

ME9101 用户手册

Ver 4.0

南京微盟电子有限公司

2018.03

目录

1. 简介	5
1.1 概述	5
1.2 结构框图	6
1.3 封装形式	7
1.4 引脚说明	8
2. 8051 CPU 核.....	9
2.1 概述	9
2.2 寻址方式	9
2.3 指令	10
2.4 SFR 寄存器	13
3. 存储器结构.....	16
3.1 简介	16
3.2 程序存储器	16
3.3 内部 RAM	16
4. 定时器.....	18
4.1 概述	18
4.2 结构框图	18
4.3 控制寄存器	18
4.4 电路说明	22
5. 看门狗定时器 WDT	27
5.1 概述	27
5.2 结构框图	27
5.3 控制寄存器	27
5.4 WDT 的操作	28
6. 模数转换器.....	29
6.1 概述	29
6.2 结构框图	29
6.3 控制寄存器	30
6.4 ADC 操作说明和注意事项	32
7. I/O 输入输出端口.....	34
7.1 概述	34
7.2 I/O 控制寄存器	34
7.3 编程注意事项.....	38
8. 系统时钟	39

8.1 概述	39
8.2 系统时钟框图.....	39
8.3 控制寄存器	39
8.4 指令周期 CLK_MCU.....	41
8.5 系统高速时钟.....	41
8.6 系统低速时钟.....	43
9. 检测保护功能	44
9.1 简介	44
9.2 控制寄存器	44
9.3 工作原理	45
10. 复位和初始化	46
10.1 简介	46
10.2 上电复位	46
10.3 看门狗复位	46
10.4 外部复位	47
11. 系统工作模式	50
11.1 概述	50
11.2 低速模式	51
11.3 高速模式	51
11.4 空闲模式	51
11.5 停止模式	52
12. 中断	53
12.1 概述	53
12.2 硬件结构框图.....	54
12.3 中断寄存器	54
12.4 中断处理	56
12.5 中断屏蔽	57
12.6 中断优先级	57
12.7 中断采样	57
12.8 中断延迟	57
13. OTP 烧录	58
13.1 概述	58
13.2 OTP 烧录信息	58
14. SFR 寄存器.....	59
14.1 寄存器汇总	59
14.2 寄存器表	60
15. 电特性.....	61

15.1 极限参数	61
15.2 DC 特性	61
15.3 ADC 特性	62
16. 封装信息	63
16.1 封装类型: SOP 14 单位:mm(inch)	63
16.2 封装类型: SSOP 16 单位:mm(inch)	64

1. 简介

1.1 概述

ME9101是一款8位高性能、高效率的混合信号微控制器（MCU）。可编程的4K Byte OTP ROM给用户使用提供了极大的方便。多通道的12位A/D转换器可用于直接处理模拟信号，例如直接连接传感器。该MCU带有多个使用灵活的定时器模块，可提供定时功能、脉冲产生功能及PWM/Buzzer产生功能。内部看门狗定时器、低电压检测、温度检测等内部保护特性，外加优秀的抗干扰和ESD保护性能，确保单片机在恶劣的电磁干扰环境下可靠地运行。

该MCU提供了高速时钟HIRC和低速时钟LIRC振荡器功能选项，且内建完整的系统振荡器，无需外接元件。其不同工作模式之间动态切换的能力，为用户提供了一个优化单片机操作和减少功耗的手段。

有了集成的A/D和PWM等功能的优势，再加上低功耗、高性能、灵活控制的输入/输出和低成本等特性，此MCU可广泛被应用在电子测量仪器、环境监控、手持式测量工具、家庭应用、电子控制工具、马达控制等场合。

ME9101的主要特性如下：

(1) MCU 核：

- 8 位 8051 兼容的 MCU，运行速度比标准的 8051 快 3 倍。
- 4 个时钟长度的指令周期，相对标准 8051，平均指令执行时间缩短 2.5 倍。
- 双数据指针。
- 位操作指令。

(2) 内部存储：

- 4K Bytes OTP ROM。
- 128 Bytes RAM。

(3) 12 个具有上拉功能的 IO 端口：

- 输入输出双向端口：P0.x(P0.4 除外)、P4.x、P5.x。
- 单向输入引脚：P0.4，与复位引脚共用。
- 具有唤醒功能的端口：P0.X（P0.0、P0.1、P0.2、P0.3）。
- 4 个外部中断输入端口。

(4) 5+1 通道 12 位 A/D 转换器

- 5 个外部 ADC 输入。
- 一个内部电池检测。
- 内部 AD 参考电压（VDD、4V、3V、2V）或外部 P4.0 输入参考电压。

(5) 2 个 16 位 Timer：

- 具有预分频器(Prescaler)及中断功能的定时器。
- 脉冲宽度调制 PWM 输出。
- BUZZER 输出。

(6) 8 位的看门狗定时器(WDT)。

(7) 内部、外部双时钟系统：

- 外部高速时钟：
 - RC 模式：高达 10 MHz。
 - 晶体模式：高达 16 MHz。
 - 内部高速时钟：RC 模式，16MHz。
 - 内部低速时钟：RC 模式，32KHz。
- (8) 具有低电压检测 LVD、温度检测等保护功能。
- (9) 下列特性可以节省功耗：
- 空闲模式
 - 停止模式
- (10) 工作电压： 2.2V ~ 5.5 V
- (11) 温度范围： -40°C ~ +85°C

1.2 结构框图

ME9101是一个混合信号SOC，提供了多个功能模块。图1.1为其结构框图。

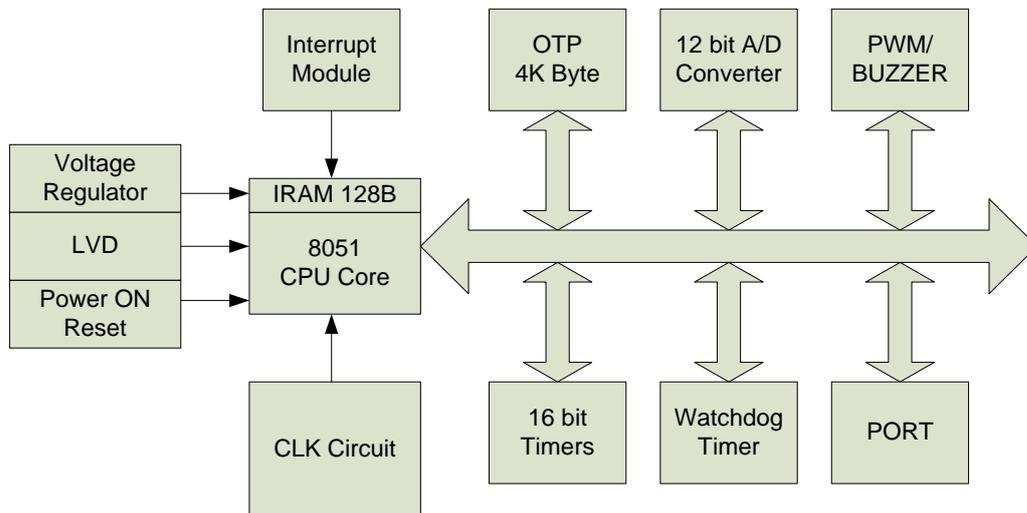


图 1.1 ME9101 结构框图

1.3 封装形式

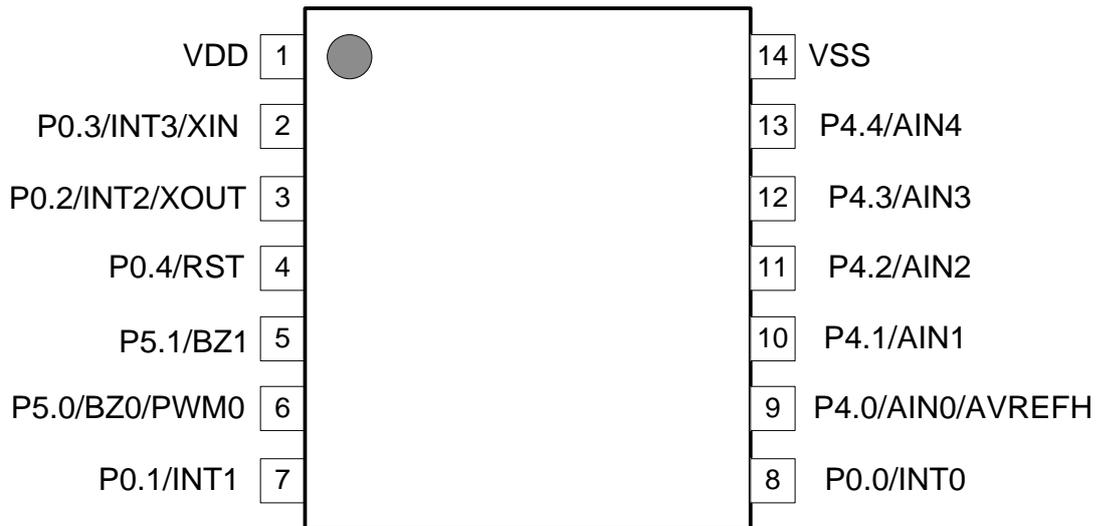


图 1.2 SOP14 (top view)

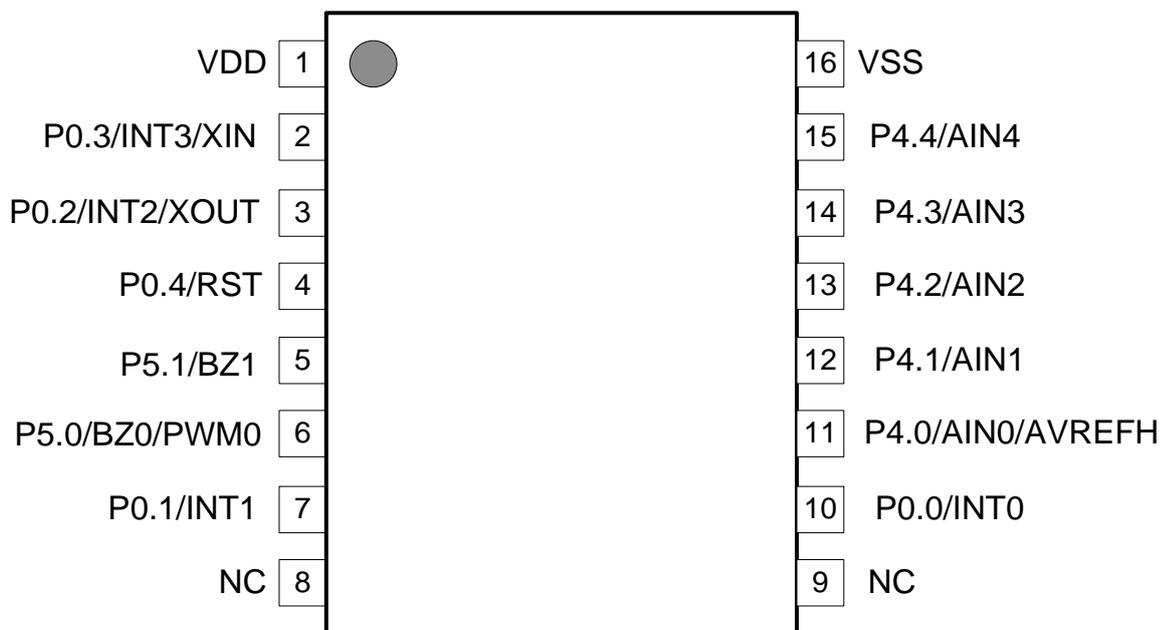


图 1.3 SSOP 16 (top view)

注：如您需要其他封装形式的产品，请联系我司销售人员

1.4 引脚说明

Table 1.1: 引脚定义

Name	Pin		Type	Description
	14pins	16pins		
VDD	1	1	P	电源输入端
P0.3/INT3/XIN	2	2	I/O	P0.3: 双向输入/输出引脚。内置上拉电阻。 INT3: 外部中断输入引脚, 具有唤醒功能。 XIN: 使能外部振荡电路(晶体/RC 振荡电路)时为振荡信号输入引脚。
P0.2/INT2/ XOUT	3	3	I/O	P0.2: 双向输入/输出引脚。内置上拉电阻。 INT2: 外部中断输入引脚, 具有唤醒功能。 XOUT: 使能外部晶体振荡器时为振荡器输出引脚。
P0.4/RST	4	4	I	P0.4: 禁止外部复位时为单向输入引脚, 施密特触发, 无内置上拉电阻。作普通输入引脚使用时, 用户需在单片机的 P0.4 外面串接一个 100 欧姆的电阻(电阻要尽可能的靠近单片机)。 RST: 系统复位输入引脚, 施密特结构, 低电平触发, 通常保持高电平。
P5.1/BZ1	5	5	I/O	P5.1: 双向输入/输出引脚。内置上拉电阻。 BZ1: Buzzer 1 输出引脚。
P5.0/BZ0/PWM0	6	6	I/O	P5.0: 双向输入/输出引脚。内置上拉电阻。 PWM0: PWM0 输出引脚。 BZ0: Buzzer0 输出引脚。
P0.1/INT1	7	7	I/O	P0.1: 双向输入/输出引脚。内置上拉电阻。 INT1: 外部中断输入引脚, 具有唤醒功能。
P0.0/INT0	8	10	I/O	P0.0: 双向输入/输出引脚。内置上拉电阻。 INT0: 外部中断输入引脚, 具有唤醒功能。
P4.0/AIN0/ AVREFH	9	11	I/O	P4.0: 双向输入/输出引脚。内置上拉电阻。 AIN0: ADC 输入通道。 AVREFH: ADC 参考电压的高电平输入引脚。
P4.1/AIN1	10	12	I/O	P4.1: 双向输入/输出引脚。内置上拉电阻。 AIN1: ADC 输入通道。
P4.2/AIN2	11	13	I/O	P4.2: 双向输入/输出引脚。内置上拉电阻。 AIN2: ADC 输入通道。
P4.3/AIN3	12	14	I/O	P4.3: 双向输入/输出引脚。内置上拉电阻。 AIN3: ADC 输入通道。
P4.4/AIN4	13	15	I/O	P4.4: 双向输入/输出引脚。内置上拉电阻。 AIN4: ADC 输入通道。
VSS	14	16	P	电源地
NC	-	8、9	-	Not Connection

2. 8051 CPU 核

2.1 概述

ME9101的CPU核是一个高性能的8051，完全兼容803x/805x控制器。其技术优势如下：

- 4 个时钟的指令周期（标准 8051 的指令周期为 12 个时钟）
 - 平均运行速度是标准 8051 的 3 倍
- 灵活可调的存储器周期
 - 应用软件可通过调整来适应外部 RAM 的工作速度
 - MOVX 小至 8 个时钟周期
- 双数据指针
 - 当移动大数据块时可提高效率
- 低功耗、低电磁干扰(EMI)

2.2 寻址方式

寻址方式是寻找操作数或操作数地址的方式。8051指令集中，寻址模式有如下几种：

1) **立即寻址**：在指令中直接给出操作数。常使用于常数操作。

```
MOV A, #80H 8 位操作数
```

2) **直接寻址**：指令中直接给出操作数地址，只有内部 RAM 的空间 00H-7FH 和 SFR 可以直接寻址。因此常使用于 SFR 和内部 RAM 操作。

```
MOV PSW, #50H
```

```
MOV A, 30H
```

3) **间接寻址**：以寄存器中的内容作为操作数的地址。内部 RAM 的空间 80H-FFH 和外部 RAM 可以间接寻址。外部数据 RAM 寻址指令上采用 MOVX。常使用于内部 RAM 和外部 RAM 操作。

能够用于寄存器间接寻址的寄存器有：R0, R1, DPTR, SP。

8 位寻址的地址寄存器可以选用寄存器 R0、R1 或堆栈指针 SP。16 位寻址的地址寄存器只能是 16 位的数据指针寄存器 DPTR。

```
MOV @R0, A
```

```
MOVX A, @R1
```

```
MOVX @DPTR, A
```

4) **寄存器寻址**：以通用寄存器的内容作为操作数（通用寄存器包括 A, B, DPTR, R0~R7）。

```
INC DPTR
```

注意：A、B 既是通用寄存器，也是 SFR（直接寻址）

5) **变址寻址**：以基址寄存器 PC 或者 DPTR 与变址寄存器 A 中的内容之和作为操作数的地址。变址寻址只能对程序存储器中的数据进行寻址，由于程序存储器是只读的，因此变址寻址只有读操作，指令上采用 MOVC。

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

```
MOVC A, @A+PC
```

- 6) **相对寻址:** 用于修改 PC 的值, 使得 PC 加上指令中给出的一字节的偏移量。
 由于转移指令有两字节和三字节这两种形式, 因此偏移量的范围分别为-126~+129和-125~+130。

SJMP 80H

- 7) **位寻址:** 以位地址中的内容为操作数。

SETB 20H

2.3 指令

所有8051指令为二进制代码, 完全与标准的8051功能相同。这些指令的bits、标志位及状态功能的效果也与标准的8051相同。但指令的执行时间是不同的, 无论是每个指令周期的时钟数, 还是指令周期的时间。

表2.1: 指令集中符号说明

符号	功能
A	Accumulator
Rn	Register R0-R7
direct	Internal register address
@Ri	Internal register pointed to by R0 or R1 (except MOVX)
rel	Two's complement offset byte
bit	Direct bit address
#data	8-bit constant
#data 16	16-bit constant
addr 16	16-bit destination address
addr 11	11-bit destination address

表2.2 列出了CPU核的指令集及相关的操作码、时钟数和字节数。

表 2.2: 指令集

Mnemonic	Description	Byte	Instr. Cycles	Hex Code
Arithmetic				
ADD A, Rn	Add register to A	1	1	28-2F
ADD A, direct	Add direct byte to A	2	2	25
ADD A, @Ri	Add data memory to A	1	1	26-27
ADD A, #data	Add immediate to A	2	2	24
ADDC A, Rn	Add register to A with carry	1	1	38-3F
ADDC A, direct	Add direct byte to A with carry	2	2	35
ADDC A, @Ri	Add data memory to A with carry	1	1	36-37
ADDC A, #data	Add immediate to A with carry	2	2	34

SUBB A, Rn	Subtract register from A with borrow	1	1	98-9F
SUBB A, direct	Subtract direct byte from A with borrow	2	2	95
SUBB A, @Ri	Subtract memory from A with borrow	1	1	96-97
SUBB A, #data	Subtract immediate from A with borrow	2	2	94
INC A	Increment A	1	1	04
INC Rn	Increment register	1	1	08-F
INC direct	Increment direct byte	2	2	05
INC @Ri	Increment data memory	1	1	06-07
DEC A	Decrement A	1	1	14
DEC Rn	Decrement register	1	1	18-1F
DEC direct	Decrement direct byte	2	2	15
DEC @Ri	Decrement data memory	1	1	16-17
INC DPTR	Increment data pointer	1	3	A3
MUL AB	Multiply A by B	1	5	A4
DIV AB	Divide A by B	1	5	84
DA A	Decimal adjust A	1	1	D4
Logical				
ANL A, Rn	AND register to A	1	1	58-5F
ANL A, direct	AND direct byte to A	2	2	55
ANL A, @Ri	AND data memory to A	1	1	56-57
ANL A, #data	AND immediate to A	2	2	54
ANL direct, A	AND A to direct byte	2	2	52
ANL direct, #data	AND immediate data to direct byte	3	3	53
ANL A, Rn	OR register to A	1	1	48-4F
ANL A, direct	OR direct byte to A	2	2	45
ANL A, @Ri	OR data memory to A	1	1	46-47
ANL A, #data	OR immediate to A	2	2	44
ANL direct, A	OR A to direct byte	2	2	42
ANL direct, #data	OR immediate data to direct byte	3	3	43
XRL A, Rn	Exclusive-OR register to A	1	1	68-6F
XRL A, direct	Exclusive-OR direct byte to A	2	2	65
XRL A, @Ri	Exclusive-OR data memory to A	1	1	66-67
XRL A, #data	Exclusive-OR immediate to A	2	2	64
XRL direct, A	Exclusive-OR A to direct byte	2	2	62
XRL direct, #data	Exclusive-OR immediate data to direct byte	3	3	63
CLR A	Clear A	1	1	E4
CPL A	Complement A	1	1	F4
SWAP A	Swap nibbles of A	1	1	C4
RL A	Rotate A left	1	1	23
RLC A	Rotate A left through carry	1	1	33
RR A	Rotate A right	1	1	03
RRC A	Rotate A right through carry	1	1	13

Data Transfer				
MOV A, Rn	MOV register to A	1	1	E8-EF
MOV A, direct	MOV direct byte to A	2	2	E5
MOV A, @Ri	MOV data memory to A	1	1	E6-E7
MOV A, #data	MOV immediate to A	2	2	74
MOV Rn, A	MOV A to register	1	1	F8-FF
MOV Rn, direct	MOV direct byte to register	2	2	A8-AF
MOV Rn, #data	MOV immediate to register	2	2	78-7F
MOV direct, A	MOV A to direct byte	2	2	F5
MOV direct, Rn	MOV register to direct byte	2	2	88-8F
MOV direct, direct	MOV direct byte to direct byte	3	3	85
MOV direct, @Ri	MOV data memory to direct byte	2	2	86-87
MOV direct, #data	MOV immediate to direct byte	3	3	75
MOV @Ri, A	MOV A to data memory	1	1	F6-F7
MOV @Ri, direct	MOV direct byte to data memory	2	2	A6-A7
MOV @Ri, #data	MOV immediate to data memory	2	2	76-77
MOV DPTR, #data	MOV immediate to data pointer	3	3	90
MOVC A, @A+DPTR	MOV code byte relative DPTR to A	1	3	93
MOVC A, @A+PC	MOV code byte relative PC to A	1	3	83
MOVX A, @Ri	MOV external data (A8) to A	1	2-9*	E2-E3
MOVX A, @DPTR	MOV external data (A16) to A	1	2-9*	E0
MOVX @Ri, A	MOV A to external data (A8)	1	2-9*	F2-F3
MOVX @DPTR, A	MOV A to external data (A8)	1	2-9*	F0
PUSH direct	Push direct byte onto stack	2	2	C0
POP direct	Pop direct byte from stack	2	2	D0
XCH A, Rn	Exchange A and register	1	1	C8-CF
XCH A, direct	Exchange A and direct byte	2	2	C5
XCH A, @Ri	Exchange A and data memory	1	1	C6-C7
XCHD A, @Ri	Exchange A and data memory nibble	1	1	D6-D7
*具体的周期数用户可选(请参考 2.4 节的“(6) 弹性存储器周期”)				
Boolean				
CLR C	Clear carry	1	1	C3
CLR bit	Clear direct bit	2	2	C2
SETB C	Set carry	1	1	D3
SETB bit	Set direct bit	2	2	D2
CPL C	Complement carry	1	1	B3
CPL bit	Complement direct bit	2	2	B2
ANL C, bit	AND direct bit to carry	2	2	82
ANL C, /bit	AND direct bit inverse to carry	2	2	B0
ORL C, bit	OR direct bit to carry	2	2	72
ORL C, /bit	OR direct bit inverse to carry	2	2	A0

MOV C, bit	MOV direct bit to carry	2	2	A2
MOV bit, C	MOV carry to direct bit	2	2	92
Branching				
ACALL addr, 11	Absolute call to subroutine	2	3	11-F1
LCALL addr, 16	Long call to subroutine	3	4	12
RET	Return from subroutine	1	4	22
RETI	Return from interrupt	1	4	32
AJMP addr, 11	Absolute jump unconditional	2	3	01-E1
LJMP addr, 16	Long jump unconditional	3	4	02
SJMP rel	Short jump (relative address)	2	3	80
JC rel	Jump on carry =1	2	3	40
JNC rel	Jump on carry =0	2	3	50
JB bit, rel	Jump on direct bit =1	3	4	20
JNB bit, rel	Jump on direct bit =0	3	4	30
JBC bit, rel	Jump on direct bit =1 and clear	3	4	10
JMP @A+DPTR	Jump indirect relative DPTR	1	3	73
JZ rel	Jump on accumulator =0	2	3	60
JNZ rel	Jump on accumulator /=0	2	3	70
CJNE A, direct, rel	Compare A, direct JNE relative	3	4	B5
CJNE A, #d, rel	Compare A, immediate JNE relative	3	4	B4
CJNE Rn, #d, rel	Compare reg, direct JNE relative	3	4	B8-BF
CJNE @Ri, #d, rel	Compare ind, direct JNE relative	3	4	B6-B7
DJNZ Rn, rel	Decrement register. JNZ relative	2	3	D8-DF
DJNE direct, rel	Decrement direct byte. JNZ relative	3	4	D5
Miscellaneous				
NOP	No operation	1	1	00

注：有一个额外的保留操作码（A5），执行与NOP相同的功能。

2.4 SFR 寄存器

下面介绍与CPU操作和程序执行相关的SFR。

(1) ACC (累加器 A)

ACC 是累加寄存器，一般用来储存指令执行过程的中间结果，简称为 A。

(2) B 寄存器

B 寄存器主要用于乘、除法运算。对于其它的指令，它可以被视为一个暂存寄存器。

(3) 堆栈指针 (SP)

堆栈指针是个 8 位寄存器。PUSH 和 CALL 指令执行期间，数据被存储前它是递增的。复位后堆栈指针初始化为 07H，这也导致堆栈是从 08H 开始的。但是堆栈可以驻留在 RAM 中的任何地方。

(4) 双数据指针

8051 采用双数据指针来加快数据内存块移动。标准 8051 的数据指针（DPTR）16 位，用于外部数据存储或外设的寻址。8051 在 SFR 的位置 82H 和 83H，保持标准的数据指针 DPTR0。

因而使用 DPTR0 不需要修改代码。

8051 外加第二个数据指针 (DPTR1), SFR 位置为 84H、85H。由 DPTR 选择寄存器 DPS (SFR 86h) 内的 SEL bit 选择使用的指针。当 SEL = 0, 用 DPTR 指令将使用 DPL0 和 DPH0。当 SEL = 1, 用 DPTR 指令将使用 DPL1 和 DPH1。SEL 是 SFR 86H 的第 0 bit。SFR 86H 的其它 bit 没使用。

数据指针 (DPTR) 由一个高字节和一个低字节组成, 占据一个 16 位的地址。它可以作为一个 16 位的寄存器使用, 也可作为两个独立的 8 位寄存器使用。

双数据指针的 SFR 位置为:

- 82h: DPL0 - DPTR0 低字节
- 83h: DPH0 - DPTR0 高字节
- 84h: DPL1 - DPTR1 低字节
- 85h: DPH1 - DPTR1 高字节
- 86h: DPS - DPTR 选择 (LSB)

名称	地址	7	6	5	4	3	2	1	0
DPS	86H	0	0	0	0	0	0	0	SEL
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R	R	R	R/W

简称	位置	功能
SEL	Bit0	数据指针选择位。 SEL =0: 使用 DPTR 的指令使用 DPL0 和 DPH0。 SEL =1: 使用 DPTR 的指令使用 DPL1 和 DPH1。

(5) 程序状态字 (PSW)

程序状态字 (PSW) 包含反映 CPU 当前状态的多个状态位。

名称	地址	7	6	5	4	3	2	1	0
PSW	D0H	CY	AC	F0	RS1	RS0	OV	F1	P
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R/W	R/W	R/W	R	R/W	R

简称	位置	功能
CY	Bit7	进位标志位。当最后一个算术运算导致进位 (在加法) 或借位 (在减法) 时, 设置为 1; 否则清零。
AC	Bit6	辅助进位标志位。当最后一个算术运算发生低四位向高四位进位 (在加法) 或借位 (在减法) 时, 设置为 1; 否则清零。
F0	Bit5	用户标志位 0。位寻址。软件控制的通用标志位。该位为用户定义的状态标志, 用户可根据需要用软件对其置位或清零, 也可以用软件测试 F0 的状态来实现分支转移。

RS1, RS0	Bit4, Bit3	工作寄存器组选择控制位，定义如下 RS1 RS0 寄存器组选择 0 0: 寄存器组 0, 地址 00h-07h 0 1: 寄存器组 1, 地址 08h-0Fh 1 0: 寄存器组 2, 地址 10h-17h 1 1: 寄存器组 3, 地址 18h-1Fh
OV	Bit2	溢出标志位。当最后的算术运算导致进位（加）、借位（减法）或溢出（乘或除）时，设置为 1，否则清零。
F1	Bit1	用户标志位 1。位寻址。软件控制的通用标志位。
P	Bit0	奇偶标志位。若累加器 A 中 1 的个数为奇数，设置为 1，否则为 0。

(6) 弹性存储器周期

弹性存储器周期实现了用户通过应用软件来调整数据存储器的访问速度。8051可以在短短的一两个指令周期执行MOVX指令，然而，有时需要控制调整这个速度，例如，访问慢速存储器或慢速外设（如串行接口）。

时钟控制寄存器CKCON（SFR位置8EH）的三个最低位控制这个弹性值。你可以使用0到7之间的控制数值。数值0增加零个指令周期，即MOVX执行需要两个指令周期。数值7增加了七个指令周期，即MOVX执行需要九个指令周期。在程序控制下，弹性值可以动态变化。

默认情况下，弹性值设置为1（3周期MOVX）。全速数据存储器存取时，软件必须设置弹性值为0。弹性值只会影响数据的访问。减小程序存储器（ROM）访问速度的唯一途径是使用较慢的时钟。

弹性值影响读/写使能和所有相关的时间宽度。使用较大弹性值将使读/写使能信号更宽，允许存储器或外设有更多的时间来响应。

表2.3列出了弹性值0~7所对应的数据存储器的访问速度。MD2 ~ MD0是时钟控制寄存器的低三位(CKCON.2~0)。

表 2.3: 数据存储器弹性值

MD2	MD1	MD0	指令周期	读/写使能宽度(Clocks)
0	0	0	2	2
0	0	1	3(default)	4
0	1	0	4	8
0	1	1	5	12
1	0	0	6	16
1	0	1	7	20
1	1	0	8	24
1	1	1	9	28

3. 存储器结构

3.1 简介

ME9101包含三个不同功能的存储器:

- 4K 字节的程序存储器
- 128字节的内部数据存储器
- 128字节的特殊功能寄存器

图3.1所示为内存地址空间。

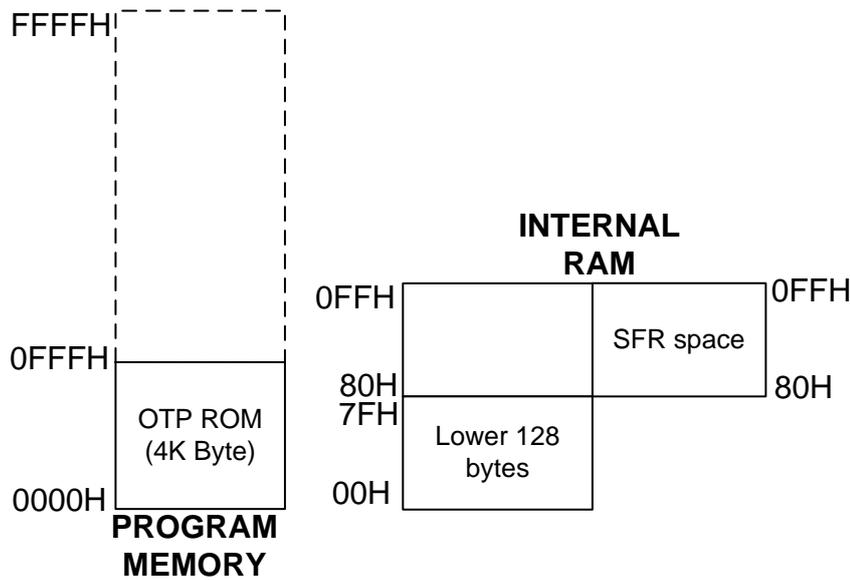


图 3.1 内存地址空间

程序存储器区域为0000H~0FFFH。

作为数据存储器，内部RAM的地址与XRAM低位地址有重叠，但它们是通过不同的指令类型来访问的。特殊功能寄存器（SFR）的地址范围为80HH~0FFH，它是直接寻址方式寻址。

3.2 程序存储器

ME9101内含的4K字节程序存储器是一次性可编程存储器（OTPROM）。

3.3 内部 RAM

内部 RAM 如图 3.2 所示，由下列两部分组成：

- (1) 128 字节的 IRAM 地址为 00H ~ 7FH，可通过直接或间接寻址访问。
- (2) 128 字节的特殊功能寄存器（SFR）地址为 80H ~ FFH，只能通过直接寻址访问。

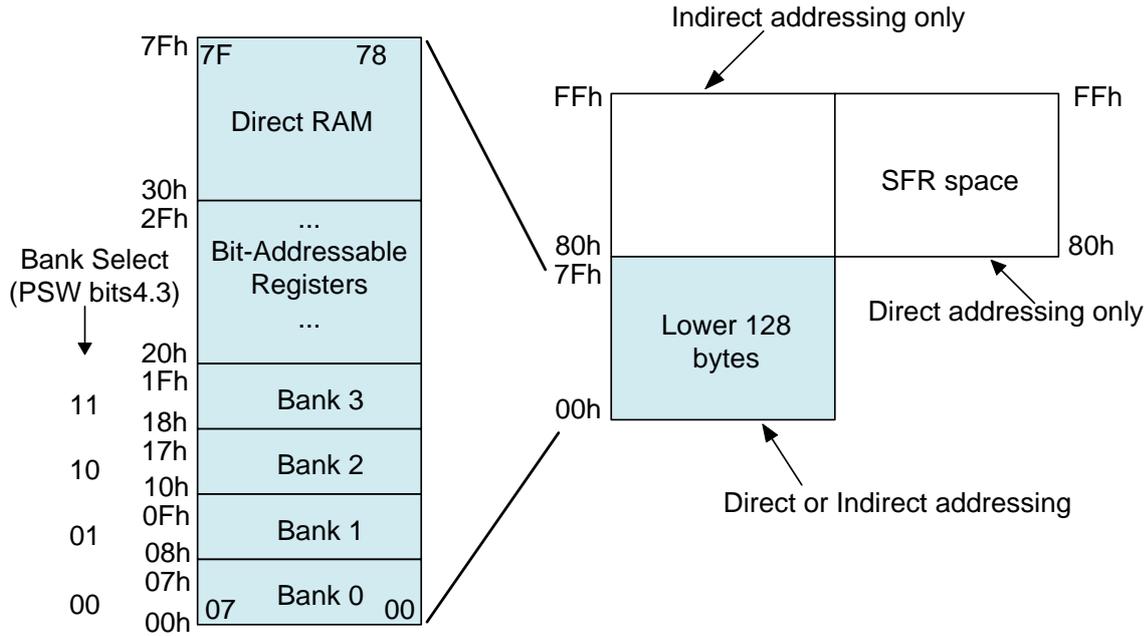


图3.2: 内部RAM结构

内部RAM的128字节组成如图3.2所示。低端的32个字节组成5个八位寄存器（R0-R7）的寄存器组，由程序状态字PSW中的2 bit选择哪个寄存器组在使用，这样可更有效地使用代码空间，因为寄存器指令比使用直接寻址的指令更短。接下来的十六个字节20H-2FH组成一块能够位寻址的存储空间。所有的低128字节的都可以在iram_bus总线上直接或间接寻址。

特殊功能寄存器SFR的地址范围为80H ~ FFh，在sfr_bus总线上直接寻址方式访问。以0H、8H结束的SFR地址是位寻址的。

4. 定时器

4.1 概述

ME9101 包含两个 16 位定时/计数器(Timer 0 和 Timer 1)。

Timer 0 和 Timer 1 有四种工作模式：

- 模式0: 13bit 计数器
- 模式1: 16bit 计数器
- 模式2: 8bit 自动加载计数器
- 模式3: 两个8bit 计数器（只有Timer 0可设置此模式）

Timer 0 和 Timer 1 的主要功能为：

- 定时器功能：时钟由系统MCU时钟提供。
- PWM功能：在P5.0端口产生周期和占空比可调的脉冲调制波形。
- 蜂鸣器BUZZER功能：在端口P5.0和P5.1产生周期可调、占空比固定的波形。

4.2 结构框图

图4.1所示为定时器的结构框图。

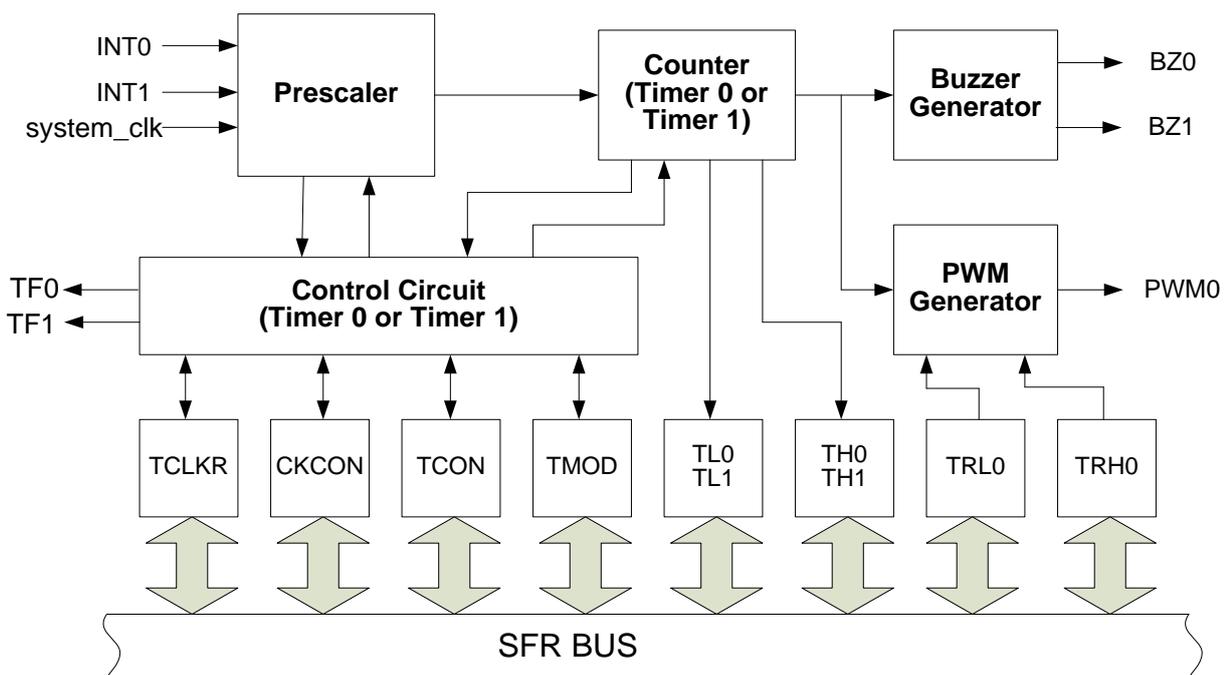


图 4.1: 定时器的结构框图

4.3 控制寄存器

定时器模块内的控制寄存器说明如下：

4.3.1 定时器控制寄存器 (TCON)

名称	地址	7	6	5	4	3	2	1	0
TCON	88H	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W							

简称	位置	功能
TF1	Bit7	定时器 1 溢出标志位。定时器 1 溢出时，设置为“1”，中断处理后清除。 TF1 =0: 定时器 1 无溢出。 TF1 =1: 定时器 1 溢出。
TR1	Bit6	定时器 1 运行控制位。设置为“1”使定时器 1 开始计数。 TR1 =0: 定时器 1 停止工作。 TR1 =1: 定时器 1 启动。
TF0	Bit5	定时器 0 溢出标志位。定时器 0 溢出时，设置为“1”，中断处理后清除。 TF0 =0: 定时器 0 无溢出。 TF0 =1: 定时器 0 溢出。
TR0	Bit4	定时器 0 运行控制位。设置为“1”使定时器 0 开始计数。 TR0 =0: 定时器 0 停止工作。 TR0 =1: 定时器 0 启动。
IE1	Bit3	外部中断 1 检测。 如果外部中断 1 被配置为沿敏感 (IT1 = 1)，当 int1_n 引脚检测到下降沿时，IE1 设置为“1”。当 CPU 执行对应的中断服务程序时，IE1 自动清除。在沿敏感模式，IE1 也可通过软件清除。 如果外部中断 1 被配置为电平敏感 (IT1 = 0)，当 int1_n 引脚检测到低电平时，IE1 设置为“1”。当 int1_n 引脚检测到高电平时，IE1 自动清除。在电平敏感模式，软件无法写 IE1。
IT1	Bit2	外部中断 1 类型选择。 当 IT1 = 1，8051 在下降沿检测 int1_n (沿敏感)。 当 IT1 = 0，8051 低电平检测 int1_n (电平敏感)。
IE0	Bit1	外部中断 0 检测。 如果外部中断 0 被配置为沿敏感 (IT0 = 1)，当 int0_n 引脚检测到下降沿时，IE0 设置为“1”。当 CPU 执行对应的中断服务程序时，IE0 自动清除。在沿敏感模式，IE0 也可通过软件清除。 如果外部中断 0 被配置为电平敏感 (IT0 = 0)，当 int0_n 引脚检测到低电平时，IE0 设置为“1”。当 int0_n 引脚检测到高电平时，IE0 自动清除。在电平敏感模式，软件无法写 IE0。
IT0	Bit0	外部中断 0 类型选择。 当 IT0 = 1，8051 在下降沿检测 int0_n (沿敏感)。 当 IT0 = 0，8051 低电平检测 int0_n (电平敏感)。

4.3.2 定时器模式寄存器 (TMOD)

名称	地址	7	6	5	4	3	2	1	0
TMOD	89H	0	0	M11	M10	0	0	M01	M00
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R/W	R/W	R	R	R/W	R/W

简称	位置	功能
M11, M10	Bit5, Bit4	定时器 1 模式选择位。 M11 M10 0 0: 模式 0: 13bit 计数器 0 1: 模式 1: 16bit 计数器 1 0: 模式 2: 8bit 自动加载计数器 1 1: 模式 3: 定时器 1 停止
M01, M00	Bit1, Bit0	定时器 0 模式选择位。 M01 M00 0 0: 模式 0: 13bit 计数器 0 1: 模式 1: 16bit 计数器 1 0: 模式 2: 8bit 自动加载计数器 1 1: 模式 3: 两个 8bit 计数器

4.3.3 定时器 0 计数器的低字节 (TL0)

名称	地址	7	6	5	4	3	2	1	0
TL0	8AH	TL07	TL06	TL05	TL04	TL03	TL02	TL01	TL00
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W							

简称	位置	功能
TL07 ~TL00	Bit7~Bit0	定时器 0 的计数值

4.3.4 定时器 0 计数器的高字节/ 加载寄存器 (TH0)

名称	地址	7	6	5	4	3	2	1	0
TH0	8CH	TH07	TH06	TH05	TH04	TH03	TH02	TH01	TH00
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W							

简称	位置	功能
TH07 ~	Bit7~Bit0	模式 2, 通过写这个寄存器, 设置定时器 0 的加载值。

TH00	模式 0 / 1 / 3, 定时器 0 的计数值
------	--------------------------

4.3.5 定时器 1 计数器的低字节(TL1)

名称	地址	7	6	5	4	3	2	1	0
TL1	8BH	TL17	TL16	TL15	TL14	TL13	TL12	TL11	TL10
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W							

简称	位置	功能
TL17 ~ TL10	Bit7~Bit0	模式 0/1/2, 定时器 1 的计数值 模式 3, 没有使用。

4.3.6 定时器 1 计数器的高字节/ 加载寄存器(TH1)

名称	地址	7	6	5	4	3	2	1	0
TH1	8DH	TH17	TH16	TH15	TH14	TH13	TH12	TH11	TH10
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W							

简称	位置	功能
TH17 ~ TH10	Bit7~Bit0	模式 2, 通过写这个寄存器, 设置定时器 1 的加载值。 模式 0 / 1, 定时器 1 的计数值 模式 3, 没有使用。

4.3.7 时钟控制寄存器 (CKCON)

与标准的8051相同, ME9101的定时器默认的时钟是每个增量为12个时钟周期。使用默认的速率(每个定时器增量为12个时钟), 实时的应用程序代码可正常运行, 如波特率。然而, 8051核的指令周期是4个时钟周期。

因此, 为了满足快速定时应用的需要, 设置时钟控制寄存器(CKCON)的控制位可选择定时器每个增量为4个时钟周期。

名称	地址	7	6	5	4	3	2	1	0
CKCON	8EH	0	0	0	T1M	T0M	MD2	MD1	MD0
Reset Values:		0	0	0	0	0	0	0	1
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
T1M	Bit4	定时器 1 时钟选择位。 T1M = 0: 定时器 1 使用与 8051 的兼容性的 mcuclk/12。 T1M = 1: 定时器 1 使用 mcuclk/4。
T0M	Bit3	定时器 0 时钟选择位。

		TOM = 0: 定时器 0 使用与 8051 的兼容性的 mcuclk/12。 TOM = 1: 定时器 0 使用 mcuclk/4。
MD2~MD0	Bit2~Bit0	控制用于外部 MOVX 指令的周期数。详细请参照第二章有关内容。

4.3.8 PWM0 占空比配置低字节 (PWM0L)

名称	地址	7	6	5	4	3	2	1	0
PWM0L	B1H	PWM0L7	PWM0L6	PWM0L5	PWM0L4	PWM0L3	PWM0L2	PWM0L1	PWM0L0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W							

简称	位置	功能
PWM0L7 ~ PWM0L0	Bit7~Bit0	PWM0 占空比配置寄存器的低字节。

4.3.9 PWM0 占空比配置高字节 (PWM0H)

名称	地址	7	6	5	4	3	2	1	0
PWM0H	B2H	PWM0H7	PWM0H6	PWM0H5	PWM0H4	PWM0H3	PWM0H2	PWM0H1	PWM0H0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W							

简称	位置	功能
PWM0H7 ~ PWM0H0	Bit7~Bit0	PWM0 占空比配置寄存器的高字节。

4.4 电路说明

4.4.1 模式 0: 定时器 0/1

模式0对于定时器0和定时器1而言是相同的。

模式0时，定时器的低位字节TL0或TL1会从0计数到31，当从31继续递增计数时，上述寄存器被清零，且高位字节TH0或TH1加1。在这一模式下，定时器仅使用了13位，即TLx (x=0~1) 的低5位 (Bit0~Bit4) 和THx (x=0~1) 的全部8位。

模式0时，TL0或TL1的高三位是不确定的，当使用软件读取该寄存器的值时，这三位必须被屏蔽。

在寄存器TCON中，通过配置控制位TR0/TR1 (TCON.4/TCON.6)，用户可以开启定时器。

当13位定时器从1FFFh继续递增计数时，计数器变为全零，定时器计时溢出，TCON 中的TF0 (TCON.5) 或者TF1 (TCON.7) 置位，并且向CPU提起中断。

4.4.2 模式 1: 定时器 0/1

定时器0和定时器1的模式1工作是相同的。

模式1时，Timer0/1被配置成16位计数器。因为定时器的低位字节 (TL0和TL1) 的全部8位都有效，所以TLx (x=0~1) 从0开始递增计数到255。当TLx (x=0~1) 从255继续递增计数时，定时

器的低位字节会被清零，而高位字节（TH0和TH1）会加1。定时器从FFFFh继续递增时，定时器计时溢出，THx和TLx（x=0~1）全部清零。其他情况，模式1的操作与模式0相同。

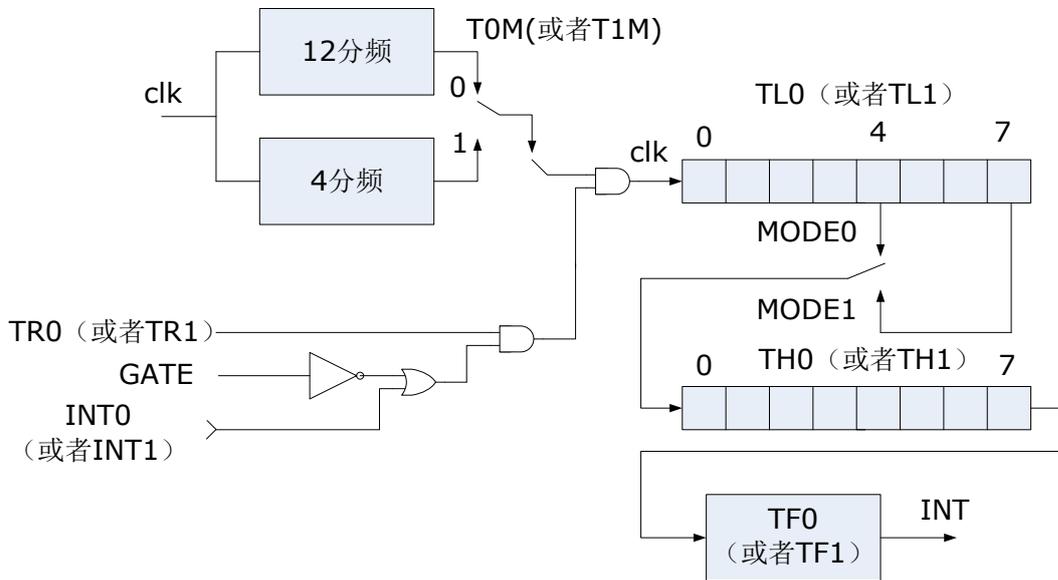


图 4.2: 模式0/1实现框图

4.4.3 模式 2: 定时器 0/1

定时器0和定时器1的模式2工作是相同的。

模式2时，Timer0/1被配置成可以加载初值的8位定时器。其中，定时器的低位字节TL0或TL1是定时器计数值，而高位字节TH0或TH1保存的是加载初值。

如下图所示，模式2的控制部分与模式0和模式1基本相同。只是模式2时，当TLx（x=0~1）从255继续递增计数时，TLx的值会重置为THx（x=0~1）中保存的初值，然后，TLx从该初值开始重新计数，而THx值保持不变。

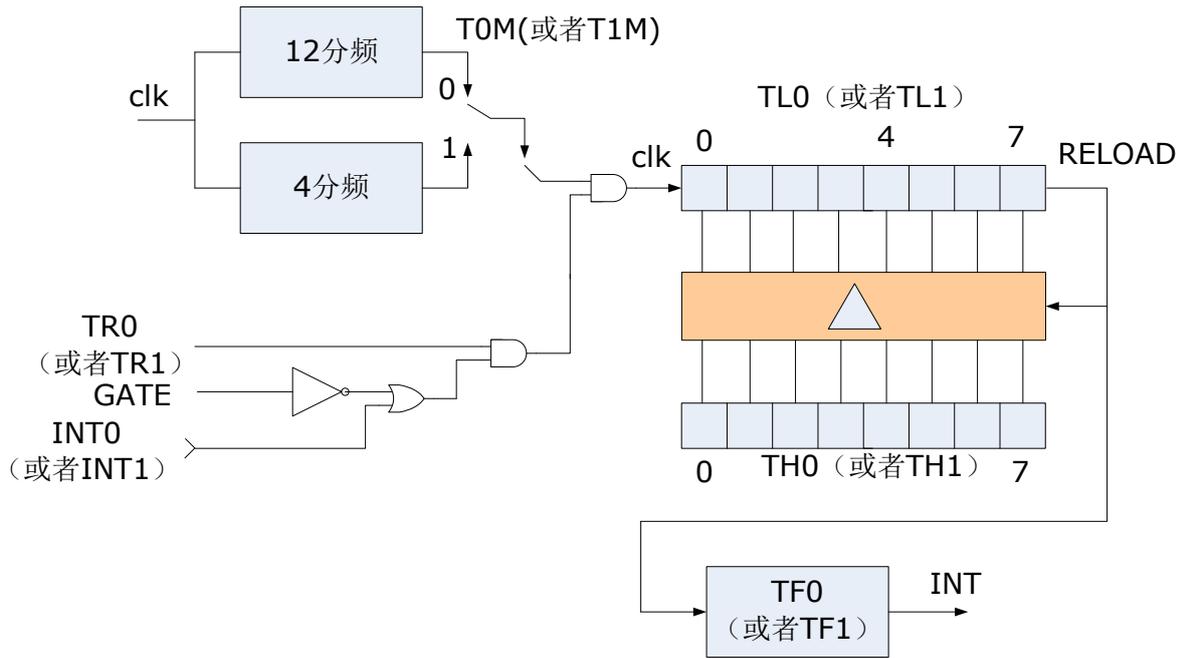


图 4.3: 模式 2 实现框图

4.4.4 模式 3: 定时器 0

模式3时，定时器0的高位字节与低位字节分别作两个独立的8位定时器使用。定时器1停止计数，TL1 SFR和TH1 SFR保持原有的值不变。

当定时器0工作于模式3时，TL0通过正常定时器0的控制位配置成8位计数器。此时，用户仍可通过配置T1M0（TMOD.4）和T1M1（TMOD.5）使定时器1工作于模式0/1/2下，但此时，定时器1不能向CPU提起溢出中断，因为TF1位为定时器0所用，如下图所示。

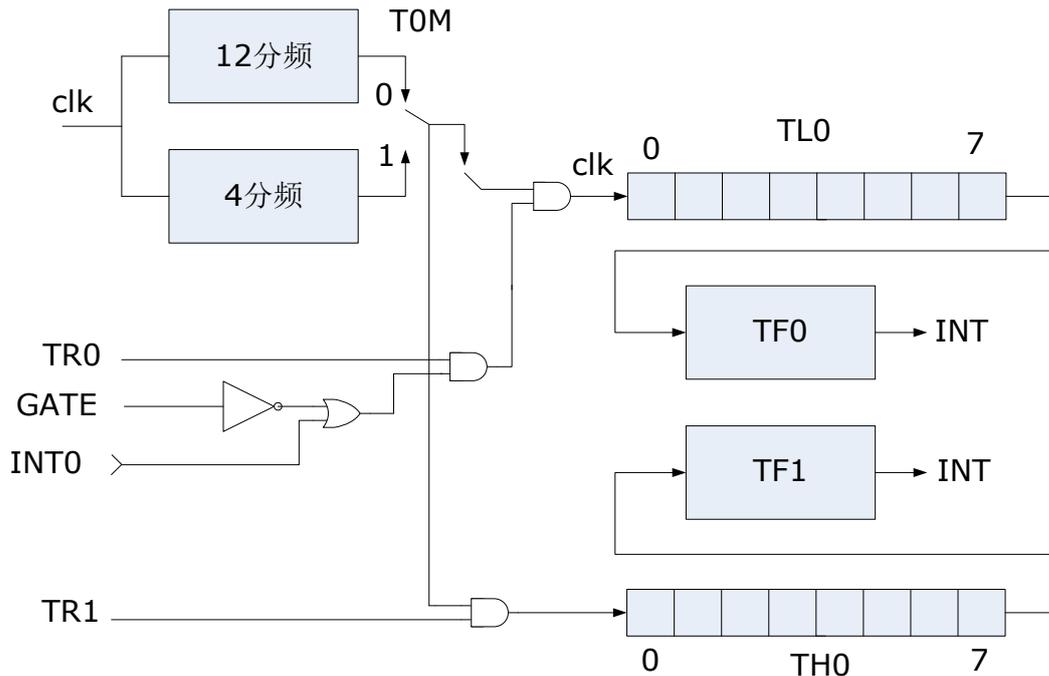


图 4.4: 定时器 0 的模式 3 实现框图

4.4.5 PWM

可编程控制占空比/周期的PWM 可以提供不同的PWM 信号。周期可通过调整MCU的时钟分频值以及驱动TIMER的时钟分频寄存器CKCON来决定；占空比由占空比配置寄存器PWM0H、PWM0L决定。

使能定时器（TR0）且设置P5端口功能选择配置寄存器P5FUNCFGR为PWM 输出，引脚开始输出PWM 信号。PWM 首先输出高电平，然后输出低电平。使能定时器时，设置定时器计数器的初始值为0。当计数值等于PWM0H、PWM0L的设置值时，PWM 输出低电平；计数器溢出时（计数值从0FFH 到00H），整个PWM 周期完成，并进入下一个周期，PWM 再次输出高电平。在PWM 输出的过程中，由程序更改PWM 的周期，则在本周期开始输出新占空比的PWM 信号。

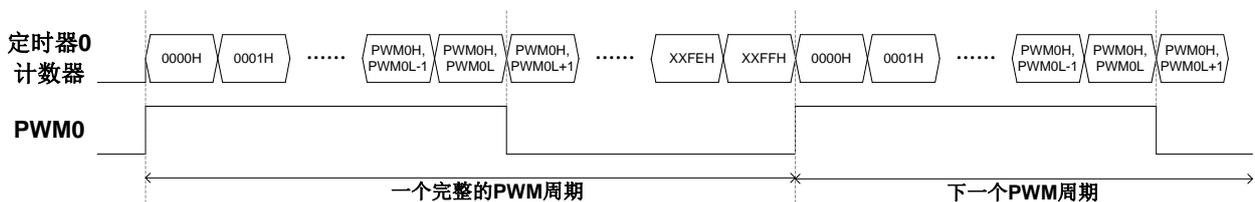


Figure 4.5: PWM 输出示意图

PWM输出引脚与GPIO 引脚共用，通过P5端口功能选择配置寄存器P5FUNCFGR设置PWM 输出。

禁止PWM输出后，该引脚自动返回到最后一个GPIO 模式。

4.4.6 BUZZER

Buzzer是在计数器发生溢出时将输出反转，其周期是计数器溢出周期的2倍。设置TIMER0/1

的溢出周期即可设置Buzzer的周期。

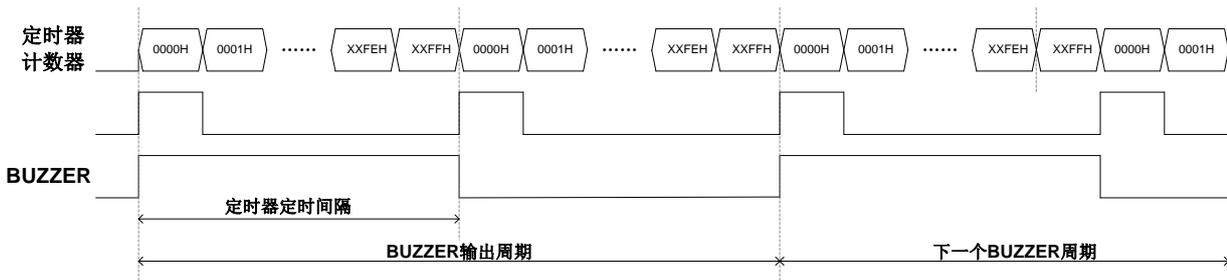


Figure 4.6: Buzzer 输出示意图

Buzzer输出引脚与GPIO 引脚共用，通过P5端口功能选择配置寄存器P5FUNCFGR设置 Buzzer 输出。

禁止Buzzer 输出后，该引脚自动返回到最后一个GPIO 模式。

5. 看门狗定时器 WDT

5.1 概述

由于电磁干扰等外部不可控事件影响的存在，可能造成CPU执行程序的不正常。为增强系统可靠性，芯片应能监测到这一异常事件的发生，并有手段恢复到正常状态。因此在ME9101中内置了看门狗定时器。正常执行的程序需要定时的清除看门狗计数，若未定时清除（有可能是由于外部环境导致的程序执行异常），看门狗定时器会给出复位信号，MCU将恢复到上电状态，重新从头开始执行程序。

5.2 结构框图

图5.1所示为WDT的结构框图。

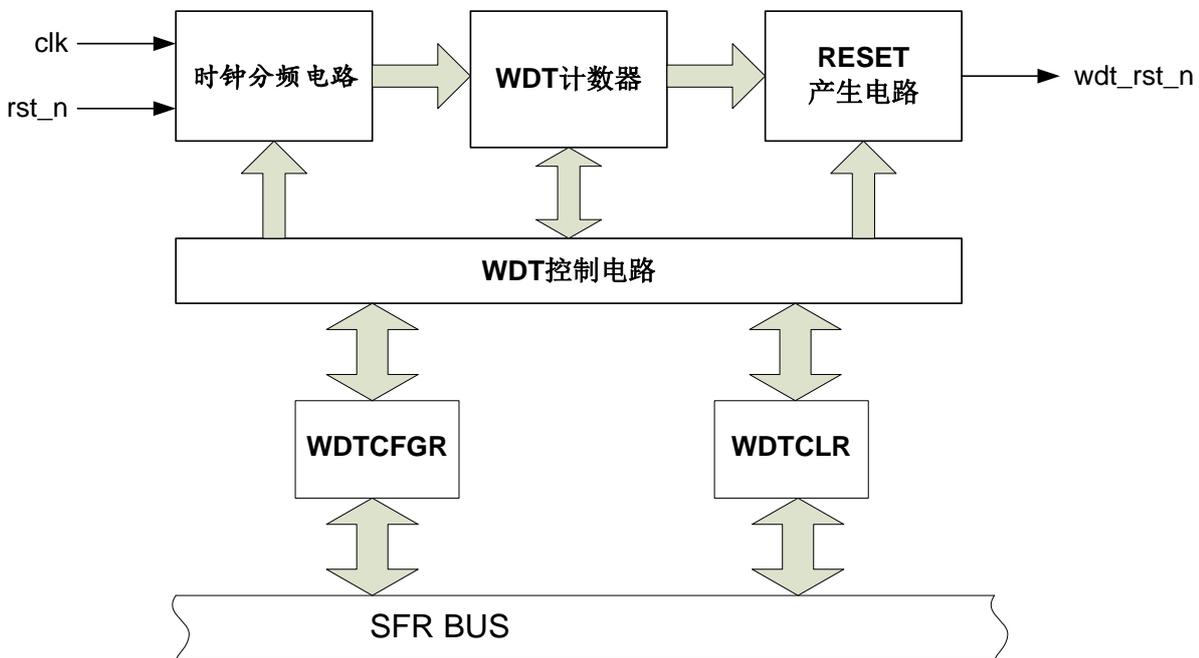


Figure 5.1: WDT 的结构框图

5.3 控制寄存器

WDT定时器模块内的控制寄存器说明如下：

(a) WDT 配置寄存器 WDTCFGR

名称	地址	7	6	5	4	3	2	1	0
WDTCFGR	9EH	0	0	0	0	WDTDIV2	WDTDIV1	WDTDIV0	WDTGT
Reset Values:		0	0	0	0	1	0	0	0
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
WDTDIV2 ~	Bit3~Bit1	配置 WDT 时钟的工作频率。

WDTDIV0		WDTDIV[2 ~0]=000: 不分频; WDTDIV[2 ~0]=001: 2 分频; WDTDIV[2 ~0]=010: 4 分频; WDTDIV[2 ~0]=011: 8 分频。 WDTDIV[2 ~0]=100: 16 分频。 WDTDIV[2 ~0]=101: 32 分频。 WDTDIV[2 ~0]=110: 64 分频。 WDTDIV[2 ~0]=111: 128 分频。
WDTGT	Bit0	门控 WDT 时钟源。 WDTGT =0: WDT 时钟被门控。 WDTGT =1: WDT 时钟未被门控。 注: 当工作模式切换为空闲模式时, WDT 时钟自动被门控。

(b) WDT 清零寄存器 WDTCLR

名称	地址	7	6	5	4	3	2	1	0
WDTCLR	9FH	WDTR7	WDTR6	WDTR5	WDTR4	WDTR3	WDTR2	WDTR1	WDTR0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		W	W	W	W	W	W	W	W

简称	位置	功能
WDTR7 ~ WDTR0	Bit7~Bit0	看门狗计数器清零寄存器。只有当 WDTR 写入清零控制字“5AH”时, 看门狗计数器清除为 0。

5.4 WDT 的操作

WDT使能时 (WDTGT =1), 计数器将递增, 直到溢出, 或称“超时”, 这时WDT将产生复位信号wdt_rst_n, 强制芯片复位。

这也就意味着正常工作期间, 用户需在应用程序中, 在看门狗溢出前有策略地清除看门狗定时器, 以防止其产生复位。这样的话, 无论什么原因, 程序失常跳转到一个未知的地址或进入一个死循环, 这些清除指令都不能被正确执行, 此种情况下, 看门狗将溢出以使单片机复位。

实现看门狗清除的方法是: 写数据“5AH”到SFR寄存器WDTCLR,看门狗定时器将清零。

6. 模数转换器

6.1 概述

对于大多数的电子系统而言，处理真实世界的模拟信号是共同的需求。为了完全由单片机来处理这些信号，首先必须通过A/D 转换器将模拟信号转换成数字信号。将A/D 转换器电路集成入单片机，可有效的减少外部器件，随之而来，具有低成本和减少器件空间需求的优势。系统包含了6个模拟通道的SAR A/D 转换器，它们可以直接接入外部模拟信号（来自传感器或其它控制信号）并直接将这此信号转换成12 位的数字量。

6.2 结构框图

ADC的结构框图如图6.1所示。

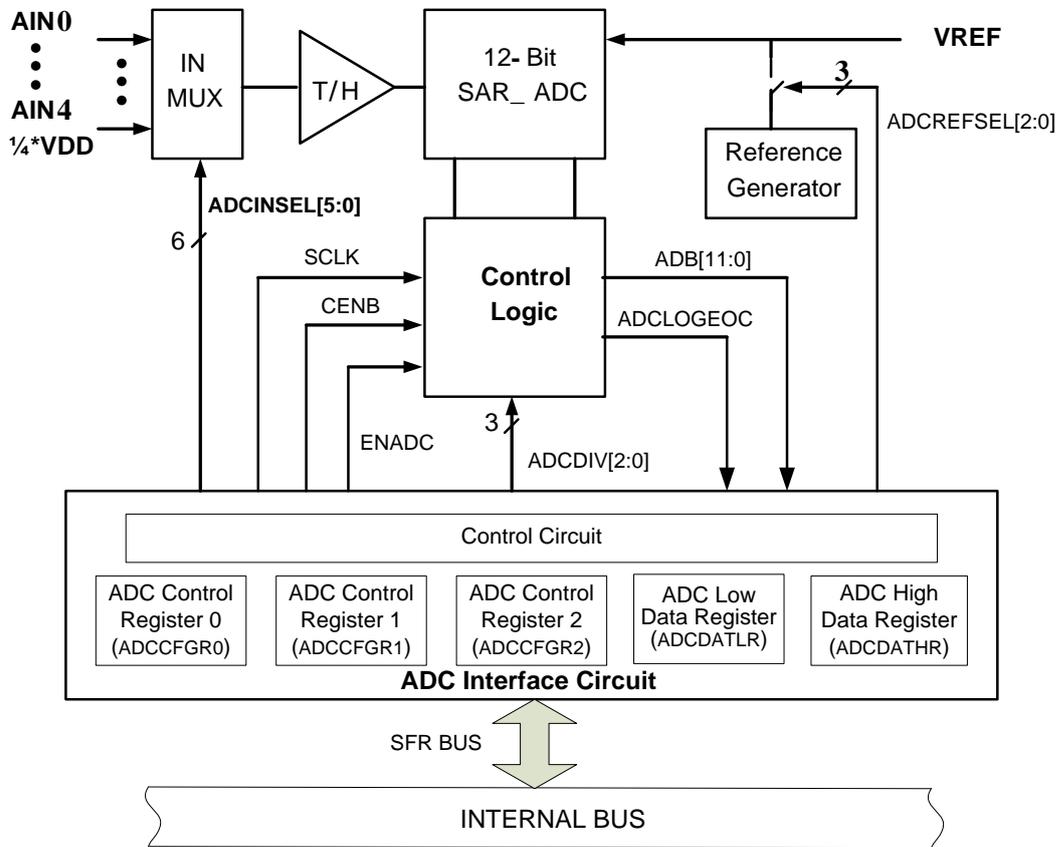


图6.1: ADC的结构框图

ADC是通过ADCCFGR0寄存器的ADCINSEL [5:0]来选择模拟信号输入引脚:(AINx(x=0~4)) 引脚)以及内部1/4*VDD 电压源，使模拟信号输入至SAR ADC。ADC的输入引脚必须由程序设为不带上拉电阻的输入引脚。CHS 位使能全部ADC通道。

ADC 的分辨率为12 位，可以通过ADCDIV [2:0]位选择ADC 的转换速率以决定ADC的转换时间。ADC 参考电压的高电平包括2 种：包括AVDD、4V、3V、2V (ADCREFSSEL2=0) 的内部参考源和由P4.0 提供的外部参考源 (ADCREFSSEL2=1) 。

设置好ENADC、ADCCENB 和ADCGT位后，ADC 开始转换，转换结束时，ADC 电路将

ADCFLGEOC 和ADC_INT（外部中断标志寄存器EXIF）置1，并将转换结果存入ADCDATLR和ADCDATHR寄存器中。若EADI=1，ADC 请求中断，AD 转换完成后，ADC_INT=1 时，程序计数器跳转中断向量地址（ORG 0053H）执行中断服务程序。

6.3 控制寄存器

ADC模块由下列所示寄存器控制：

6.3.1 ADC 输入通道选择寄存器 ADCCFGR0

名称	地址	7	6	5	4	3	2	1	0
ADCCFGR0	B3H	0	0	ADCINS EL5	ADCINS EL4	ADCINS EL3	ADCINS EL2	ADCINS EL1	ADCINS EL0
Reset Values:		0	0	0	0	0	0	0	1
User R/W:		R	R	R/W	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
ADCINSEL [5:0]	Bit2~Bit0	ADC 输入通道选择位。 ADCINSEL [5:0]=000001：选择 ADC 的输入为 AIN0； ADCINSEL [5:0]=000010：选择 ADC 的输入为 AIN1； ADCINSEL [5:0]=000100：选择 ADC 的输入为 AIN2； ADCINSEL [5:0]=001000：选择 ADC 的输入为 AIN3； ADCINSEL [5:0]=010000：选择 ADC 的输入为 AIN4； ADCINSEL [5:0]= 100000：选择内部 1/4*VDD 电压源。

6.3.2 ADC 系统控制寄存器 ADCCFGR1

名称	地址	7	6	5	4	3	2	1	0
ADCCFGR1	B4H	0	0	ADCFL GEOC	ADCRE FSEL2	ADCRE FSEL1	ADCRE FSEL0	ENADC	ADCCE NB
Reset Values:		0	0	0	0	1	1	0	1
Designer R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能									
ADCFLGEOC	Bit5	ADC 状态控制位。读此寄存器自动清除此状态位。 ADCFLGEOC = 0：ADC 转换进行中。 ADCFLGEOC = 1：ADC 转换结束，数据有效，可以读取数据。									
ADCREFSEL2	Bit4	ADC 内部参考电压控制位。 ADCREFSEL2 =0：允许 ADC 内部 VREFH 功能。 ADCREFSEL2 = 1：禁止 ADC 内部 VREFH 功能，VREFH 从 PAD4.0 输入。									
ADCREFSEL [1:0]	Bit3, Bit2	ADC 内部参考电压选择位。 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>ADCREFSEL1</th> <th>ADCREFSEL0</th> <th>内部 VREFH 电压</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>2.0V</td> </tr> <tr> <td>0</td> <td>1</td> <td>3.0V</td> </tr> </tbody> </table>	ADCREFSEL1	ADCREFSEL0	内部 VREFH 电压	0	0	2.0V	0	1	3.0V
ADCREFSEL1	ADCREFSEL0	内部 VREFH 电压									
0	0	2.0V									
0	1	3.0V									

		1	0	4.0V
		1	1	VDD
ENADC	Bit1	ADC 使能控制位。睡眠模式下，禁止 ADC 以省电。配置为正常工作后，需要一段建立时间之后，ADC 才会工作稳定，所以开启 ADC 使能后，用户需等待一段时间，才能去读 ADC 数据寄存器里的数据。 ENADC = 0: ADC 禁止工作 ENADC = 1: ADC 使能工作		
ADCCENB	Bit0	ADC 数字部分启动位。为 0 时开始转换；为 1 时转换停止。		

6.3.3 ADC 时钟控制寄存器 ADCCFGR2

名称	地址	7	6	5	4	3	2	1	0
ADCCFGR2	B5H	0	0	0	0	ADCDIV 2	ADCDIV 1	ADCDIV 0	ADCGT
Reset Values:		0	0	0	0	1	0	0	0
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
ADCDIV [2:0]	Bit3~Bit1	ADC 时钟源选择位。 ADCDIV [2:0]=000: Fcpu。 ADCDIV [2:0]=001: Fcpu/2。 ADCDIV [2:0]=010: Fcpu/4。 ADCDIV [2:0]=011: Fcpu/8。 ADCDIV [2:0]=100: Fcpu/16。 ADCDIV [2:0]=101: Fcpu/32。 ADCDIV [2:0]=110: Fcpu/64。 ADCDIV [2:0]=111: Fcpu/128。
ADCGT	Bit0	ADC 门控时钟信号。 ADCGT=0: 被门控。 ADCGT=1: 未被门控。

6.3.4 ADC 数据寄存器 ADCDATLR/ADCDATHR

完成转换过程后所获得的 12 位的数字量，将储存在高位寄存器 ADCDATHR 和低位寄存器 ADCDATLR 内。在转换过程发生后，CPU 可以直接读取这两个寄存器，以获得数字化的转换值。要注意的是，只有高位寄存器 ADCDATHR 完全利用了 8 位 (bit11~bit4)。而低位寄存器 ADCDATLR 只利用了 8 位中 4 位 (bit3~bit0)，它包含的只是 12 位转换值中低的四位。

8 位 ADC 模式下，ADC 数据存放于 ADCDATHR 寄存器中。12 位 ADC 模式下，ADC 数据存放于 ADCDATHR 和 ADCDATLR 寄存器中。

名称	地址	7	6	5	4	3	2	1	0
ADCDATHR	B7H	ADB11	ADB10	ADB9	ADB8	ADB7	ADB6	ADB5	ADB4
Reset Values:		X	X	X	X	X	X	X	X
User R/W:		R	R	R	R	R	R	R	R

简称	位置	功能
ADB[11:4]	Bit7~Bit0	ADC 12 位分辨率的高字节数据寄存器或 8 位分辨率的数据寄存器。

名称	地址	7	6	5	4	3	2	1	0
ADCDATLR	B6H	0	0	0	0	ADB3	ADB2	ADB1	ADB0
Reset Values:		0	0	0	0	X	X	X	X
User R/W:		R	R	R	R	R	R	R	R

简称	位置	功能
ADB[3:0]	Bit3~Bit0	ADC 12 位分辨率的低字节数据寄存器。

6.4 ADC 操作说明和注意事项

6.4.1 ADC 信号格式

ADC 采样电压范围为参考电压高/低电平之间，ADC 参考低电压为VSS，高电压包括VDD/4V/3V/2V，外部参考电压由P4.0/AVREFH 引脚提供（由ADCREFSSEL2 控制）。ADCREFSSEL2 = 0 时，ADC 参考电压选择内部参考源；ADCREFSSEL2 = 1 时，ADC 参考电压选择外部参考源（P4.0/AVREFH）。ADC 参考电压的范围为：（ADC 参考高电压-ADC 参考低电压） \geq 2V，ADC 参考低电压为VSS=0V，故ADC 参考高电压范围为2V~VDD，外部参考电压需在此范围之内。

- ADC 内部参考低电压=0V。
- ADC 内部端口低电压=VDD/4V/3V/2V。（ADCREFSSEL2 = 0）
- ADC 外部参考电压=2V~VDD。（ADCREFSSEL2 = 1）

ADC 采样输入信号电压必须在ADC 参考低电压和ADC 参考高电压之间，若ADC 输入信号的电压不在此范围内，则ADC 的转换结果会出错（满量程或者为0）。

- ADC 参考低电压 \leq ADC 采用输入信号电压 \leq ADC 参考高电压

6.4.2 ADC 启动

ADC启动前，应先设置ADCGT=1、ADCCENB=1，之后，设置ENADC =1。

6.4.3 ADC 转换时间

ADC 转换时间是指从ENADC =1（开始ADC）到ADCFLGEOC=1（ADC 结束）所用的时间，由ADC 分辨率和ADC 时钟速率决定，转换时间为（1/ADC 时钟*13）S。ADC 的时钟源为Fcpu，包括Fcpu/1、Fcpu/2、Fcpu/8、Fcpu/16、Fcpu/32、Fcpu/64、Fcpu/128，由ADCDIV [2:0]位控制。

ADC 的转换时间会影响ADC 的性能，如果输入高Rate 的模拟信号，必须要选择一个高Rate

的ADC 转换Rate。如果ADC 的转换时间比模拟信号的转换Rate 慢，则ADC 的结果出错。故选择合适的ADC 时钟速率和ADC 分辨率才能得到合适的ADC 转换Rate。

6.4.4 ADC 引脚配置

ADC输入引脚与P4口共用，ADC 输入通道的选择由ADCINSEL [5:0]控制，ADCINSEL [5:0]=000000 时选择AIN0，ADCINSEL [5:0]=000001 时选择AIN1.....同一时间设置P4 口的一个引脚作为ADC 的输入引脚，该引脚必须设置为输入引脚，禁止内部上拉。通过ADCINSEL [5:0]选择好ADC 输入通道后，GCHS 置1 以使能ADC功能。

- ADC 输入引脚为GPIO 引脚时必须设为输入模式。
- 必须禁止ADC 输入引脚的内部上拉电阻。
- 通过ADCINSEL [5:0]选择好ADC 输入通道。

ADCREFSSEL2 = 1 时，P4.0/AIN0 为ADC 外部参考源的输入引脚，此时，P4.0 必须设为输入模式，并禁止其上拉电阻。

- ADC 外部参考源输入引脚为GPIO 引脚时必须设为输入模式。
- 必须禁止ADC 外部参考源输入引脚的内部上拉电阻。

ADC 输入引脚与普通I/O 引脚共用。当输入一个模拟信号到CMOS 结构端口时，尤其当模拟信号为1/2 VDD 时，可能产生额外的漏电流。当P4 输入多个模拟信号时，也会产生额外的漏电流。睡眠模式下，上述漏电流会严重影响到系统的整体功耗。将ADCINSEL [5:0]的相应位设置为1，其对应的P4 引脚将被设为纯模拟信号输入引脚，从而避免上述漏电流的产生。

7. I/O 输入输出端口

7.1 概述

I/O (Input/Output) 端口是单片机必须具备的最基本外设功能。在ME9101芯片内，有P0、P4，P5共3组通用I/O口。所有I/O都是通用的，称为GPIO (General Purpose Input/Output)。I/O模块包含以下特性：

- 可编程控制IO中断：
 - 边沿触发（上升沿、下降沿）
 - 电平触发（高电平、低电平）
- 输入/输出可承受5V
- 模拟输入
- 可编程控制GPIO管脚配置：
 - 弱上拉电阻
 - 数字输入/输出使能

ME9101的输入/ 输出口控制具有很大的灵活性。大部分引脚可在用户程序控制下被设定为输入或输出。所有引脚的上拉电阻设置以及指定引脚的特定功能设置也都由软件控制，这些特性也使得ME9101在广泛应用上都能符合用户开发的需求。

7.2 I/O 控制寄存器

对每一个GPIO，都由设置输入数据寄存器、输出数据寄存器、上拉电阻配置寄存器、输出使能配置寄存器、功能选择寄存器控制。除电源、地引脚外，每一GPIO和其它功能引脚都是复用的。上电复位、RST输入复位、WDT溢出复位都会将IO控制寄存器复位到默认值。

7.2.1 PO 端口的控制寄存器

(a) P0端口输入数据寄存器P0DINR

名称	地址	7	6	5	4	3	2	1	0
P0DINR	A2H	0	0	0	P0DIN4	P0DIN3	P0DIN2	P0DIN1	P0DIN0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R	R	R	R

简称	位置	功能
P0DIN4 ~ P0DIN0	Bit4~Bit0	P0 端口 (P0.4~P0.0) 的输入数据缓存。当端口设置为 GPIO，输出为高阻态，此寄存器采样 P0 端口输入的数据。

(b) P0端口输出数据寄存器P0DOUR

名称	地址	7	6	5	4	3	2	1	0
P0DOUR	A3H	0	0	0	0	P0DO3	P0DO2	P0DO1	P0DO0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
P0DO3 ~ P0DO0	Bit3~Bit0	P0 端口 (P0.3~P0.0) 的输出数据缓存。

(c) P0端口输出使能寄存器P0OENR

名称	地址	7	6	5	4	3	2	1	0
P0OENR	A4H	0	0	0	0	P0OEN3	P0OEN2	P0OEN1	P0OEN0
Reset Values:		0	0	0	0	1	1	1	1
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
P0OEN3 ~ P0OEN0	Bit3~Bit0	P0 端口 (P0.3~P0.0) 的输出使能控制信号。当输出使能配置为 0 时，端口为输出模式；配置为 1 时，高阻态输出，可从端口输入信号。

(d) P0端口上拉电阻使能寄存器P0PULLUPR

名称	地址	7	6	5	4	3	2	1	0
P0PULLUPR	A5H	0	0	0	0	P0PUP3	P0PUP2	P0PUP1	P0PUP0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
P0PUP3 ~ P0PUP0	Bit3~Bit0	P0 端口 (P0.3~P0.0) 的上拉电阻使能控制信号。当配置为 0 时，使能上拉电阻；配置为 1 时，不使能上拉电阻。复位时使能上拉电阻。

(e) P0端口功能选择配置寄存器P0FUNCFR

名称	地址	7	6	5	4	3	2	1	0
P0FUNCFR	A6H	0	0	0	P0FSEL4	P0FSEL3	P0FSEL2	P0FSEL1	P0FSEL0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
P0FSEL4	Bit4	P0.4 端口功能控制位。若为 0，此端口作为 GPIO 输入使用；若为 1，此端口作为 RESET 端口使用。
P0FSEL3	Bit4	P0.3 端口功能控制位。若为 0，此端口作为 GPIO 使用；若为 1，此端口作为外部中断 int3_n 使用。
P0FSEL2	Bit4	P0.2 端口功能控制位。若为 0，此端口作为 GPIO 使用；若为 1，此端口作为外部中断 int2 使用。
P0FSEL1	Bit4	P0.1 端口功能控制位。若为 0，此端口作为 GPIO 使用；若为 1，此端口作为外部中断 int1_n 使用。
P0FSEL0	Bit0	P0.0 端口功能控制位。若为 0，此端口作为 GPIO 使用；若为 1，此端口作为外部中断 int0_n 使用。

7.2.2 P4 端口的控制寄存器

(a) P4端口输入数据寄存器P4DINR

名称	地址	7	6	5	4	3	2	1	0
P4DINR	A7H	0	0	0	P4DIN4	P4DIN3	P4DIN2	P4DIN1	P4DIN0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R	R	R	R

简称	位置	功能
P4DIN4 ~ P4DIN0	Bit4~Bit0	P4 端口 (P4.4~P4.0) 的输入数据缓存。当端口设置为 GPIO, 输出为高阻态, 此寄存器采样 P4 端口输入的数据。

(b) P4端口输出数据寄存器P4DOUR

名称	地址	7	6	5	4	3	2	1	0
P4DOUR	A9H	0	0	0	P4DO4	P4DO3	P4DO2	P4DO1	P4DO0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
P4DO4 ~ P4DO0	Bit4~Bit0	P4 端口 (P4.4~P4.0) 的输出数据缓存。

(c) P4端口输出使能寄存器P4OENR

名称	地址	7	6	5	4	3	2	1	0
P4OENR	AAH	0	0	0	P4OEN4	P4OEN3	P4OEN2	P4OEN1	P4OEN0
Reset Values:		0	0	0	1	1	1	1	1
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
P4OEN4 ~ P4OEN0	Bit4~Bit0	P4 端口 (P4.4~P4.0) 的输出使能控制信号。当输出使能配置为 0 时, 端口为输出模式; 配置为 1 时, 高阻态输出, 可从端口输入信号。

(d) P4端口上拉电阻使能寄存器P4PULLUPR

名称	地址	7	6	5	4	3	2	1	0
P4PULLUPR	ABH	0	0	0	P4PUP4	P4PUP3	P4PUP2	P4PUP1	P4PUP0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
P4PUP4 ~ P4PUP0	Bit4~Bit0	P4 端口 (P4.4~P4.0) 的上拉电阻使能控制信号。当配置为 0 时, 使能上拉电阻; 配置为 1 时, 不使能上拉电阻。

复位时使能上拉电阻。

7.2.3 P5 端口的控制寄存器

(a) P5端口输入数据寄存器P5DINR

名称	地址	7	6	5	4	3	2	1	0
P5DINR	ACH	0	0	0	0	0	0	P5DIN1	P5DIN0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R	R	R	R

简称	位置	功能
P5DIN1, P5DIN0	Bit1, Bit0	P5 端口 (P5.1, P5.0) 的输入数据缓存。当端口设置为 GPIO, 输出为高阻态, 此寄存器采样 P5 端口输入的数据。

(b) P5端口输出数据寄存器P5DOUR

名称	地址	7	6	5	4	3	2	1	0
P5DOUR	ADH	0	0	0	0	0	0	P5DO1	P5DO0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R	R	R/W	R/W

简称	位置	功能
P5DO1, P5DO0	Bit1, Bit0	P5 端口 (P5.1, P5.0) 的输出数据缓存。

(c) P5端口输出使能寄存器P5OENR

名称	地址	7	6	5	4	3	2	1	0
P5OENR	AEH	0	0	0	0	0	0	P5OEN1	P5OEN0
Reset Values:		0	0	0	0	0	0	1	1
User R/W:		R	R	R	R	R	R	R/W	R/W

简称	位置	功能
P5OEN1, P5OEN0	Bit1, Bit0	P5 端口 (P5.1, P5.0) 的输出使能控制信号。当输出使能配置为 0 时, 端口为输出模式; 配置为 1 时, 高阻态输出, 可从端口输入信号。

(d) P5端口上拉电阻使能寄存器P5PULLUPR

名称	地址	7	6	5	4	3	2	1	0
P5PULLUPR	AFH	0	0	0	0	0	0	P5PUP1	P5PUP0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R	R	R/W	R/W

简称	位置	功能
P5PUP1,	Bit1, Bit0	P5 端口 (P5.1, P5.0) 的上拉电阻使能控制信号。当配置为 0 时,

P5PUP0		使能上拉电阻；配置为 1 时，不使能上拉电阻。 复位时使能上拉电阻。
--------	--	---------------------------------------

(e) P5端口功能选择配置寄存器P5FUNCFR

名称	地址	7	6	5	4	3	2	1	0
P5FUNCFR	BAH	0	0	0	0	P5FSEL3	P5FSEL2	P5FSEL1	P5FSEL0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
P5FSEL3, P5FSEL2	Bit3, Bit2	P5.1 端口功能控制位。 P5FSEL[3:2]=00: 端口作为正常 GPIO 使用。 P5FSEL[3:2]=01: 设置禁止。 P5FSEL[3:2]=10: 端口为 BUZZER 1 输出。 P5FSEL[3:2]=11: 端口作为正常 GPIO 使用。
P5FSEL1, P5FSEL0	Bit1, Bit0	P5.0 端口功能控制位。 P5FSEL[1:0]=00: 端口作为正常 GPIO 使用。 P5FSEL[1:0]=01: 端口为 PWM0 输出。 P5FSEL[1:0]=10: 端口为 BUZZER 0 输出。 P5FSEL[1:0]=11: 端口作为正常 GPIO 使用。

7.3 编程注意事项

在编程时，最先要考虑的是端口的初始化。所有输入/ 输出引脚默认为输入状态，且选择了上拉电阻。

8. 系统时钟

8.1 概述

系统内置双时钟系统：高速时钟和低速时钟。高速时钟包括内部高速时钟和外部高速时钟，低速时钟由内部低速振荡器提供，高、低速时钟都可以作为系统时钟源使用。

- 内部高速时钟：由内部高速振荡器产生，可高达16MHz，定义为HIRC。
- 内部低速时钟：由内部低速振荡器产生，频率为32kHz±20%，定义为LIRC。
- 外部晶体时钟：包括晶体（4MHz、12MHz、32KHz）振荡器和RC 振荡器，定义为XTAL。

8.2 系统时钟框图

ME9101的时钟结构如图8.1所示。

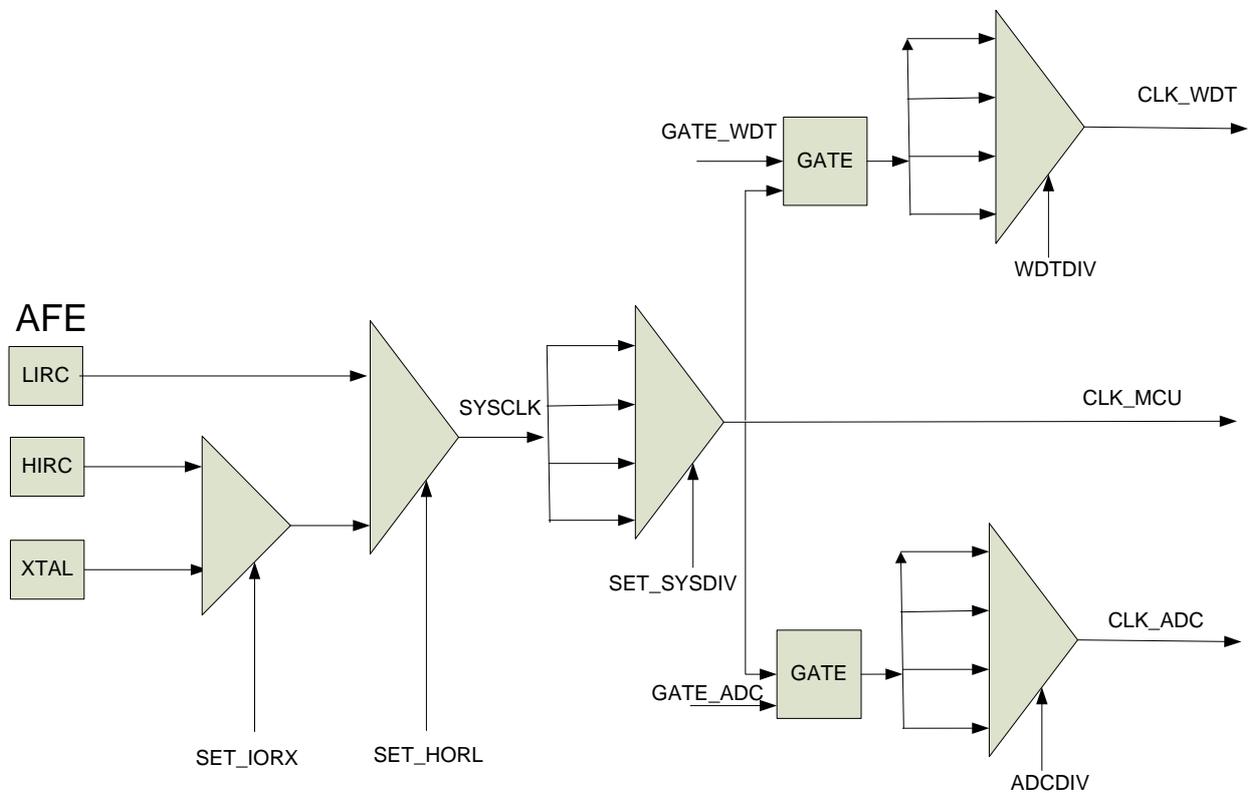


图 8.1: ME9101 的时钟结构框图

8.3 控制寄存器

(1) 高频RC配置寄存器HIRCCFGR

名称	地址	7	6	5	4	3	2	1	0
HIRCCFGR	9BH	0	0	0	HIRCADJ3	HIRCADJ2	HIRCADJ1	HIRCADJ0	HIRCEN
Reset Values:		0	0	0	0	1	1	1	0
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
HIRCADJ[3:0]	Bit4~Bit1	HIRC 频率调整控制位，步长为 2%。 HIRCADJ[3:0]=0000: -14% HIRCADJ[3:0]=0001: -12% HIRCADJ[3:0]=0111: 0% HIRCADJ[3:0]=1000: +2% HIRCADJ[3:0]=1110: +14% HIRCADJ[3:0]=1111: +16%
HIRCEN	Bit0	高频 RC 振荡器使能控制位。 HIRCEN=1: 使能。 HIRCEN=0: 不使能。

(2) 低频RC配置寄存器LIRCCFGR

名称	地址	7	6	5	4	3	2	1	0
LIRCCFGR	9CH	0	0	0	0	0	LIRCADJ1	LIRCADJ0	LIRCEN
Reset Values:		0	0	0	0	0	0	1	1
User R/W:		R	R	R	R	R	R/W	R/W	R/W

简称	位置	功能
LIRCADJ[1:0]	Bit2, Bit1	LIRC 频率调整控制位，步长为 10%。 LIRCADJ[1:0]=00: -10% LIRCADJ[1:0]=01: 0% LIRCADJ[1:0]= 10: +10% LIRCADJ[1:0]= 11: +20%
HIRCEN	Bit0	低频 RC 振荡器使能控制位。 LIRCEN=1: 使能。 LIRCEN=0: 不使能。

(3) 外部晶体配置寄存器XTALCFGR

名称	地址	7	6	5	4	3	2	1	0
XTALCFGR	9AH	0	0	0	0	XTALSEL 2	XTALSEL 1	XTALSEL 0	XTALEN
Reset Values:		0	0	0	0	0	0	1	0
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
XTALSEL [2:0]	Bit3~ Bit1	外接振荡器时钟源选择控制位。 XTALSEL [2:0]=001: 时钟来自 16M、12M 等外界晶体 XTALSEL [2:0]=010: 时钟来自 8K、32K 外接晶体 XTALSEL [2:0]=100: 时钟来自外界 RC 振荡电路
XTALEN	Bit0	外部晶体振荡器使能控制位。 XTALEN =1: 使能。 XTALEN =0: 不使能。

(4) 系统管理配置寄存器SYSCFGR

名称	地址	7	6	5	4	3	2	1	0
SYSCFGR	9DH	0	RSTFLG1	RSTFLG0	PDTCEN	SET_SYS DIV1	SET_SYS DIV0	SET_IOR X	SET_HOR L
Reset Values:		0	0	0	1	0	0	0	0
Designer R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W
User R/W:		R	R	R	R/W	R/W	R/W	R/W	R/W

简称	位置	功能
RSTFLG [1:0]	Bit6, Bit5	复位状态位，显示复位来源。 RSTFLG[1:0]=2'b00: 发生上电复位。 RSTFLG[1:0]=2'b01: 发生 IO 复位。 RSTFLG[1:0]=2'b10: 发生 WDT 复位。
PDTCEN	Bit4	编程检测电路使能控制位。缺省开启。为降低系统功耗，编程结束后用户程序最好设置此位为 0。 PDTCEN =1: 使能编程检测电路 PDTC。 PDTCEN =0: 关闭编程检测电路 PDTC。
SET_SYSDIV [1:0]	Bit3, Bit2	系统时钟工作频率控制位。 SET_SYSDIV [1:0]=2'b00: 不分频。 SET_SYSDIV [1:0]=2'b01: 2 分频。 SET_SYSDIV [1:0]=2'b10: 4 分频。 SET_SYSDIV [1:0]=2'b11: 8 分频。
SET_IORX	Bit1	高频时钟来源控制位。 SET_IORX =1: 高频时钟来源于外部振荡器 XTAL。 SET_IORX =0: 高频时钟来源于内部高速 RC 时钟 HIRC。
SET_HORL	Bit0	高速/低速模式控制位。 SET_HORL =1: 高速模式。 SET_HORL =0: 低速模式。

8.4 指令周期 CLK_MCU

系统时钟速率，即机器周期（CLK_MCU），从系统时钟源分离出来，决定系统的工作速率。Fcpu 的速率由SET_SYSDIV [1:0]决定，正常模式下，CLK_MCU =SYSCLK/1~SYSCLK/16。若高速时钟源为外部4MHz 振荡器，SET_SYSDIV [1:0]=10时，则CLK_MCU频率为4MHz/4=1MHz。低速模式下，SET_SYSDIV [1:0]=10时，CLK_MCU频率为32KHz/4=8KHz。

指令周期，一个指令周期是4个机器周期。例如，一个三字节指令MOVX需要3个指令周期，即12个机器周期。

高干扰环境下，强烈建议设置SET_SYSDIV [1:0]=10/11，以减少干扰的影响。

8.5 系统高速时钟

系统高速时钟包括外部高速时钟和内部高速时钟，由SET_IORX实现选择控制。外部高速时钟包括4MHz、12MHz、32KHz晶体/陶瓷和RC振荡器，由XTALSEL [2:0]选择，而XTALEN可控制其开关。内部高速时钟振荡器为16MHz，HIRCEN可控制其开关。

8.5.1 高速时钟选择

对应不同的时钟功能，ME9101提供多种高速时钟选项，如HIRC_16M、RC、32K X'tal和12M X'tal，以支持对不同带宽振荡器的需求。

- **HIRC_16M:** 系统高速时钟源来自内部高速16MHz RC振荡器。此时XIN/XOUT 作为普通的I/O引脚使用，不连接任何外部振荡设备。
- **RC:** 系统高速时钟源来自廉价的RC振荡电路，RC振荡电路只需要和XIN 引脚连接，XOUT 作为普通的I/O 引脚使用。
- **32K X'tal:** 系统高速时钟源来自外部低频32768Hz 振荡器。该选项用于支持RTC 正常工作所需的32768Hz晶体振荡器。
- **12M X'tal:** 系统高速时钟源来自外部高频晶体/陶瓷振荡器，其带宽为1MHz~16MHz。

8.5.2 内部高速 RC 振荡器 (HIRC)

内部高速16MHz RC振荡器，普通环境下精确度为 $\pm 2\%$ ，当HIRCEN置“1”时，内部高速振荡器工作。

HIRC_16M: 系统高速时钟为内部16MHz RC振荡器，XIN/XOUT 作为普通I/O 引脚使用。

8.5.3 外部高速振荡器

外部高速振荡器包括12MHz、32KHz和RC。12MHz和32KHz可以使用晶体和陶瓷振荡器，这时XIN/XOUT和GND之间需连接一个20pF的电容。廉价的RC振荡电路只需要和XIN 引脚连接，电容的容值不能低于100pF，电阻的阻值决定频率。

(a) XIN/XOUT 可接 32.768KHz、4MHz、12MHz 的晶体和陶瓷振荡器，XIN/XOUT 和 GND 之间需连接一个 20pF 的电容。连接关系如下图所示：

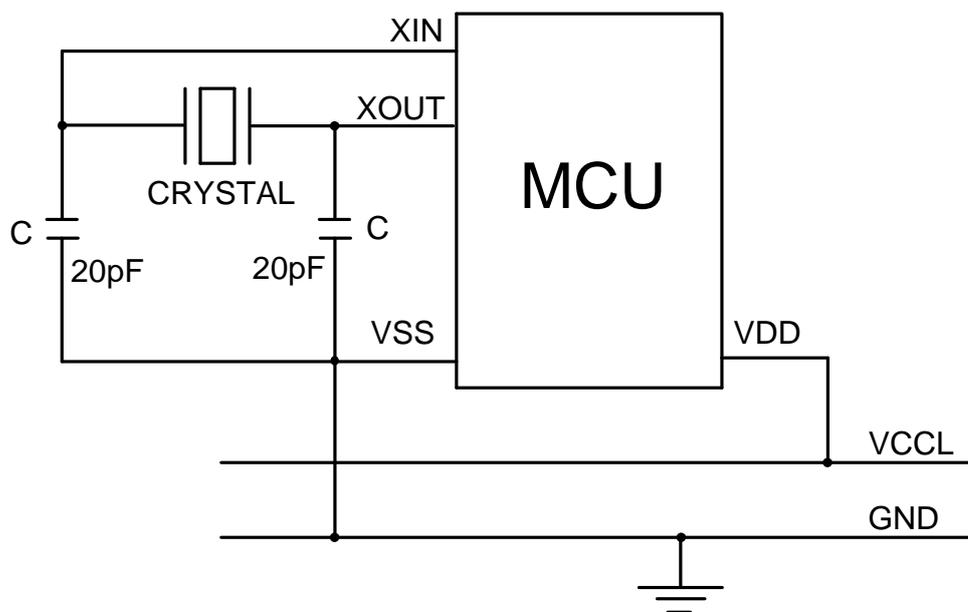


Figure 8.2: 晶体和陶瓷连接电路

注：晶体/陶瓷和电容C要尽可能的靠近单片机的XIN/XOUT/VSS；电阻R和电容C要尽可能的靠近单片机的VDD。

(b) XIN 可接 RC 震荡电路，XOUT 做为普通的 IO 引脚使用。电容的容值不能低于 100pF,电阻的阻值决定频率。连接关系如下图所示：

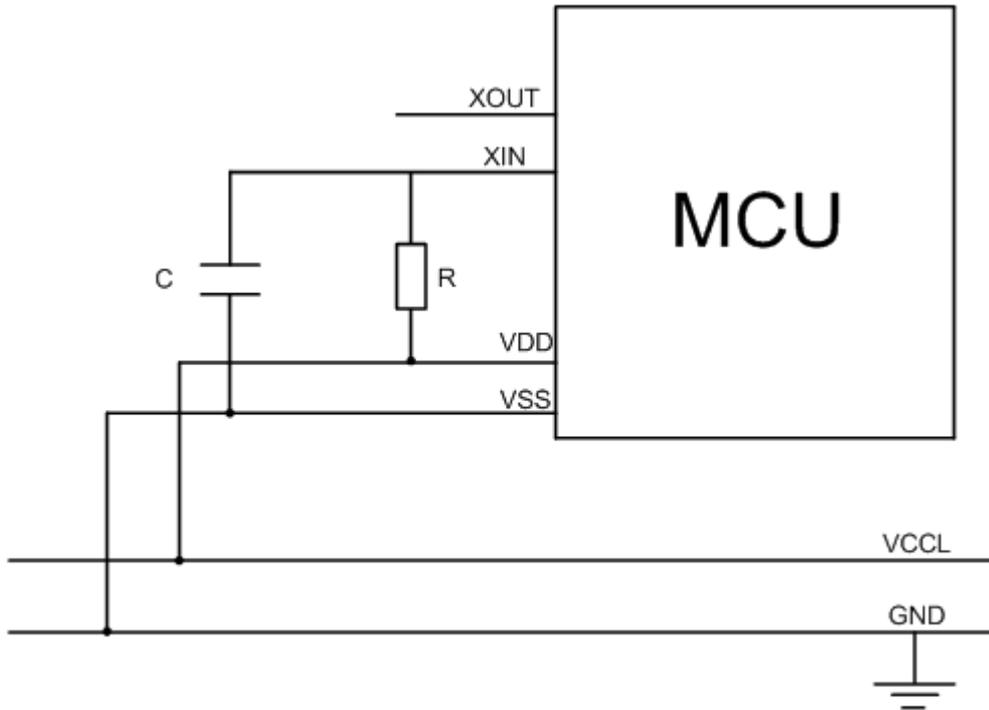


Figure 8.3: RC 连接电路

8.6 系统低速时钟

系统低速时钟源即内置的低速振荡器，采用RC振荡电路，产生稳定的 $32\text{KHz}\pm 20\%$ 的时钟信号，用于系统待机及低速处理时使用。低速时钟的输出频率基本不受系统电压的影响，但受环境温度的影响。

HIRCEN可以使能低速RC振荡器。低速RC振荡器缺省时为使能状态。

9. 检测保护功能

9.1 简介

ME9101包含如下二个检测电路：

- 低电压检测：检测电源电压，若低于报警阈值电压，则报警。通过寄存器可以配置报警阈值电压，阈值可分别设置为3.6V、 2.4V和2.0V；
- 温度检测：用于保护芯片的温度检测以及过温报警电路。过温报警阈值在160℃左右。

9.2 控制寄存器

9.2.1 LVD 配置寄存器 LVDCFGR

LVD检测电源电压，若电压低于报警的阈值电压，则报警。通过LVD配置寄存器LVDCFGR，配置报警阈值电压。

名称	地址	7	6	5	4	3	2	1	0
LVDCFGR	BBH	0	LVD20B	LVD24B	LVD36B	LVDH	LVDM	LVDL	LVDEN
Reset Values:		0	0	0	0	1	0	0	1
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
LVD20B	Bit6	LVD 工作电压状态位。 LVD20B=0: 系统工作电压 AVDD 低于 2.0V。 LVD20B=1: 系统工作电压 AVDD 超过 2.0V。
LVD24B	Bit5	LVD 工作电压状态位。 LVD24B=0: 系统工作电压 AVDD 低于 2.4V。 LVD24B=1: 系统工作电压 AVDD 超过 2.4V。
LVD36B	Bit4	LVD 工作电压状态位。 LVD36B=0: 系统工作电压 AVDD 低于 3.6V。 LVD36B=1: 系统工作电压 AVDD 超过 3.6V。
LVDH	Bit3	LVD 检测 3.6V 阈值设定控制位。 LVDH=1: 3.6V 阈值设定有效。 LVDH=0: 3.6V 阈值设定无效。
LVDM	Bit2	LVD 检测 2.4V 阈值设定控制位。 LVDM=1: 2.4V 阈值设定有效。 LVDM=0: 2.4V 阈值设定无效。
LVDL	Bit1	LVD 检测 2.0V 阈值设定控制位。 LVDL=1: 2.0V 阈值设定有效。 LVDL=0: 2.0V 阈值设定无效。
LVDEN	Bit0	低压检测功能使能控制位。 LVDEN =1: 使能。 LVDEN =0: 不使能。

9.2.2 温度检测配置寄存器 TPTCFGR

温度检测电路提供过温报警电路，用于保护芯片。过温报警阈值在160℃左右。

名称	地址	7	6	5	4	3	2	1	0
TPTCFGR	A1H	0	0	TPTFLG	0	0	0	TPTSEN	TPTCOMPEN
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R	R	R	R	R	R	R/W	R/W

简称	位置	功能
TPTFLG	Bit5	温度状态位。 TPTFLG=0: 芯片温度低于 160℃ TPTFLG=1: 芯片温度超过 160℃
TPTSEN	Bit1	温度检测功能使能控制位。 TPTSEN =1: 使能。 TPTSEN =0: 不使能。
TPTCOMPEN	Bit0	高频振荡器的温度补偿功能使能控制位。 TPTCOMPEN =1: 使能。 TPTCOMPEN =0: 不使能。

9.3 工作原理

9.3.1 低电压检测 LVD

低电压检测LVD是系统内置的低电压检测保护装置，当VDD跌落并低于LVD 检测电压值时，LVD将通过SFR的状态位显示VDD状态。

LVD设计为三层结构（2.0V/2.4V/3.6V），由LVD配置寄存器LVDCFGR控制。各控制位的关系如下表所示。

	LVDH		LVDM		LVDL	
	0	1	0	1	0	1
LVD20B	-	-	-	-	无效	有效
LVD24B	-	-	无效	有效	-	有效
LVD36B	无效	有效	-	有效	-	有效

9.3.2 温度检测

若温度检测配置寄存器TPTCFGR的TPTSEN使能，则该寄存器将实时显示芯片的温度。

10. 复位和初始化

10.1 简介

ME9101有以下几种复位方式：

- 1) 上电复位；
- 2) 看门狗复位；
- 3) 外部复位（仅在外部复位引脚处于使能状态时）。

上述任一种复位发生时，所有的系统寄存器恢复默认状态，程序停止运行，同时程序计数器PC清零。复位结束后，系统从向量0000H处重新开始运行。

任何一种复位情况都需要一定的响应时间，系统提供完善的复位流程以保证复位动作的顺利进行。对于不同类型的振荡器，完成复位所需要的时间也不同。因此，VDD的上升速度和不同晶振的起振时间都不固定。RC振荡器的起振时间最短，晶体振荡器的起振时间则较长。客户在使用的过程中，应注意考虑主机对上电复位时间的要求。

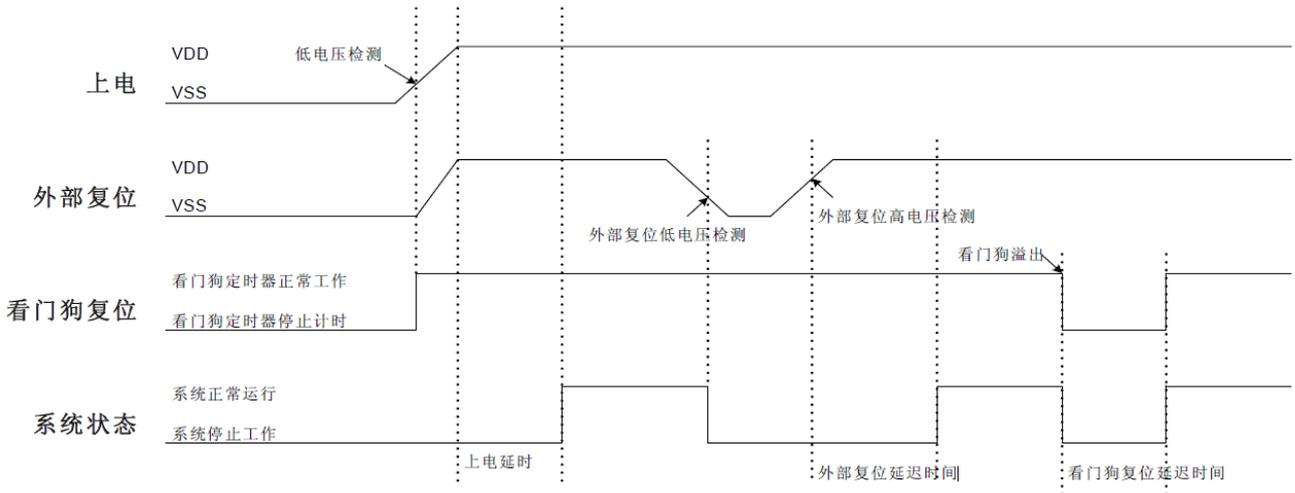


图10.1: 各种复位的响应时间示意图

10.2 上电复位

上电复位与LVD动作密切相关。系统上电的过程呈逐渐上升的曲线形式，需要一定时间才能达到正常电平值。下面给出上电复位的正常时序：

- 上电：系统检测到电源电压上升并等待其稳定；
- 外部复位（仅限于外部复位引脚使能状态）：系统检测外部复位引脚状态。如果不为高电平，系统保持复位状态直到外部复位引脚释放；
- 系统初始化：所有的系统寄存器被置为初始值；
- 振荡器开始工作：振荡器开始提供系统时钟；
- 执行程序：上电结束，程序开始运行。

10.3 看门狗复位

看门狗复位是系统的一种保护设置。在正常状态下，由程序将看门狗定时器清零。若出错，系

统处于未知状态，看门狗定时器溢出，此时系统复位。看门狗复位后，系统重启进入正常状态。看门狗复位的时序如下：

- 看门狗定时器状态：系统检测看门狗定时器是否溢出，若溢出，则系统复位；
- 系统初始化：所有的系统寄存器被置为默认状态；
- 振荡器开始工作：振荡器开始提供系统时钟；
- 执行程序：上电结束，程序开始运行。

看门狗定时器应用注意事项：

- 对看门狗清零之前，检查I/O口的状态和RAM的内容可增强程序的可靠性；
- 不能在中断中对看门狗清零，否则无法侦测到主程序跑飞的状况；
- 程序中应该只在主程序中有一次清看门狗的动作，这种架构能够最大限度的发挥看门狗的保护功能。

10.4 外部复位

外部复位由P0.4/RST/VPP管脚输入。复位引脚处于高电平时，系统正常运行。当复位引脚输入低电平信号时，系统复位。需要注意的是，在系统上电完成后，外部复位引脚必须输入高电平，否则系统将一直保持在复位状态。外部复位的时序如下：

- 外部复位（当且仅当外部复位引脚为使能状态）：系统检测复位引脚的状态，如果复位引脚不为高电平，则系统会一直保持在复位状态，直到外部复位结束；
- 系统初始化：初始化所有的系统寄存器；
- 振荡器开始工作：振荡器开始提供系统时钟；
- 执行程序：上电结束，程序开始运行。

外部复位可以在上电过程中使系统复位。良好的外部复位电路可以保护系统避免进入未知的工作状态，如AC应用中的掉电复位等。

10.5.1 RC 复位电路

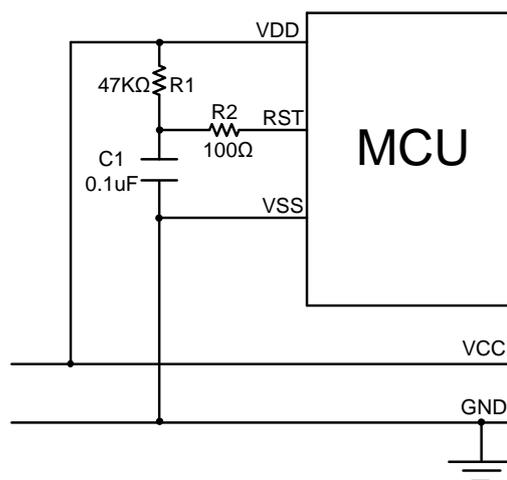


图10.2: RC复位电路

图10.2为一个由电阻R1和电容C1组成的基本RC复位电路，它在系统上电的过程中能够为复位引脚提供一个缓慢上升的复位信号。这个复位信号的上升速度低于VDD的上电速度，为系统提供合理的复位时序，当复位引脚检测到高电平时，系统复位结束，进入正常工作状态。

注：此RC复位电路不能解决非正常上电和掉电复位问题。

10.5.2 二极管及 RC 复位电路

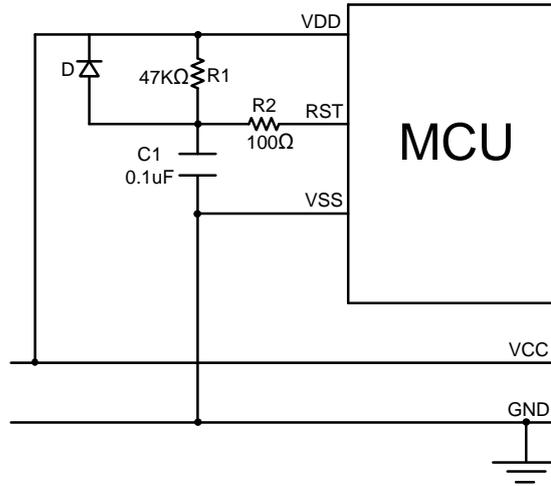


图10.3: 二极管及RC复位电路

图10.3中，R1和C1同样是为复位引脚提供输入信号。对于电源异常情况，二极管正向导通使C1快速放电并与VDD保持一致，避免复位引脚持续高电平、系统无法正常复位。

注：“基本RC复位电路”和“二极管及RC复位电路”中的电阻R2都是必不可少的限流电阻，以避免复位引脚ESD（Electrostatic Discharge）或EOS（Electrical Over-stress）击穿。

10.5.3 稳压二极管复位电路

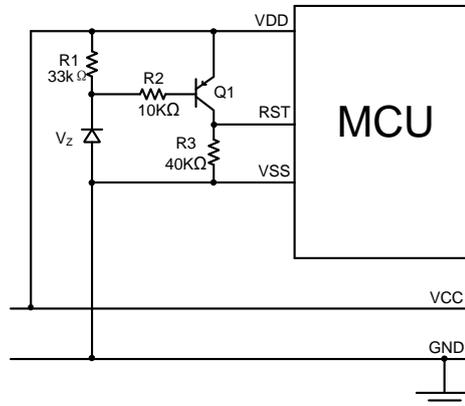


Figure 10.4: 稳压二极管复位电路

稳压二极管复位电路是一种简单的LVD电路，基本上可以完全解决掉电复位问题。图10.4电路中，利用稳压管的击穿电压作为电路复位检测值，当VDD高于“ $V_z + 0.7V$ ”时，三极管集电极输出高电平，单片机正常工作；当VDD低于“ $V_z + 0.7V$ ”时，三极管集电极输出低电平，单片机复位。稳压管规格不同则电路复位检测值不同，应根据电路的要求选择合适的二极管。

10.5.4 电压偏置复位电路

电压偏置复位电路是一种简单的LVD电路，基本上可以完全解决掉电复位问题。与稳压二极管复位电路相比，这种复位电路的检测电压值的精确度有所降低。电路中，R1和R2构成分压电路，当VDD高于和等于分压值“ $0.7V \times (R1 + R2) / R1$ ”时，三极管集电极C输出高电平，单片机正常工作；

VDD低于“ $0.7V \times (R1 + R2) / R1$ ”时，集电极C输出低电平，单片机复位。

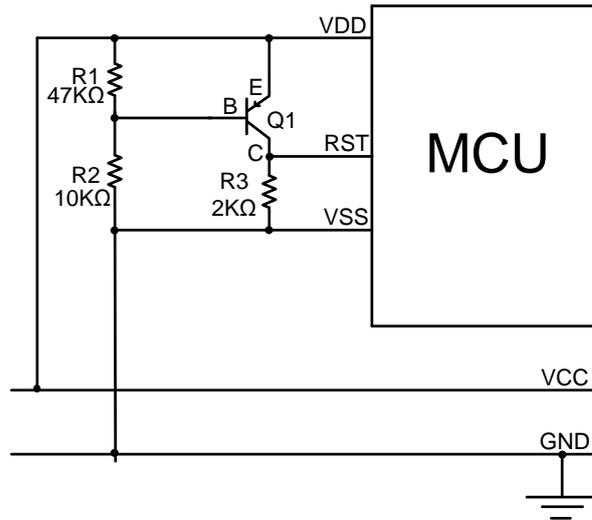


图10.5: 电压偏置复位电路

对于不同应用需求，选择适当的分压电阻。单片机复位引脚上电压的变化与VDD电压变化之间的差值为0.7V。如果VDD跌落并低于复位引脚复位检测值，那么系统将被复位。如果希望提升电路复位电平，可将分压电阻设置为 $R2 > R1$ ，并选择VDD与集电极之间的结电压高于0.7V。分压电阻R1和R2在电路中要耗电，此处的功耗必须计入整个系统的功耗中。

注：在电源不稳定或掉电复位的情况下，“稳压二极管复位电路”和“偏压复位电路”能够保护电路在电压跌落时避免系统出错。当电压跌落至低于复位检测值时，系统将被复位。从而保证系统正常工作。

10.5.5 外部IC复位

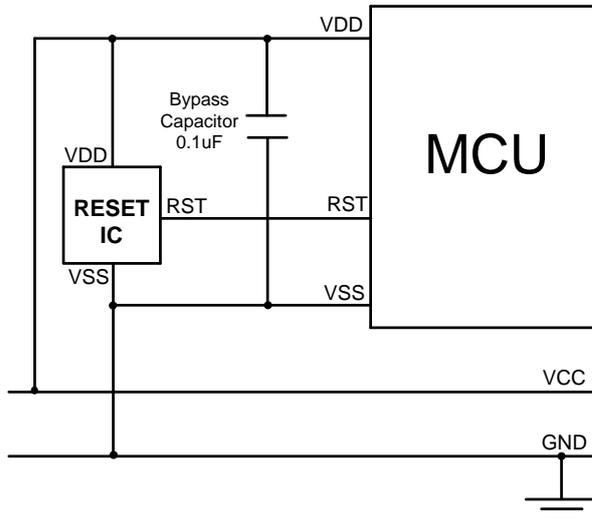


图10.6: 外部IC复位

外部复位也可以选用IC进行外部复位，但是这样一来系统成本将会增加。针对不同的应用要求选择适当的复位IC，如图10.6所示的外部IC复位电路，能够有效的降低电源变化对系统的影响。

11. 系统工作模式

11.1 概述

ME9101可以在以下4种工作模式下以不同的时钟频率工作，这些模式可以控制振荡器的工作、程序的执行以及模拟电路的功能损耗。

- 高速模式：系统普通工作模式；
- 低速模式：系统低速工作模式；
- 空闲模式；
- 停止模式。

几个工作模式切换转移图如下图所示。

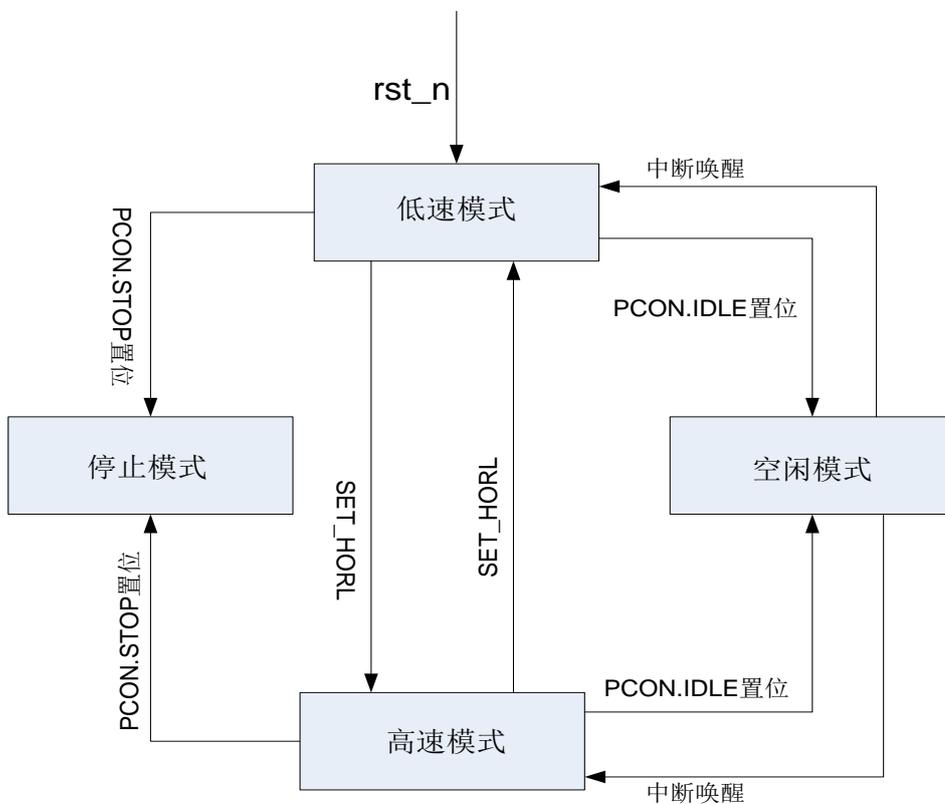


Figure 11.1: 工作模式切换转移图

各模式下的工作情况总结如下表所示。

工作模式	低速模式	高速模式	空闲模式	停止模式
低速振荡器 LIRC	工作	可设置 LIRCCFGR.LIRCEN 位关闭		不工作
高速振荡器 HIRC	可设置 HIRCCFGR.HIRCEN 位关闭	若选择 XTAL, HIRC 可关闭		不工作
晶体振荡器	可设置	若选择 HIRC, XTAL		不工作

XTAL	XTALCFGR.XTALEN 位关闭	可关闭		
MCU 时钟	LIRC	HIRC 或者 XTAL	未被门控	不工作
WDT 时钟	可设置 WDTCFG.WDTGT 位 门控	可设置 WDTCFG.WDTGT 位门控	自动被门控	不工作
ADC 时钟	可设置 ADCCFG2.ADTGT 位门控	可设置 ADCCFG2.ADCGT 位门控	自动被门控	不工作
LDO1.8V	供电	供电	供电	不供电

11.2 低速模式

低速模式是MCU系统上电后的默认模式，即系统时钟源由低速振荡器LIRC提供。上电复位或任何复位（WDT复位、外部手动复位）触发后，系统进入低速模式执行程序。

低速模式时，芯片状态总结如下：

- 程序被执行，所有的功能都可控制。
- 系统速率为低速。
- 内部低速RC振荡器正常工作，高速振荡器由HIRCCFGR.HIRCEN控制。低速模式下，强烈建议停止高速振荡器。
- 通过相应寄存器，低速模式可以切换进入其它的工作模式。
- 从低速模式切换到空闲模式，唤醒后返回到低速模式。
- 系统从停止模式唤醒后进入低速模式。

11.3 高速模式

高速模式是MCU系统时钟源由高速振荡器（XTAL或者HIRC，具体由SYSCTRL.SET_IORX位决定）提供。程序通过配置系统控制寄存器SYSCTRL.SET_HORL位为1，切换到高速模式。

高速模式时，芯片状态总结如下：

- 程序被执行，所有的功能都可控制。
- 系统速率为高速。
- 高速振荡器和内部低速RC振荡器都正常工作。高速模式下，建议停止低速振荡器。
- 通过相应的寄存器，系统可以从高速模式切换到其它任何一种工作模式。
- 低速模式可以切换到高速模式。
- 从高速模式切换到空闲模式，唤醒后返回到高速模式。

11.4 空闲模式

设置IDLE位(PCON.0) 为“1”的指令完成后，芯片即进入空闲模式。空闲模式时，CPU处理暂停，内部寄存器保持目前的数据，CPU核的idle_mode_n输出低电平。如果外部时钟没有停止，定时器和ADC工作。与标准的8051不同，内部时钟不禁用。

有下列两种方法退出空闲模式：

- 中断发生

- 复位 Reset

任何中断的激活都使硬件清除空闲模式控制位，终止空闲模式。CPU执行与接收到的中断相关的中断服务程序（ISR）。ISR结束时的RETI指令，将使CPU回到芯片进入空闲模式时的指令。

激活复位使芯片退出空闲模式时，复位内部模块，并从标准复位向量地址0000H开始执行程序。

11.5 停止模式

设置PCON.STOP位为“1”的指令执行将使芯片进入停止模式。进入停止模式，CPU处理停止，CPU核的stop_mode_n输出低电平。然而，与标准的8051不同，内部时钟不禁用。在5V电压域，模拟接口电路观察到内核进入stop模式后，会关闭提供数字1.8V电源的LDO，此时数字单元完全断电。因此，能显著减少功耗。

退出停止模式的方法是复位发生外部IO复位或者重新上电，这时1.8V LDO重新开始供电，CPU执行复位处理程序，即程序从地址0000H开始执行。

12. 中断

12.1 概述

ME9101提供下列7个中断源:

- INT0 interrupt
- INT1 interrupt
- INT2 interrupt
- INT3 interrupt
- TC0 interrupt
- TC1 interrupt
- ADC interrupt

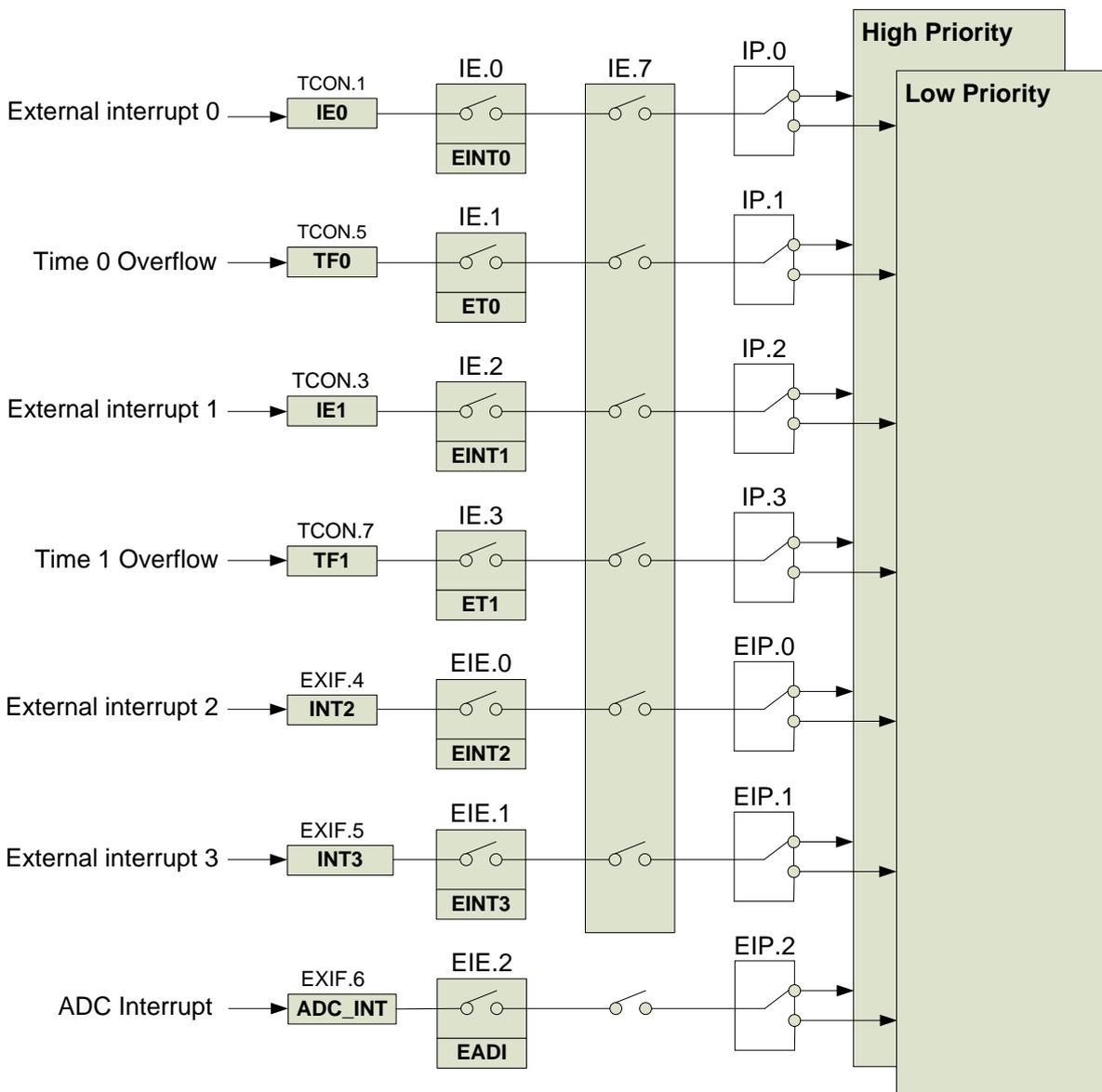


Figure 12.1: 中断硬件结构框图

12.2 硬件结构框图

图12.1所示为中断系统的硬件结构框图。

12.3 中断寄存器

中断系统的SFR主要包括：中断使能寄存器IE、中断优先级寄存器IP、外部中断标志寄存器EXIF、外部中断使能寄存器EIE和外部中断优先级寄存器EIP，这些寄存器为中断单元提供了中断使能、优先级控制和标志位等。

12.3.1 中断使能寄存器 (IE)

名称	地址	7	6	5	4	3	2	1	0
IE	A8H	EA	0	0	0	ET1	EINT1	ET0	EINT0
Reset Values:		0	0	0	0	0	0	0	0
User R/W:		R/W	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
EA	Bit7	全局中断使能位。 EA = 0: 禁止所有中断(EA 凌驾于单独中断使能位)。 EA = 1: 每一个中断的使能或者屏蔽由这些中断自己的使能位决定。
ET1	Bit3	Timer1 中断使能位。 ET1=0: 禁止 Timer1 中断(TF1)。 ET1=1: 使能由 TF1 标志位产生的中断。
EINT1	Bit0	P0.1 (Int1_n) 中断使能位。 EINT1=0: 禁止 P0.1 中断。 EINT1=1: 使能 P0.1 中断。
ET0	Bit1	Timer 0 中断使能位。 ET0=0: 禁止 Timer0 中断(TF0)。 ET0=1: 使能由 TF0 标志位产生的中断。
EINT0	Bit0	P0.0 (Int0_n) 中断使能位。 EINT0=0: 禁止 P0.0 中断。 EINT0=1: 使能 P0.0 中断。

12.3.2 中断优先级寄存器 (IP)

名称	地址	7	6	5	4	3	2	1	0
IP	B8H	1	1	1	1	PT1	PINT1	PT0	PINT0
Reset Values:		1	1	1	1	0	0	0	0
User R/W:		R	R	R	R	R/W	R/W	R/W	R/W

简称	位置	功能
PT1	Bit3	Timer 1 中断优先级控制位。

		PT1=0: 设置 Timer 1 (TF1)中断优先级为低。 PT1=1: 设置 Timer 1 (TF1)中断优先级为高。
PINT1	Bit2	P0.1 (Int1_n) 中断优先级控制位。 PINT1 =0: 设置 P0.1 (Int1_n) 中断优先级为低。 PINT1 =1: 设置 P0.1 (Int1_n) 中断优先级为高。
PT0	Bit1	Timer 0 中断优先级控制位。 PT0=0: 设置 Timer 0 (TF0)中断优先级为低。 PT0=1: 设置 Timer 0 (TF0)中断优先级为高。
PINT0	Bit0	P0.0 (Int0_n) 中断优先级控制位。 PINT0 =0: 设置 P0.0 (Int0_n) 中断优先级为低。 PINT0 =1: 设置 P0.0 (Int0_n) 中断优先级为高。

12.3.3 外部中断标志寄存器 (EXIF)

名称	地址	7	6	5	4	3	2	1	0
EXIF	91H	1	ADC_INT	INT3	INT2	1	0	0	0
Reset Values:		1	0	1	0	1	0	0	0
User R/W:		R	R/W	R/W	R/W	R	R	R	R

简称	位置	功能
ADC_INT	Bit6	ADC 中断标志位。必须软件清除。若使能，使用软件设置也会产生中断。 ADC_INT=0: 没有 ADC 中断产生 ADC_INT=1: ADC 中断产生
INT3	Bit5	INT3 中断标志位。INT3 = 1, 表示 INT3 端口输入信号有下降沿出现, 中断产生。必须软件清除。若使能，使用软件设置也会产生中断。
INT2	Bit4	INT2 中断标志位。INT2 = 1, 表示 INT2 端口输入信号有上升沿出现, 中断产生。必须软件清除。若使能，使用软件设置也会产生中断。

12.3.4 外部中断使能寄存器 (EIE)

名称	地址	7	6	5	4	3	2	1	0
EIE	E8H	1	1	1	0	0	EADI	EINT3	EINT2
Reset Values:		1	1	1	0	0	0	0	0
User R/W:		R	R	R	R	R	R/W	R/W	R/W

简称	位置	功能
EADI	Bit2	ADC 中断使能位。 EADI =0: 禁止 ADC 中断。 EADI =1: 使能 ADC 中断。
EINT3	Bit1	P0.3 (Int3_n) 中断使能位。 EINT3=0: 禁止 P0.3 (Int3_n) 中断。

		EINT3=1: 使能 P0.3 (Int3_n) 中断。
EINT2	Bit1	P0.2 (Int2) 中断使能位。 EINT2=0: 禁止 P0.2 (Int2) 中断。 EINT2=1: 使能 P0.2 (Int2) 中断。

12.3.5 外部中断优先级寄存器 (EIP)

名称	地址	7	6	5	4	3	2	1	0
EIP	F8H	1	1	1	1	1	PADI	PINT3	PINT2
Reset Values:		1	1	1	1	1	0	0	0
User R/W:		R	R	R	R	R	R/W	R/W	R/W

简称	位置	功能
PADI	Bit2	ADC 中断优先级控制位。 PADI =0: 设置 ADC 中断优先级为低。 PADI =1: 设置 ADC 中断优先级为高。
PINT3	Bit1	P0.3 (Int3_n) 中断优先级控制位。 PINT3 =0: 设置 P0.3 (Int3_n) 中断优先级为低。 PINT3 =1: 设置 P0.3 (Int3_n) 中断优先级为高。
PINT2	Bit0	P0.2 (Int2) 中断优先级控制位。 PINT2 =0: 设置 P0.2 (Int2) 中断优先级为低。 PINT2 =1: 设置 P0.0 (Int2) 中断优先级为高。

12.4 中断处理

当中断发生时，CPU向量指向与下表所示的该中断相关联的中断服务程序（ISR）地址。CPU 执行ISR完成，除非另一个高优先级的中断发生。每个ISR的最后指令为RETI（中断返回指令）。执行RETI后，该CPU返回到执行中断前的下一个指令。

中断向量和优先级中断

中断	描述	缺省的优先级	中断向量
Int0_n	外部中断 0	1	03H
TF0	Time 0 中断	2	0BH
Int1_n	外部中断 1	3	13H
TF1	Time 1 中断	4	1BH
Int2	外部中断 2	5	43H
Int3_n	外部中断 3	6	4BH
adc_int	ADC 中断	7	53H

ISR的执行只能被一个更高优先级的中断打断。

在执行中断服务前，CPU要完成正在处理的指令。如果正在进行的指令为RETI，或对IP、IE、EIP、EIE中的任何一个寄存器进行写访问，CPU在执行中断服务前，要多执行一个指令。

12.5 中断屏蔽

寄存器IE的EA (IE.7) 是一个全局的中断使能控制位。当EA = 1, 每个中断使能或屏蔽由各自的使能位控制。当EA = 0, 所有的中断被屏蔽。

下表简要概括了芯片中的中断源、标志位、使能位和优先级。

中断源、标志位、使能位和优先级

中断	描述	标志位	使能位	优先级控制
Int0_n	外部中断 0	TCON.1	IE.0	IP.0
TF0	Time 0 中断	TCON.5	IE.1	IP.1
Int1_n	外部中断 1	TCON.3	IE.2	IP.2
TF1	Time 1 中断	TCON.7	IE.3	IP.3
Int2	外部中断 2	EXIF.4	EIE.0	EIP.0
Int3_n	外部中断 3	EXIF.5	EIE.1	EIP.1
adc_int	ADC 中断	EXIF.6	EIE.2	EIP.2

12.6 中断优先级

芯片提供两级中断优先级的设定：中断级和缺省优先级，即除了可设定的优先级（高或低），每个中断具有如前一个表所示的缺省优先级。中断级（高或低）优先于缺省优先级。若具有相同中断级的中断（例如都设为高）同时发生时，中断的处理将根据缺省优先级的先后依次进行。例如，如果TF0和TF1都编程为高优先级，若中断同时发生，TF0中断优先执行。

一旦一个中断正在进行，只有一个更高优先级的中断能打断这个正在处理的中断服务程序的执行。

12.7 中断采样

定时器通过设置各自的SFR中断标志位来产生中断。

int0_n和int1_n都为低有效，通过SFR TCON的ITO、IT1位可编程为电平敏感或沿敏感。例如，当ITO = 0, int0_n为电平敏感，当int0_n引脚采样到低电平时，CPU设置IE0标志位。当ITO = 1, int0_n为沿敏感，当int0_n引脚采样到高到低变化的连续信号时，CPU设置IE0标志位。

ME9101的外部中断皆为沿敏感的，int2和ADC高有效，而int3_n低有效。

12.8 中断延迟

中断响应时间取决于CPU的当前状态。最快的响应时间是五个指令周期：一个检测中断，四个进行ISR的LCALL。

当CPU执行后面紧跟MUL或DIV指令的RETI时，将发生最大延迟：十三个指令周期。在这种情况下，十三个指令周期分别为：一个检测中断，三个完成RETI，五个执行MUL或DIV，四个执行ISR的LCALL。最大延迟的情况下，响应时间为 $13 \times 4 = 52$ 个时钟周期。

中断结构提供了一种进行单步执行程序的方法。当执行RETI指令退出ISR时，CPU总是至少执行一个指令的服务程序。因此，一旦进入一个中断服务程序，只有至少一个程序指令被执行后，它才能再此进入。

13. OTP 烧录

13.1 概述

ME9101内含4K字节的一次性可编程存储器（OTP ROM）。用户在使用芯片前，需要使用OTP烧录器写入程序数据后，才能使用。数据一旦写入，不能随意改写，在切断电源之后，数据也不会消失，即具有非易失性。

13.2 OTP 烧录信息

ME9101采用串行烧录方式实现OTP的编程数据写入，所需信号如下表所示。

Name	Type	Description
VDD	P	电源输入端（2.2V ~ 5.5 V）。
VSS	P	电源地。
P0.4/VPP	P	OTP 烧录高电压引脚 VPP（7.5V ~ 8.0 V）。
P4.1/PCK	I	时钟信号 PCK。
P4.2/PDAT	I	串行编程数据输入口。
P4.3/PWE	I	编程写信号 PWE。
P4.4/PPROG	I	编程信号 PPROG。

14. SFR 寄存器

特殊功能寄存器SFR用来控制ME9101的各种功能。不像通用寄存器，每个SFR都各自具有特殊的功能。它的地址被分配在80H到FFH的128字节区域。

14.1 寄存器汇总

下表所示为ME9101的SFR:

Symbol	Name	Address	Reset Value
SP	堆栈指针	81H	07H
DPL0	数据指针 0 低字节	82H	00H
DPH0	数据指针 0 高字节	83H	00H
DPL1	数据指针 1 低字节	84H	00H
DPH1	数据指针 1 高字节	85H	00H
DPS	数据指针 DPTR 选择寄存器	86H	00H
PCON	电源模式控制寄存器	87H	30H
TCON	定时器控制寄存器	88H	00H
TMOD	定时器模式控制寄存器寄存器	89H	00H
TL0	定时器 0 计数器的低字节寄存器	8AH	00H
TL1	定时器 1 计数器的低字节寄存器	8BH	00H
TH0	定时器 0 计数器的高字节寄存器	8CH	00H
TH1	定时器 1 计数器的高字节寄存器	8DH	00H
CKCON	时钟控制寄存器	8EH	01H
EXIF	外部中断标志寄存器	91H	08H
XTALCFGR	外部晶体配置寄存器	9AH	02H
HIRCCFGR	高频 RC 配置寄存器	9BH	0EH
LIRCCFGR	低频 RC 配置寄存器	9CH	03H
SYSCFGR	系统管理配置寄存器	9DH	10H
WDTCFGR	WDT 配置寄存器	9EH	08H
WDTCLR	WDT 清零寄存器	9FH	00H
TPTCFGR	温度检测配置寄存器	A1H	00H
P0DINR	P0 端口输入数据寄存器	A2H	00H
P0DOUTR	P0 端口输出数据寄存器	A3H	00H
P0OENR	P0 端口输出使能寄存器	A4H	0FH
P0PULLUPR	P0 端口上拉电阻使能寄存器	A5H	00H
P0FUNCFGR	P0 端口功能选择配置寄存器	A6H	00H
P4DINR	P4 端口输入数据寄存器	A7H	00H
IE	中断使能控制寄存器	A8H	00H
P4DOUTR	P4 端口输出数据寄存器	A9H	00H
P4OENR	P4 端口输出使能寄存器	AAH	1FH
P4PULLUPR	P4 端口上拉电阻使能寄存器	ABH	00H
P5DINR	P5 端口输入数据寄存器	ACH	00H

P5DOUTR	P5 端口输出数据寄存器	ADH	00H
P5OENR	P5 端口输出使能寄存器	AEH	03H
P5PULLUPR	P5 端口上拉电阻使能寄存器	AFH	00H
PWM0L	PWM0 占空比配置低字节寄存器	B1H	00H
PWM0H	PWM0 占空比配置高字节寄存器	B2H	00H
ADCCFGR0	ADC 输入通道选择寄存器	B3H	01H
ADCCFGR1	ADC 系统控制寄存器	B4H	0DH
ADCCFGR2	ADC 时钟控制寄存器	B5H	08H
ADCDATLR	ADC 数据低字节寄存器	B6H	XXH
ADCDATHR	ADC 数据高字节寄存器	B7H	XXH
IP	中断优先级控制寄存器	B8H	80H
P5FUNCFGR	P5 端口功能选择配置寄存器	BAH	00H
LVDCFGR	LVD 配置寄存器	BBH	09H
PSW	程序状态字	D0H	00H
ACC	累加器 A	E0H	00H
EIE	外部中断使能控制寄存器	E8H	E0H
B	B 寄存器	F0H	00H
EIP	外部中断优先级控制寄存器	F8H	E0H

14.2 寄存器表

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
F8	EIP								FF
F0	B								F7
E8	EIE								EF
E0	ACC								E7
D8									DF
D0	PSW								D7
C8									CF
C0	IOPORT								C7
B8	IP		P5FUNCFGR	LVDCFGR					BF
B0		PWM0L	PWM0H	ADCCFGR0	ADCCFGR1	ADCCFGR2	ADCDATLR	ADCDATHR	B7
A8	IE	P4DOUTR	P4OENR	P4PULLUPR	P5DINR	P5DOUTR	P5OENR	P5PULLUPR	AF
A0		TPTCFGR	P0DINR	P0DOUTR	P0OENR	P0PULLUPR	P0FUNCFGR	P4DINR	A7
98			XTALCFGR	HIRCCFGR	LIRCCFGR	SYSCFGR	WDTCFGR	WDTCLR	9F
90		EXIF							97
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON		8F
80		SP	DPL0	DPH0	DPL1	DPH1	DPS	PCON	87

↑
Bit Addressable

图14.1: 寄存器表

15. 电特性

15.1 极限参数

Parameter	Symbol	Limit Values		Unit
		Min.	Max.	
Supply voltage	VDD	-0.3	7.0	V
Input voltage	VIN	-0.3	VDD+0.3	V
Operating temperature	T _A	-25	70	°C
Storage temperature	T _s	-40	125	°C

15.2 DC 特性

Parameter	Symbol	Conditions	Limit Values			Unit
			Min.	Typ.	Max.	
Supply voltage	VDD		2.2	5.0	5.5	V
Supply current (Disable ADC)	I _{DD1}	Run Mode (4MHz)	-	2.5	5	mA
		Run Mode (4MHz; VDD=3V)	-	1	2	mA
	I _{DD2}	Run Mode (32KHz; VDD=5V)	-	20	40	uA
		Run Mode (32KHz; VDD=3V)	-	5	10	uA
	I _{DD3}	Idle Mode (4MHz; VDD=5V)	-	0.6	1.2	mA
		Idle Mode (4MHz; VDD=3V)	-	0.25	0.5	mA
	I _{DD4}	Idle Mode (32KHz; VDD=5V)	-	15	30	uA
		Idle Mode (32KHz; VDD=3V)	-	3	6	uA
	I _{DD5}	Stop Mode (VDD=5V)	-	0.8	1.6	uA
Stop Mode (VDD=3V)		-	0.7	1.4	uA	
Supply current spikes	I _{DDD}	T<400ns; E=20nAs	-	-	80	mA
Internal High Oscillator Freq.	Fhirc	Internal High RC	15.68	16	16.32	MHz
Internal Low Oscillator Freq.	Flirc		-	32	-	KHz
LVD Voltage	Vdet0	Low voltage indicator level 0.	1.7	2.0	2.3	V
	Vdet1	Low voltage indicator level 1.	2.0	2.4	3.0	V
	Vdet2	Low voltage indicator level 2.	2.9	3.6	4.5	V
Input Low Voltage	V _{IL}				0.3VDD	V
Input High Voltage	V _{IH}		0.7VDD			V
Pull-Up Resistance	R _{pu}	Set Pull-Up enable		50		kΩ
Output Low Voltage	V _{OL}	Without current driving			0.1VDD	V
Output High Voltage	V _{OH}	Without current driving	0.9VDD			V
Output Source Current	I _{OH}	V _O ≥VDD-0.5V		16		mA

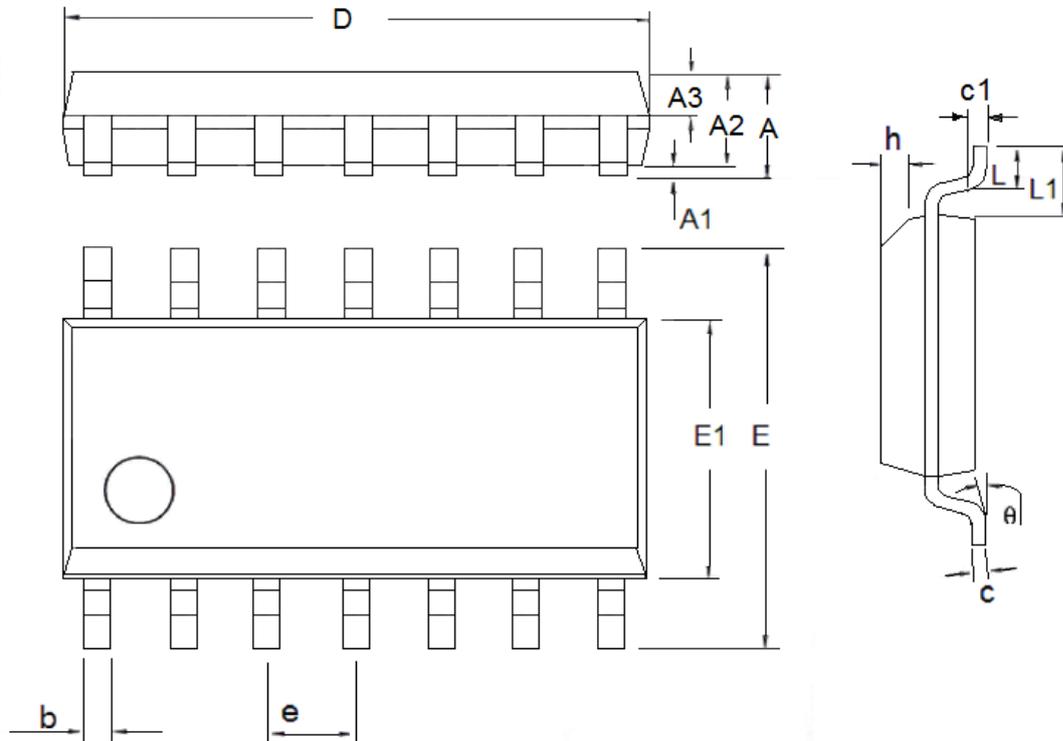
Output Sink Current	I_{OL}	$V_O \leq 0.5V$		16		mA
Leakage Current	I_{lk}	Floating Pin without Pull-Up			1	μA
ESD	HBM		2000			V
	MM		300			V

15.3 ADC 特性

Parameters	Symbol	Conditions	Min.	Typ.	Max.	Unit
RESOLUTION			-	12		bits
THROUGHOUT RATE	F_s		-	-	200	KHz
DC ACCURACY						
Differential Nonlinearity	DNL	External VREF	-	-	± 1	LSB
Integral Nonlinearity	INL		-	-	± 2	LSB
Offset Error	E_{zs}		-	-	± 10	LSB
Gain Error	E_{FS}		-	-	± 10	LSB
Internal REFERENCE VOLTAGE						
Internal Reference Voltage	VREF		2.0	-	AVDD	V
External REFERENCE VOLTAGE						
External Reference Voltage	VREF		2.0	-	AVDD	V
Current draw into VREF pin	IREF		-	-	150	μA
ANALOG INPUT						
Input Voltage Range	AIN		0	-	VREF	V
Input Capacitance	C_{in}	Switched	-	-	15	pF
Analog Source Impedance	R_s		-	-	1000	ohm
Input Bandwidth (-3dB)	BW		-	100	-	KHz
POWER SUPPLY						
Analog Power Voltage	AVDD		2.2	5	5.5	V
POWER CONSUMPTION						
Internal VREF Mode			-	-	700	μA
External VREF Mode			-	-	500	μA
Stand-By Mode Consumption		Internal VREF	-	-	300	μA
		External VREF	-	-	100	μA
Sleep Mode Consumption	Ppd		1	-	-	μA
DYNAMIC PERFORMANCE (AIN = - 0.5dBFS)						
Signal-To-Noise & Distortion	SINAD	$f_{IN} = 10KHz$	62	-	-	dB
Effect Bits	ENOB	$f_{IN} = 10KHz$	10.5	-	-	Bits
Signal-To-Noise Ratio	SNR	$f_{IN} = 10KHz$	66	-	-	dB
Total Harmonic Distortion	THD	$f_{IN} = 10KHz$	-	-	-62	dB
Spurious-Free Dynamic Range	SFDR	$f_{IN} = 10KHz$	62	-	-	dB
CLOCK						
Clock Frequency	fclk		0.1	2	2.6	MHz

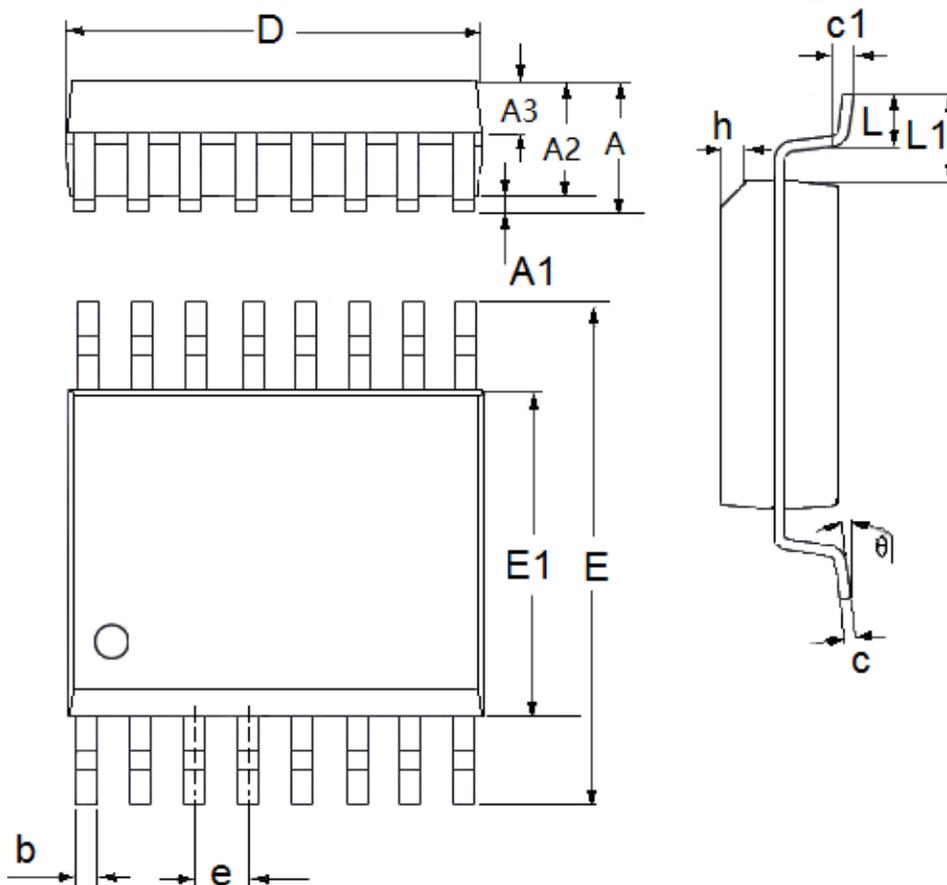
16. 封装信息

16.1 封装类型: SOP 14 单位:mm(inch)



参数	尺寸 (mm)		尺寸 (Inch)	
	最小值	最大值	最小值	最大值
A	1.45	1.85	0.0571	0.0728
A1	0.05	0.25	0.002	0.0098
A2	1.25	1.6	0.0492	0.063
A3	0.6	0.7	0.0236	0.0276
b	0.31	0.51	0.0122	0.0201
c	0.1	0.25	0.0039	0.0098
D	8.4	8.91	0.3307	0.3508
E	5.8	6.2	0.2283	0.2441
E1	3.8	4	0.1496	0.1575
e	1.27(TYP)		0.0500(TYP)	
h	0.25	0.5	0.0098	0.0197
L	0.4	1.27	0.0157	0.05
L1	1.05(TYP)		0.0413(TYP)	
θ	0	8°	0	8°
c1	0.25(TYP)		0.0098(TYP)	

16.2 封装类型: SSOP 16 单位:mm(inch)



参数	尺寸 (mm)		尺寸 (Inch)	
	最小值	最大值	最小值	最大值
A	1.3	1.8	0.0512	0.0709
A1	0.1	0.25	0.0039	0.0098
A2	1.3	1.55	0.0512	0.061
A3	0.55	0.65	0.0217	0.0256
b	0.2	0.31	0.0079	0.0122
c	0.17	0.25	0.0067	0.0098
D	4.7	5.1	0.185	0.2008
E	5.8	6.2	0.2283	0.2441
E1	3.8	4	0.1496	0.1575
e	0.635(TYP)		0.025(TYP)	
h	0.25	0.5	0.0098	0.0197
L	0.4	1.27	0.0157	0.05
L1	1.05(TYP)		0.0413(TYP)	
θ	0	8°	0	8°
c1	0.25(TYP)		0.0098(TYP)	

- 本资料内容，随产品的改进，可能会有未经预告之更改。
- 本资料所记载设计图等因第三者的工业所有权而引发之诸问题，本公司不承担其责任。另外，应用电路示例为产品之代表性应用说明，非保证批量生产之设计。
- 本资料内容未经本公司许可，严禁以其他目的加以转载或复制等。
- 本资料所记载之产品，未经本公司书面许可，不得作为健康器械、医疗器械、防灾器械、瓦斯关联器械、车辆器械、航空器械及车载器械等对人体产生影响的器械或装置部件使用。
- 尽管本公司一向致力于提高质量与可靠性，但是半导体产品有可能按照某种概率发生故障或错误工作。为防止因故障或错误动作而产生人身事故、火灾事故、社会性损害等，请充分留心冗余设计、火势蔓延对策设计、防止错误动作设计等安全设计。