 字节 Flash 存储器(NVR1~NVR2 共 2 个 sector 区)可以进行 IAP (In Application Programming) 操作, 即用户可在程序中将数据写入相应 sector 区域或从该区域读出, 实现 Data EEPROM 功能。

其实 Data EEPROM 功能就是通过对地址为 8200H~85FFH 的 Flash ROM 区域进行擦读写操作来实现的。此功能允许用户程序动态把数据逐字节写入此区域或者从此区域读出。IAP 在线编程功能的实现需要调用 WDT 模块, 所以在调用 IAP 功能实现对 Data EEPROM 擦、写的时候, 不能同时使用 WDT 功能。Data EEPROM 读操作也可通过使用 MOVC 实现, 故 Data EEPROM 读操作对 WDT 使用没有限制。

### 22-1. IAP 相关控制寄存器

#### 22-1-1. IAPDATA(C2H): IAP 数据寄存器

IAP 数据寄存器: 该寄存器存放待写入 Data EEPROM 的 8 位数据

地址	名称	7	6	5	4	3	2	1	0	上电复位值
C2H	IAPDATA	IAPDATA[7:0]								FFH
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	
位编号	位符号	说明								
7~0	IAPDATA	存放待写入 Data EEPROM 的 8 位数据								

#### 22-1-2. IAPADDRH (C3H) : IAP 高位地址寄存器

IAP 高位地址寄存器: 该寄存器存放待写 Data EEPROM 地址的高 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位值
C3H	IAPADDRH	IAPADDRH[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

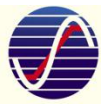
位编号	位符号	说明								
7~0	IAPADDRH	存放待写 Data EEPROM 地址高 8 位								

#### 22-1-3. IAPADDRL (C4H) : IAP 低位地址寄存器

IAP 低位地址寄存器: 该寄存器存放待写 Data EEPROM 地址的低 8 位

地址	名称	7	6	5	4	3	2	1	0	上电复位值
C4H	IAPADDRL	IAPADDRL[7:0]								00H
读/写		W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明								
7~0	IAPADDRL	存放待写 Data EEPROM 地址低 8 位								



#### 22-1-4. IAPCTRL (C5H) : IAP 控制寄存器

IAP 控制寄存器: 该寄存器用来设置 IAP 解锁功能与其他控制信息

ASM87L164 每次 IAP 操作后都会自动清除 IAPCTRL 寄存器的 IAPLOCK 位, 锁死 IAP 功能。如果需要连续进行 IAP 操作, 同时绕过每次 IAP 操作前都必须进行的解锁过程, 可以通过清除 IAP 控制器寄存器的最高位 IAPLOCKEN 来实现。这样就不需要每次 IAP 操作都要先进行解锁, 只需第一次解锁就可以了。连续 IAP 操作完成再重新置位 IAPCTRL 最高位 IAPLOCKEN, 重新启用 IAP 锁死功能保护数据。IAPCTRL 寄存器的擦除操作位和写操作位不能同时为 1, 可以同时为 0 (读操作)。也可使用 MOVC 指令执行读操作。

地址	名称	7	6	5	4	3	2	1	0	上电复位值
C5H	IAPCTRL	IAPLOCKEN	-	-	IAPLOCK	IAPER	IAPWR	IAPINF	IAPEN	80H
	读/写	W/R			W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7	IAPLOCKEN	IAP 功能解锁状态控制位 0: IAP 功能解锁状态不影响 IAP 操作 1: IAP 功能解锁状态影响 IAP 操作
6~5	-	保留
4	IAPLOCK	IAP 功能锁定使能, 0: 锁死; 1: 解锁
3	IAPER	是否执行擦除操作, 0: 不执行擦除操作; 1: 执行擦除操作
2	IAPWR	是否执行写操作, 0: 不执行写操作; 1: 执行写操作
1	IAPINF	数据区选择, 0: 操作 Data EEPROM 区(0x8200H~0x85FFH); 1: 操作程序区(0x0000~0x3FFF), 该操作有风险。
0	IAPEN	IAP 操作是否开始执行, 0: 停止操作; 1: 开始操作

注:

- (1) 其中擦除操作是针对 1K (NVR1~NVR2 共 2 个 Sector) 字节进行擦除操作的。
- (2) IAPLOCKEN 位默认处于保护状态, 此位写 1 或清 0, 必须先进行解锁操作后方便成功写 1 或写 0。

操作过程如下:

```
IAPLOCK = 0x5A; //先写 0x5A
```

```
IAPLOCK = 0xA5; //再写 0xA5
```

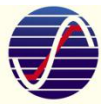
```
IAPCTRL |= 0x10; //解锁
```

```
IAPCTRL &= 0x7F; //解锁后, 此时 IAPCTRL 最高位才可清 0 或写 1
```

- (3) IAP 操作程序区时, 每次擦除、写入前需要将 IAPCTRL 寄存器 bit1 置 1

#### 22-1-5. IAPLOCK (C6H) : IAP 解锁寄存器

ASM87L164 上电复位后, 或者每次 IAP 操作结束后, IAP 功能是自动锁死的, 以免 IAP 误操作, 丢



失 Data EEPROM 内部数据。往此寄存器先写 0x5A，再写 0xA5，并且将 IAP 控制器寄存器(IAPCTRL)的 IAPLOCK 置 1，即可完成 IAP 解锁功能。使用完 IAP 功能后，建议通过将 IAPCTRL 控制寄存器将其最高位 IAPLOCKEN 置 1，并且清除 IAPCTRL 寄存器的 IAPLOCK 来重新锁死 IAP 功能，以保护 Data EEPROM 内部数据。

地址	名称	7	6	5	4	3	2	1	0	上电复位值
C6H	IAPLOCK	IAPLOCK[7:0]								00H
	读/写	W/R	W/R	W/R	W/R	W/R	W/R	W/R	W/R	

位编号	位符号	说明
7~0	IAPLOCK	解锁配置字，先写入 0x5A，再写 0xA5 解锁 Flash 的 IAP 功能

## 22-2. IAP 操作流程

IAP 写入流程如下：

- (1) 写 IAPLOCK 解锁寄存器进行 IAP 解锁，开始 IAP 操作（先写#5AH，再写#A5H）；
- (2) 将要写入 Data EEPROM 的数据写入 IAPDATA（准备好 IAP 写入的数据）；
- (3) 将要写入 Data EEPROM 的地址写入 {IAPADRH[7:0], IAPADDRL[7:0]}；
- (4) 设置 IAP 控制寄存器 IAPCTRL，确定操作(擦除或者写入)；
- (5) 重复以上过程

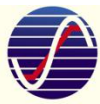
**注意：**

- 使用 IAP 对 Data EEPROM 进行擦与写操作的时候，WDT 不能使用。
- IAP 擦除、写入以及读取操作指令执行前要关闭全局中断，擦除、写入及读取操作完成后再次开启全局中断。
- 通过 MOVC 指令读取时，不需要关全局中断。
- 当高位地址 IAPADRH[7:0]=82H 时，IAP 自动指向 Data EEPROM NVR1 区域（512Bytes）。
- 当高位地址 IAPADRH[7:0]=84H 时，IAP 自动指向 Data EEPROM NVR2 区域（512Bytes）。

IAP 读取操作流程与写入类似，只是在上述(4)步时，将写操作改成读操作。另外 IAP 读取操作还可以直接使用 MOVC 指令读取 Data EEPROM 中的数据。

## 22-3. IAP 读写范例

```
#include <ASM87L164.h>
#include <INTRINS.H>
uchar code *addr;
uchar data_1,data_2;
void main(void)
{
    IAPDATA = 0xa5;
    IAPADDRH = 0x82;
    IAPADDRL = 0x00; //页地址
    _nop_();
}
```



```

IAPLOCK = 0x5a; //解锁, 先写入 0x5A,再写入 0xA5
IAPLOCK = 0xa5;
IAPCTRL |= 0x10; //IAP 解锁
    _nop_();
EA = 0; //擦除指令执行前, 关闭全局中断
IAPCTRL |= 0x09; //擦除操作+IAP 开始操作
    _nop_();
EA = 1; //擦除动作完成后, 打开全局中断
IAPCTRL = 0x80;//以上为擦除一页操作。
//注: 因为是页擦除, 实际使用前应先一一读取保存原先存储的数据再擦
//除, 之后再逐一写入之前的数据达到恢复现场的目的。

    .....
IAPDATA = 0xa5; //写 0xa5 到 Data EEPROM 的 0x8201 地址
IAPADDRH = 0x82;
IAPADDRL = 0x01; //地址 8201
    _nop_();
IAPLOCK = 0x5a;
IAPLOCK = 0xa5;
IAPCTRL |= 0x10;
    _nop_();
EA = 0; //写入操作指令执行前, 关闭全局中断
IAPCTRL |= 0x05; //写入操作+IAP 开始操作
    _nop_();
EA = 1; //写入动作完成后, 打开全局中断
IAPCTRL = 0x80;

    .....
addr=0x8201;//从地址 0x8201 的 Data EEPROM 读取数据
    _nop_();
IAPLOCK = 0x5a;
IAPLOCK = 0xa5;
IAPCTRL |= 0x10;
    _nop_();
EA = 0; //IAP 读操作前, 关闭全局中断
IAPCTRL |= 0x01;
    _nop_();
data_1=*addr;
EA = 1; //IAP 读操作结束后, 打开全局中断
IAPCTRL = 0x80;
}

```



## 23. 电气特性

### 23-1. 极限参数

参数	最小	最大	单位
电源电压 VDD	-0.3	6.0	V
工作温度 T <sub>OPR</sub>	-40	85	°C
存储温度 T <sub>STG</sub>	-55	125	°C
任意管脚对地的输入电压 V <sub>I</sub>	-0.3	VDD+0.3	V

### 23-2. 推荐工作条件

参数	最小	最大	单位
电源电压 VDD	1.8	5.5	V
工作温度 T <sub>OPR</sub>	-40	85	°C

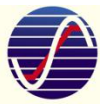
### 23-3. ESD 特性

参数	最小	最大	单位	规范	等级
VESD(HBM)		8	KV	ANSI/ESDA/JEDEC	Class 3B(最高)
VESD(CDM)		2	KV	C JS-001-2017	CLASS C3(最高)
VESD(MM)		600	V	JEDEC-STD	Class 3(最高)
EFT		4.8	KV	IEC61000-4-4	Class 4(最高)
I <sub>latchup</sub>		400	mA	JEDEC-STD 78E	Class I(高)

### 23-4. 直流电气特性

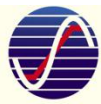
(VDD = 2.0~ 5V, TOPG = 25°C, 如无其他说明则都是此条件)

符号	参数	最小值	典型值	最大值	单位	条件
VDD	工作电压	1.7		5.5	V	1 ≤ F <sub>sys</sub> ≤ 24MHz
F <sub>EXT</sub>	外接晶体	1	16	24	MHz	外接 1MHz 晶振时, 外接电容为 20pF; 其它为 10pF
I <sub>OP1</sub>	工作电流 1		2.3		mA	VDD=5V, 系统时钟选择内部振荡 16MHz, P1 输出方波, 其他 IO 处于输入上拉, 关闭 LVR、定时器、WDT、ADC。
			2.2		mA	VDD=3.3V, 系统时钟选择内部振荡 16MHz, P1 输出方波, 其他 IO 处于输入上



						拉, 关闭 LVR、定时器、WDT、ADC。
I <sub>OP2</sub>	工作电流 2		1.35		mA	VDD=5V, 系统时钟选择内部振荡 8MHz, P1 输出方波, 其他 IO 处于输入上拉, 关闭 LVR、定时器、WDT、ADC。
			1.33		mA	VDD=3.3V, 系统时钟选择内部振荡 8MHz, P1 输出方波, 其他 IO 处于输入上拉, 关闭 LVR、定时器、WDT、ADC。
I <sub>OP3</sub>	工作电流 3		0.86		mA	VDD=5V, 系统时钟选择内部振荡 4MHz, P1 输出方波, 其他 IO 处于输入上拉, 关闭 LVR、定时器、WDT、ADC。
			0.85		mA	VDD=3.3V, 系统时钟选择内部振荡 4MHz, P1 输出方波, 其他 IO 处于输入上拉, 关闭 LVR、定时器、WDT、ADC。
I <sub>OP4</sub>	工作电流 4		0.47		mA	VDD=5V, 系统时钟选择内部振荡 1MHz, P1 输出方波, 其他 IO 处于输入上拉, 关闭 LVR、定时器、WDT、ADC。
			0.47		mA	VDD=3.3V, 系统时钟选择内部振荡 1MHz, P1 输出方波, 其他 IO 处于输入上拉, 关闭 LVR、定时器、WDT、ADC。
I <sub>SB2</sub>	待机电流 (深度睡眠模式)		0.8		uA	VDD=5V, 所有 IO 设为准双向、关闭上下拉, 端口置为 1。工作模式参见“深度睡眠模式章节”。所有功能模块处于关闭状态, 包括内置 LDO 与 Flash 供电模块。
			0.6		uA	VDD=3V, 所有 IO 设为准双向、关闭上下拉, 端口置为 1。工作模式参见“深度睡眠模式章节”。所有功能模块处于关闭状态, 包括内置 LDO 与 Flash 供电模块。
			0.5		uA	VDD=1.8V, 所有 IO 设为准双向、关闭上下拉, 端口置为 1。工作模式参见“深度睡眠模式章节”。所有功能模块处于关闭状态, 包括内置 LDO 与 Flash 供电模块。
V <sub>IL</sub>	低电平输入电压			0.3V <sub>DD</sub>	V	P0 (P0.4、P0.3 与 P0.2 用作 IO 口)、P1 与 P2 设为输入
V <sub>IH</sub>	高电平输入电压	0.7V <sub>DD</sub>		V <sub>DD</sub>	V	P0 (P0.4、P0.3 与 P0.2 用作 IO 口)、P1 与





						P2 设为输入
I <sub>OL1</sub>	10mA IO 输出低电平驱动		10		mA	VDD=5V, V <sub>OL</sub> =0.6V, 推挽输出
I <sub>OL2</sub>	20mA IO 输出低电平驱动		20		mA	VDD=5V, V <sub>OL</sub> =0.6V, 推挽输出
I <sub>OH1</sub>	10mA IO 输出高电平驱动		10		mA	VDD=5V, V <sub>OH</sub> =4.4V, 推挽输出
I <sub>OH2</sub>	20mA IO 输出高电平驱动		20		mA	VDD=5V, V <sub>OH</sub> =4.4V, 推挽输出
R <sub>RST</sub>	RST 上拉电阻	6		10	MΩ	VDD=5V

### 23-5. IRC 电气特性

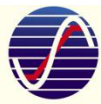
(VDD = 1.8~ 5.5V, TOPG = 25°C, 如无其他说明则都是此条件)

符号	参数	条件	最小	典型值	最大	单位
T <sub>HIRC</sub>	内部高频振荡器起振时间	IRC=1MHz		10	20	us
F <sub>HIRC</sub>	内部高频振荡器稳定性	VDD=2.0V~5.5V; TA=25°C	-1%	16	+1%	MHz
		VDD=5V; TA=-40°C ~ 85°C	-2%	16	+2%	MHz
		VDD=2.0V~5.5V; TA=-40°C ~ 85°C	-2%	16	+2%	MHz
F <sub>LIRC</sub>	内部高频振荡器稳定性	VDD=2.0V~5.5V; TA=-40°C ~ 85°C	-50%	32	+50%	KHz

### 23-6. ADC 电气特性

(VDD = 1.8~ 5V, TOPG = 25°C, 如无其他说明则都是此条件)

符号	参数	条件	最小	典型值	最大	单位
VDDA	ADC 供电电压		2.5		5.5	V
VREF <sup>1</sup>	ADC 外部参考电压		2.0		5.5	V
	ADC 内部参考电压	可选择 2V、3V、4V 以及内部 VDD	-1%		+1%	
AIN	ADC 输入电压		GND		VREF	V
I <sub>ADC</sub>	ADC 转换电流	ADC 模块使能; 使用内部参考; VDD=5V		1.3		mA
DNL	微分非线性误差	VREF=VDD; VDD=5V;		±1.5	±2.5	LSB



		TCLK=2MHz				
INL	积分非线性误差	VREF=VDD; VDD=5V; TCLK=2MHz		±3	±4	LSB
TCLK	ADC 输入时钟	VDD=5V			16	MHz
T <sub>ADC12</sub>	12bit ADC 转换时间		16			ADC 时钟 个数
T <sub>ADC10</sub>	10bit ADC 转换时间		14			
ENOB	有效位	VREF=内部 4V, VDD=5V Tsh=8 个 ADC Clock , ADC Clock=1MHz  <i>Ain=1V~1.5V</i>		10		Bit
		VREF=内部 4V, VDD=5V Tsh=8 个 ADC Clock , ADC Clock=1MHz  <i>Ain &lt; 1V</i>		10.3		Bit
		VREF=内部 2V, VDD=5V Tsh=8 个 ADC Clock , ADC Clock=1MHz  <i>Ain &lt; 500mv</i>		10.3		Bit

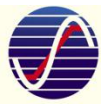
注:

- 1) 在选择 ADC 内部参考电压源时, 其电源电压 VDDA 至少为 VREF+0.5V。
- 2) ADC 部分参数未经过量产测试, 仅作为设计参考之用。

### 23-7. PGA 电气特性

(VDD = 2~ 5.5V, TOPG = 25°C, 如无其他说明则都是此条件)

Parameters	Condition	Min.	Typ.	Max.	Unit	备注
Power Supply(VDDP)		2	3.3	5.5	V	
Operation Temperature		-40	25	85	C	
Input Common-mode		0		VDDP-1	V	

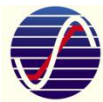


Voltage Range(Vcm)						
Gain		1		70		
Gain Error	DC			1	%	
	Gain=1~40(fin ≤ 40KHz)			5	%	
	Gain=50~70(fin ≤ 40KHz)			8	%	
input Offset				+/-6	mV	
PSRR			90		dB	
Current load drive(Iout)	VDDP=5V		80		uA	
Capacitive load drive(Cload)				100	pF	
Quiescent Current	PD=0 (VDDP=5V)		160		uA	
Power Down Current	PD=1 (VDDP=5V)			100	nA	

### 23-8. LVD 电气特性

(VDD =5V, TOPG = 25°C, 如无其他说明则都是此条件)

符号	参数	条件	最小	典型值	最大	单位
Vlevel	LVD 检测电压	LVDS[2:0]=3'b000		1.8		V
		LVDS[2:0]=3'b001		2		V
		LVDS[2:0]=3'b010		2.4		V
		LVDS[2:0]=3'b011		2.6		V
		LVDS[2:0]=3'b100		3		V
		LVDS[2:0]=3'b101		3.6		V
		LVDS[2:0]=3'b110		4		V
		LVDS[2:0]=3'b111		4.5		V
Vhyste	迟滞电压			130		Mv

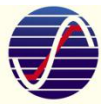


Vdev	检测电压点精度			2	4	%
------	---------	--	--	---	---	---

### 23-9. LVR 电气特性

(VDD =5V, TOPG = 25°C, 如无其他说明则都是此条件)

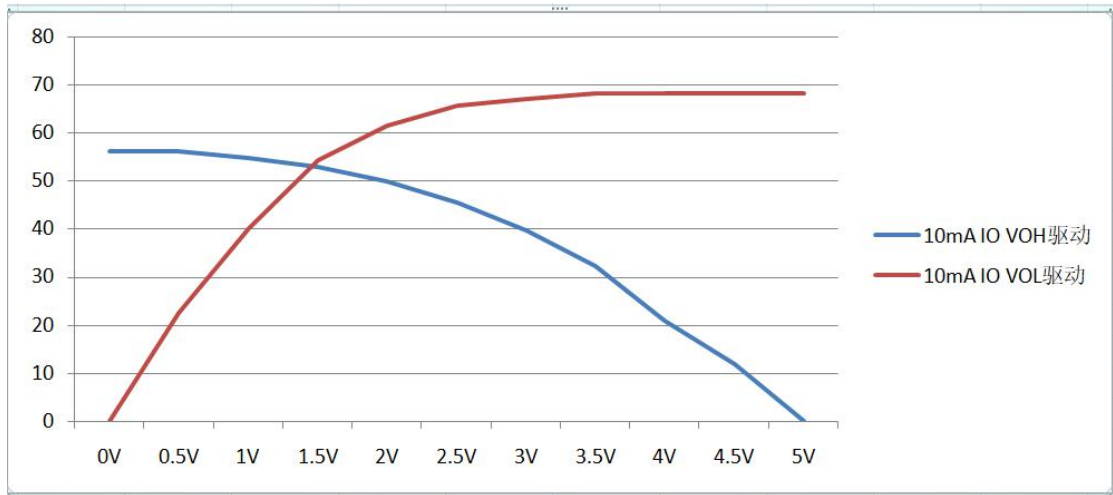
符号	参数	条件	最小	典型值	最大	单位
Vlevel	LVR 检测电压	LVRSEL[3:0]=4'b0000		1.6		V
		LVRSEL[3:0]=4'b0001		1.7		V
		LVRSEL[3:0]=4'b0010		1.8		V
		LVRSEL[3:0]=4'b0011		2		V
		LVRSEL[3:0]=4'b0100		2.2		V
		LVRSEL[3:0]=4'b0101		2.4		V
		LVRSEL[3:0]=4'b0110		2.6		V
		LVRSEL[3:0]=4'b0111		2.8		V
Vhyste	迟滞电压			200		Mv
Vdev	检测电压点精度			2	4	%



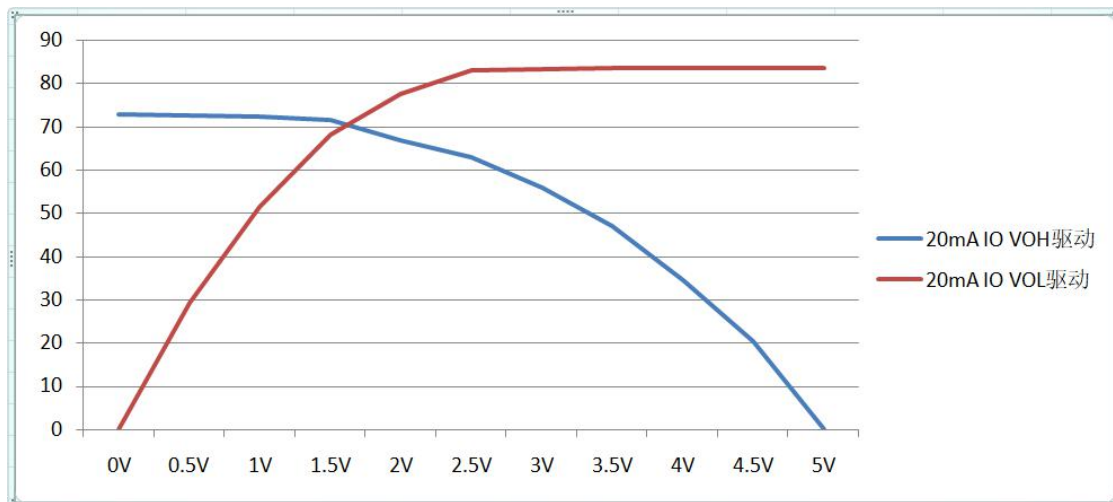
## 23-10. 各项曲线图

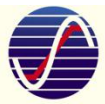
以下各项测试条件，如不涉及温度，皆为常温 25℃。

### 23-10-1. IO 口推挽输出驱动能力 (10mA IO)

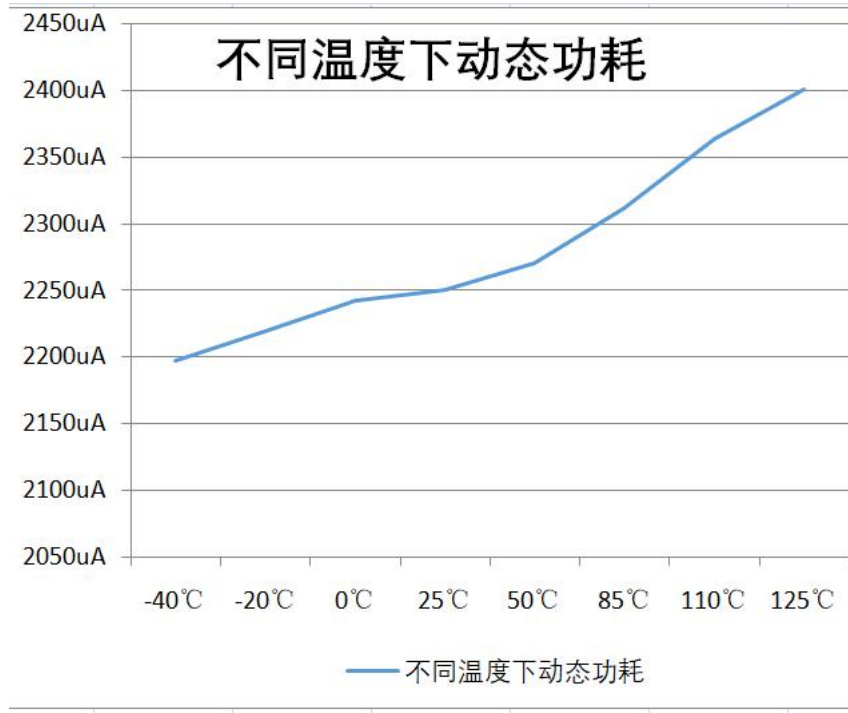


### 23-10-2. IO 口推挽输出驱动能力 (20mA IO)

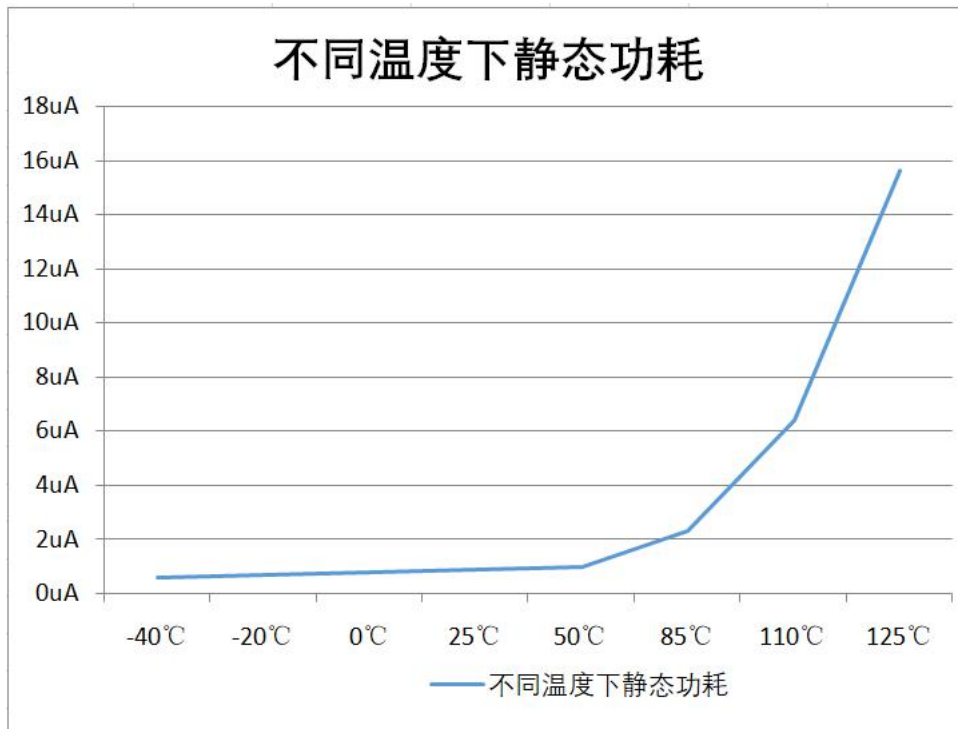


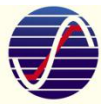


23-10-3. 动态功耗 (内部 RC) VS 环境温度



23-10-4. 静态功耗 VS 环境温度

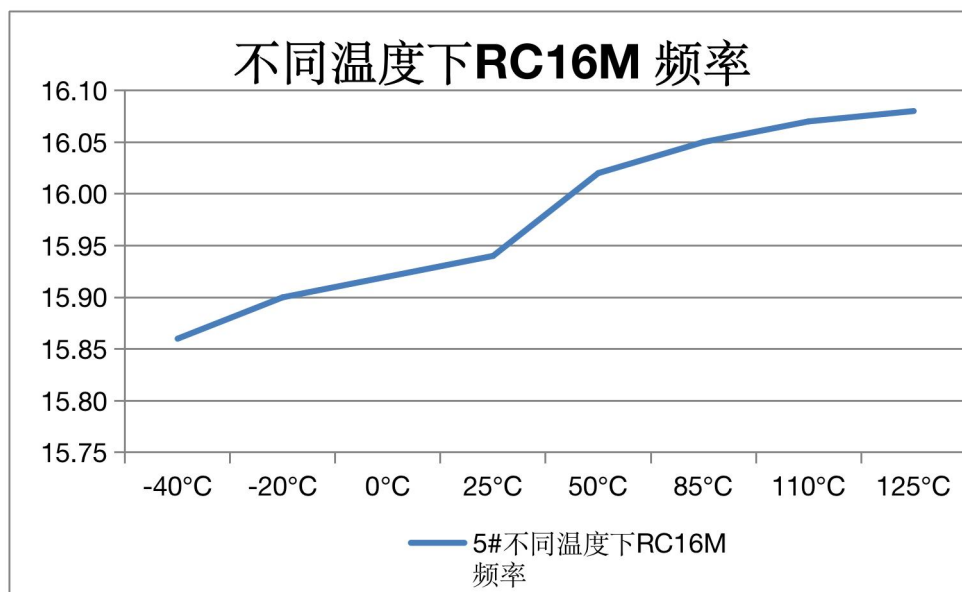


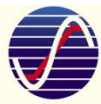


23-10-5. 内部 HIRC 频率 VS 电源电压

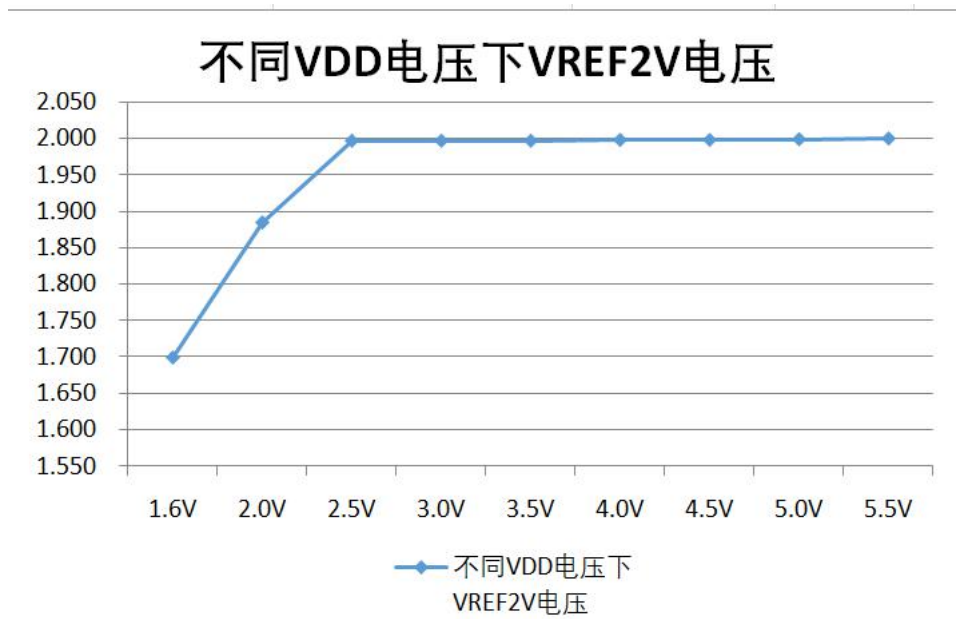


23-10-6. 内部 HIRC 频率 VS 环境温度

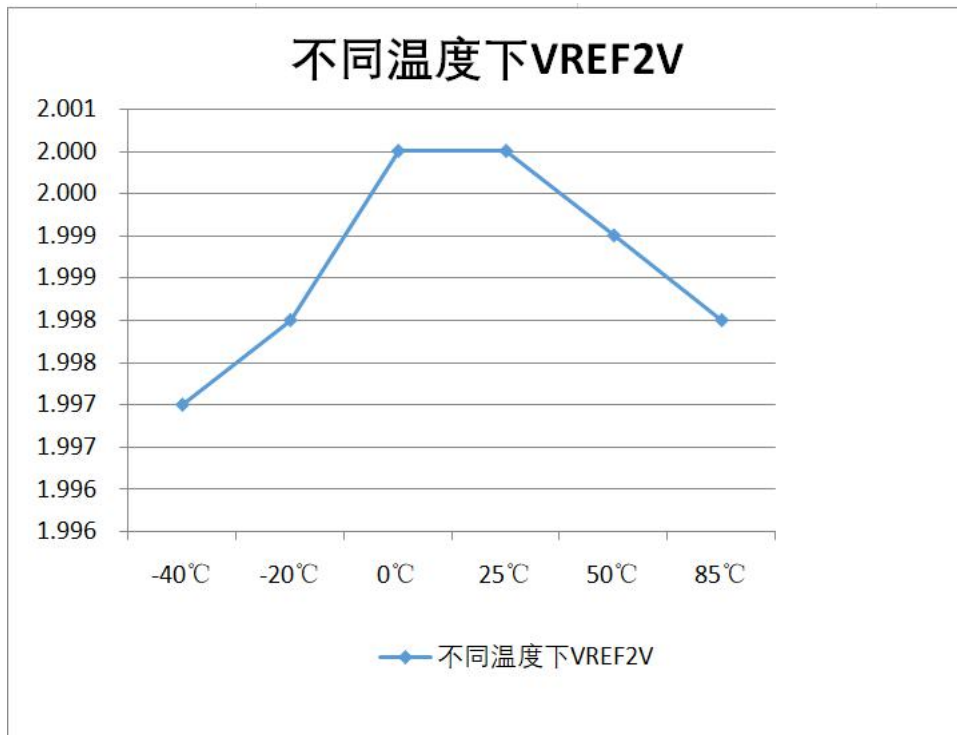




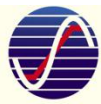
23-10-7. AD 内部参考 (2V) VS 电源电压



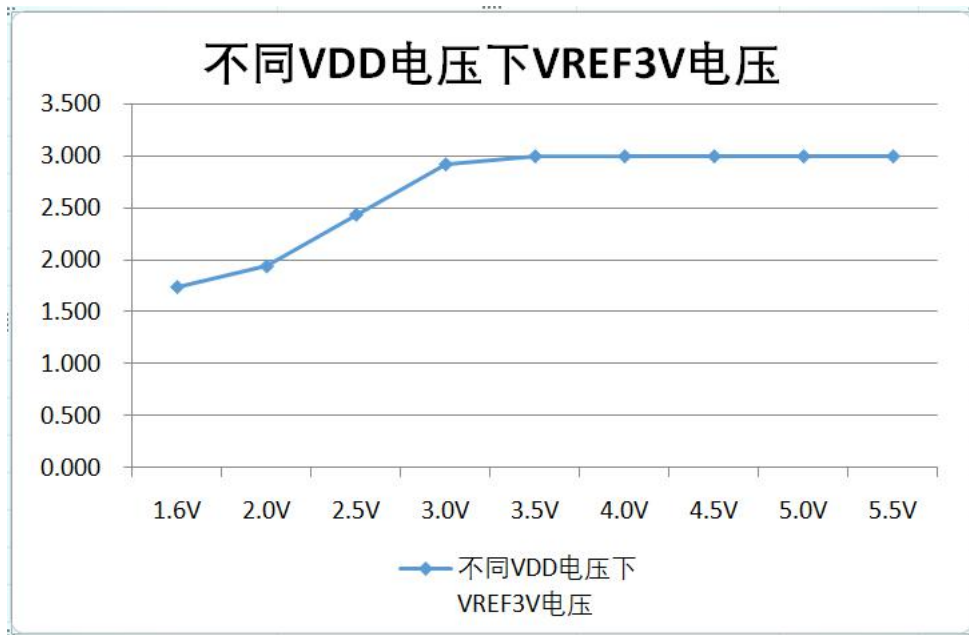
23-10-8. AD 内部参考 (2V) VS 环境温度



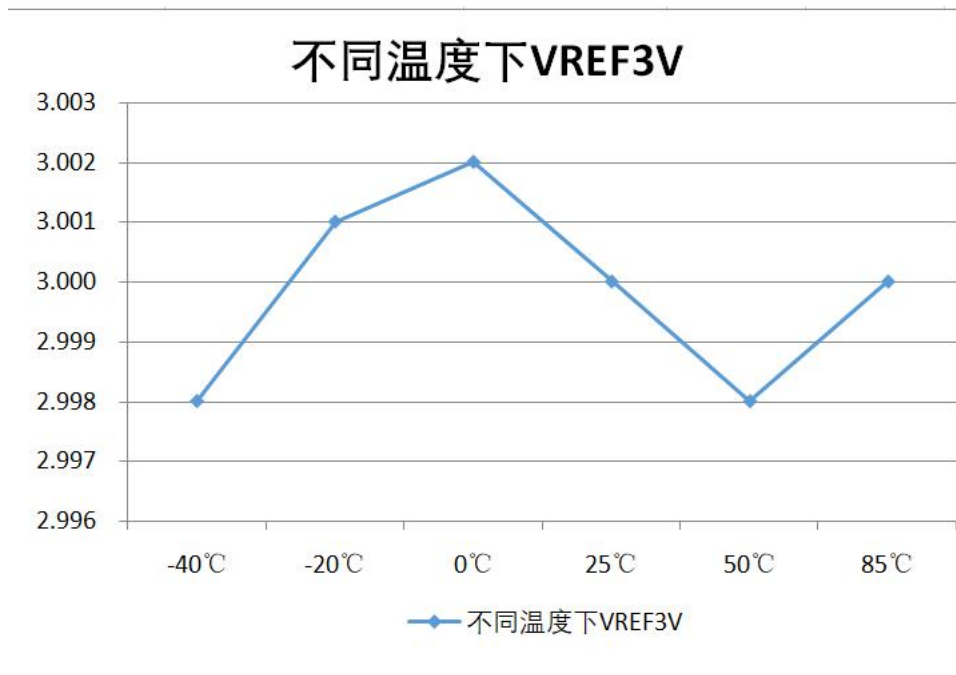


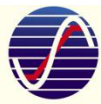


23-10-9. AD 内部参考 (3V) VS 电源电压

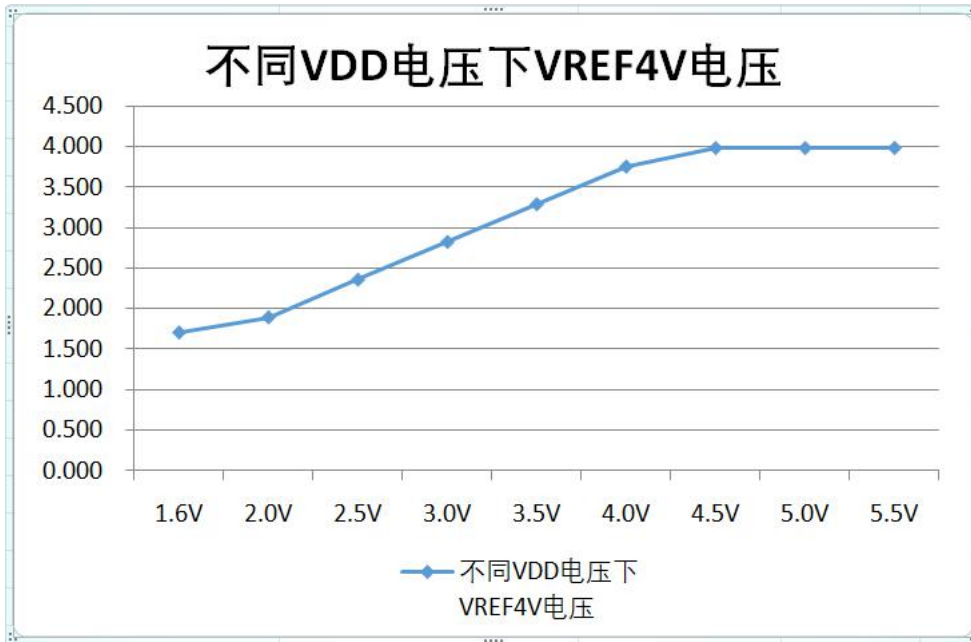


23-10-10. AD 内部参考 (3V) VS 环境温度

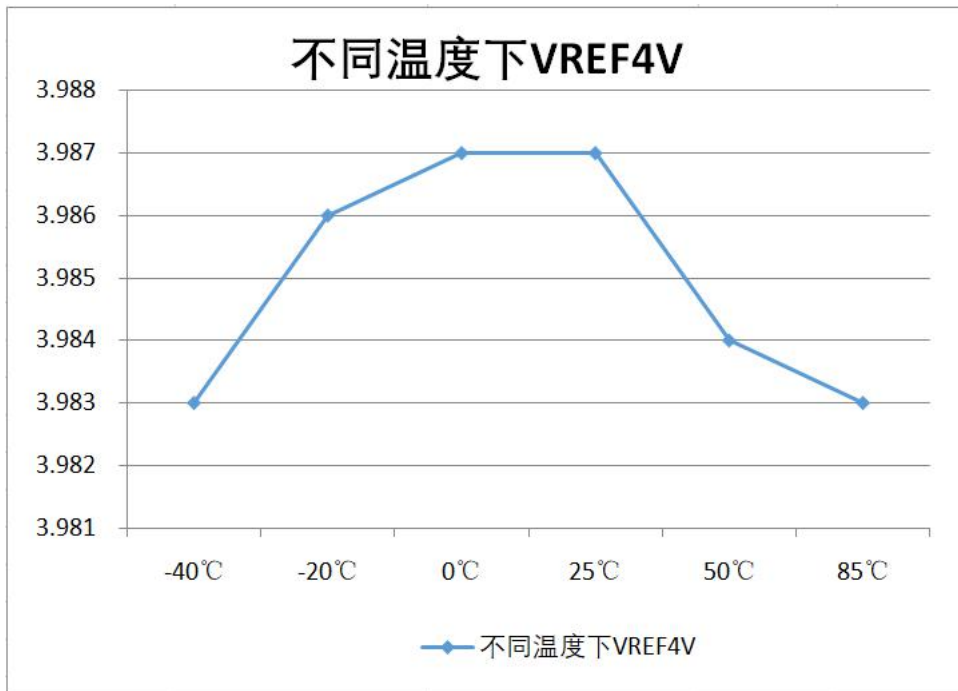




23-10-11. AD 内部参考 (4V) VS 电源电压

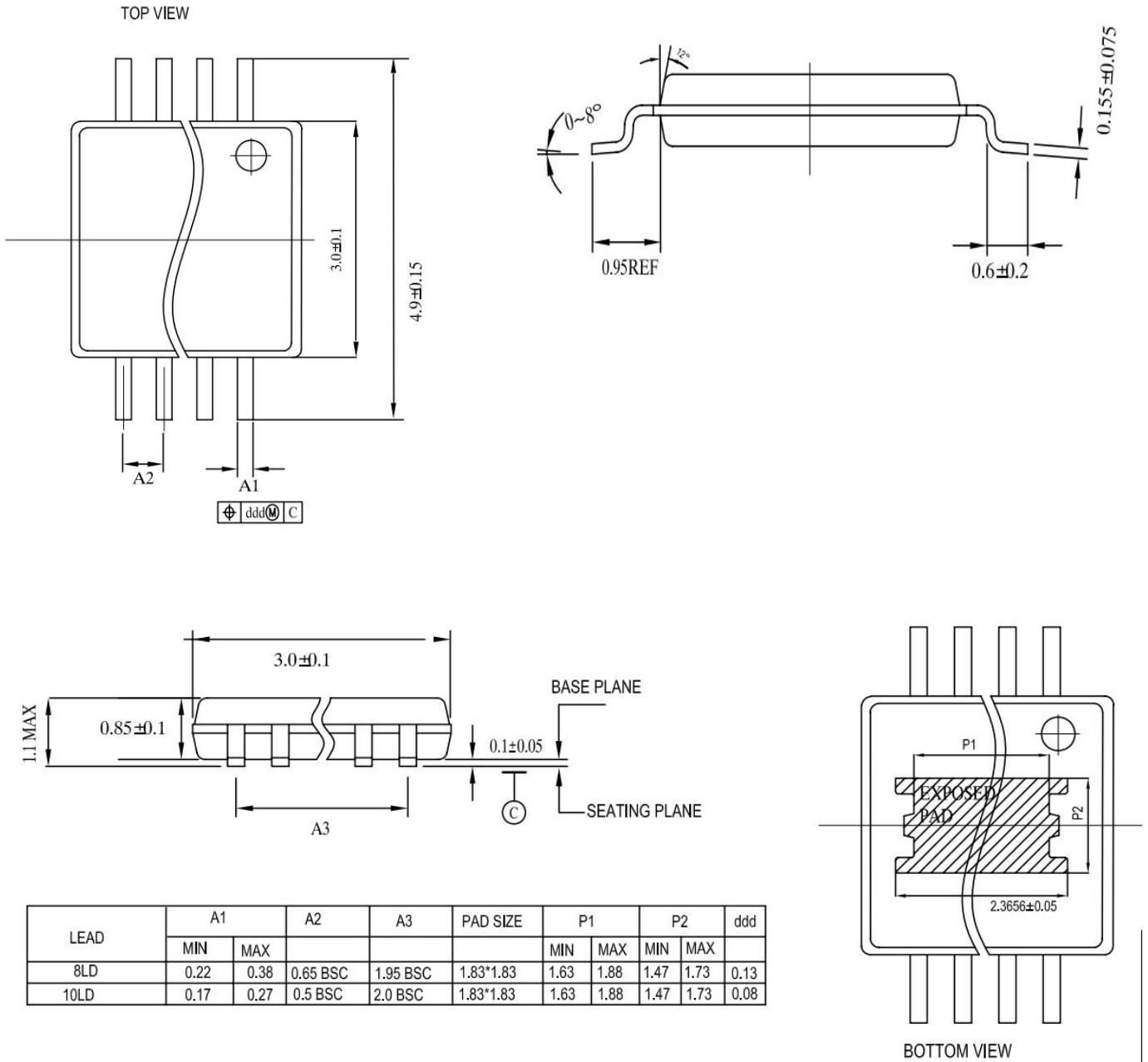


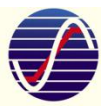
23-10-12. AD 内部参考 (4V) VS 环境温度



## 24. 封装外形

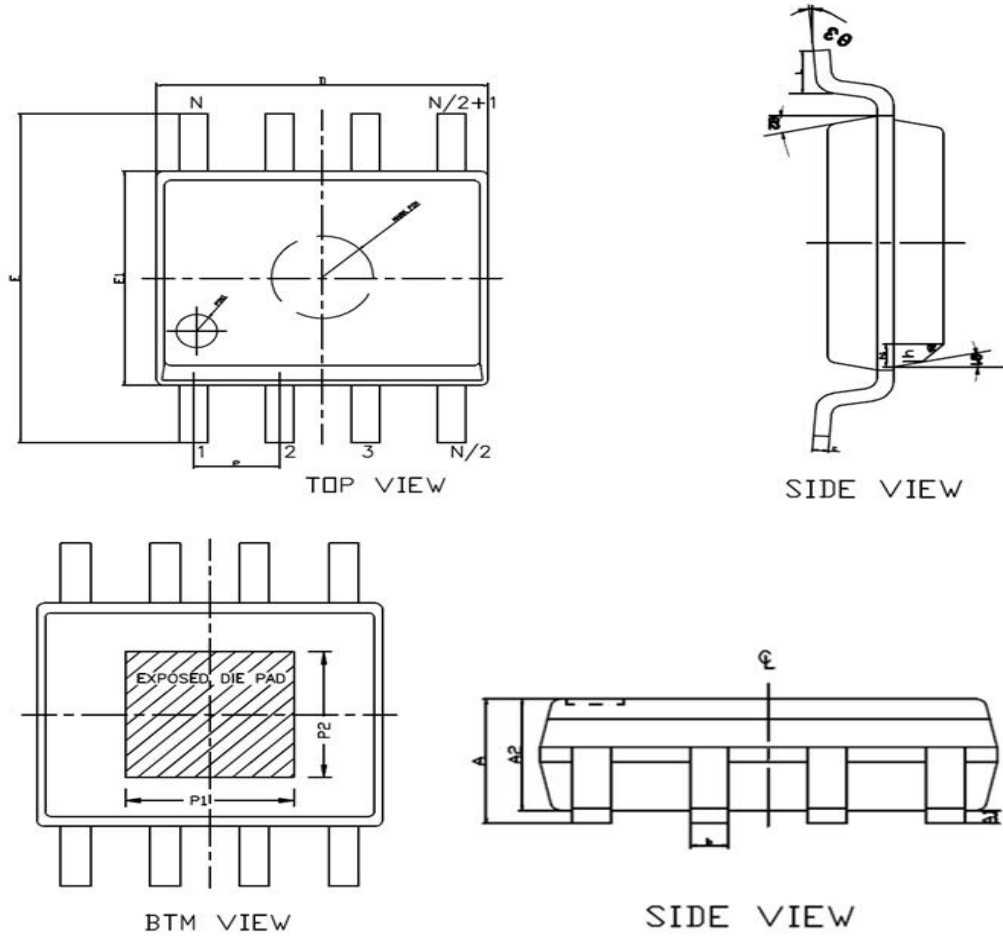
MSOP10 外形尺寸 (单位: mm)





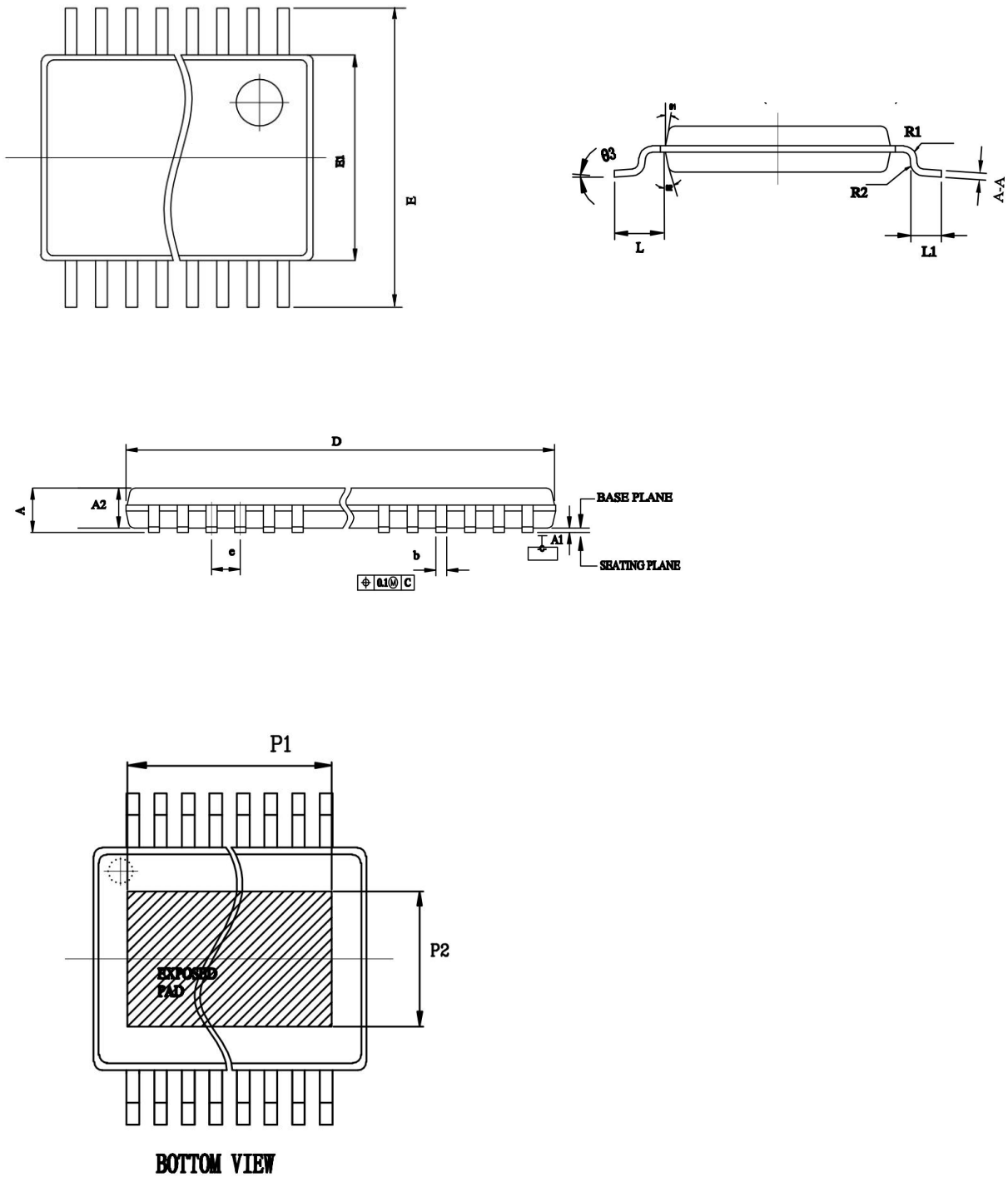
SOIC14/16 (150mil) 外形尺寸

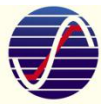
SOIC20 (300mil) 外形尺寸 (单位: mm)



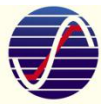
Symbol	符号	SOIC8(150MIL)		SOICW8(208MIL)		SOIC14(150MIL)		SOIC16(150MIL)		SOICW16(300MIL)		SOIC20(300MIL)	
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
Pad Size	衬底尺寸	-	-	-	-	-	-	-	-	-	-	-	-
P1	散热片长	-	-	-	-	-	-	-	-	-	-	-	-
P2	散热片宽	-	-	-	-	-	-	-	-	-	-	-	-
A	总高	1.499	1.750	1.751	2.154	1.499	1.750	1.499	1.750	2.388	2.650	2.388	2.650
A1	站高	0.102	0.249	0.050	0.249	0.102	0.249	0.102	0.249	0.102	0.300	0.102	0.300
A2	塑封体高	1.397	-	1.701	-	1.397	-	1.397	-	2.286	-	2.286	-
E	跨度	5.842	6.198	7.747	8.077	5.842	6.198	5.842	6.198	10.109	10.566	10.109	10.566
E1	塑封体宽	3.877	3.997	5.182	5.385	3.915	3.998	3.861	3.998	7.442	7.595	7.442	7.595
D	塑封体长	4.852	4.952	5.130	5.334	8.658	8.738	9.804	9.957	10.262	10.414	12.687	12.891
L	脚长	0.406	0.889	0.508	0.762	0.406	0.889	0.406	0.889	0.406	0.889	0.406	0.889
e	脚间距	1.27TYP		1.27TYP		1.27TYP		1.27TYP		1.27TYP		1.27TYP	
b	引脚宽度	0.406TYP		0.406TYP		0.406TYP		0.406TYP		0.406TYP		0.406TYP	
c	引脚厚度	0.2TYP		0.2TYP		0.2TYP		0.2TYP		0.254TYP		0.254TYP	
θ1	脱模斜度	12°TYP		7°TYP		7°TYP		7°TYP		7°TYP		7°TYP	
θ2	脱模斜度	12°TYP		7°TYP		7°TYP		7°TYP		7°TYP		7°TYP	
θ3	引脚角度	0	8	0	8	0	8	0	8	0	8	0	8
θ4	脱模倒角度	45		-		45		45		45		45	
h	脱模倒角1	-	-	-		-	-	-	-	-	-	-	-
h1	脱模倒角2	0.254	0.457	-		0.254	0.457	0.254	0.457	0.254	0.457	0.254	0.457

TSSOP-20 外形尺寸 (单位: mm)





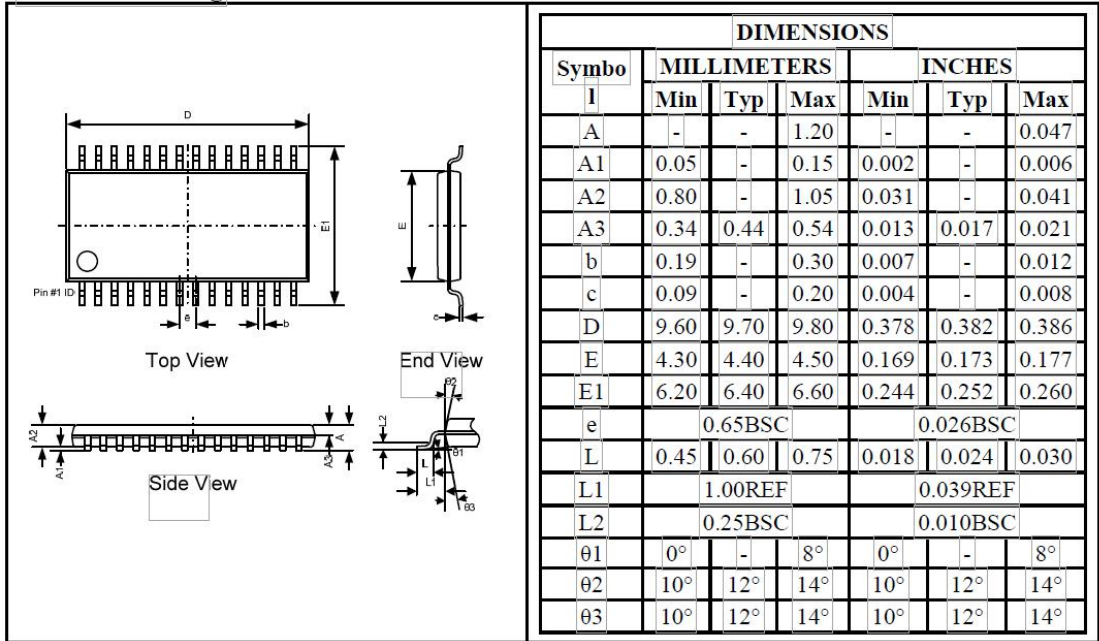
Symbol	符号	TSSOP8LD		TSSOP14LD		TSSOP20	
		Min	Max	Min	Max	Min	Max
A	总高		1.2		1.2		1.2
A1	站高	0.05	0.15	0.05	0.15	0.05	0.15
A2	塑封体高	0.8	1.05	0.8	1.05	0.8	1.05
E	跨度	6.25	6.55	6.25	6.55	6.25	6.55
E1	塑封体宽	4.3	4.5	4.3	4.5	4.3	4.5
D	塑封体长	2.9	3.1	4.9	5.1	6.4	6.6
L	脚长		1		1		1
L1		0.45	0.75	0.45	0.75	0.45	0.75
e	脚间距	0.65		0.65		0.65	
b	脚宽	0.19	0.3	0.19	0.3	0.19	0.3
R1		0.15TYP		0.15TYP		0.15TYP	
R2		0.15TYP		0.15TYP		0.15TYP	
A-A		0.09	0.2	0.09	0.2	0.09	0.2
$\theta 1$	脱模斜度	12° TYP		12° TYP		12° TYP	
$\theta 2$	脱模斜度	12° TYP		12° TYP		12° TYP	
$\theta 3$	引脚角度	0	8	0	8	0	8



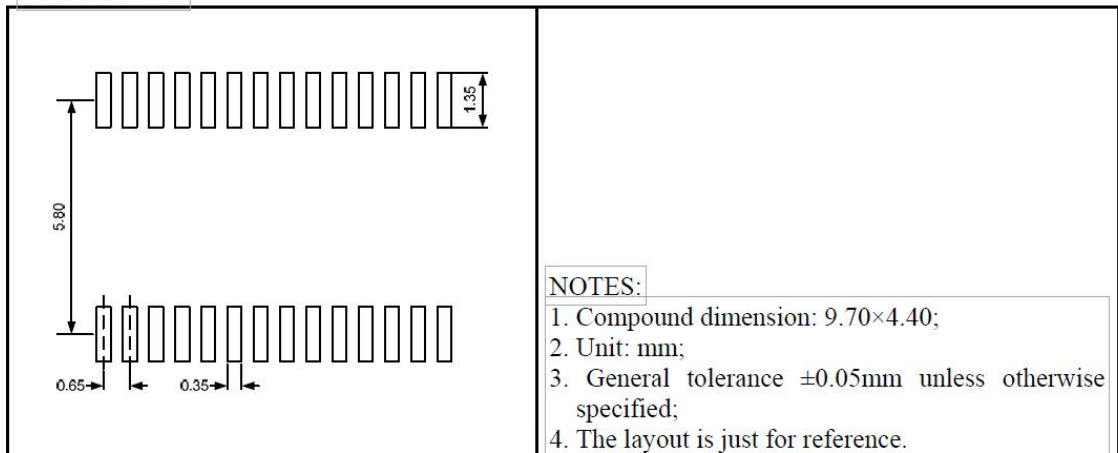
TSSOP-28 外形尺寸 (单位: mm)

TSSOP28

Outline Drawing



Land Pattern





## 25. 修订记录

版本	更改概要	修改日期
V0.1	初始版本	2019/05/06
V0.2	增加电气特性参数	2019/09/24
V0.3	修改完善	2019/10/11
V0.4	增加 TSSOP28 封装	2019/11/18
V0.5	Timer2/PWM/PGA/IAP 章节完善	2019/12/10



## **X-ON Electronics**

Largest Supplier of Electrical and Electronic Components

*Click to view similar products for [sine microelectronics](#) manufacturer:*

Other Similar products are found below :

[ASM87L164](#) [ASM6050AD](#) [ASM87A164](#)