内置 12 位 ADC / 16 位 PWM / 触摸 / 8K MTP 8 位微控制器

CA51M151 系列
中文用户手册

REV1.3

深圳市锦锐科技股份有限公司
电话：0755-83949938
传真：0755-83949977
http://www.cachip.com.cn
地址：中国广东省深圳市南山区沙河西路深圳湾科技生态园一区 2 栋 B 座 5 层

重要声明：本公司保留对以下所有产品在可靠性、功能和设计方面作进一步说明的权利，同时保留在未通知的情况下，对本产品所有文档做更改的权利。客户在使用此产品时，请向我公司销售人员索取最新文档。特此声明！

目　　录

1 概述 5
2 基本特性 5
3 芯片型号功能介绍 8
4 系统框图 9
5 引脚封装及其描述 10
5.1 封装定义 10
6 中央处理器（CPU） 21
6.1 CPU 简介 21
6.2 寄存器描述 21
7 存储器系统 25
7.1 随机数据存储器（RAM） 25
7.2 特殊功能寄存器（SFR） 26
7.3 EEPROM 和 MTP 存储器 27
7.3.1 功能简介 27
7.3.2 EEPROM 和 MTP 寄存器描述 27
7.3.4 EEPROM 控制例程 30
8 中断系统 32
8.1 功能简介 32
8.2 中断逻辑 32
8.3 中断向量表 33
8.4 中断控制寄存器 33
8.5 外部中断 35
8.5.1 外部中断介绍 35
8.5.2 外部中断寄存器 35
8.5.3 外部中断控制例程 37
9 时钟系统 38
9.1 时钟系统介绍 38
9.1.1 时钟专用名称定义 39
9.1.2 内置 8MHz RC 振荡器（IRCH） 39
9.1.2 内置 24MHz RC 振荡器（TKRC） 39
9.1.3 内置 128KHz RC 振荡器（IRCL） 39
9.1.4 外部 32.768KHz 晶体谐振器（XOSCL） 39
9.2.3 时钟控制寄存器描述 40
9.3 系统时钟 41
9.3.1 系统时钟结构图 41
9.3.2 系统时钟控制寄存器描述 41
9.3.3 系统时钟控制方法及例程 43
10 供电和复位系统 44
10.1 供电系统 44
10.1.2 内部基准电压控制寄存器 45
10.2 复位系统 46
11 功耗管理 48

11.1 IDLE 模式 48
11.2 STOP 模式 48
11.3 低速运行模式 48
11.4 低功耗相关寄存器描述 49
11.5 低功耗模式控制例程 50
12 通用定时器（定时器 0,定时器 1,定时器 2） 52
12.1 定时器 0 52
12.1.1 定时器 0 介绍 52
12.1.2 定时器 0 寄存器描述 53
12.2 定时器 1 55
12.2.1 定时器 1 介绍 55
12.2.2 定时器 1 寄存器描述 56
12.3 定时器 2 57
12.3.1 定时器 2 介绍 57
12.3.2 定时器 2 寄存器描述 57
13 看门狗定时器（WDT） 59
13.1 看门狗定时器（WDT）功能简介 59
13.2 看门狗定时器（WDT）寄存器描述 59
13.3 看门狗定时器控制例程 61
14 TMC 定时器 62
14.1 TMC 功能简介 62
14.2 TMC 寄存器描述 62
14.3 TMC 控制例程 63
15 通用输入输出口（GPIO）及复用定义 64
15.1 功能简介 64
15.2 引脚寄存器描述 65
15.3 引脚控制例程 75
16 通用串行接口（UART1/UART2） 76
16.1 功能简介 76
16.2 UARTx 寄存器描述 77
17 I^2C 接口 81
17.1 功能简介 81
17.2 I^2C 主要特点 81
17.3 I^2C 功能描述 81
17.4 I^2C 通信引脚的映射 83
17.5 寄存器描述 83
18 PWM 87
18.1 PWM 功能简介 87
18.2 PWM 功能描述 87
18.3 PWM 寄存器描述 87
19 SPI 接口 90
19.1 功能简介 90
19.2 寄存器描述 92
20 模/数字转换器（ADC） 94
20.1 功能简介 94

20.2 主要特性 94
20.3 结构框图 94
20.4 功能描述 95
20.5 寄存器描述....95
21 电容式触摸按键（Touch Key） 98
21.1 功能简介 98
21.2 主要特性 98
21.3 功能描述 98
21.4 寄存器描述 98
21 低电压检测（LVD） 103
21.1 功能简介 103
21.2 功能描述 103
21.3 寄存器描述....104
21.4 LVD 控制例程 105
22 程序下载和仿真....106
22.1 程序下载 106
22.2 在线仿真 106
23 电气特性....107
23.1 极限参数 107
23.2 直流电气特性....107
23.3 ESD/EFT 特性 109
23.4 交流电气特性....109
23.5 内部 RC 时钟温度特性....110
24 封装类型....111
25 附录 118
附录 1 指令集速查表 118

1 概述

CA51M151 系列芯片是基于 1T 8051 内核的 8 位微控制器，不仅保留了传统 8051 芯片的基本特性，通常情况下运行速度比传统的 8051 芯片快 10 倍，性能更加优越。芯片内置 8 KB MTP 程序存储器，256Byte 内部 RAM，512Byte 外部 RAM 及 512Byte EEPROM。还集成了 26 路 12 位 ADC、 26 路 Touch Key（不需外接电容）、16 位 PWM、 I2C、2 路 UART、SPI、TMC、低电压检测(LVD)等功能模块。支持 PWM、I2C、UART 外设可以映射到任意管脚。I/O 推挽输出模式推电流 4 级可选、灌电流 2 级可选，软件可实现双扫 LED 屏显示功能。还支持 IDLE、STOP 和低速运行三种省电模式以适应不同功耗要求的应用。强大的功能及优越的抗干扰性能使其可广泛应用于各类家电及消费类产品。

2 基本特性

内核

- CPU：1T 8051，最高速度比传统 8051 快 10 倍
- 兼容 8051 指令集，双 DPTR 工作模式

存储器

- MTP 程序存储器: 8KB，大于 1000 次重复擦写
- EEPROM: 512 Byte，用于存储掉电需要保存的数据，大于 100000 次重复擦写
- RAM: 256 Byte 内部 RAM，512 Byte 外部 RAM

◆ 工作电压

- 工作电压：2.7V ~ 5.5V@Fosc=8MHz
  2.2V ~ 5.5V@Fosc=4MHz

◆ 工作温度

- 工作温度：-40℃~+85℃

◆ 时钟系统

- 外部低速振荡器： 32.768KHz
- 内置低速 RC 振荡器：128KHz
- 内置高速 RC 振荡器：8MHz，精度为±2%@5V/25℃（出厂校正）

◆ 定时器

- 3 个 16 位通用定时器: 定时器 0，定时器 1，定时器 2

◆ 通用输入输出口（GPIO）

- 支持最多 26 个 GPIO 口（不同型号会有不同）
- 支持推挽、开漏、上拉、下拉、高阻模式
- 推挽输出时，GPIO 的推电流有 4 级可选：12mA、10mA、8mA、6mA ; @VDD=5V
- 推挽输出时，GPIO 的灌电流有 2 级可选：16mA、8mA; @VDD=5V
- GPIO 可软件模拟成 1/2 BIAS 的 LCD 驱动，不需外接电阻

- **TMC 功能**
  - TMC 定时器的时钟源可选 IRCL 或 XOSCL
  - 中断时间最小单位，当时钟源选 IRCL 时为 512 个 IRCL 时钟周期，当时钟源选 XOSCL 时为 128 个 XOSCL 时钟周期
  - 可配置中断时间为 1-256 个最小单位时间
- **中断系统**
  - 7 个有效中断源
  - 两级中断优先级，支持中断嵌套
  - 5 个外部中断源 INT0~INT4
  - 外部中断触发沿选择：INT0~1（上升沿、下降沿）、INT2~4（上升沿、下降沿、双沿）
  - 中断输入引脚选择：INT0(P0.0)、INT1(P0.1)、INT2(P0.2)、INT3(P0.3)，INT4 可选除 P0.0~P0.3 外的任意 GPIO 引脚为中断输入引脚
- **模/数转换器（ADC）**
  - 支持 26 路 12 位 SAR ADC（不同型号会有不同）
  - 支持 2 种基准电压源：VDD 和内部基准
  - 选择内部电压为基准，可测量 VDD 电压
- **触摸按键（Touch Key）**
  - 内置触摸感应控制器，不需外接电容
  - 最大支持 26 路触摸通道（不同型号会有不同）
  - 高抗干扰性能，符合 EMC(CS 动态 10V，EFT 4KV)标准
  - 支持触摸中断
  - 支持触摸通道并联，可用于实现低功耗模式
- **PWM**
  - 支持 4 路 PWM 输出，每路可单独控制，在 16 位范围内可任意配置周期和占空比
  - 支持可直接输出内部时钟功能
  - 支持 PWM 中断功能
  - 每路 PWM 输出引脚可映射至不同的 GPIO 引脚
- **通用串行接口（UART1/UART2）**
  - 支持 2 路全双工串口
  - 支持 1 字节接收缓存
  - UART1/UART2 的 TX/RX 功能可映射至不同的 GPIO 引脚
- **I^2C 接口**
  - 内置 1 路 I^2C 接口，支持主从模式，支持标准/快速/高速模式
  - I^2C 引脚 SCL/SDA 功能可映射到任意的 GPIO 引脚
- **SPI 接口**
  - 内置 1 个 4 线 SPI 接口，支持主从模式
- **低电压检测（LVD）**
  - 检测电压可设置为 2.2V、2.5V、2.7V、3.0V、3.3V、3.6V、3.9V 和 4.2V。
  - 可设置低电压复位或中断

- 复位模式
  - 芯片支持多种复位源：上电/掉电复位，软复位，看门狗复位，低电压检测复位
- 看门狗
  - 27 位看门狗定时器，16 位调节精度，可配置看门狗复位或中断
- 程序下载和仿真
  - 支持 ISP 和 IAP
  - 支持在线仿真功能
- 低功耗
  - STOP 模式，电流<7uA
  - IDLE 模式，电流<33uA
- 封装类型：SOP28 /SSOP28/QFN28/QFN20/SOP20/TSSOP20/SOP16

锦锐科技

3 芯片型号功能介绍

表 3-1 CA51M151 系列具体型号功能特点

芯片型号	MTP ROM 容量[BYTE]	SRAM 容量[BYTE]	EEPROM 容量 [BYTE]	内部高速 RC 振荡器	内部低速 RC 振荡器	外部振荡器[32.768KHz]	GPIO 数量	通用 16 位定时器数量	UART 数量	I²C	SPI	16 bit PWM 通道数量	电容式触摸按键	12 位 ADC 通道数量	ISP 方式下载程序	片上仿真功能	工作电压[V]	封装形式
CA51M151S6A	8K	768	512	√	√	√	26	3	2	1	1	4	26	26	√	√	2.2~5.5	SOP28
CA51M151S6B	8K	768	512	√	√	√	26	3	2	1	1	4	26	26	√	√	2.2~5.5	SOP28
CA51M151P6A	8K	768	512	√	√	√	26	3	2	1	1	4	26	26	√	√	2.2~5.5	SSOP28
CA51M151N3	8K	768	512	√	√	√	26	3	2	1	1	4	26	26	√	√	2.2~5.5	QFN28
CA51M151S4A	8K	768	512	√	√	√	18	3	2	1	1	4	18	18	√	√	2.2~5.5	SOP20
CA51M151T3A	8K	768	512	√	√	√	18	3	2	1	1	4	18	18	√	√	2.2~5.5	TSSOP20
CA51M151N2	8K	768	512	√	√	√	18	3	2	1	1	4	18	18	√	√	2.2~5.5	QFN20
CA51M151S3A	8K	768	512	√	√	√	14	3	2	1	1	4	14	14	√	√	2.2~5.5	SOP16

4 系统框图

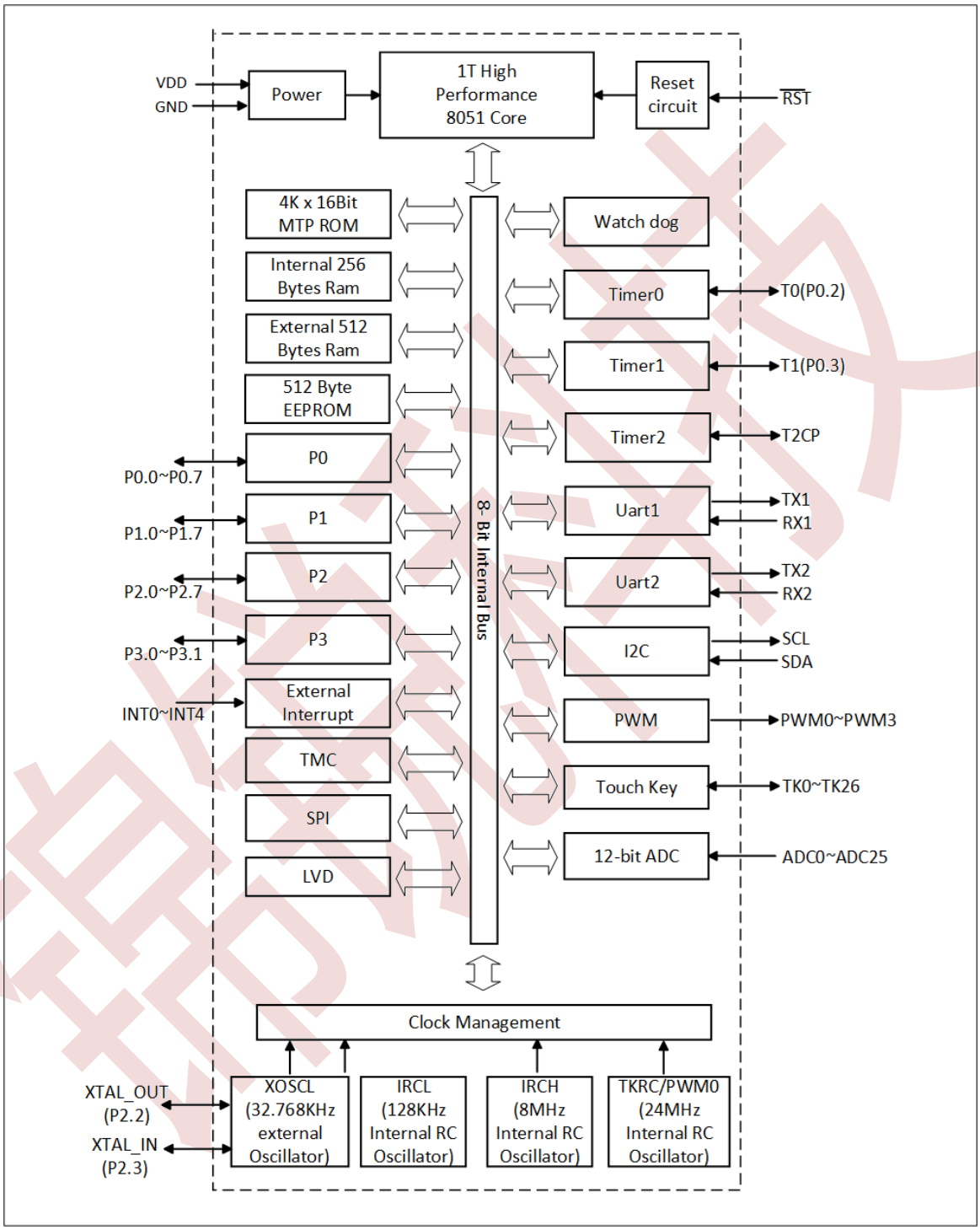


图 4-1-1 芯片框图

5 引脚封装及其描述

5.1 封装定义

备注：

(1)为便于设计应用，以下所有型号，UART1/UART2 的 RX/TX 通过设置可以映射至不同的 GPIO 管脚，详细见“15-2-7 引脚复用功能映射表”描述。

(2)为便于设计应用，以下所有型号，IIC 的 SCL/SDA 通过设置可以映射到任意的 GPIO 管脚，详细见“15-2-7 引脚复用功能映射表”描述。

型号：**CA51M151S6A/CA51M151P6A**

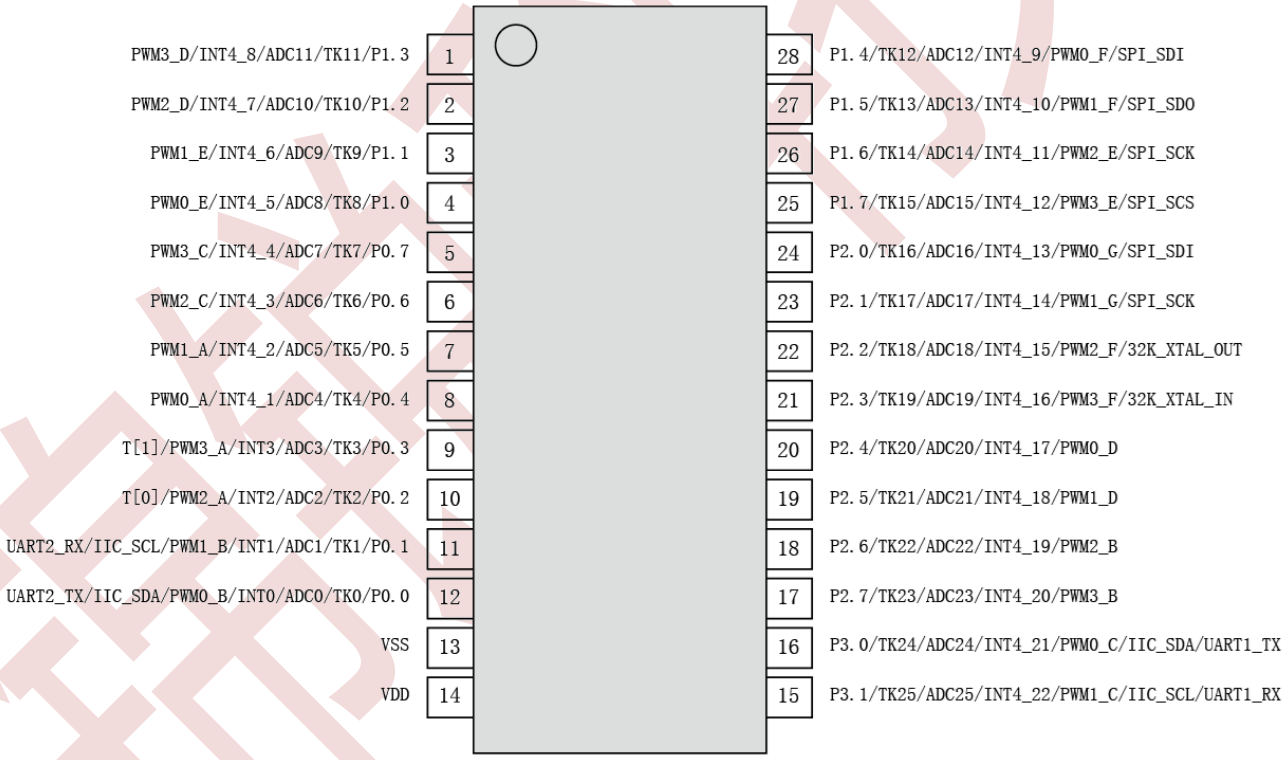


图 5-1-1 SOP28/SSOP28 封装引脚图

*备注：CA51M151S6A/CA51M151P6A 程序下载及仿真接口：*P1.4(IIC_SDA)、P1.6(IIC_SCL)。

型号：**CA51M151S6B**

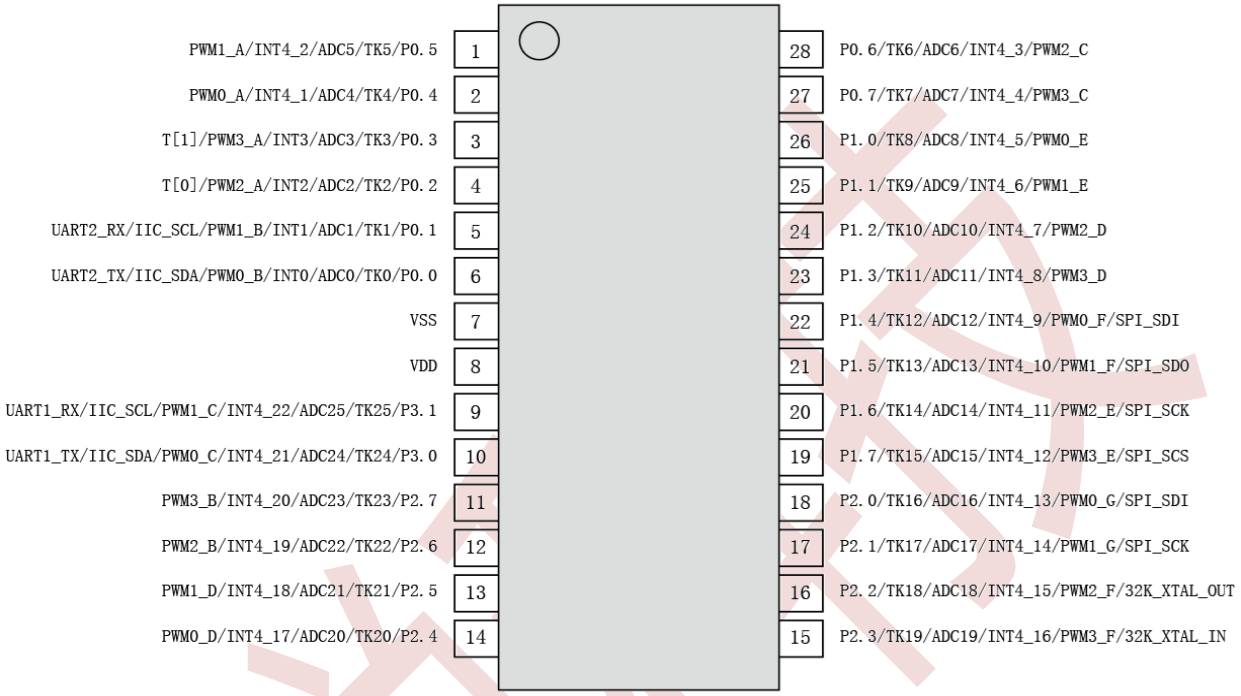


图 5-1-2 SOP28 封装引脚图

*备注：**CA51M151S6B** 程序下载及仿真接口：***P3.1(IIC_SDA)、P3.0(IIC_SCL)。**

型号：**CA51M151N3**

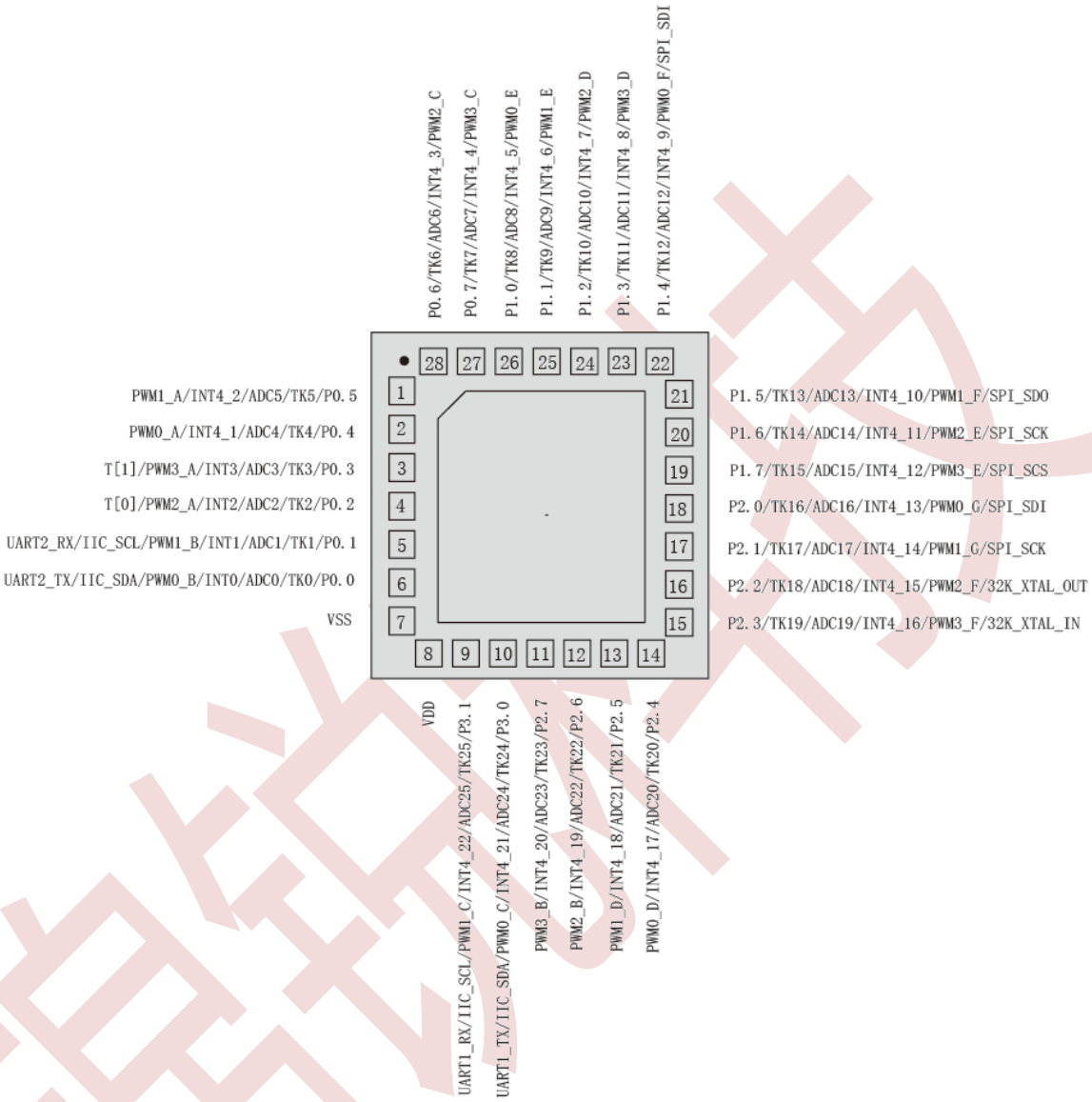


图 5-1-3 QFN28 封装引脚图

*备注：**CA51M151N3** 程序下载及仿真接口：***P0.0(IIC_SDA)、P0.1(IIC_SCL)。

型号：**CA51M151S4A/CA51M151T3A**

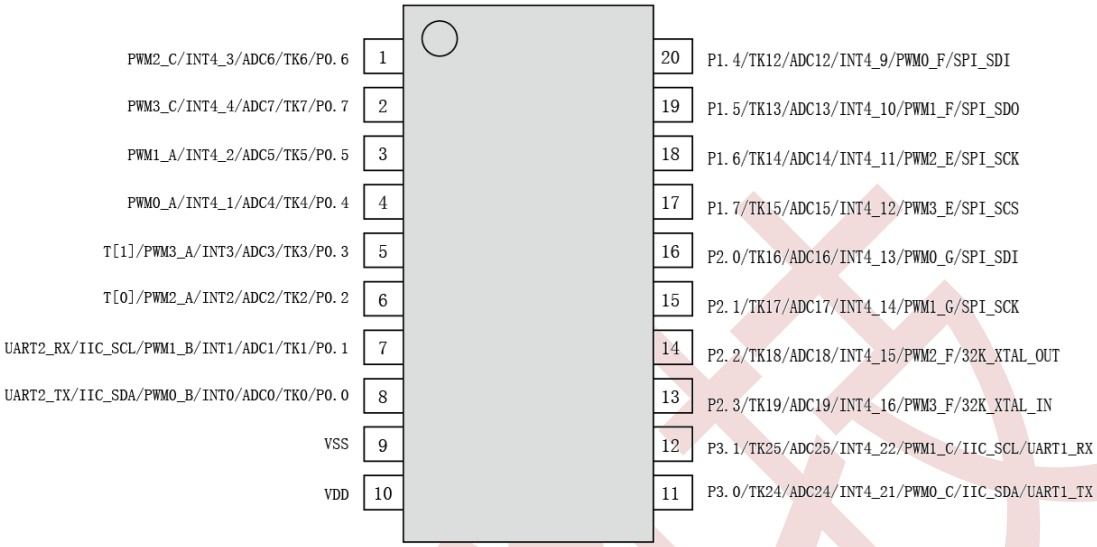


图 5-1-4 SOP20/TSSOP20 封装引脚图

备注：　CA51M151S4A 程序下载及仿真接口：**P1.4(IIC_SDA)、P1.6(IIC_SCL)。**

CA51M151T3A 程序下载及仿真接口：**P3.0(IIC_SDA)、P3.1(IIC_SCL)。**

型号：**CA51M151N2**

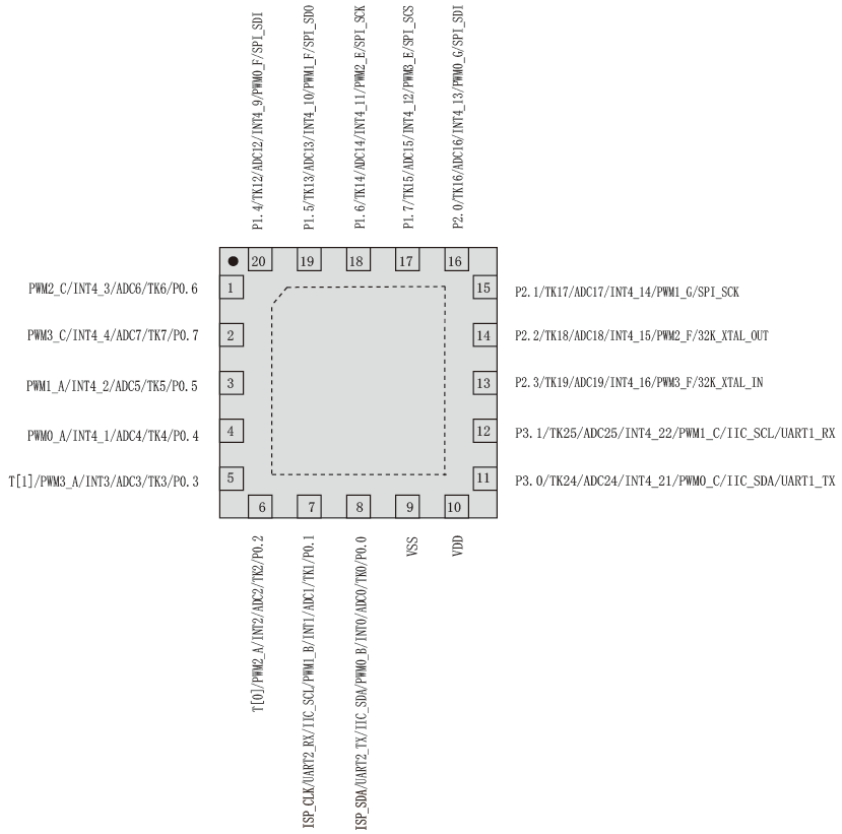


图 5-1-5 QFN20 封装引脚图

*备注：CA51M151N2 程序下载及仿真接口：*P0.0(IIC_SDA)、P0.1(IIC_SCL)。

型号：**CA51M151S3A**

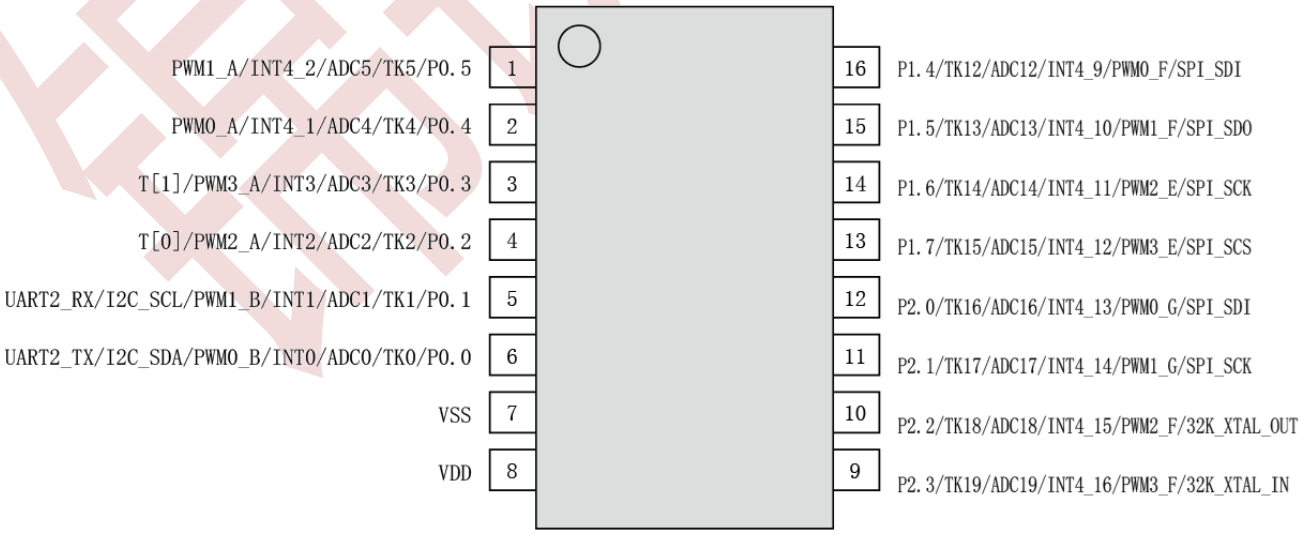


图 5-1-6 SOP16 封装引脚图

*备注：CA51M151S3A 程序下载及仿真接口：*P1.4(IIC_SDA)，P1.6(IIC_SCL)。

5.2 引脚描述

表 5-2-1 引脚描述

引脚序号					管脚名称	管脚功能	默认功能
SOP28 SSOP28 (A)	SOP28 (B)	QFN28	TSSOP20 (A) SOP20 QFN20	SOP16 (A)			
7	1	1	3	1	P0.5/INT4_2/PWM1_A/ADC5/TK5/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM1 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART2 传输口 T2CP 信号输入	通用双向 I/O 口
8	2	2	4	2	P0.4/INT4_1/PWM0_A/ADC4/TK4/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM0 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART2 传输口 T2CP 信号输入	通用双向 I/O 口
9	3	3	5	3	P0.3/INT3/T1/PWM3_A/ADC3/TK3 / I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT3 信号输入 T1 输入端口 PWM3 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART2 传输口 T2CP 信号输入	通用双向 I/O 口
10	4	4	6	4	P0.2/INT2/T0/PWM2_A/ADC2/TK2 / I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT2 信号输入 T0 输入端口 PWM2 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART2 传输口 T2CP 信号输入	通用双向 I/O 口
11	5	5	7	5	P0.1/INT1/PWM1_B/ADC1/TK1/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT1 信号输入 PWM1 信号输出 ADC 模拟通道输入	通用双向 I/O 口

						触摸按键模拟通道输入 I²C 传输口 UART2 传输口 T2CP 信号输入	
12	6	6	8	6	P0.0/INT0/PWM0_B/ADC0/TK0/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT0 信号输入 PWM0 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART2 传输口 T2CP 信号输入	通用双向 I/O 口
13	7	7	9	7	VSS	电源地管脚	电源地引脚
14	8	8	10	8	VDD	芯片供电管脚	芯片供电管脚
15	9	9	12	-	P3.1/INT4_22/PWM1_C/ADC25/TK 25/I2C_SDA/I2C_SCL/UART1_RX/ UART1_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM1 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入	CA51M151S6B / CA51M151T3 为 I²C 传输口； 其它型号为通 用双向 I/O 口
16	10	10	11	-	P3.0/INT4_21/PWM0_C/ADC24/TK 24/I2C_SDA/I2C_SCL/UART1_RX/ UART1_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM0 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入	CA51M151S6B / CA51M151T3 为 I²C 传输口； 其它型号为通 用双向 I/O 口
17	11	11	-	-	P2.7/INT4_20/PWM3_B/ADC23/TK 23/I2C_SDA/I2C_SCL/UART1_RX/ UART1_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM3 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入	通用双向 I/O 口
18	12	12	-	-	P2.6/INT4_19/PWM2_B/ADC22/TK 22/I2C_SDA/I2C_SCL/UART1_RX/ UART1_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM2 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入	通用双向 IO 口

						I²C 传输口 UART1 传输口 T2CP 信号输入	
19	13	13	-	-	P2.5/INT4_18/PWM1_D/ADC21/TK21/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM1 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入	通用双向 IO 口
20	14	14	-	-	P2.4/INT4_17/PWM0_D/ADC20/TK20/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM0 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入	通用双向 IO 口
21	15	15	13	9	P2.3/INT4_16/PWM3_F/ADC19/TK19/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP/XTAL_IN	通用双向 I/O 口 INT4 信号输入 PWM3 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入 外部低速晶振输入端口	通用双向 IO 口
22	16	16	14	10	P2.2/INT4_15/PWM2_F/ADC18/TK18/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP/XTAL_OUT	通用双向 I/O 口 INT4 信号输入 PWM2 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入 外部低速晶振输出端口	通用双向 IO 口
23	17	17	15	11	P2.1/INT4_14/PWM1_G/ADC17/TK17/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP/SPI_SCK	通用双向 I/O 口 INT4 信号输入 PWM1 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入	通用双向 IO 口

						SPI_SCK 端口	
24	18	18	16	12	P2.0/INT4_13/PWM0_G/ADC16/TK16/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP/SPI_SDI	通用双向 I/O 口 INT4 信号输入 PWM0 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入 SPI_SDI 端口	通用双向 IO 口
25	19	19	17	13	P1.7/INT4_12/PWM3_E/ADC15/TK15/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP/SPI_SCS	通用双向 I/O 口 INT4 信号输入 PWM3 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入 SPI_SCS 端口	通用双向 IO 口
26	20	20	18	14	P1.6/INT4_11/PWM2_E/ADC14/TK14/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP/SPI_SCK	通用双向 I/O 口 INT4 信号输入 PWM2 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入 SPI_SCK 端口	CA51M151S6B / CA51M151T3 为通用双向 IO 口；其它型号为 I²C 传输口
27	21	21	19	15	P1.5/INT4_10/PWM1_F/ADC13/TK13/I2C_SDA/I2C_SCL/UART1_RX/UART1_TX/T2CP/SPI_SDO	通用双向 I/O 口 INT4 信号输入 PWM1 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART1 传输口 T2CP 信号输入 SPI_SDO 端口	通用双向 IO 口
28	22	22	20	16	P1.4/INT4_9/PWM0_F/ADC12/TK12/ I2C_SDA/I2C_SCL/UART2_RX/UART2_TX/T2CP/SPI_SDI	通用双向 I/O 口 INT4 信号输入 PWM0 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口	CA51M151S6B / CA51M151T3 为通用双向 IO 口；其它型号为 I²C 传输口

						UART2 传输口 T2CP 信号输入 SPI_SDI 端口	
1	23	23	-	-	P1.3/INT4_8/PWM3_D/ADC11/TK11/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM3 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I^2C 传输口 UART2 传输口 T2CP 信号输入	通用双向 IO 口
2	24	24	-	-	P1.2/INT4_7/PWM2_D/ADC10/TK10/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM2 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I^2C 传输口 UART2 传输口 T2CP 信号输入	通用双向 IO 口
3	25	25	-	-	P1.1/INT4_6/PWM1_E/ADC9/TK9/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM1 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I^2C 传输口 UART2 传输口 T2CP 信号输入	通用双向 IO 口
4	26	26	-	-	P1.0/INT4_5/PWM0_E/ADC8/TK8/ I2C_SDA/I2C_SCL/UART2_RX /T2CP	通用双向 I/O 口 INT4 信号输入 PWM0 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I^2C 传输口 UART2 数据接收端口 T2CP 信号输入	通用双向 IO 口
5	27	27	2	-	P0.7/INT4_4/PWM3_C/ADC7/TK7/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM3 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I^2C 传输口 UART2 传输口 T2CP 信号输入	通用双向 IO 口

6	28	28	1	-	P0.6/INT4_3/PWM2_C/ADC6/TK6/ I2C_SDA/I2C_SCL/UART2_RX/ UART2_TX/T2CP	通用双向 I/O 口 INT4 信号输入 PWM2 信号输出 ADC 模拟通道输入 触摸按键模拟通道输入 I²C 传输口 UART2 传输口 T2CP 信号输入	通用双向 IO 口

备注：信号引脚复用功能设置方法详见表 15-2-7

锐芯科技

6 中央处理器（CPU）

6.1 CPU 简介

CA51M151 系列芯片采用单周期 8051 CPU，与原来的 MCS-51 指令集完全兼容。CPU 采用流水线结构，通常情况下，单周期 8051 CPU 的运行速度比标准 8051 处理器快 10 倍。

CPU 有以下特性：

- 1T 8051 CPU
- 兼容 8051 指令集，见指令集附录
- 双 DPTR，可用于数据快速搬移

6.2 寄存器描述

程序计数器 PC

程序计数器 PC 寄存器为 16 位，是专门用来控制指令执行顺序的寄存器，它没有寄存器地址。单片机上电或复位后，PC 值为 0，单片机从零地址开始执行程序。

累加器 ACC

累加器 ACC 是一个常用的专用寄存器，指令系统中采用 A 作为累加器的助记符，常用于存放算术或逻辑运算的操作数及运算结果。

通用寄存器 B

B 在乘除法运算中需要和 ACC 配合使用。MUL AB 指令把 ACC 和 B 中 8 位无符号数相乘，所得的 16 位乘积的低字节存放在 A 中，高字节存放在 B 中。DIV AB 指令用 B 除以 A，整数商存放在 A 中，余数存放在 B 中。寄存器 B 还可以用作通用暂存寄存器。

堆栈指针 SP

堆栈指针 SP 是一个 8 位专用寄存器。它指示出堆栈顶部在内部 RAM 块中的位置。系统 复位后，SP 初始化位 07H，使得堆栈事实上由 08H 单元开始，考虑 08H~1FH 单元分别 属于工作寄存器组 1~3，若在程序设计中用到这些区，则最好 SP 改变为 80H 或更大的为宜。

数据指针 DPTR

数据指针 DPTR0/DPTR1 是两个 16 位专用寄存器，它们的高位字节寄存器用 DP0H/DP1H 表示，低位字节寄存器用 DP0L/DP1L 表示，通过 DPS(PSW.1)可选择使用 DPTR0/DPTR1。每个 DPTR 既可以作为一个 16 位寄存器来处理，也可以作为 2 个独立的 8 位寄存器 DP0H/DP1H 和 DP0L/DP1L 来处理。

状态寄存器 PSW

状态寄存器 PSW 是 CPU 的状态寄存器。在 CPU 做算术运算或者逻辑运算时，对应的 PSW 状态位会发生改变。

表 6-2-1 累加器 ACC

E0H	7	6	5	4	3	2	1	0
ACC	ACC[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

表 6-2-2 通用寄存器 B

F0H	7	6	5	4	3	2	1	0
B	B[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

表 6-2-3 堆栈指针 SP

81H	7	6	5	4	3	2	1	0
SP	SP[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	1	1	1

表 6-2-4 数据指针 DP0L

82H	7	6	5	4	3	2	1	0
DP0L	DP0L[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

表 6-2-5 数据指针 DP0H

83H	7	6	5	4	3	2	1	0
DP0H	DP0H[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

表 6-2-6 数据指针 DP1L

84H	7	6	5	4	3	2	1	0
DP1L	DP1L[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

表 6-2-7 数据指针 DP1H

85H	7	6	5	4	3	2	1	0
DP1H	DP1H[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

表 6-2-8 状态寄存器 PSW

D0H	7	6	5	4	3	2	1	0
PSW	CY	AC	F0	RS[1:0]		OV	DPS	P
R/W	R/W	R/W	R/W	R/W		R/W	R	R
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	CY	进位标志位 0：算术或逻辑运算中，没有进位或借位发生 1：算术或逻辑运算中，有进位或借位发生
6	AC	辅助进位标志位 0：算术或逻辑运算中，没有辅助进位或借位发生 1：算术或逻辑运算中，有辅助进位或借位发生
5	F0	F0 标志位 用户自定义标志位
4~3	RS	R0~R7 寄存器页选择位 00：页 0（映射到 00H-07H) 01：页 1（映射到 08H-0FH) 10：页 2（映射到 10H-17H) 11：页 3（映射到 18H-1FH)
2	OV	溢出标志位 0：没有溢出发生 1：有溢出发生
1	DPS	DPTR 选择寄存器，0 为选择 DPTR0，1 为选择 DPTR1
0	P	奇偶校验位 0：累加器 A 值为 1 的位数为偶数 1：累加器 A 值为 1 的位数为奇数

表 6-2-9 寄存器 SPMAX

F3H	7	6	5	4	3	2	1	0
SPMAX	SPMAX[7:0]							
R/W	R	R	R	R	R	R	R	R
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	SPMAX	寄存器 SPMAX 用于记录 SP 的最大值，用户在应用程序中可查看此寄存器来判断堆栈有没有溢出风险						

锐芯科技

7 存储器系统

7.1 随机数据存储器（RAM）

CA51M151 系列芯片提供了 256 字节内部 RAM 和 512 字节外部 RAM，存储器地址分配如下：

- 低位 128 字节的内部 RAM（地址：00H ~ 7FH）可直接寻址或间接寻址。
- 高位 128 字节的内部 RAM（地址：80H ~ FFH）只能间接寻址。
- 外部 512 字节外部 RAM（地址：0000H ~ 01FFH）可通过 MOVX 指令间接寻址。

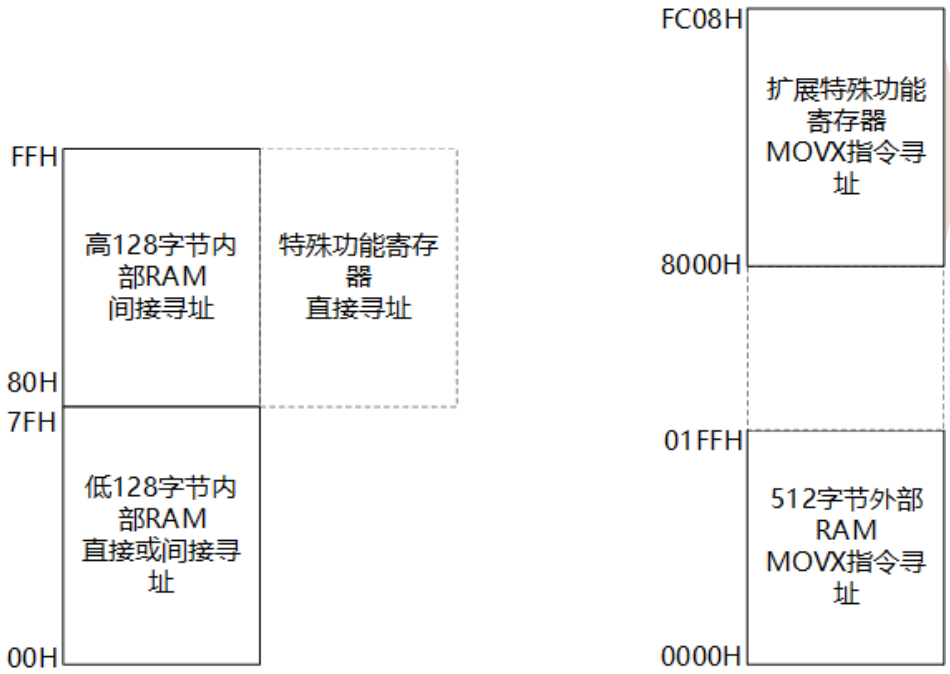


图 7-1-1　RAM 组织结构图

7.2 特殊功能寄存器（SFR）

CA51M151 系列芯片提供了兼容传统 8051 的 SFR 分布，SFR 和高 128 字节内部 RAM 共用地址 80H ~ FFH，只能直接寻址，SFR 映射如表 7-2-1 所示。

表 7-2-1 特殊功能寄存器（SFR）映射表

	可位寻址	不可位寻址						
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
F8H	TKST	TKCFG1	TKCFG2	TKCFG3	TKDL	TKDH	TKPULL	TKCFG4
F0H	B	CBYTE	TKPULLTRIM	SPMAX	TKIE	-	-	-
E8H	LVDCON	-	-	-	-	-	-	-
E0H	ACC	-	-	-	-	-	-	-
D8H	UDCKS1	-	-	-	-	-	-	-
D0H	PSW	EP0CON	EP1CON	EP2CON	EPIF	TMCON	TMSNU	-
C8H	CKCON	CKDIV	IHCFG	TKCCFG	-	-	-	-
C0H	I2CCON	I2CADR	I2CADM	I2CCCR	I2CDAT	I2CSTA	I2CFLG	-
B8H	IP	S2CON	S2BUF	S2RELL	S2RELH	UDCKS2	-	-
B0H	P3	-	-	-	-	-	-	-
A8H	IE	SPCON	SPDAT	SPSTA	-	-	-	-
A0H	P2	WDFLG	WDVTHL	WDVTHH	T2CON	T2CRH	T2CRL	WDCON
98H	S1CON	S1BUF	PWMDUTH	PWMDUTL	PWMAIF	PWMBIF	PWMCIF	INDEX
90H	P1	PWMEN	PWMIF	PWMCMAX	PWMCON	PWMCKD	PWMDIVL	PWMDIVH
88H	TCON	TMOD	TL0	TL1	TH0	TH1	IDLST	STPST
80H	P0	SP	DP0L	DP0H	DP1L	DP1H	PWCON	PCON

由于 SFR 地址空间有限，CA51M151 系列芯片在外部 RAM 地址空间增加了扩展特殊功能寄存器，扩展特殊功能寄存器映射如图表 7-2-2 所示。

表 7-2-2 扩展特殊功能寄存器映射表

	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
8000H	P00F	P01F	P02F	P03F	P04F	P05F	P06F	P07F
8008H	P10F	P11F	P12F	P13F	P14F	P15F	P16F	P17F
8010 H	P20F	P21F	P22F	P23F	P24F	P25F	P26F	P27F
8018H	P30F	P31F	-	-	-	-	-	-
8060H	ADCON	ADCFGL	ADCDL	ADCDH	ADCALL	ADCALH	-	-
8068 H	S1RELL	S1RELH	-	-	-	-	-	-
8120H	P00C	P01C	P02C	P03C	P04C	P05C	P06C	P07C
8128H	P10C	P11C	P12C	P13C	P14C	P15C	P16C	P17C
8130H	P20C	P21C	P22C	P23C	P24C	P25C	P26C	P27C
8138H	P30C	P31C	-	-	-	-	-	-
FC00H	MECON	FSCMD	FSDAT	LOCK	-	PTSL	PTSH	-
FC08H	-	-	-	-	-	-	-	-

7.3 EEPROM 和 MTP 存储器

7.3.1 功能简介

MTP 程序存储器 8KB，大于 1000 次重复擦写。内置 512Byte EEPROM，大于 100000 次重复擦写，用于存储掉电需要保存的数据。

- MTP 程序存储器由若干个页组成，页是进行擦除操作的最小单位，每页大小为 32 字节。写操作以页为单位进行，必须一次性写入 32 字节，不支持单字节写入，每次写操作都是完成一个页的数据。读操作不需要以页为单位，可读单字节或连续读取多字节。
- EEPROM 由若干个页组成，页是进行擦除操作的最小单位，每页大小为 32 字节。写操作以页为单位进行，必须一次性写入 32 字节，不支持单字节写入，每次写操作都是完成一个页的数据。读操作不需要以页为单位，可读单字节或连续读取多字节。

7.3.2 EEPROM 和 MTP 寄存器描述

表 7-3-2-1 寄存器 MECON

FC00H	7	6	5	4	3	2	1	0
MECON	-	DPSTB	-	-	-	-	-	BOOT
R/W	-	R/W	-	-	-	-	-	R/W
初始值	-	0	-	-	-	-	-	0

位编号	位符号	说明
7	-	-
6	DPSTB	IDLE/STOP 模式下 EEPROM 和 MTP 进入睡眠模式控制位 0：IDLE/STOP 模式下，EEPROM 和 MTP 处于正常工作模式 1：IDLE/STOP 模式下，EEPROM 和 MTP 进入睡眠模式 *备注：如果 DPSTB=1，当芯片进入 IDLE/STOP 模式，*EEPROM 和 MTP *也同时进入睡眠模式，当芯片退出 IDLE/STOP 模式，*EEPROM 和 MTP *也同时退出睡眠模式。*
5~1	-	-
0	BOOT	设置软复位后程序启动空间选择位域 0：软复位后程序从 MTP 启动运行 1：软复位后程序从 XRAM 启动运行

表 7-3-2-2 寄存器 FSCMD

FC01H	7	6	5	4	3	2	1	0
FSCMD	-	-	-	-	CLRPL	CMD[2:0]		
R/W	-	-	-	-	0	R/W		
初始值	-	-	-	-	0	0	0	0

位编号	位符号	说明
7	-	-
6~4	-	-
3	CLRPL	清除 EEPROM/MTP latch 中数据
2~0	CMD	命令寄存器 000：无操作 100：NC 001：读 EEPROM 010：写 EEPROM 011：擦除 EEPROM 一个页 101：读 MTP 110：写 MTP 111：擦除 MTP 一个页 *备注：* *1. 擦除和写命令执行后 CMD 自动清零。* *2. 读命令写入后 CMD 保持不变然后通过读写 FSDAT 完成。* *3. EEPROM 擦除后为不定态*

表 7-3-3-3 寄存器 FSDAT

FC02H	7	6	5	4	3	2	1	0
FSDAT	FSDAT[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	FSDAT	EEPROM/MTP 数据寄存器

表 7-3-2-4 寄存器 LOCK

FC03H	7	6	5	4	3	2	1	0
LOCK								
R	-	REPE	-	-	FLKF	PLKF	DLKF	ILKF
W	LOCK[7:0]							
初始值	-	0	-	-	0	0	0	0

位编号	位符号	说明
写操作		
7~0	LOCK	28H：对 EEPROM/MTP 可编程区解锁 29H：对 MTP 解锁 2AH：对 EEPROM 解锁 AAH：EEPROM/MTP 加锁，不能进行写擦操作
读操作		

7~4	-	
3	FLKF	可编程区解锁标志，1 表示已解锁
2	PLKF	程序区解锁标志，1 表示已解锁
1	DLKF	数据区解锁标志，1 表示已解锁
0	-	-

表 7-3-2-5 寄存器 PTS

FC05H	7	6	5	4	3	2	1	0
PTSL	PTS[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
FC06H	7	6	5	4	3	2	1	0
PTSH	-	-	PTS[13:8]					
R/W	-	-	R/W					
初始值	-	-	0	0	0	0	0	0

位编号	位符号	说明
15~14	-	-
13~0	PTS	目标地址指针寄存器，写 FSDAT 操作时，数据会写入 PTS[5:0]相关的 EEPROM/MTP 锁存器中暂存，PTS[5:0]对应实际写操作的低 6 位；发送写命令时，需要设置相关的 page 地址 PTS[13:6]。 每次读、写、擦操作时最好重新配置一次 PTS 地址，连续读取操作时，可以只设置连续读取操作的首地址操作即可。

7.3.4 EEPROM 控制例程

◆ **EEPROM 页擦除**

例如，需要擦除 EEPROM 第 n 页，程序如下：

```
---------------------------------------------------------------------------------
unsigned char i;
unsigned int address;
address = 0x20*n;
FSCMD = 8;                                  //擦除 Latch 中的数据
while(FSCMD != 0){}
PTSH = (unsigned char)(address>>8);         //填写高位地址
PTSL = (unsigned char)(address);            //填写低位地址
for(i=0; i<0x20; i++)
{
    FSDAT = 0xAA;
}
PTSH = (unsigned char)(address>>8);         //填写高位地址
PTSL = (unsigned char)(address);            //填写低位地址
FSCMD = 0;                                  //设置 CMD 为 0
LOCK = 0x2A;                                //EEPROM 解锁
PTSH = (unsigned char)(address >>8);        //填写高位地址
PTSL = (unsigned char)(address);            //填写低位地址
FSCMD = 3;                                  //设置数据区擦除命令
LOCK = 0xAA;                                //EEPROM 加锁
---------------------------------------------------------------------------------
```

备注：页序号 n=0、1、2……。

◆ **EEPROM 页写入数据**

例如，往 EEPROM 第 n 页写入数据 0xAA，程序如下：

```
---------------------------------------------------------------------------------
unsigned char i;
unsigned int address;
address = 0x20*n;
FSCMD = 0;                                  //设置 CMD 为 0
LOCK = 0x2A;                                //数据空间解锁
PTSH = 0;                                   //设置 page latch 起始地址
PTSL = 0;                                   //设置 page latch 起始地址
FSCMD = 8;                                  //设置擦除 latch
for(i=0; i<0x20; i++)
{
    FSDAT = 0xAA;                           //连续写入 1 page 的数据
}
PTSH = (unsigned char)(address >>8);        //设置数据首地址高 8 位
PTSL = (unsigned char)(address);            //设置数据首地址低 8 位
FSCMD = 2;                                  //设置写命令
LOCK = 0xAA;                                //EEPROM 加锁
---------------------------------------------------------------------------------
```

备注：

1. 页序号 n=0、1、2……。

2. 当连续写入数据时，只需设置首地址，每次写 FSDAT 后，数据指针寄存器 PTS 会自动累加。

3. 数据写入只能以页为单位，每次必须写入 32 字节。

◆ **EEPROM 读出数据**

例如，从地址 n 的位置开始，读出 Length 个数据到 dataBuf，程序如下：

```
---------------------------------------------------------------------------------------
unsigned char dataBuf[32];
unsigned int i;
FSCMD = 0;
PTSH = (unsigned char)(n>>8);                    //填写高位地址
PTSL = (unsigned char)(n);                       //填写低位地址
FSCMD = 1;                                       //执行读操作
for(i = 0; i < Length; i++)
{
    dataBuf[i]= FSDAT;
}
FSCMD = 0;
LOCK = 0xAA;                                     //对 FLASH 加锁
---------------------------------------------------------------------------------------
```

备注：

1. 地址 n = 0~0x1FF，长度 Length = 1~0x200。

2. 当连续读出数据时，只需设置首地址，每次读 FSDAT 后，数据指针寄存器 PTS 会自动累加。

3. 数据读出不需要以页为单位，可以连续读取多字节数据。

8 中断系统

8.1 功能简介

CA51M151 系列芯片有一个增强的中断控制系统，共有 7 个中断入口，每个中断入口有若干中断源，每个中断源有 2 级中断优先级。每个中断源都有独立的中断向量、优先级设置位、中断使能位、中断标志。CPU 在响应中断后，进入该中断对应的中断服务程序，接到 RETI 指令后将返回中断前状态。如果同时有多个有效中断产生中断请求，CPU 将根据设置的中断优先级依次响应；如果优先级相同，则根据它们的自然优先级（中断入口地址从低到高）依次响应。

8.2 中断逻辑

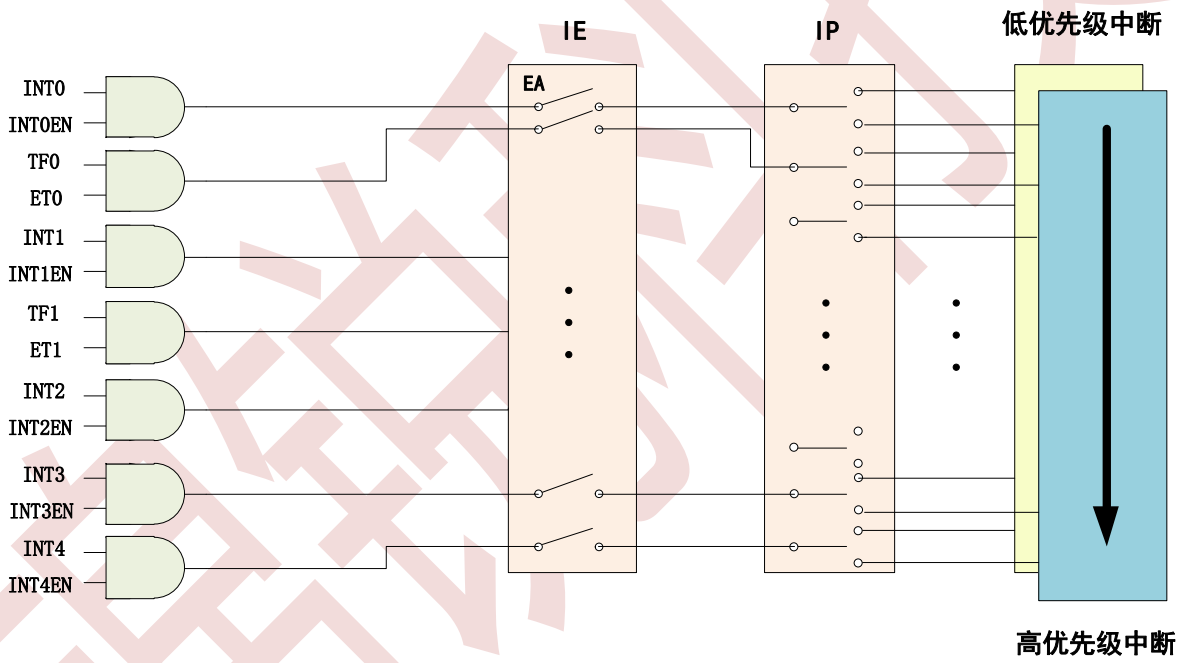


表 8-2-1 中断逻辑图

8.3 中断向量表

表 8-3-1 中断向量表

中断	中断源	向量	默认优先级
INT0	INT0	03H	0
TF0	定时器 0	0BH	1
INT1	INT1	13H	2
TF1	定时器 1	1BH	3
INT2	外部中断 2/UART1/ ADC 中断/PWM 中断/SPI 中断	23H	4
INT3	外部中断 3/定时器 2/触摸中断/TMC 中断	2BH	5
INT4	外部中断 4/ UART2/WDT 中断/I2C 中断/LVD 中断	33H	6

8.4 中断控制寄存器

表 8-4-1 寄存器 IE

A8H	7	6	5	4	3	2	1	0
IE	EA	INT4EN	INT3EN	INT2EN	ET1	EX1	ET0	EX0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	EA	全局中断使能控制位 0：关闭 1：打开
6	INT4EN	中断 4 使能控制位（中断 4 用于 UART2/WDT/I2C/LVD/外部中断 4） 0：关闭 1：打开
5	INT3EN	中断 3 使能控制位（中断 3 用于定时器 2/ TK/ TMC/外部中断 3） 0：关闭 1：打开
4	INT2EN	中断 2 使能控制位（中断 2 用于 UART1/ ADC/ PWM/ SPI/外部中断 2） 0：关闭 1：打开
3	ET1	定时器 1 中断使能控制位 0：关闭 1：打开
2	EX1	中断 1 使能控制位（中断 1 用于外部中断 1） 0：关闭 1：打开

1	ET0	定时器 0 中断使能控制位 0：关闭 1：打开
0	EX0	中断 0 使能控制位（中断 0 用于外部中断 0） 0：关闭 1：打开

备注：IE 的使能控制位是对应中断向量的，各中断源的中断开关也要另外打开。例如：要开启外部中断 2 的中断，除了设置 INT2EN 为 1，EPIE0（外部中断 2 使能位）也要设为 1。

表 8-4-2 寄存器 IP

B8H	7	6	5	4	3	2	1	0
IP	-	PX4	PX3	PX2	PT1	PX1	PT0	PX0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	0	0	0	0	0	0	0

位编号	位符号	说明
7	-	-
6	PX4	中断 INT4 优先级控制位 0：低优先级 1：高优先级
5	PX3	中断 INT3 优先级控制位 0：低优先级 1：高优先级
4	PX2	中断 INT2 优先级控制位 0：低优先级 1：高优先级
3	PT1	定时器 1 优先级控制位 0：低优先级 1：高优先级
2	PX1	外部中断 1 优先级控制位 0：低优先级 1：高优先级
1	PT0	定时器 0 优先级控制位 0：低优先级 1：高优先级
0	PX0	外部中断 0 优先级控制位 0：低优先级 1：高优先级

8.5 外部中断

8.5.1 外部中断介绍

除了标准 8051 的 INT0 和 INT1 以外，系统还扩展了 3 个中断入口 INT2~INT4 作为外部中断。每个外部中断都可以用于 STOP 模式唤醒。EPIF 为 INT2~INT4 外部中断状态寄存器。INT2~INT4 对应的各个配置寄存器为 EP0CON~EP2CON。

中断输入引脚选择：INT0(P0.0)、INT1(P0.1)、INT2(P0.2)、INT3(P0.3)， INT4 可选择除 P0.0~ P0.3 外的其它任意 I/O 引脚为中断触发源输入引脚。INT0~INT1 可选择上升沿或下降沿触发中断，INT2~INT4 可选择上升沿、下降沿或双沿触发中断。

备注：

1. INT0 和 INT1 可选择上升沿或下降沿触发，选择位分别为 IT0 和 IT1，详见寄存器 TCON 相关描述。

8.5.2 外部中断寄存器

表 8-5-1 寄存器 EPIF

D4H	7	6	5	4	3	2	1	0
EPIF	-	-	-	-	-	EPIF2	EPIF1	EPIF0
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

位编号	位符号	说明
7~3	-	-
2	EPIF2	外部中断 4 中断标志位，写 1 清零
1	EPIF1	外部中断 3 中断标志位，写 1 清零
0	EPIF0	外部中断 2 中断标志位，写 1 清零

表 8-5-2 寄存器 EPCON

D1H	7	6	5	4	3	2	1	0
EP0CON	EPIE0	EPPL0		-	-	-	-	-
R/W	R/W	R/W	RW	-	-	-	-	-
初始值	0	0	0	-	-	-	-	-

D2H	7	6	5	4	3	2	1	0
EP1CON	EPIE1	EPPL1		-	-	-	-	-
R/W	R/W	R/W	R/W	-	-	-	-	-
初始值	0	0	0	-	-	-	-	-

D3H	7	6	5	4	3	2	1	0
EP2CON	EPIE2	EPPL2		EPPS2				
R/W	R/W	R/W	R/W	R/W				
初始值	0	0	0	0	0	0	0	0

备注：下表中的“n”表示 0/1/2

位编号	位符号	说明
7	EPIEn	外部中断使能位 0：关闭 1：打开 *备注：n=0/1/2 分别对应外部中断 2/3/4。*
6~5	EPPLn	外部中断触发沿选择位 00：上升沿 01：下降沿 1x：双沿 *备注：n=0/1/2 分别对应外部中断 2/3/4。* *x=0/1*
4~0	EPPS2	外部中断 4 选择中断输入口 5’h00：表示选择 P1.0； 5’h01：表示选择 P1.1； 5’h02：表示选择 P1.2； 5’h03：表示选择 P1.3； 5’h04：表示选择 P1.4； 5’h05：表示选择 P1.5； 5’h06：表示选择 P1.6； 5’h07：表示选择 P1.7； 5’h08：表示选择 P2.0； 5’h09：表示选择 P2.1； 5’h0A：表示选择 P2.2； 5’h0B：表示选择 P2.3； 5’h0C：表示选择 P2.4； 5’h0D：表示选择 P2.5； 5’h0E：表示选择 P2.6； 5’h0F：表示选择 P2.7； 5’h10：表示选择 P0.4； 5’h11：表示选择 P0.5； 5’h12：表示选择 P0.6； 5’h13：表示选择 P0.7； 5’h14：表示选择 P3.0； 5’h15 及其它：表示选择 P3.1；

锦锐科技

8.5.3 外部中断控制例程

◆ **外部中断 0 控制例程**

例如，使能外部中断 0，程序如下:

```
-----------------------------------------------------------------------------
void INT0_init(void)
{
    P00F = 1;       //外部中断 0 的中断引脚为 P0.0，设置 P0.0 为输入功能
    EX0 = 1;        //INT0 中断使能
    IE0 = 1;        //外部中断 0 使能
    IT0 = 1;        //设置为下降沿中断
    EA = 1;         //总中断使能
}
void INT0_ISR (void) interrupt 0
{
    //外部中断 0 中断服务程序
}
-----------------------------------------------------------------------------
```

◆ **外部中断 4 控制例程**

以外部中断 4 为例，设置 P1.0 为外部中断 4 中断输入引脚并开启外部中断 4，程序如下:

```
-----------------------------------------------------------------------------
void INT4_init(void)
{
    P10F = 1;                       //设置 P1.0 为输入引脚
    EP2CON = (1<<7) | (0<<5) | 0;  //设置为上升沿触发并设置中断引脚索引编号，0 对应 P1.0，如果设置为下
                                    //降沿触发则设置为 EP2CON = (1<<7) | (1<<5) | 0;
    INT4EN = 1;                     //INT4 中断使能
    EA = 1;                         //总中断使能
}
void INT4_ISR (void) interrupt 6
{
    if(EPIF & 0x04)             //判断外部中断 4 中断标志
    {
        EPIF = 0x04;            //中断标志写 1 清 0
        //外部中断 4 中断服务程序
        ......
    }
}
-----------------------------------------------------------------------------
```

9 时钟系统

9.1 时钟系统介绍

CA51M151 系列芯片共支持以下时钟源：

- 内置 8MHz RC 振荡器
- 内置 128KHz RC 振荡器
- 内置 24MHz RC 振荡器
- 支持外部 32.768KHz 晶体振荡器

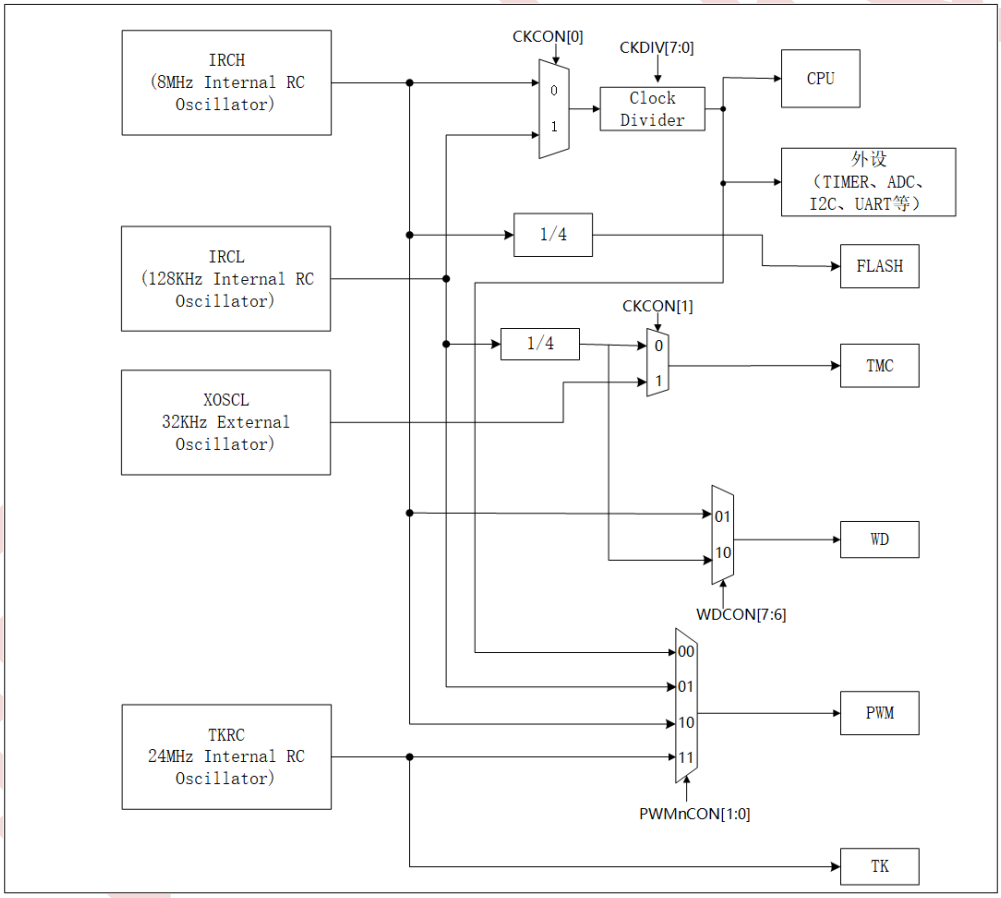


图 9-1-1 时钟源示意图

用户可独立的管理各个时钟源，每个时钟源都可以单独打开或关闭，从而可以灵活控制功耗。IRCH/IRCL 时钟源可设置为系统时钟，也可分配到各种外设中，作为外设的时钟源，详细请参考外设部分介绍。

9.1.1 时钟专用名称定义

名称缩写	描述
IRCH	内置 8MHz RC 振荡器
IRCL	内置 128KHz RC 振荡器
TKRC	内置 24MHz RC 振荡器
XOSCL	外部 32.768KHz 晶体振荡器

9.1.2 内置 8MHz RC 振荡器（IRCH）

IRCH 是芯片上电后默认的系统时钟，可通过寄存器 CKCON 的 IHCKE 位打开或关闭。芯片出厂后，IRCH 的频率校正为 8MHz@5V/25℃，时钟精度为±2%。

9.1.2 内置 24MHz RC 振荡器（TKRC）

TKRC 可通过寄存器 CKCON 的 TKCKE 位打开或关闭，为触摸模块专用时钟。

9.1.3 内置 128KHz RC 振荡器（IRCL）

IRCL 可通过寄存器 CKCON 的 ILCKE 位打开或关闭。IRCL 设为系统时钟可实现系统低功耗。

9.1.4 外部 32.768KHz 晶体谐振器（XOSCL）

XOSCL 主要是作为 TMC 的时钟源，用于实时计时，实现产品的时钟功能。XOSCL 通过寄存器 CKCON 的 XLCKE 位打开或关闭，要注意的是，XOSCL 起振时间比较长，大约需要 1 秒左右才能达到稳定，在应用时需要等待 XOSCL 时钟稳定后才可以使用。

备注：外部 32.768KHz 低速时钟(XOSCL)不支持设置为系统时钟。

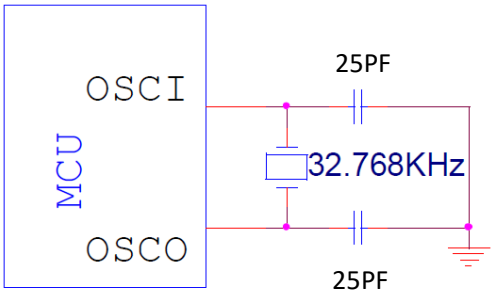


图 9-1-4-1 XOSCL 典型电路图

重要提醒： *1. 硬件设计时晶振负载电容地必须和芯片地连接，晶振补偿电容尽量靠近芯片 VSS 引脚。32.768KHz 石英晶振要求使用直径 3mmx8mm 的晶振规格。*

2. 以上电路及元件参数仅供参考，使用不同厂家晶振在电路使用中参数可能需要修改。

9.2.3 时钟控制寄存器描述

表 9-2-3-1 寄存器 IHCFG

CAH	7	6	5	4	3	2	1	0
IHCFG	IHCFG[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	IHCFG	IRCH 频率调整寄存器						

表 9-2-3-2 寄存器 TKCCFG

CBH	7	6	5	4	3	2	1	0
TKCCFG	TKCCFG [7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	TKCCFG	TKRC 频率调整寄存器						

锦锐科技

9.3 系统时钟

系统时钟控制由寄存器 CKCON、CKDIV 完成。通过这些寄存器组，可以单独设置各时钟源的开关、系统时钟的切换和分频等操作。

系统时钟有两个时钟可选：IRCH 和 IRCL，上电后，默认的系统时钟是 IRCH，并且 CKDIV 值为 1，即系统时钟上电默认为 IRCH 的二分频，如 CPU 需运行于更高频率，软件可设置 CKDIV 为 0。

9.3.1 系统时钟结构图

系统时钟结构图见图 9-3-1。

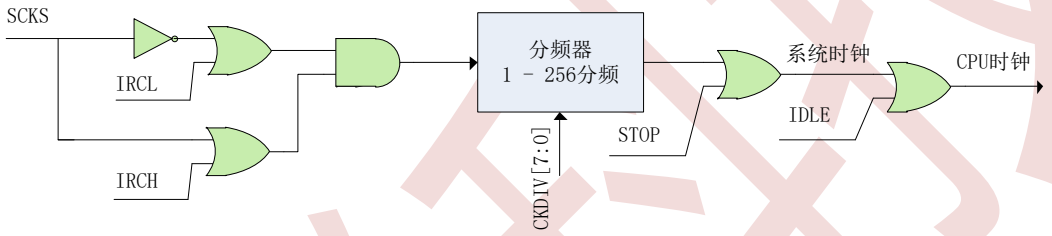


图 9-3-1 系统时钟结构图

9.3.2 系统时钟控制寄存器描述

表 9-3-2-1 寄存器 CKCON

C8H	7	6	5	4	3	2	1	0
CKCON	IHCKE	ILCKE	TKCKE	XLCKE	-	-	TMCS	SCKS
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
初始值	0	0	0	0	-	-	0	0

位编号	位符号	说明
7	IHCKE	IRCH 使能控制位 1：打开 0：关闭 *备注：* *该位为 1 时，时钟模块打开，但是该位为 0 时，如果系统或者其他模块选择了该时钟源，该时钟仍然会被打开。*
6	ILCKE	IRCL 使能控制位 1：打开 0：关闭

		备注： *该位为 1 时，时钟模块打开，但是该位为 0 时，如果系统或者其他模块选择了该时钟源，该时钟仍然会被打开。*
5	TKCKE	TK 时钟使能控制位 1：打开 0：关闭
4	XLCKE	XOSCL 时钟使能控制位 0：XOSCL 时钟关闭 1：XOSCL 时钟打开
3~2	-	-
1	TMCS	TMC 计数时钟选择 0：选择 IRCL 1：选择 XOSCL
0	SCKS	系统时钟选择位 0：选择 IRCH 1：选择 IRCL

表 9-3-2-2 寄存器 CKDIV

C9H	7	6	5	4	3	2	1	0
CKDIV	CKDIV[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	1
位编号	位符号	说明						
7~0	CKDIV	系统时钟分频： 00H：不分频 01H：2 分频 02H：3 分频 03H：4 分频 …… FFH：256 分频						

备注：上电后系统时钟默认为 2 分频，如果需要修改为 8MHz 时钟，需要将 CKDIV 设置为 8'h00；

9.3.3 系统时钟控制方法及例程

◆ **设置系统时钟为 IRCH**

设置系统时钟为 IRCH，程序如下：

```
-----------------------------------------------------------------------------
#define IHCKE            (1<<7)
#define CKSEL_IRCH          0
void Sys_Clk_Set_IRCH(void)
{
    CKCON |= IHCKE;                          //打开 IRCH 时钟
    CKCON = (CKCON&0xFE) | CKSEL_IRCH;       //设置系统时钟为 IRCH
}
-----------------------------------------------------------------------------
```

◆ **设置系统时钟为 IRCL**

设置系统时钟为 IRCL，程序如下：

```
-----------------------------------------------------------------------------
#define ILCKE            (1<<6)
#define CKSEL_IRCL          1
void Sys_Clk_Set_IRCL(void)
{
    CKCON |= ILCKE;                          //打开 IRCL 时钟
    Delay_ms(1);                             //使能 IRCL 后延时 1ms，等待 IRCL 稳定
    CKCON = (CKCON&0xFE) | CKSEL_IRCL;       //设置系统时钟为 IRCL
}
-----------------------------------------------------------------------------
```

10 供电和复位系统

10.1 供电系统

在 CA51M151 系列芯片 VDD 和 VSS 引脚间接入 2.2V - 5.5V 的电源，此电源可直接给芯片内部数字及模拟系统供电。需要注意的是，不同的供电电压条件下，芯片支持运行的最高频率和功耗并不相同，具体请查看电气特性章节。

芯片内部还设计了 BANDGAP 基准电压，作为 ADC 内部参考电压、LVD 电压等的基准电压源。

图 10-1-1 芯片供电典型电路图

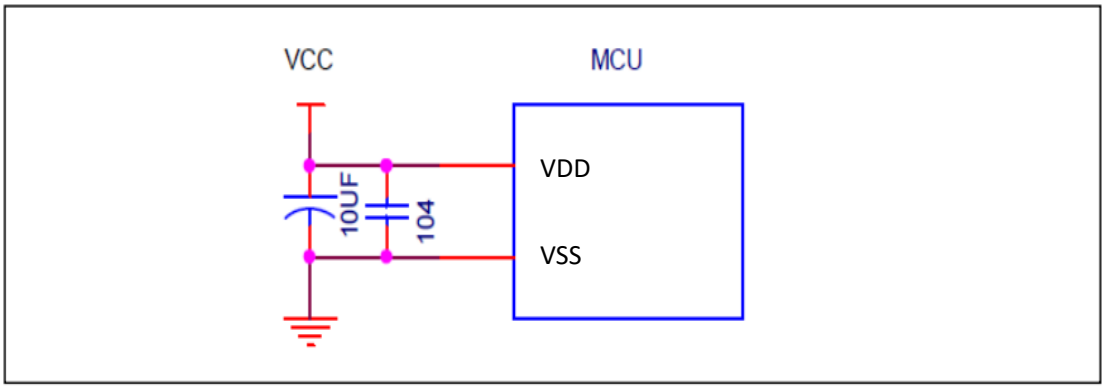


图 10-1-1 芯片供电典型电路图

重要提醒： *1. 以上电路中，滤波电容 10uF 和 104 为芯片供电电路标配，不可省略，此电容须靠近芯片电源引脚摆放，否则有可能会导致芯片工作异常。*

2. 以上电路及元件参数仅供参考，根据外围工作环境及不同电压供电参数可能需要修改。

10.1.2 内部基准电压控制寄存器

表 10-1-2-1 寄存器 PWCON

86H	7	6	5	4	3	2	1	0
PWCON	FLEVEL[3:0]				VREFS	-	-	-
R/W	R/W				R/W	-	-	-
初始值	0	1	1	1	0	-	-	-

位编号	位符号	说明
7~4	FLEVEL	内部基准电压（Bandgap）输出调整位域 0000：0.825V 0001：0.850V 0010：0.875V 0011：0.900V 0100：0.925V 0101：0.950V 0110：0.975V 0111：1.000V 1000：1.025V 1001：1.050V 1010：1.075V 1011：1.100V 1100：1.125V 1101：1.150V 1110：1.175V 1111：1.200V *备注：此基准电压出厂经校准后精度为±30mV，校准值上电自动加载，用户不允许更改。*
3	VREFS	参考电压的驱动选择：
2-0		

锐芯科技

10.2 复位系统

CA51M151 系列芯片有多个内部和外部复位源，如图 10-2-1 所示。

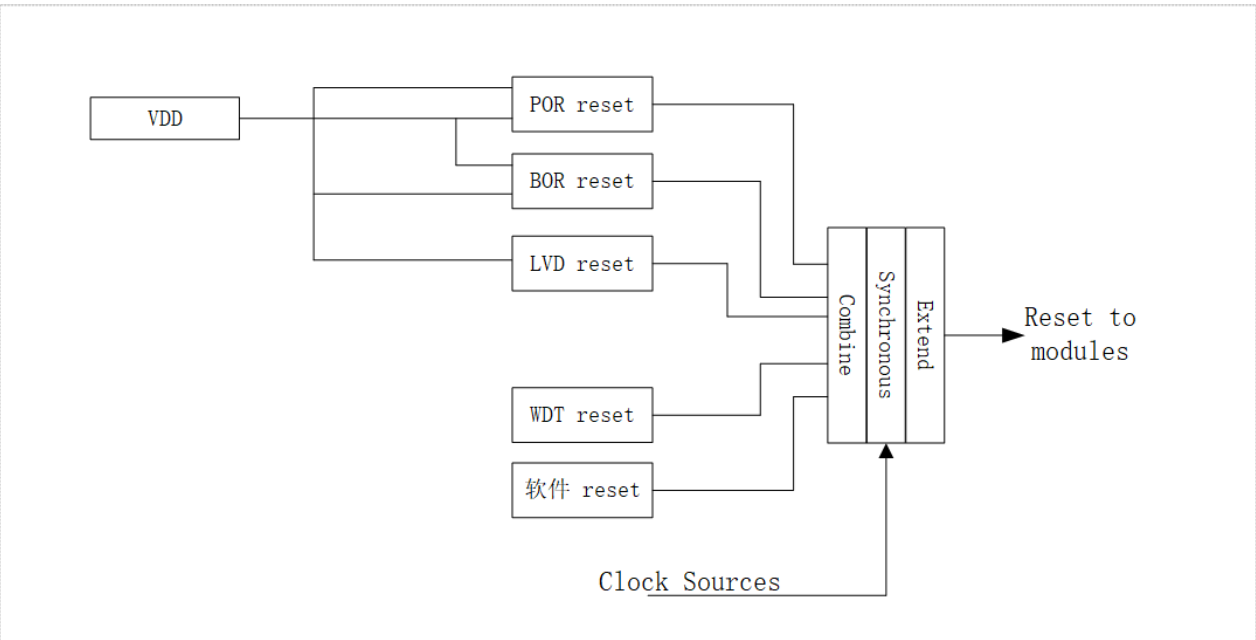
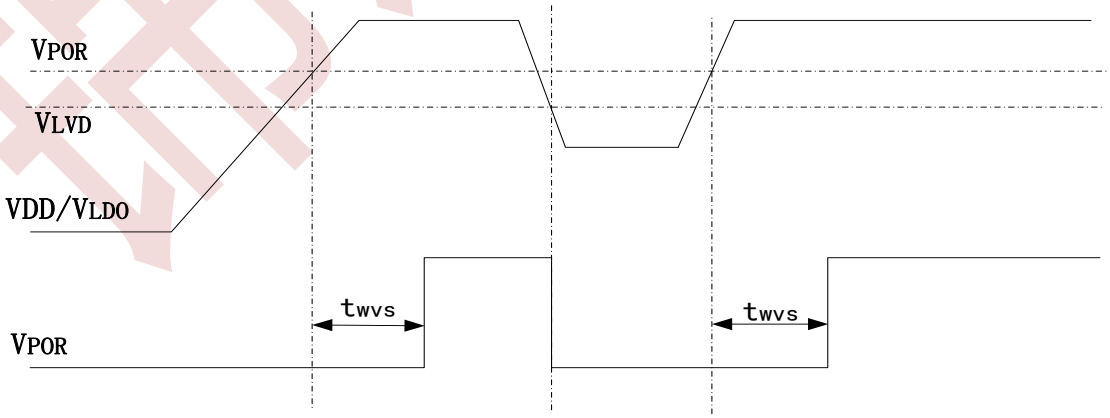


图 10-2-1 复位系统结构图

- **上电复位（POR）**

系统上电呈现逐渐上升的曲线形式，需要一定时间才能达到正常的工作电压。上电复位是基于电源电压 VDD，当电压低于检测阈值时，上电复位信号有效。

上电复位电路能够保证芯片在上电过程中处于复位状态，芯片上电后能够从一个已知的稳定的状态开始运行。上电复位信号也会被芯片内部的计数器展宽，以保证上电后内部的各种模拟模块能够进入稳定的工作状态。



twvs：等待电压稳定时间

图 10-2-2 上电复位电路示例及上电过程

- **掉电复位（BOR）**

  利用掉电复位，可以为芯片提供电源跌落(例如受到干扰或者负载变化)的预警信号。一旦发现电源电压 VDD 下降到某一个阈值时，就使芯片及时复位以免系统工作状态不正常或者程序执行错误。

- **低电压复位**

  低电压检测（LVD）可以在多种工作模式下持续监控电源电压 VDD。当 VDD 低于 LVD 设定的域值电压超过 20us 就可以产生复位信号（前提是 LVD 设置为复位模式）。

- **看门狗复位**

  看门狗定时器负责监控处理器执行指令的情况，通过合适的配置，如果看门狗定时器在特定时间段内未被刷新，则可以产生复位信号。上电复位后，看门狗定时器是关闭的，用户需要时，再配置开启。

- **软复位**

  芯片可以在程序控制下执行软复位。通过对 PCON 寄存器中的 SWRST 位写 1，CPU 可以发出复位指令。

上电掉电复位将复位所有的电路，LVD 和 WDT 的复位不能复位其本身电路，但可以复位其他电路（例如：WDT 复位产生后，WDT 模块电路没有复位，WDT 寄存器还保持复位之前的状态，但 WDT 之外的电路已经复位了）。LVD/WDT 和软复位都不能复位存储控制电路。软复位后，程序将从 BOOT 配置指向的位置开始运行。所有复位产生之后，PC 都将指向地址 0。

11 功耗管理

CA51M151 系列芯片有三种不同的低功耗模式：IDLE 模式、STOP 模式、低速运行模式。IDLE 模式时系统功耗小于 33uA，STOP 模式时系统功耗小于 7uA。

11.1 IDLE 模式

在 IDLE 模式下，CPU 将停止工作。进入 IDLE 模式前，除了主时钟，其他的时钟源根据需要都可选择关闭，以便节省功耗。同样地，进入 IDLE 模式前，可根据需要设定芯片某些外设的开关。打开的外设在 IDLE 状态下仍然可以正常工作。

设置进入 IDLE 模式前，需要先查看一下寄存器 IDLST（IDLSTH 和 IDLSTL），如果所有位都为 0，则设置进入 IDLE 模式后，CPU 将正常进入 IDLE 模式。如果 IDLST 的位不全为 0，即使有设置进入 IDLE 模式的操作，CPU 也不会进入 IDLE 模式，而是继续停留在正常工作模式。此时用户需先把 IDLST 对应位的中断处理完成，再重新设置进入 IDLE 模式的动作。

所有复位事件和任何中断事件都将唤醒芯片。中断唤醒 CPU 后，芯片首先将恢复时钟，然后响应该中断，进入该中断的服务程序。退出中断服务程序后，芯片将执行置位 IDLE 指令后面的指令。退出 IDLE 模式时，IDLE 位将自动清零。

需要注意的是，在置位 IDLE 的指令后面需要紧接两条 nop 指令，防止程序出错。

11.2 STOP 模式

STOP 模式是比 IDLE 更深层次的低功耗模式。STOP 模式可以停止所有时钟（包括主时钟）和时钟产生电路。如果 WDT 和 TMC 处于打开状态，则它们使用的时钟模块将处于工作状态，可以有选择地关闭 WDT 和 TMC 以节省功耗。

类似于 IDLE 模式，进入 STOP 模式前，需要先查看 STPST（STPSTH 和 STPSTL）寄存器，若有置 1 的位存在，需要先行处理，以确保能顺利进入 STOP 模式。

STOP 模式可以通过外部中断、LVD 中断或复位、TMC 中断、WDT 中断或复位、时钟监控中断、触摸中断来唤醒。如果是中断唤醒，那么唤醒 MCU 后，芯片首先将恢复时钟，然后响应该中断，进入该中断的服务程序。退出中断服务程序后，芯片将执行置位 STOP 指令后面的指令。退出 STOP 模式时，STOP 位将自动清零。

为了更好的唤醒芯片，推荐在进入 STOP 模式前切换系统时钟到内部时钟，因为唤醒时，外部时钟需要更多时间去等待稳定。

在进入 STOP 模式时，最后一个时钟沿将关闭系统时钟，然后芯片完全进入 STOP 模式。需要注意的是，在置位 STOP 的指令后面需要紧接三条 nop 指令，防止程序出错。

11.3 低速运行模式

由于芯片的功耗与运行速度直接相关，所以把主时钟切换到低速时钟运行也可以显著降低功耗。

11.4 低功耗相关寄存器描述

表 11-4-1 寄存器 PCON

87H	7	6	5	4	3	2	1	0
PCON	-	-	SWRST	-	-	TSMODE	STOP	IDLE
R/W	-	-	W	-	-	R	W	W
初始值	-	-	0	-	-	0	0	0

位编号	位符号	说明
7	-	-
6	-	-
5	SWRST	软复位控制位，1 有效 设置 SWRST=1 产生软复位，复位产生后自动清 0。
4~3	-	-
2	TSMODE	在线仿真模式标志位，为 1 表示芯片正工作于在线仿真模式
1	STOP	STOP 模式控制位，1 有效 当设置 STOP=1 且 STPST 为 0 时，芯片进入 STOP 模式，退出 STOP 模式后自动清 0
0	IDLE	IDLE 模式控制位，1 有效 当设置 IDLE=1 且 IDLST 为 0 时，芯片进入 IDLE 模式，退出 IDLE 模式后自动清 0

表 11-4-2 寄存器 IDLST

8EH	7	6	5	4	3	2	1	0
IDLST	-	IDLSTL[6:0]						
R/W	-	R						
初始值	-	0	0	0	0	0	0	0

位编号	位符号	说明
7	-	-
6	I2CINT/WDIF/LVDINT/UART2/EPIF[2]	IDLE 模式时，I²C/WDT/LVD/UART2/外部中断 4 的中断状态
5	TKINT/TMINT/TIMER2/EPIF[1]	IDLE 模式时，触摸按键/TMC/定时器 2/外部中断 3 的中断状态
4	UART1/ADC/PWM/SPI/EPIF[0]	IDLE 模式时，UART1/PWM/SPI/ADC/外部中断 2 的中断状态
3	TF1	IDLE 模式时，定时器 1 的中断状态
2	PIF[1]	IDLE 模式时，外部中断 1 的中断状态
1	TF0	IDLE 模式时，定时器 0 的中断状态
0	PIF[0]	IDLE 模式时，外部中断 0 的中断状态

表 11-4-3 寄存器 STPST

8FH	7	6	5	4	3	2	1	0
STPST	-	STPSTL [6:0]						

R/W	-	R						
初始值	-	0	0	0	0	0	0	0

位编号	位符号	说明
7	-	-
6	WDTWKF/LVDWKF/I2CWKF	STOP 模式时，WDT/LVD/I²C 的中断状态
5	TKWKF/TMWKF	STOP 模式时，触摸按键/TMC 的中断状态
4	EPWKF[2]	STOP 模式时，外部中断 4 的中断状态
3	EPWKF[1]	STOP 模式时，外部中断 3 的中断状态
2	EPWKF[0]	STOP 模式时，外部中断 2 的中断状态
1	PWKF[1]	STOP 模式时，外部中断 1 的中断状态
0	PWKF[0]	STOP 模式时，外部中断 0 的中断状态

11.5 低功耗模式控制例程

◆ **STOP 模式例程**

STOP 模式程序如下：

```
-----------------------------------------------------------------------------------
#define IHCKE           (1<<7)
#define ILCKE           (1<<6)

#define CKSEL_IRCH      0
#define CKSEL_IRCL      1

void Stop(void)
{
    bit IE_EA;
    unsigned char ck_bak;
    I2CCON = 0;             //关闭 I2C 功能，因为 I2C 默认是使能的，如果 I2C 不关闭将无法关闭 IRCH 时钟
    MECON |= (1<<6);        //设置 FLASH 进入深度睡眠状态
    ck_bak = CKCON & 0xFE;          //备份时钟状态
    CKCON = 0;                      //关闭所有时钟
    IE_EA = EA;                     //保存全局中断使能位状态
    EA = 0;
    PCON = (PCON&0x04)|0x02;        //进入 STOP 模式
    _nop_();                        //在 STOP 的指令后面需要紧接三条 nop 指令，防止程序出错。
    _nop_();
    _nop_();
    EA = IE_EA;                     //恢复原全局中断开关状态
    Sys_Clk_Set_IRCH();
    CKCON |= ck_bak;                //恢复所关闭的时钟
}
```

测试条件：所有时钟关闭，所有输出引脚无负载，所有数字输入引脚不浮动，所有外设关闭， Flash 进入深度睡眠模式，CPU 进入 STOP 模式。

◆ **IDLE 模式例程**

IDLE 模式程序如下：

```
#define IHCKE           (1<<7)
#define ILCKE           (1<<6)

#define CKSEL_IRCH      0
#define CKSEL_IRCL      1

void Idle(void)
{
    unsigned char ck_bak;
    I2CCON = 0;         //关闭 I2C 模块，因为 I2C 默认是使能的，如果 I2C 不关闭将无法关闭 IRCH 时钟
    ck_bak = CKCON & 0xFE;              //备份时钟状态
    CKCON = (CKCON&0x41) | ILCKE;       //IRCL 时钟使能,并关闭其它时钟
    Delay_ms(1);                        //使能 IRCL 后延时 1ms，等待 IRCL 稳定
    CKCON = (CKCON&0xFE) | CKSEL_IRCL; //系统时钟切换到 IRCL
    MECON |= (1<<6);                    //设置 FLASH 进入深度睡眠状态
    while(IDLST&0x7F);                  //如果有中断未响应,等待中断被响应
    PCON = (PCON&0x04)|0x01;            //进入 IDLE 模式
    _nop_();
    _nop_();
    Sys_Clk_Set_IRCH();
    CKCON |= ck_bak;                    //恢复所关闭的时钟
}
```

备注：由于进入 IDLE 后，主时钟仍是打开的，如果进入 IDLE 前主时钟是高速时钟，进入 IDLE 模式后功耗仍会很大，所以进入 IDLE 之前需要把主时钟切换到低速时钟。

---

12 通用定时器（定时器 0,定时器 1,定时器 2）

12.1 定时器 0

12.1.1 定时器 0 介绍

定时器或计数器功能通过 CT0 位（TMOD[2]）来选择，CT0=0 选择为定时器，CT0=1 选择为计数器。作为定时器时，时钟是系统时钟的 12 分频。作为计数器时，时钟是 T0 的输入时钟。由于检测 T0 输入边沿变化需要 2 个时钟周期，所以作为计数器时最大的输入波特率是内部系统时钟频率的 1/2。T0 输入信号在占空比上没有限制，然而为了完全识别 0 或 1 的状态，信号至少需要保持 1 个内部系统时钟周期时间。定时器 0 有 4 个工作模式，通过 T0M0、T0M1 位(TMOD[1:0])来选择。

- **模式 0**

在此模式下，定时器 0 作为 13 位定时器/计数器，TH0 存放 13 位定时器/计数器的高 8 位，TL0[4:0]存放低 5 位，而 TL0[7:5]是无效的，在读取时应被忽略。当定时器 0 溢出，中断标志位 TF0（TCON[5]）会被置 1。中断被响应后，TF0 位会自动清 0。当 GATE0（TCON[3]）=0 时，定时器/计数器由 TR0（TCON[4]）位使能计数，当 GATE0=1 时，定时器/计数器由引脚 INT0 控制使能，INT0 为高电平时计数，INT0 为低电平则停止计数。

- **模式 1**

此模式下，定时器 0 作为 16 位定时器/计数器，除此之外，功能与模式 0 完全相同。

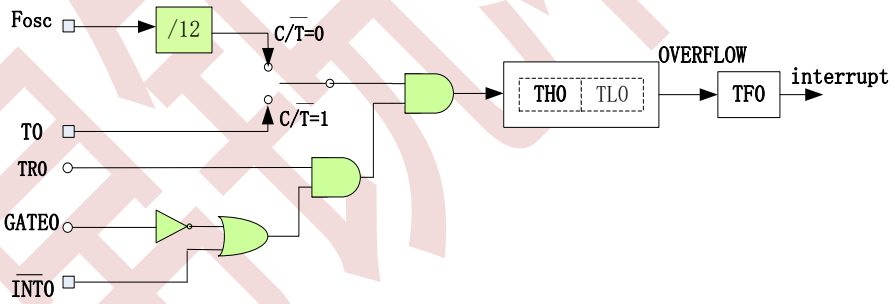


图 12-1-1-1 定时器 0 的模式 0 与 1

- **模式 2**

在此模式中，定时器 0 作为 8 位自动重载定时器/计数器，只有 TL0 自动累加。当 TL0 计数溢出时，不但产生中断标志 TF0，而且从 TH0 中自动装载计数初始值到 TL0。其他设置方法和模式 0、1 相同。

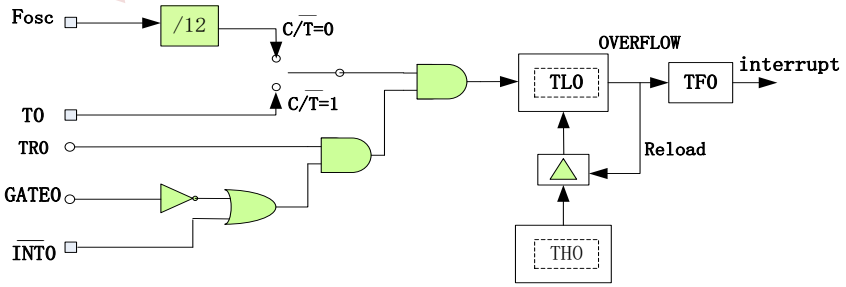


图 12-1-1-2 定时器 0 的模式 2

- **模式 3**

在此模式中，TL0 和 TH0 作为两个独立的 8 位定时器/计数器。TL0 可以作为定时器或计数器，而 TH0 只能作为定时器。其中 TL0 占用定时器 0 的控制位 CT0、GATE0、TR0、TF0、INT0，而 TH0 只能占用定时器 1 的控制位 TR1、TF1。其他控制方法和模式 0、1 相同。当定时器 0 工作于模式 3 时，定时器 1 和 TH0 共用控制位 TR1，但定时器 1 由于 TF1 已被 TH0 占用，所以只能工作于不需要产生中断的场合，例如作为 UART 的波特率产生器。

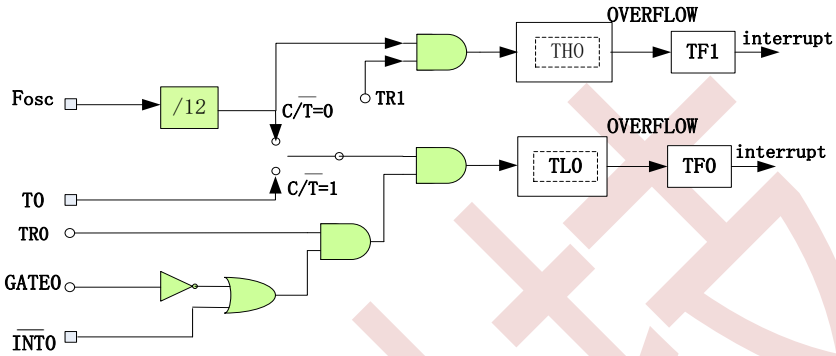


图 12-1-1-3 定时器 0 的模式 3

12.1.2 定时器 0 寄存器描述

表 12-1-2-1 寄存器 TCON

88H	7	6	5	4	3	2	1	0
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	TF1	定时器 0 模式 3 的 TH0 溢出/定时器 1 溢出标志位，中断被响应后自动清 0.
6	TR1	定时器 1 运行控制位，1 有效
5	TF0	定时器 0 溢出标志位，中断被响应后自动清 0.
4	TR0	定时器 0 运行控制位，1 有效
3	IE1	外部中断 1 使能位，1 有效
2	IT1	外部中断 1 触发类型控制位 0：外部中断 1 在输入管脚上升沿时触发 1：外部中断 1 在输入管脚下降沿时触发
1	IE0	外部中断 0 使能位，1 有效
0	IT0	外部中断 0 触发类型控制位 0：外部中断 0 在输入管脚上升沿时触发 1：外部中断 0 在输入管脚下降沿时触发

表 12-1-2-2 寄存器 TMOD

89H	7	6	5	4	3	2	1	0
TMOD	GATE1	CT1	T1M1	T1M0	GATE0	CT0	T0M1	T0M0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	GATE1	定时器 1 门控控制位，1 有效。有效时定时器 1 由 INT1 控制开关
6	CT1	定时器 1 计数器/定时器选择位 0：定时器，时钟为系统时钟 12 分频 1：计数器，时钟为 T1 输入时钟
5	T1M1	[T1M1,T1M0]为定时器 1 模式选择位
4	T1M0	00：模式 0，TL1 和 TH1 组成 13 位定时器/计数器 01：模式 1，TL1 和 TH1 组成 16 位定时器/计数器 10：模式 2，TL1 作为 8 位定时器/计数器，TH1 作为自动重载寄存器 11：模式 3，此模式会锁住 TH1/TL1，等效于 TR1=0
3	GATE0	定时器 0 门控控制位，1 有效。有效时定时器 0 由 INT0 控制开关
2	CT0	定时器 0 计数器/定时器选择位 0：定时器，时钟为系统时钟 12 分频 1：计数器，时钟为 T0 输入时钟
1	T0M1	[T0M1,T0M0]为定时器 0 模式选择位
0	T0M0	00：模式 0，TL0 和 TH0 组成 13 位定时器/计数器 01：模式 1，TL0 和 TH0 组成 16 位定时器/计数器 10：模式 2，TL0 作为 8 位定时器/计数器，TH0 作为自动重载寄存器 11：模式 3，TL0 和 TH0 作为两个完全独立的 8 位定时器/计数器

表 12-1-2-3 寄存器 TL0

8AH	7	6	5	4	3	2	1	0
TL0	TL0							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	TL0	定时器 0 模式 0/1 计数值的低字节，模式 2/3 计数值

表 12-1-2-4 寄存器 TH0

8CH	7	6	5	4	3	2	1	0
TH0	TH0							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	TH0	定时器 0 模式 0/1 计数值的高字节，模式 2 重载值，模式 3 计数值

12.2 定时器 1

12.2.1 定时器 1 介绍

定时器或计数器功能通过 CT1 位（TMOD[6]）来选择，CT1=0 选择为定时器，CT1=1 选择为计数器。作为定时器时，时钟是系统时钟的 12 分频。作为计数器时，时钟是 T1 的输入时钟。由于检测 T1 输入边沿变化需要 2 个时钟周期，所以作为计数器时最大的输入波特率是内部系统时钟频率的 1/2。T1 输入信号在占空比上没有限制，然而为了完全识别 0 或 1 的状态，信号至少需要保持 1 个内部系统时钟周期时间。定时器 1 有 4 个工作模式，通过 T1M0、T1M1 位(TMOD[5:4])来选择。

- **模式 0**

  在此模式下，定时器 1 作为 13 位定时器/计数器，TH1 存放 13 位定时器/计数器的高 8 位，TL1[4:0]存放低 5 位，而 TL1[7:5]是无效的，在读取时应被忽略。当定时器 1 溢出，中断标志位 TF1（TCON[7]）会被置 1。中断被响应后，TF1 位会自动清 0。当 GATE1（TCON[7]）=0 时，定时器/计数器由 TR1（TCON[6]）位使能计数，当 GATE1=1 时，定时器/计数器由引脚 INT1 控制使能，INT1 为高电平时计数，INT1 为低电平则停止计数。

- **模式 1**

  在此模式下，定时器 1 作为 16 位定时器/计数器，TH1 存放 16 位定时器/计数器的高 8 位，TL1 存放低 8 位。当定时器 1 溢出，中断标志位 TF1（TCON[7]）会被置 1。中断被响应后，TF1 位会自动清 0。当 GATE1（TCON[7]）=0 时，定时器/计数器由 TR1（TCON[6]）位使能计数，当 GATE1=1 时，定时器/计数器由引脚 INT1 控制使能，INT1 为高电平时计数，INT1 为低电平则停止计数。

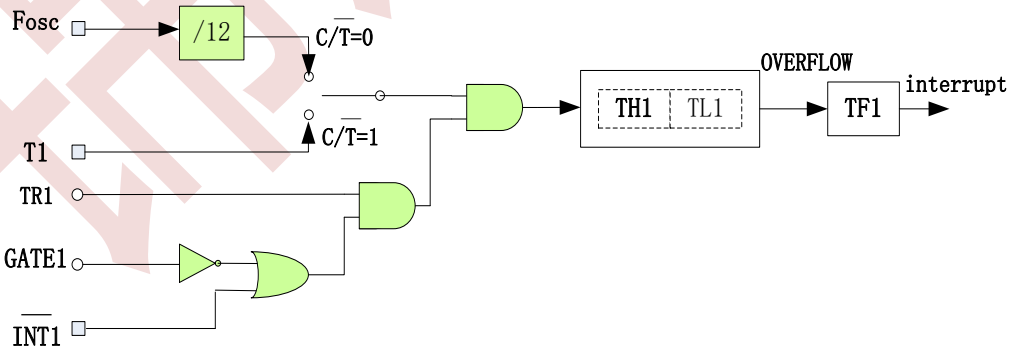


图 12-2-1 定时器 1 的模式 0 和 1

- **模式 2**

  在此模式中，定时器 1 作为 8 位自动重载定时器/计数器，只有 TL1 自动累加。当 TL1 计数溢出时，不但产生中断标志 TF1，而且从 TH1 中自动装载计数初始值到 TL1。其他设置方法和模式 0、1 相同。

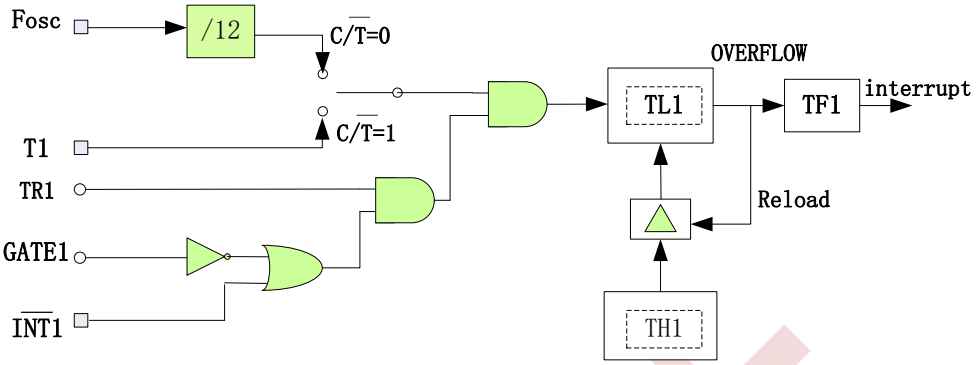


图 12-2-2 定时器 1 的模式 2

- **模式 3**

  此模式下，TH1、TL1 会被锁住，等效于 TR1=0。

12.2.2 定时器 1 寄存器描述

寄存器 TCON 和 TMOD 见表 12-1-2-1 和表 12-1-2-2。

表 12-2-2-1 寄存器 TL1

8BH	7	6	5	4	3	2	1	0
TL1	TL1							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	TL1	定时器 1 模式 0/1 计数值的低字节，模式 2/3 计数值						

表 12-2-2-2 寄存器 TH1

8DH	7	6	5	4	3	2	1	0
TH1	TH1							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	TH1	定时器 1 模式 0/1 计数值的高字节，模式 2 重载值，模式 3 计数值						

12.3 定时器 2

12.3.1 定时器 2 介绍

定时器 2 是一个 16 位的定时器，定时器 2 时钟为系统时钟（注意：和定时器 0、1 不同的是，时钟没有经过 12 分频）。通过 T2M 设置不同的工作模式。

当 T2M=0 时，定时器 2 工作于定时模式，T2CRH、T2CRL 存放计数上限值。当定时器 2 使能时，16 位计数器自动累加，当定时器计数值达到设定的上限值后产生定时中断标志 TF2，16 位计数器清零，中断标志 TF2 通过写 1 清 0.

当 T2M=1 时，定时器 2 工作于捕获模式。当引脚 T2CP 触发沿发生时，定时器 2 的计数值被锁存到 T2CRH、T2CRL，触发沿可通过 CPEDGESEL 位设置，当捕获事件产生后，捕获中断标志 CF2 置 1，如果定时器 2 中断使能会触发捕获中断，CF2 通过写 1 清 0。在捕获模式下，计数器溢出会产生溢出中断标志位 OF2，通过对 OF2 写 1 清除 OF2 中断标志。

备注：T2CP 引脚可映射至不同的 GPIO 引脚，详细见"15-2-7 引脚复用功能映射表"描述。

12.3.2 定时器 2 寄存器描述

表 12-3-2-1 寄存器 T2CON

A4H	7	6	5	4	3	2	1	0
T2CON	TR2	IE2	TF2	OF2	CF2	CPEDGESEL		T2M
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	TR2	定时器 2 工作使能，1 有效
6	IE2	定时器 2 中断使能，1 有效
5	TF2	定时中断标志位，写 1 清 0
4	OF2	溢出中断标志位，写 1 清 0
3	CF2	捕获中断标志位，写 1 清 0
2~1	CPEDGESEL	捕获模式触发沿选择： 0，3：双沿触发 1：下降沿 2：上升沿
0	T2M	定时/捕获功能选择： 0：定时模式 1：捕获模式

表 12-3-2-2 寄存器 T2CRL

A6H	7	6	5	4	3	2	1	0
T2CRL	TL2							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	T2CRL	T2M=0: 定时器计数上限值低字节 T2M=1: 捕获结果低字节						

表 12-3-2-3 寄存器 T2CRH

A5H	7	6	5	4	3	2	1	0
T2CRH	TH2							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	T2CRH	T2M=0: 定时器计数上限值高字节 T2M=1: 捕获结果高字节						

13 看门狗定时器（WDT）

13.1 看门狗定时器（WDT）功能简介

看门狗定时器是一个可选时钟源的 27 位减法计数器，时钟为 8MHz 下计数时间范围为 0.256ms ~16.778s，有 16 位调节精度。看门狗主要用于监控系统，避免 CPU 因为外界干扰出现死机。如果软件不能在溢出前刷新看门狗定时器，看门狗将产生内部复位或者中断。写 A5H 到寄存器 WDFLG 将刷新看门狗，读 WDFLG 可得到看门狗状态。在 STOP 模式下，如果看门狗处于使能状态，则看门狗所选的时钟源正常工作，此时如果看门狗设为中断，看门狗中断可唤醒 CPU。

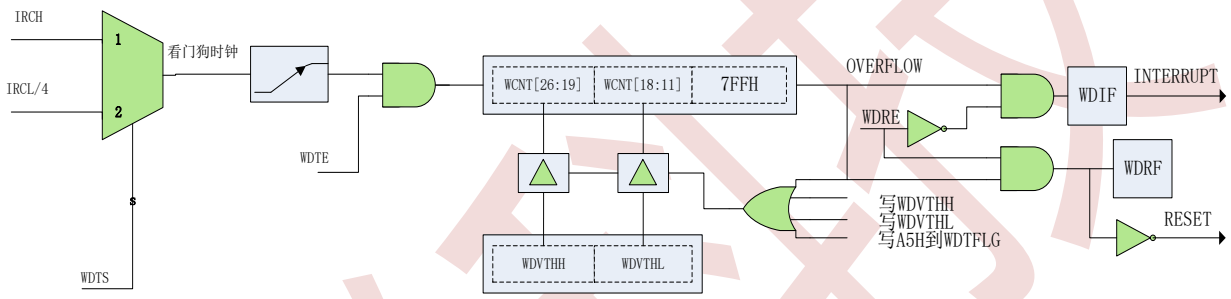


图 13-1-1 看门狗模块结构图

13.2 看门狗定时器（WDT）寄存器描述

表 13-2-1 寄存器 WDCON

A7H	7	6	5	4	3	2	1	0
WDCON	WDTS[1:0]		-	-	-	-	-	WDRE
R/W	R/W		-	-	-	-	-	R/W
初始值	0	0	-	-	-	-	-	0

位编号	位符号	说明
7~6	WDTS	WDT 时钟选择位 01：选择 IRCH 10：选择 IRCL 四分频 其他：WDT 关闭
5~1	-	-
0	WDRE	WDT 功能选择位 0：WDT 溢出后产生中断 1：WDT 溢出后产生复位

表 13-2-2 寄存器 WDFLG

A1H	7	6	5	4	3	2	1	0
WDFLG							WDIF	WDRF
R/W	-						R/W	R/W
初始值	-	-	-	-	-	-	0	0

位编号	位符号	说明
7~2	-	-
1	WDIF	WDT 中断标志，写 A5H 时将清除该标志
0	WDRF	WDT 复位标志，写 A5H 时将清除该标志

表 13-2-3 寄存器 WDVTHL、WDVTHH

A2H	7	6	5	4	3	2	1	0
WDVTHL	WDVTH[7:0]							
R/W	R/W							
初始值	1	1	1	1	1	1	1	1
A3H	7	6	5	4	3	2	1	0
WDVTHH	WDVTH[15:8]							
R/W	R/W							
初始值	1	1	1	1	1	1	1	1

位编号	位符号	说明
15~0	WDVTH	WDT 阈值设置寄存器，计算公式如下： WDT 触发时间＝ (WDVTH * 800H+7FFH) * clock cycle

13.3 看门狗定时器控制例程

◆ **看门狗中断模式例程**

例如，看门狗时钟设置为 IRCH，IRCH 的频率为 8MHz，看门狗设置为中断模式，溢出时间为 1 秒，程序如下：

```
-----------------------------------------------------------------------------
void WDT_init(void)
{
    WDCON = (1<<6) | 0;             //设置看门时钟为 IRCH，看门狗中断模式
    WDVTHH = 0x0F;                  //设置看门狗时间为 1 秒
    WDVTHL = 0x41;
    WDFLG = 0xA5;                   //刷新看门狗
    INT4EN = 1;                     //开启看门狗中断
    EA = 1;                         //开启总中断
}
void WDT_ISR (void) interrupt 6
{
    if(WDFLG & 0x02)
    {
        WDFLG = 0xA5;               //刷新看门狗
    }
}
-----------------------------------------------------------------------------
```

◆ **看门狗复位模式例程**

例如，看门狗时钟设置为 IRCH，IRCH 的频率为 8MHz，看门狗设置为复位模式，溢出时间为 1 秒，程序如下：

```
-----------------------------------------------------------------------------
void WDT_init(void)
{
    WDCON = (1<<6) | 1;             //设置看门时钟为 IRCH，看门狗复位模式
    WDVTHH = 0x0F;                  //设置看门狗时间为 1 秒
    WDVTHL = 0x41;
    WDFLG = 0xA5;                   //刷新看门狗
}
-----------------------------------------------------------------------------
```

14 TMC 定时器

14.1 TMC 功能简介

TMC 定时器的时钟源可选 IRCL 或 XOSCL。时钟源为 IRCL 时，中断时间最小单位为 512 个时钟周期。时钟源为 XOSCL 时，中断的最小单位为 128 个时钟周期，可配置中断时间为 1~256 个最小单位时间。在 STOP/IDLE 模式下，TMC 中断可唤醒 CPU。

14.2 TMC 寄存器描述

表 14-2-1 寄存器 TMCON

D5H	7	6	5	4	3	2	1	0
TMCON	TME	-	-	-	-	-	-	TMF
R/W	R/W	-	-	-	-	-	-	R
初始值	0	-	-	-	-	-	-	0

位编号	位符号	说明
7	TME	TME 模块使能，1 有效
6~1	-	-
0	TMF	TMC 中断标志，1 有效，写 1 清 0

表 14-2-2 寄存器 TMSNU

D6H	7	6	5	4	3	2	1	0
TMSNU	TMSNU[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	-

位编号	位符号	说明
7~0	TMSNU	TMC 中断时间配置寄存器 当 TMC 定时器的时钟源为 IRCL 时，TMC 的中断时间为（TMSNU+1）×512×Tircl； 当 TMC 定时器的时钟源为 XOSCL 时，TMC 的中断时间为（TMSNU+1）×128×Txoscl

14.3 TMC 控制例程

程序如下：

```
-----------------------------------------------------------------------------
#define TME(N)              (N<<7)    //N=0-1 TME 模块使能，1 有效
#define TMF                 (1<<0)    //TMC 中断标志，1 有效，写 1 清 0

#define XLCKE               (1<<4)    //XLOSC 时钟使能控制位
#define ILCKE               (1<<6)    //IRCL 使能控制位
#define TMCS(N)             (N<<1)    //N=0-1,TMC 计数时钟选择
#define TMC_CLK_SELECT      ILCKE
void INT3_ISR (void) interrupt 5
{
    if(TMCON & TMF)                   //判断 TMC 中断标志
    {
        TMCON |= TMF;                 //清除 TMC 中断标志

    }
}

void TMC_Init(void)
{
#if (TMC_CLK_SELECT == ILCKE)
    CKCON |= ILCKE;                   //打开 IRCL 时钟
    CKCON |= TMCS(0);                 //TMC 计数时钟选 IRCL 时钟
#elif (TMC_CLK_SELECT == XLCKE)
    P22F = 10;                        //设置 P2.2 为晶振引脚
    P23F = 10;                        //设置 P2.3 为晶振引脚
    CKCON |= XLCKE;                   //使能 XOSCL 时钟
    CKCON |= TMCS(1);                 //TMC 计数时钟 XOSCL 时钟
#endif
    TMCON = TME(1);                   //TMC 使能
    TMSNU = 0;                        //设置中断时间
    INT3EN =1;                        //开启 TMC 中断
    EA = 1;                           //开启总中断
}
-----------------------------------------------------------------------------
```

15 通用输入输出口（GPIO）及复用定义

15.1 功能简介

CA51M151 系列芯片最大封装有 26 个 I/O 引脚，每个引脚都是复用功能引脚，不仅能独立编程为输入/输出口，而且还能设置为其他功能引脚。每个引脚都分配了一个功能设置寄存器 PnxF（分别对应引脚 Pnx，其中 n=0、1、2、3，代表 P0、P1、P2、P3，x=0~7，代表 Pn.0~Pn.7），用户可通过寄存器 PnxF 配置引脚的主功能和其他选项。详见寄存器部分介绍。

GPIO 的主要特性如下：

- 可配置为高阻模式
- I/O 结构可独立设置上拉下拉电阻
- 输出模式可选开漏输出或推挽输出
- 数据输出锁存支持读-修改-写
- 推挽输出时，GPIO 的推电流有 4 级可选：12mA、10mA、8mA、6mA；@VDD=5V
- 灌电流有 2 级可选：16mA、8mA；@VDD=5V
- GPIO 可软件模拟成 1/2 BIAS 的 LCD 驱动，不需外接电阻

GPIO 推挽模式结构图如图 15-1-1 所示

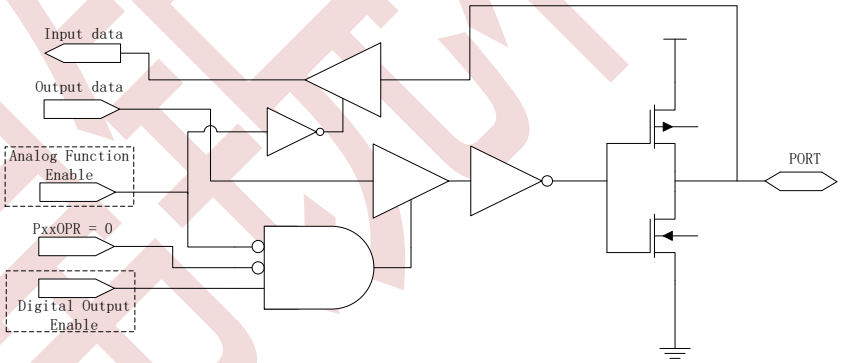


图 15-1-1 I/O 推挽模式结构示意图

GPIO 开漏模式结构图如图 15-1-2 所示

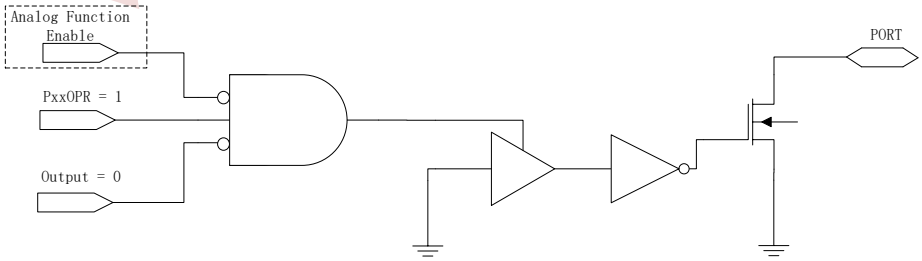


图 15-1-2 I/O 开漏模式结构示意图

GPIO 下拉结构图如图 15-1-3 所示

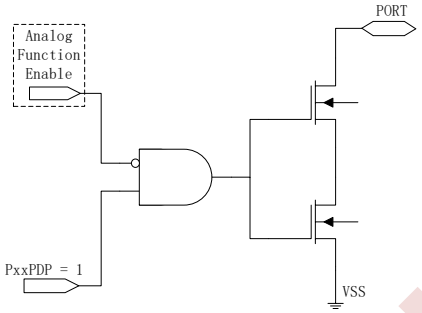


图 15-1-3 I/O 下拉模式结构示意图

GPIO 上拉结构图如图 15-1-4 所示

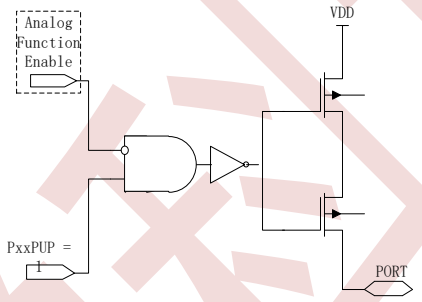


图 15-1-4 I/O 上拉模式结构示意图

15.2 引脚寄存器描述

表 15-2-1 寄存器 P0

80H	7	6	5	4	3	2	1	0
P0	P07	P06	P05	P04	P03	P02	P01	P00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	P0x	引脚 P0x 的数据寄存器，管脚功能设置为 GPIO 时有效 0：设为输入时 P0x 电平为低，设为输出时 P0x 输出低电平 1：设为输入时 P0x 电平为高，设为输出时 P0x 输出高电平

表 15-2-2 寄存器 P1

90H	7	6	5	4	3	2	1	0
P1	P17	P16	P15	P14	P13	P12	P11	P10
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	P1x	引脚 P1x 的数据寄存器，管脚功能设置为 GPIO 时有效 0：设为输入时 P1x 电平为低，设为输出时 P1x 输出低电平 1：设为输入时 P1x 电平为高，设为输出时 P1x 输出高电平

表 15-2-3 寄存器 P2

A0H	7	6	5	4	3	2	1	0
P2	P27	P26	P25	P24	P23	P22	P21	P20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	P1x	引脚 P2x 的数据寄存器，管脚功能设置为 GPIO 时有效 0：设为输入时 P2x 电平为低，设为输出时 P2x 输出低电平 1：设为输入时 P2x 电平为高，设为输出时 P2x 输出高电平

表 15-2-4 寄存器 P3

B0H	7	6	5	4	3	2	1	0
P3	-	-	-	-	-	-	P31	P30
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

位编号	位符号	说明
7~2	-	-
1~0	P3x	引脚 P3x 的数据寄存器，管脚功能设置为 GPIO 时有效 0：设为输入时 P3x 电平为低，设为输出时 P3x 输出低电平 1：设为输入时 P3x 电平为高，设为输出时 P3x 输出高电平

表 15-2-5 引脚功能控制寄存器

8000H	7	6	5	4	3	2	1	0
P00F	P00PUP	P00PDP	P00OPR	-	P00S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8001H	7	6	5	4	3	2	1	0
P01F	P01PUP	P01PDP	P01OPR	-	P01S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8002H	7	6	5	4	3	2	1	0
P02F	P02PUP	P02PDP	P02OPR	-	P02S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8003H	7	6	5	4	3	2	1	0
P03F	P03PUP	P03PDP	P03OPR	-	P03S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8004H	7	6	5	4	3	2	1	0
P04F	P04PUP	P04PDP	P04OPR	-	P04S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8005H	7	6	5	4	3	2	1	0
P05F	P05PUP	P05PDP	P05OPR	-	P05S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8006H	7	6	5	4	3	2	1	0
P06F	P06PUP	P06PDP	P06OPR	-	P06S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8007H	7	6	5	4	3	2	1	0
P07F	P07PUP	P07PDP	P07OPR	-	P07S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8008H	7	6	5	4	3	2	1	0
P10F	P10PUP	P10PDP	P10OPR	-	P10S			

R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8009H	7	6	5	4	3	2	1	0
P11F	P11PUP	P11PDP	P11OPR	-	P11S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
800AH	7	6	5	4	3	2	1	0
P12F	P12PUP	P12PDP	P12OPR	-	P12S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
800BH	7	6	5	4	3	2	1	0
P13F	P13PUP	P13PDP	P13OPR	-	P13S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
800CH	7	6	5	4	3	2	1	0
P14F	P14PUP	P14PDP	P14OPR	-	P14S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
备注：CA51M151S6A/CA51M151S6A/CA51M151S4A/CA51M151S3 初始值 P14S=6，其它型号芯片初始值 P14S=0								
800DH	7	6	5	4	3	2	1	0
P15F	P15PUP	P15PDP	P15OPR	-	P15S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
800EH	7	6	5	4	3	2	1	0
P16F	P16PUP	P16PDP	P16OPR	-	P16S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
备注：CA51M151S6A/CA51M151S6A/CA51M151S4A/CA51M151S3 初始值 P16S=7，其它型号芯片初始值 P16S=0								
800FH	7	6	5	4	3	2	1	0
P17F	P17PUP	P17PDP	P17OPR	-	P17S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8010H	7	6	5	4	3	2	1	0
P20F	P20PUP	P20PDP	P20OPR	-	P20S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0

8011H	7	6	5	4	3	2	1	0
P21F	P21PUP	P21PDP	P21OPR	-	P21S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8012H	7	6	5	4	3	2	1	0
P22F	P22PUP	P22PDP	P22OPR	-	P22S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8013H	7	6	5	4	3	2	1	0
P23F	P23PUP	P23PDP	P23OPR	-	P23S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8014H	7	6	5	4	3	2	1	0
P24F	P24PUP	P24PDP	P24OPR	-	P24S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8015H	7	6	5	4	3	2	1	0
P25F	P24PUP	P25PDP	P25OPR	-	P25S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8016H	7	6	5	4	3	2	1	0
P26F	P26PUP	P26PDP	P26OPR	-	P26S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-		0	0	0
8017H	7	6	5	4	3	2	1	0
P27F	P27PUP	P27PDP	P27OPR	-	P27S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
8018H	7	6	5	4	3	2	1	0
P30F	P30PUP	P30PDP	P30OPR	-	P30S			
R/W	R/W	R/W	R/W	-	R/W			
初始值	0	0	0	-	0	0	0	0
备注：CA51M151S6B 初始值 P30S=7，CA51M151T3A 初始值 P30S=6，其它型号芯片初始值 P30S=0								
8019H	7	6	5	4	3	2	1	0
P31F	P31PUP	P31PDP	P31OPR	-	P31S			
R/W	R/W	R/W	R/W	-	R/W			

初始值	0	0	0	-	0	0	0	0
备注：CA51M151S6B 初始值 P31S=6，CA51M151T3A 初始值 P31S=7，其它型号芯片初始值 P31S=0								
位编号	位符号	说明						
7	PnxPUP	上拉电阻使能控制位 0: 上拉电阻关闭 1: 上拉电阻打开						
6	PnxPDP	下拉电阻使能控制位 0: 下拉电阻关闭 1: 下拉电阻打开						
5	PnxOPR	开漏使能控制位，引脚设为数字输出时才有效 0: 开漏关闭 1: 开漏打开						
4	-	-						
3~0	PnxS	引脚复用功能设置详见表 15-2-7						

备注：Pnx → n=0、1、2、3，代表 P0、P1、P2、P3
x=0~7，代表 Pn.0~Pn.7

表 15-2-6 寄存器 PnxC

8120H	7	6	5	4	3	2	1	0
P00C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8121H	7	6	5	4	3	2	1	0
P01C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8122H	7	6	5	4	3	2	1	0
P02C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8123H	7	6	5	4	3	2	1	0
P03C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8124H	7	6	5	4	3	2	1	0
P04C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W5		R/W
初始值	-	1	-	-	-	0	0	0
8125H	7	6	5	4	3	2	1	0
P05C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8126H	7	6	5	4	3	2	1	0
P06C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8127H	7	6	5	4	3	2	1	0
P07C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8128H	7	6	5	4	3	2	1	0
P10C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8129H	7	6	5	4	3	2	1	0
P11C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
812AH	7	6	5	4	3	2	1	0
P12C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
812BH	7	6	5	4	3	2	1	0
P13C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
812CH	7	6	5	4	3	2	1	0
P14C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
812DH	7	6	5	4	3	2	1	0
P15C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
812EH	7	6	5	4	3	2	1	0
P16C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

812FH	7	6	5	4	3	2	1	0
P17C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8130H	7	6	5	4	3	2	1	0
P20C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8131H	7	6	5	4	3	2	1	0
P21C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8132H	7	6	5	4	3	2	1	0
P22C	-	SMIT_EN	-	-	-	DRV		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8133H	7	6	5	4	3	2	1	0
P23C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8134H	7	6	5	4	3	2	1	0
P24C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8135H	7	6	5	4	3	2	1	0
P25C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8136H	7	6	5	4	3	2	1	0
P26C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

8137H	7	6	5	4	3	2	1	0
P27C	-	SMIT_EN	-	-	-	DRV[1:0]		SR

R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8138H	7	6	5	4	3	2	1	0
P30C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0
8139H	7	6	5	4	3	2	1	0
P31C	-	SMIT_EN	-	-	-	DRV[1:0]		SR
R/W	-	R/W	-	-	-	R/W		R/W
初始值	-	1	-	-	-	0	0	0

位编号	位符号	说明
7	-	-
6	SMIT_EN	为 1 输入的 SMIT 使能，为 0 输入是反相器使能
5~3	-	-
2~1	DRV	输出强度选择 *备注：* *1、 推电流 4 级驱动电流可选，详看电气特性章节* *2、 灌电流 2 级驱动电流可选，DRV=0/1 为同一驱动电流，DRV=2/3 为同一驱动电流，详看电气特性章节*
0	SR	输出斜率控制 0：最慢斜率控制 1：最快斜率控制

表 15-2-7 引脚复用功能映射表

取值 名称	0	1	2	3	4	5	6	7	8	9	10
P00S	高阻	数字输入	数字输出	PWM0_B	ADC0	TK0	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P01S	高阻	数字输入	数字输出	PWM1_B	ADC1	TK1	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P02S	高阻	数字输入/T0	数字输出	PWM2_A	ADC2	TK2	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P03S	高阻	数字输入/T1	数字输出	PWM3_A	ADC3	TK3	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P04S	高阻	数字输入	数字输出	PWM0_A	ADC4	TK4	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P05S	高阻	数字输入	数字输出	PWM1_A	ADC5	TK5	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P06S	高阻	数字输入	数字输出	PWM2_C	ADC6	TK6	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P07S	高阻	数字输入	数字输出	PWM3_C	ADC7	TK7	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P10S	高阻	数字输入	数字输出	PWM0_E	ADC8	TK8	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P11S	高阻	数字输入	数字输出	PWM1_E	ADC9	TK9	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P12S	高阻	数字输入	数字输出	PWM2_D	ADC10	TK10	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P13S	高阻	数字输入	数字输出	PWM3_D	ADC11	TK11	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	T2CP
P14S	高阻	数字输入	数字输出	PWM0_F	ADC12	TK12	I2C_SDA	I2C_SCL	UART2_RX	UART2_TX	SPI_SDI
P15S	高阻	数字输入	数字输出	PWM1_F	ADC13	TK13	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	SPI_SDO
P16S	高阻	数字输入	数字输出	PWM2_E	ADC14	TK14	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	SPI_SCK
P17S	高阻	数字输入	数字输出	PWM3_E	ADC15	TK15	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	SPI_SCS
P20S	高阻	数字输入	数字输出	PWM0_G	ADC16	TK16	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	SPI_SDI
P21S	高阻	数字输入	数字输出	PWM1_G	ADC17	TK17	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	SPI_SCK
P22S	高阻	数字输入	数字输出	PWM2_F	ADC18	TK18	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	32K_OUT
P23S	高阻	数字输入	数字输出	PWM3_F	ADC19	TK19	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	32K_IN
P24S	高阻	数字输入	数字输出	PWM0_D	ADC20	TK20	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	T2CP
P25S	高阻	数字输入	数字输出	PWM1_D	ADC21	TK21	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	T2CP
P26S	高阻	数字输入	数字输出	PWM2_B	ADC22	TK22	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	T2CP
P27S	高阻	数字输入	数字输出	PWM3_B	ADC23	TK23	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	T2CP
P30S	高阻	数字输入	数字输出	PWM0_C	ADC24	TK24	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	T2CP
P31S	高阻	数字输入	数字输出	PWM1_C	ADC25	TK25	I2C_SDA	I2C_SCL	UART1_RX	UART1_TX	T2CP

15.3 引脚控制例程

◆ 引脚功能设置

例如，P00 设置为推挽输出，程序如下：

```
P00F = 2;
```

P00 设置为开漏输出，程序如下：

```
P00F = (1<<5)|2;
```

P00 设置为开漏输出，并且打开上拉，程序如下：

```
P00F = (1<<7) | (1<<5) | 2;
```

P00 设置为输入功能，并且打开上拉，程序如下：

```
P00F = (1<<7) | 1;
```

例如，P00 设置为推挽输出，推电流设置为最大，程序如下：

```
P00F = 2;
P00C= (P00C&0x40) | (3<<1);
```

锐芯科技

16 通用串行接口（UART1/UART2）

16.1 功能简介

UART1 和 UART2 是设计完全相同的两个全双工异步串行数据收发器，UARTx（x=1、2,代指 UART1、UART2）也有一字节的接收缓存。UARTx 有两种不同的工作模式，如表 16-1-1 所示。

UART1/UART2 的 TX/RX 功能可映射至不同的 GPIO 引脚，详细见“15-2-7 引脚复用功能映射表”描述。

SMx	模式	描述	波特率
0	A	9 位异步模式	CPUCLK/(32*(1024-SxREL))
1	B	8 位异步模式	CPUCLK/(32*(1024-SxREL))

表 16-1-1 UARTx 工作模式

UARTx 设计了专门的波特率发生器，波特率通过寄存器 SxRELL、SxRELH 来配置。

图 16-1-1 是 UARTx 的原理示意图。

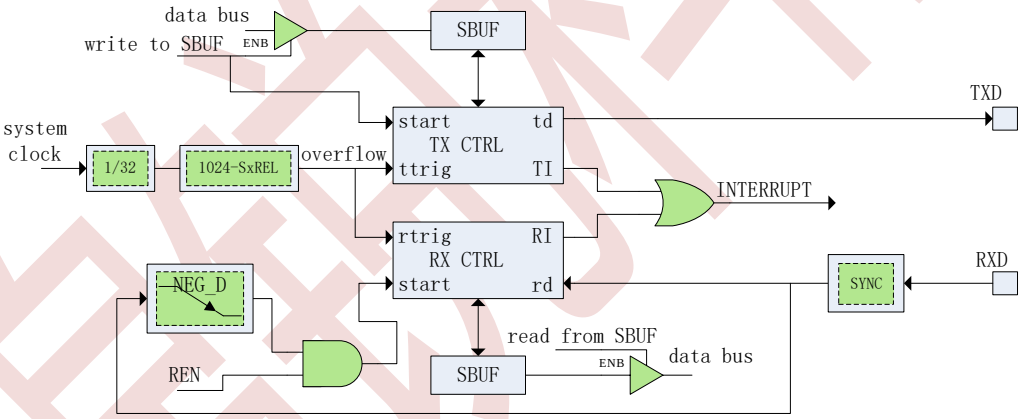


图 16-1-1 UARTx 工作原理示意图

- **模式 A**

在模式 A，UARTx 可异步同时收发 9 位数据。写入数据到寄存器 SxBUF 会启动 UARTx 数据发送。第一个传送的位是开始位（为 0），然后是 9 位数据（低位先发），第 9 位数据是寄存器 SxCON 的 TB81 位，最后传送的是停止位（为 1）。在接收状态，UARTx 通过检测引脚 RX 的下降沿来同步。传送过程完成后，低 8 位数据存放在寄存器 SxBUF，第 9 位数据存放在 RB8x 位。

- **模式 B**

模式 B 和模式 A 不同的是，模式 B 是 8 位数据传输，停止位存放的是有效停止位。其他功能和模式 A 一致。

- **UARTx 多机通信**

在 UARTx 模式 A 中有一个专门适用于多机通信的机制。当寄存器 SxCON 的 SM2x 位置 1，只有接收到第

9 位数据为 1（RB8x=1）的从机才会产生接收中断，利用这个功能可进行多机通信，从机将它们的 SM2x 位都置为 1，主机传送从机的地址时将第 9 位数据设为 1，这样所有的从机都会产生接收中断；从机的软件用它们自己的地址和接收的地址进行比较，如果一致，被寻址的从机设置 SM2x=0，然后主机继续传送后面的数据时设置第 9 位为 0，因为其他的从机 SM2x 仍然设为 1，这样就只有被寻址的从机才会产生接收中断。

- **快速波特率设置**

在标准 51 单片机的 UART 中，UART 的波特率固定为定时器溢出率的 32 分频，由于 CA51M151 系列 MCU 的 CPU 时钟为 8MHz（或 8MHz 的分频），配置的波特率和标准波特率相比有较大误差，所以在 CA51M151 系列 MCU 中，设计了校正波特率的机制，UART 的波特率不固定为定时器溢出率的 32 分频，而是可以由寄存器 UDCKS 设置。例如：当 UART 的波特率固定为定时器溢出率的 32 分频时，选择定时器 2 作为 UART 的波特率发生器，若要配置波特率为 115200，计算公式为：8000000÷32÷115200=2.17，由于定时器计数只能取整数，所以取 2（即每 2 个系统时钟周期定时器溢出一次），误差率约为 8.5%，误差率太大会导致通信不正常。由于系统时钟是固定的，要达到更准确的波特率，只能通过修改分频系数来实现。如果设置定时器 3 个时钟周期溢出，那么：8000000÷115200÷3= 23.15。取分频数为 23，那么波特率为 115942，和 115200 相比，误差率约为 0.64%，一般情况下不会影响 UART 通信。另外，更小的分频数也可以实现更高的波特率配置。
芯片默认的分频系数为 32，与标准 51 相同。如果要更改分频系数，通过设置 UDEx=1 来使能，DNUM 的数值表示不同的分频系数，详见寄存器 UDCKSx 描述。

16.2 UARTx 寄存器描述

表 16-2-1 寄存器 S1CON

98H	7	6	5	4	3	2	1	0
S1CON	SM1	U1IE	SM21	REN1	TB81	RB81	TI1	RI1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	SM1	UART1 模式选择位，详见表 16-1-1
6	U1IE	UART1 中断使能位，1 有效
5	SM21	多机通信使能位，1 有效
4	REN1	串行接收使能位，1 有效
3	TB81	发送数据的第 9 位 在模式 A，这个位用于 UART1 传送数据，对应传送数据的第 9 位 （例如奇偶校验或多主机通信），由软件控制
2	RB81	接收数据的第 9 位 在模式 A，这个位用于 UART1 接收数据，对应接收数据的第 9 位； 在模式 B，这个位是接收到的停止位
1	TI1	传送中断标志位，1 有效，写 1 清 0
0	RI1	接收中断标志位，1 有效，写 1 清 0

表 16-2-2 寄存器 S1BUF

99H	7	6	5	4	3	2	1	0
S1BUF	S1BUF[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	S1BUF	UART1 收发缓冲器 写 S1BUF 将开始发送所写的数据 读 S1BUF 将得到已经接收的数据

表 16-2-3 寄存器 UDCKS1

D8H	7	6	5	4	3	2	1	0
UDCKS1	UDE	-	-	DNUM[4:0]				
R/W	R/W	-	-	R/W				
初始值	0	-	-	0	0	0	0	0

位编号	位符号	说明
7	UDE	快速波特率配置使能控制位，1 有效 ***备注：*** *UDE=0 时 UART1 波特率按照原来的配置，UDE=1 时 UART1 波特率由 DNUM 来配置。*
6~5	-	-
4~0	DNUM	快速波特率配置寄存器，仅在 UDE=1 时有效 发送时，须满足 DNUM>=0；接收时，DNUM>=6 BR = F_{sys}*(1/((DNUM+1)*(1024-S1REL)))

表 16-2-4 寄存器 S1RELL、S1RELH

8068H	7	6	5	4	3	2	1	0
S1RELL	S1RELL[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
8069H	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
S1RELH	-	-	-	-	-	-	S1REL[9:8]	
R/W	-	-	-	-	-	-	R/W	
初始值	-	-	-	-	-	-	0	0

位编号	位符号	说明
9~0	S1REL	波特率配置寄存器 波特率为 CPUCLK/(32 * (1024 – S1REL))

表 16-2-5 寄存器 S2CON

B9H	7	6	5	4	3	2	1	0
S2CON	SM2	U2IE	SM22	REN2	TB82	RB82	TI2	RI2
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	SM2	UART2 模式选择位，详见表 16-1-1
6	U2IE	串口 2 中断使能位，1 有效
5	SM22	多机通信使能位，1 有效
4	REN2	串行接收使能位，1 有效
3	TB82	发送数据的第 9 位 在模式 A，这个位用于串口 1 传送数据，对应传送数据的第 9 位 （例如奇偶校验或多主机通信），由软件控制
2	RB82	接收数据的第 9 位 在模式 A，这个位用于 UART2 接收数据，对应接收数据的第 9 位 在模式 B，这个位是接收到的停止位
1	TI2	传送中断标志位，1 有效，写 1 清 0
0	RI2	接收中断标志，1 有效，写 1 清 0

表 16-2-6 寄存器 S2BUF

BAH	7	6	5	4	3	2	1	0
S2BUF	S2BUF[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	S2BUF	发送接收缓冲器 写 S2BUF 将启动发送所写的数据 读 S2BUF 将读取已经接收的数据

表 16-2-7 寄存器 UDCKS2

BDH	7	6	5	4	3	2	1	0
UDCKS2	UDE	-	-	DNUM[4:0]				
R/W	R/W	-	-	R/W				
初始值	0	-	-	0	0	0	0	0

位编号	位符号	说明
7	UDE	快速波特率配置使能控制位，1 有效 ***备注：***

		UDE=0 时 UART2 波特率按照原来的配置，UDE=1 时 UART2 波特率由 DNUM 来配置。
6~5	-	-
4~0	DNUM	快速波特率配置寄存器，仅在 UDE=1 时有效 发送时，须满足 DNUM>=0；接收时，DNUM>=6 $BR = F_{sys}*(1/((DNUM+1)*(1024-S2REL)))$

表 16-2-8 寄存器 S2RELL、S2RELH

BBH	7	6	5	4	3	2	1	0
S2RELL	S2RELL[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
BCH	7	6	5	4	3	2	1	0
S2RELH	-	-	-	-	-	-	S2REL[9:8]	
R/W	-	-	-	-	-	-	R/W	
初始值	-	-	-	-	-	-	0	0

位编号	位符号	说明
9~0	S2REL	波特率配置寄存器 波特率为 CPUCLK/(32 * (1024 – S2REL))

17 I²C 接口

17.1 功能简介

I²C 模块支持芯片与外围 I²C 器件以标准 I²C 协议进行串行数据传输，可设置为主机或从机模式，通过合理配置可使 I²C 支持标准/快速/高速模式。

I²C 引脚 SCL/SDA 功能可以映射到任意 GPIO 引脚，详细见“15-2-7 引脚复用功能映射表”描述。

17.2 I²C 主要特点

- 简单且强大而灵活的通讯接口，双向两线总线
- 可设置为主机或从机模式
- 可以工作于发送器模式或接收器模式
- 7 位从机地址
- 支持多主机仲裁
- 支持广播功能
- 支持 SCL/SDA 映射到任意 GPIO

17.3 I²C 功能描述

I²C 模块支持 I²C 标准总线协议。I²C 总线用 2 根线在设备间传输数据，分别为 SCL（串行时钟线）和 SDA（串行数据线），如图 17-3-1 所示。由于 I²C 端口是开漏结构，所以 I²C 总线上必须有上拉电阻，上拉电阻可以外接也可以在芯片内部打开。每个连接在总线上的设备都有一个唯一的 7 位地址。

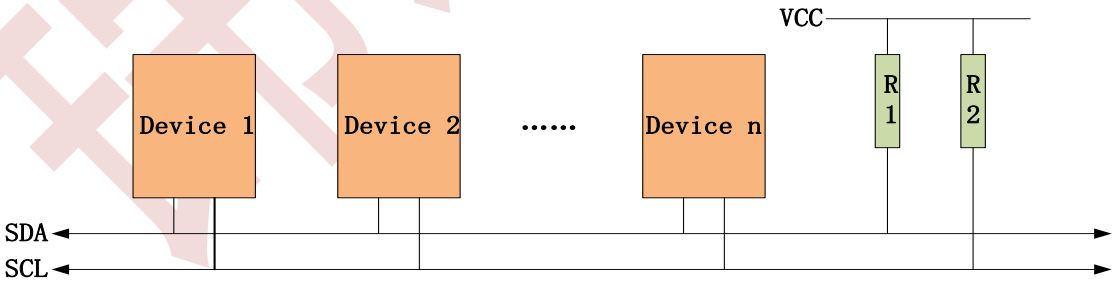


图 17-3-1 I2C 总线互连图

I²C 模块原理示意图如图图 17-3-2 所示。

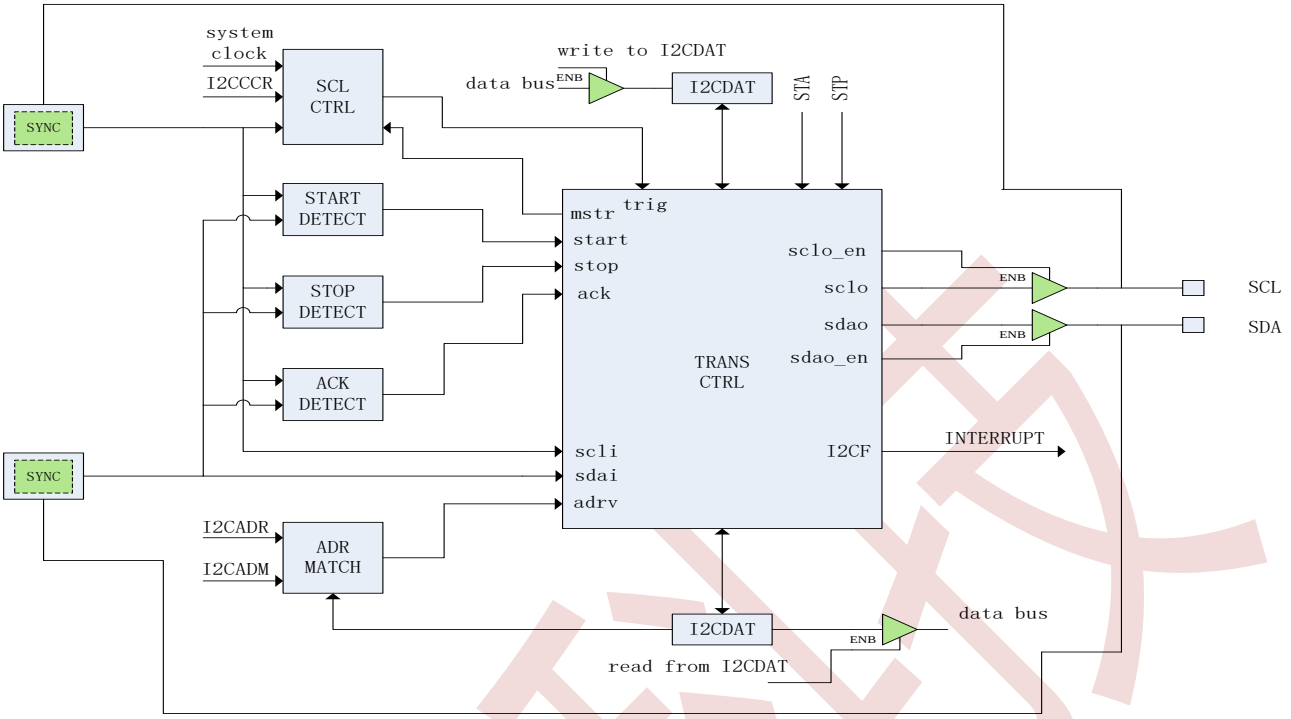


图 17-3-2　I²C 模块原理示意图

- **I²C 模式选择**

I²C 可以在以下 4 种模式中的一种运行：从机发送模式、从机接收模式、主机发送模式、主机接收模式。默认情况下，I²C 处于从机模式。I²C 在产生开始信号后自动从从机模式切换到主机模式，当仲裁失败或产生 STOP 信号后又自动切回从机模式。

- **I²C 总线数据传输格式**

一般情况下，标准的 I²C 通信由四部分组成：开始信号、从机地址传输、数据传输和结束信号。I²C 总线上传送的数据均为 8 位，高位先发，每发送一个字节后都必须跟随一个应答位，每次通信的数据字节数没有限制；在全部数据传送结束后，由主机发送停止信号，结束通信。

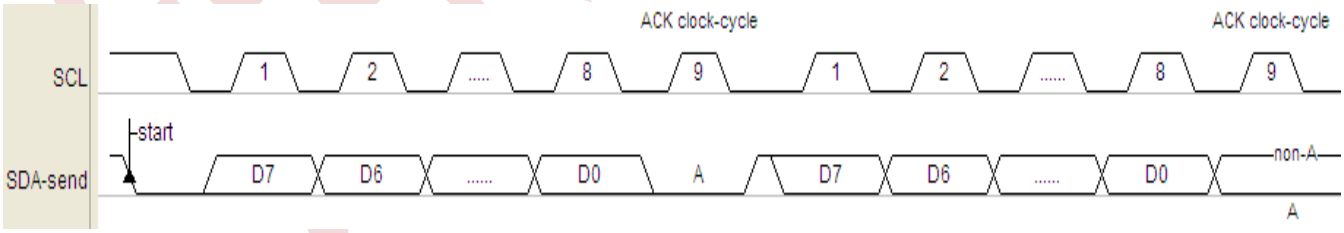


图 17-3-3　I2C 总线数据传输格式

- **通信过程**

在主机模式下，I²C 接口启动数据传输并产生时钟信号。串行数据传输总是以 START 信号开始，以 STOP 信号结束。START 信号和 STOP 信号都是在主机模式下通过软件控制产生的，START 信号通过设置 STA=1 产生，而 STOP 信号通过设置 STP=1 产生。

在从机模式下，I²C 接口能识别自身地址（7 位地址）和广播地址。软件能通过 GCE 位使能或禁止广播地址的识别。

地址和数据以字节为单位进行传输，地址会跟在 START 信号之后由主机发送。在一个字节传输的 8 个时钟后的第 9 个时钟周期内，接收器必须回送一个应答位给发送器。应答位通过 AAK 位设置，设置应答位必须在一个字节传输完之前设置，接收器完成一个字节接收时，应答信号自动产生。数据传输过程中，数据发送/接收完一字节、仲裁失败等事件都会产生中断标志 I2CF，而事件的状态则由寄存器 I2CSTA 指示（详细请参考寄存器 I2CSTA 介绍），软件应在产生中断标志后根据事件的状态设置数据传输的下一步操作，清除中断标志 I2CF 将启动下一步操作。通信结束后主机产生 STOP 信号也会在从机端产生中断标志 I2CSTP，指示通信过程的完成。当中断标志 I2CF 产生时，如果 SHD=1，在没有清除 I2CF 之前，SCL 会被从机拉低，主机检测到 SCL 被释放后才会进行下一步操作；如果 SHD=0，从机不会拉低 SCL，这样设计是为了兼容主机是软件模拟 I2C 的应用，此时，主机的软件必须等待足够长的时间让从机响应每字节数据传输的处理。

当 I²C 接口作为从机时，SCL 的时钟由主机输入，和从机的时钟配置无关。作为从机时，需要保证 SCL 为低电平的宽度最少为 6.5 个系统时钟，而高电平最少为 2.5 个系统时钟。所以，外部主机发送的 SCL 频率最高为系统时钟频率的 1/9。

17.4 I²C 通信引脚的映射

为了方便硬件设计，I²C 通信引脚可以有不同的映射，详细见“15-2-7 引脚复用功能映射表”描述。

17.5 寄存器描述

表 17-5-1 寄存器 I2CCON

C0H	7	6	5	4	3	2	1	0
I2CCON	I2CE	I2CIE	STA	STP	SHD	AAK	CBSE	STFE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	0	0	0	0	1	0	0

位编号	位符号	说明
7	I2CE	I²C 模块使能位，1 有效
6	I2CIE	I²C 中断使能位，1 有效
5	STA	I²C 发送 START 信号控制位，1 有效，检测到 START 信号后将自动清 0
4	STP	I²C 发送 STOP 信号控制位，1 有效，检测到 STOP 信号后将自动清 0
3	SHD	为 1 时，如果 I2CF 为 1，那么当 SCL 变低之后，I2CF 将会使 SCL 保持在低的状态
2	AAK	I²C 发送 ACK 信号控制位，1 有效 *备注：* *当 I²C 接口配置为从机模式时，这一位须预先置 1，否则即使地址匹配也不会回复 ACK，从而无法被寻址。*
1	CBSE	CBUS 兼容使能位 当这一位设置为 1 时，将会使传输忽略 ACK 位的状态判断，以兼容 CBUS 总线。
0	STFE	为 1 时，I²C 模块检测到 START 信号时将置位 I2CF

表 17-5-2 寄存器 I2CADR

C1H	7	6	5	4	3	2	1	0
I2CADR	GCE	I2CADRL[6:0]						
R/W	R/W	R/W						
初始值	1	0	0	0	0	0	0	0

位编号	位符号	说明
7	GCE	识别广播地址（00H）使能位，1 有效
6~0	I2CADRL	I²C 从机地址，作为从机时有效 *备注：* *（在 AAK 为 1 的前提下）7 位地址模式时，接收的第一个地址字节高 7 位和 I2CADR 匹配，则回复 ACK，进入从机模式。*

表 17-5-3 寄存器 I2CADM

C2H	7	6	5	4	3	2	1	0
I2CADM	SPFE	I2CADML[6:0]						
R/W	R/W	R/W						
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	SPFE	为 1 时，I²C 模块检测到 STOP 信号时将置位 I2CF
6~0	I2CADML	I²C 从地址按位屏蔽寄存器，为从机时有效 当 I2CADM[n](n=0~6)=1 时，对应的地址位 I2CADR[n]将不比对（即认为无论收到 1 还是 0 都算匹配）。

表 17-5-4 寄存器 I2CCCR

C3H	7	6	5	4	3	2	1	0
I2CCCR	I2CCCR[7:0]							
R/W	R/W							
初始值	0	0	1	0	0	0	0	0

位编号	位符号	说明
7~0	I2CCCR	I²C 时钟配置寄存器 采样频率为 I²C 工作时钟的 $2^{I2CCCR[7:5]}$分频，当 I2CCCR[7:5]等于 000：$F_{sample}=F_{i2cclk}$ 001：$F_{sample}=F_{i2cclk}/2$ 010：$F_{sample}=F_{i2cclk}/4$ … 111：$F_{sample}=F_{i2cclk}/128$

		输出频率为采样频率的(I2CCCR[4:0]+1)分频， $F_{scl}=F_{i2cclk}/(2^{I2CCCR[7:5]}*(I2CCCR[4:0]+1))$ 例如 I2CCCR[4:0]=9 时，当 I2CCR[7:5]等于 000：$F_{scl}=F_{i2cclk}/(1*10)$ 001：$F_{scl}=F_{i2cclk}/(2*10)$ 010：$F_{scl}=F_{i2cclk}/(4*10)$ … 111：$F_{scl}=F_{i2cclk}/(128*10)$ ***备注：*** *1. 当 I2CCCR[7:5] = 0 时，如果对 I2CCCR[4:0]写小于 9 的值，将自动按 9 的值计算。* *2. 当 I2CCCR[7:5] > 0 时，如果对 I2CCCR[4:0]写小于 7 的值，将自动按 7 的值计算。*

表 17-5-5 寄存器 I2CDAT

C4H	7	6	5	4	3	2	1	0
I2CDAT	I2CDAT[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	I2CDAT	发送和接收数据缓存 *备注：* *当 I2CF 为 1 时，建议改写/读取 I2CDAT 时，让 I2CF 保持在 1，等处理完成之后再清除 I2CF，以继续传输，这样可以避免总线发生不必要的错误。*

表 17-5-6 寄存器 I2CSTA

C5H	7	6	5	4	3	2	1	0
I2CSTA	I2CSTA[7:0]							
R/W	R							
初始值	1	1	1	1	1	0	0	0

位编号	位符号	说明
7~0	I2CSTA	I²C 状态寄存器 00H: （主/从）总线错误 08H: （主/从）检测到 START 信号（只在 STFE=1 时才有效） 18H: （主）已发送地址+写位，已接收到应答信号 20H: （主）已发送地址+写位，无接收到应答信号 28H: （主）已发送/接收一字节数据，已检测到应答信号 30H: （主）已发送/接收一字节数据，无检测到应答信号 38H: （主）失去仲裁（主机失去仲裁后会变为从机） 40H: （主）已发送地址+读位，已接收到应答信号

		48H: （主）已发送地址+读位，无接收到应答信号 60H: （从）已接收地址+写位，已发送出应答信号 70H: （主/从）已接收广播地址，已发送出应答信号（主机或从机都会变为从机） 80H: （从）已发送/接收一字节数据，已检测到应答信号 88H: （从）已发送/接收一字节数据，无检测到应答信号 A0H: （主/从）检测到 STOP 信号（只在 SPFE=1 时才有效） A8H: （从）已接收地址+读位，已发送出应答信号 F8H: （主/从）总线空闲

表 17-5-7 寄存器 I2CFLG

C6H	7	6	5	4	3	2	1	0
I2CFLG	-	-	-	-	-	-	-	I2CF
R/W	-	-	-	-	-	-	-	R
初始值	-	-	-	-	-	-	-	0

位编号	位符号	说明
7~1	-	-
0	I2CF	I²C 中断标志，1 有效，写 1 清 0 *备注：* *1. 每字节地址或数据传输完成后（收到/发送完 ACK/NAK），将置位 I2CF。* *2. 总线出错时，将置位 I2CF。* *3. 当 STFE=0 时，检测到 START 信号，I2CF 不会置 1。* *4. 当 SPFE=0 时，检测到 STOP 信号，I2CF 不会置 1。*

18 PWM

18.1 PWM 功能简介

CA51M151 系列芯片最多支持 4 路 PWM 输出，每路可单独控制，在 16 位范围内可任意配置周期和占空比。每路 PWM 输出引脚可映射至不同引脚，详细见“15-2-7 引脚复用功能映射表”描述。

18.2 PWM 功能描述

每路 PWM 通道都有一个专门的 16 位计数器，PWM 的周期通过寄存器 PWMDIV 来设置，而寄存器 PWMDUT 则对应 PWM 的占空比。PWM 通过寄存器 PWMEN 使能，寄存器 PWMEN 的每一位对应 PWM 的一个通道。PWM 可通过 PWMTOG 位设置 PWM 引脚输出反相。PWM 有多种时钟源可以选择，每路时钟源都是单独进行设置的，对应的控制寄存器为 PWMCON 的 PWMCKS 位。另外，每路 PWM 的时钟分频可通过 PWMCKD 独立设置。

值得注意的是，当 PWMDIV=0 时，PWM 引脚直接输出 PWM 时钟，如果 PWMCKD=0，PWM 引脚输出的是所选的时钟源的时钟信号；当 PWMDIV 不为 0，而 PWMDUT=0 时，PWM 引脚输出低电平（PWMTOG=0）；当 PWMDUT>=PWMDIV>0 时，PWM 引脚输出高电平（PWMTOG=0）。

- **PWM中断**

PWM 中断通过寄存器 PWMCON 的 PWMIE 位使能，PWMTOG=0 时，该通道为上升沿产生中断；PWMTOG=1 时，该通道为下降沿产生中断。寄存器 PWMIF 的低 4 为分别对应 4 个通道的中断状态。

18.3 PWM 寄存器描述

表 18-3-1 寄存器 PWMEN

91H	7	6	5	4	3	2	1	0
PWMEN	-	-	-	-	PWMEN[3:0]			
R/W	-	-	-	-	R/W			
初始值	-	-	-	-	0	0	0	0

位编号	位符号	说明
7~4	-	-
3~0	PWMEN	3~0 位分别对应 PWM 通道 3~0 的使能控制位，1 有效

表 18-3-2 寄存器 PWMCON

94H	7	6	5	4	3	2	1	0
PWMCON	PWMIE	PWMTOG	-	-	-	-	PWMCKS[1:0]	
R/W	R/W	R/W	-	-	-	-	R/W	
初始值	0	0	-	-	-	-	0	0

位编号	位符号	说明
备注： *1. PWMCON 是带索引的寄存器，设置 INDEX=0~3 分别对应 PWMCON0~3*		
7	PWMTIE	PWM 中断使能控制位，1 有效(PWMTOG=0，上升沿中断; PWMTOG=1,下降沿中断)
6	PWMTOG	PWM 输出取反使能寄存器，1 有效
5~2	-	-
1~0	PWMCKS	PWM 输入时钟选择位 00：系统时钟 01：IRCL 10：IRCH 11：INDEX=0 时，选择 TKRC；INDEX 为其他值时，选择系统时钟。

表 18-3-3 寄存器 PWMCKD

95H	7	6	5	4	3	2	1	0
PWMCKD	PWMCKD[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

备注：PWMCKD 是带索引的寄存器，设置 INDEX=0~3 分别对应 PWMCKD0~PWMCKD3

位编号	位符号	说明
7~0	PWMCKD	PWM 输入时钟预分频配置寄存器 00000000：不分频 00000001：2 分频 00000010：3 分频 …… 11111110：255 分频 11111111：256 分频

表 18-3-4 寄存器 PWMDIVL、PWMDIVH

96H	7	6	5	4	3	2	1	0
PWMDIVL	PWMDIV[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
97H	7	6	5	4	3	2	1	0
PWMDIVH	PWMDIV[15:8]							

R/W	R/W							
初始值	0	0	0	0	0	0	0	0
备注：PWMDIV 是带索引的寄存器，设置 INDEX=0~3 分别对应 PWMDIV0~PWMDIV3								
位编号	位符号	说明						
15~0	PWMDIV	PWM 周期配置寄存器						

表 18-3-5 寄存器 PWMDUTL、PWMDUTH

9BH	7	6	5	4	3	2	1	0
PWMDUTL	PWMDUT[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
9AH	7	6	5	4	3	2	1	0
PWMDUTH	PWMDUT[15:8]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
备注：PWMDUT 是带索引的寄存器，设置 INDEX=0~3 分别对应 PWMDUT0~PWMDUT3								
位编号	位符号	说明						
15~0	PWMDUT	PWM 占空比配置寄存器						

表 18-3-6 寄存器 PWMIF

92H	7	6	5	4	3	2	1	0
PWMIF	-	-	-	-	PWM3IF	PWM2IF	PWM1IF	PWM0IF
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	0	0	0	0
位编号	位符号	说明						
3	PWM3IF	PWM3 中断标志位，写 1 清 0						
2	PWM2IF	PWM2 中断标志位，写 1 清 0						
1	PWM1IF	PWM1 中断标志位，写 1 清 0						
0	PWM0IF	PWM0 中断标志位，写 1 清 0						

19 SPI 接口

19.1 功能简介

SPI 接口能够实现芯片与其他设备以半/全双工同步传输数据。外围设备可以是其它的 MCU、ADC、传感器或闪速存储器等。SPI 可以是三线或者四线，有以下特点。

- 支持主机或从机操作
- 可选择最低位或最高位优先传输
- 4 种可编程的比特率
- 可编程的极性和相位
- 发送结束中断标志
- 写入冲突标志保护机制
- 支持主模式故障出错中断

图 19-1-1 和图 19-1-2 分别是 SPI 主机模式和从机模式的原理示意图。

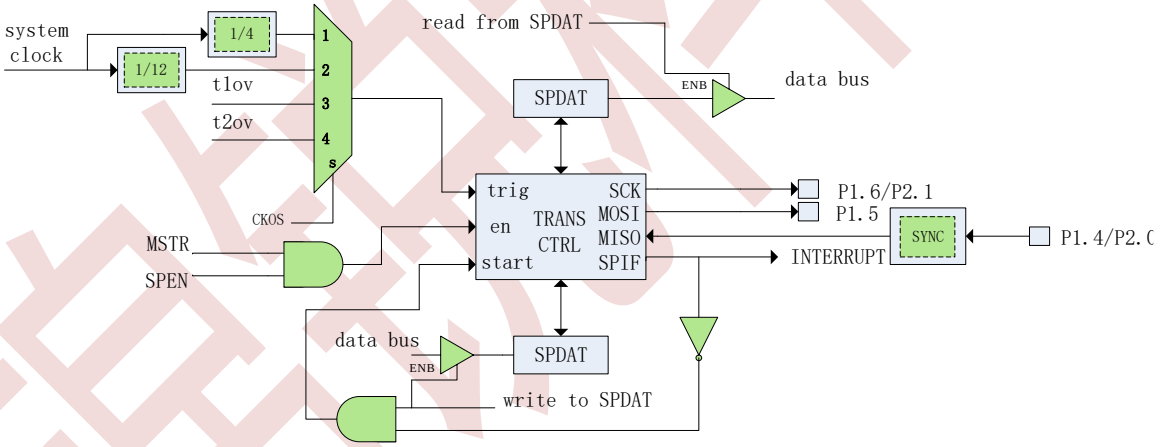


图 19-1-1 SPI 主机模式示意图

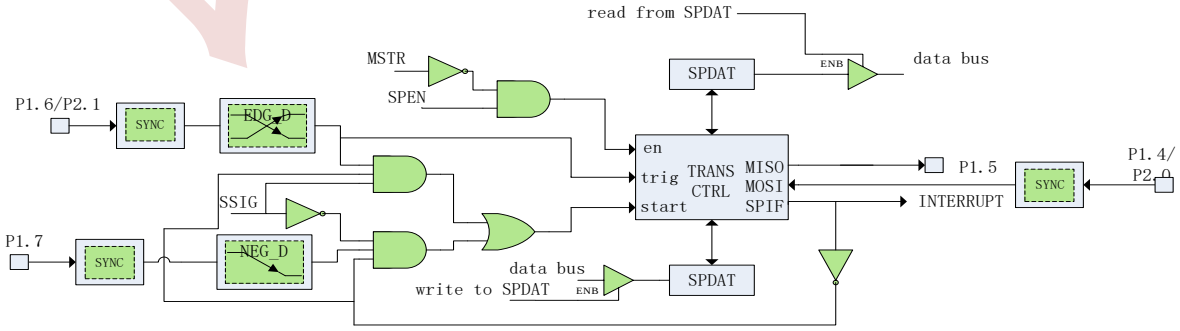


图 19-1-2 SPI 从机模式示意图

表 19-1-1 SPI 工作模式

名称	描述
主机模式	所有的传输行为都由主机发起，包括 SCK 和 SSB 信号的产生等。 当设置 MSTR（SPCON[4]）位为 1，SPI 处于主机模式。用户需要另选择一个 GPIO 作为片选引脚，连接从机 SSB，数据传输开始前，主机拉低这个引脚，传输结束后拉高。 在主机模式，写入寄存器 SPDAT 的会启动数据传输。数据在时钟有效沿从 MOSI 移位输出。
从机模式	当设置 MSTR 位为 0，SPI 处于从机模式。 当 SSIG（SPCON[5]）为 1，则 SSB 引脚无效，SPI 为三线通信，从机默认片选有效；当 SSIG 为 0，SSB 引脚有效，SSB 为低电平表示从机被片选。

表 19-1-2 SPI 接口引脚描述

名称	描述
MOSI	主机输出，从机输入 当 SPI 作为主机时该引脚为主机数据输出端口，作为从机时为从机数据输入端口
MISO	主机输入，从机输出 当 SPI 作为主机时该引脚为主机数据输入端口，作为从机时为从机数据输出端口
SCK	串行时钟 当 SPI 作为主机时该引脚为串行时钟输出端口，作为从机时为串行时钟输入端口
SSB	从机选择 当 SPI 引脚主机时该引脚为从机选择输入端口，作为从机时为从机选择输入端口

表 19-1-3 SPI 相位与极性

名称	描述
CPHA	相位控制位 0：表示在 SCK 奇数边缘（1,3,5,...,15）采样数据 1：表示在 SCK 偶数边缘（2,4,6,...,16）采样数据
CPOL	极性控制位 0：表示 SCK 空闲时处于低电平 1：表示 SCK 空闲时处于高电平

结合表 19-1-3，实际传输时的波形如图 19-1-3 和图 19-1-4 所示。

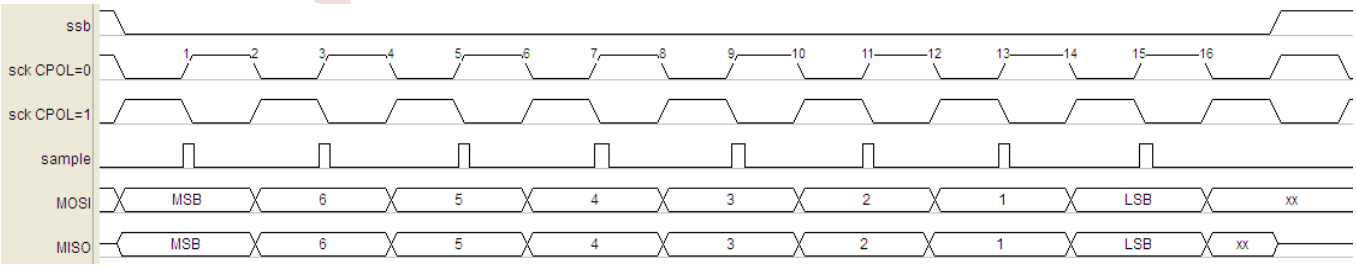


图 19-1-3 CPHA=0 时 SPI 时序图

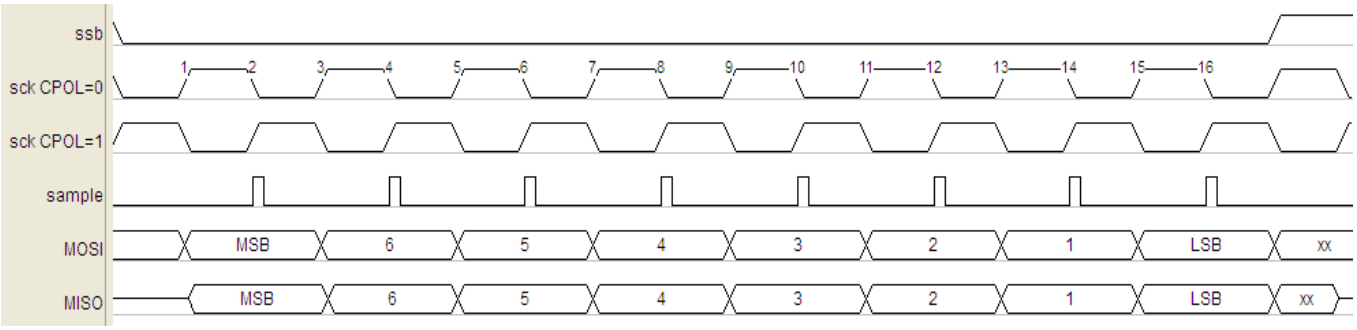


图 19-1-4 CPHA=1 时 SPI 时序图

19.2 寄存器描述

表 19-2-1 寄存器 SPCON

A9H	7	6	5	4	3	2	1	0
SPCON	SPEN	LSBF	SSIG	MSTR	CPOL	CPHA	CKOS[1:0]	
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	SPEN	SPI 模块使能位，1 有效
6	LSBF	低位或高位优先发送/接收选择位 0：高位先发 1：低位先发
5	SSIG	SSB 引脚无效控制位，默认为 0，此时 SSB 信号有效
4	MSTR	主机/从机选择位 0：从机 1：主机
3	CPOL	时钟极性选择位 0：默认情况下时钟为低 1：默认情况下时钟为高
2	CPHA	时钟相位选择位 0：在时钟离开默认情况时采样数据 1：在时钟回到默认情况时采样数据
1~0	CKOS	SPI 输出时钟选择位 00：1/8 系统时钟 01：1/24 系统时钟 10：使用定时器 1 溢出标志，每两次溢出传输一次数据

表 19-2-2 寄存器 SPDAT

AAH	7	6	5	4	3	2	1	0
SPDAT	RBUF[7:0]							
R/W	R							
初始值	0	0	0	0	0	0	0	0
SPDAT	TBUF[7:0]							
R/W	W							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~0	SPDAT	写 SPDAT 时，写入内部的 TBUF，读 SPDAT 时，从 RBUF 读出

表 19-2-3 寄存器 SPSTA

ABH	7	6	5	4	3	2	1	0
SPSTA	SPIE	-	-	-	-	WCOL	MODF	SPIF
R/W	R/W	-	-	-	-	R/W	R/W	R/W
初始值	0	-	-	-	-	0	0	0

位编号	位符号	说明
7	SPIE	SPI 中断使能位，1 有效
6~3	-	-
2	WCOL	写入冲突标志位，在数据正在发送时，如有软件有写 SPDAT 的操作，此时数据无法写入，即产生写入冲突标志。该位 1 有效，写 1 清 0，有效时不会产生中断
1	MODF	故障模式标志位，1 有效，表明 SSB 在不正确的逻辑电平下，写 1 清 0，有效时会产生中断
0	SPIF	数据传输完成标志位，1 有效，写 1 清 0，有效时会产生中断

20 模/数字转换器（ADC）

20.1 功能简介

模拟/数字转换器是 12 位逐次逼近寄存器型（SAR）ADC，最多提供多达 26 个输入通道。ADC 时钟源是系统时钟，可设置时钟预分频。ADC 有多种参考电压源可选，其中选择内部电压为参考电压时可用于检测芯片供电电压。ADC 选择内部电压为参考电压时有自动校正功能，避免芯片一致性问题。

20.2 主要特性

- 12 位的分辨率
- 最多提供多达 26 个输入通道
- 支持 ADC 中断
- 可设置 ADC 时钟预分频
- 多种参考电压可选：内部参考电压、VDD。
- 支持 VDD 和参考地电压的测量
- 选择内部参考电压时，支持自动数据校正功能
- 选择内部参考电压时,可测量 VDD 电压
- 输入电压范围：VSS<=VIN<=VDD。

20.3 结构框图

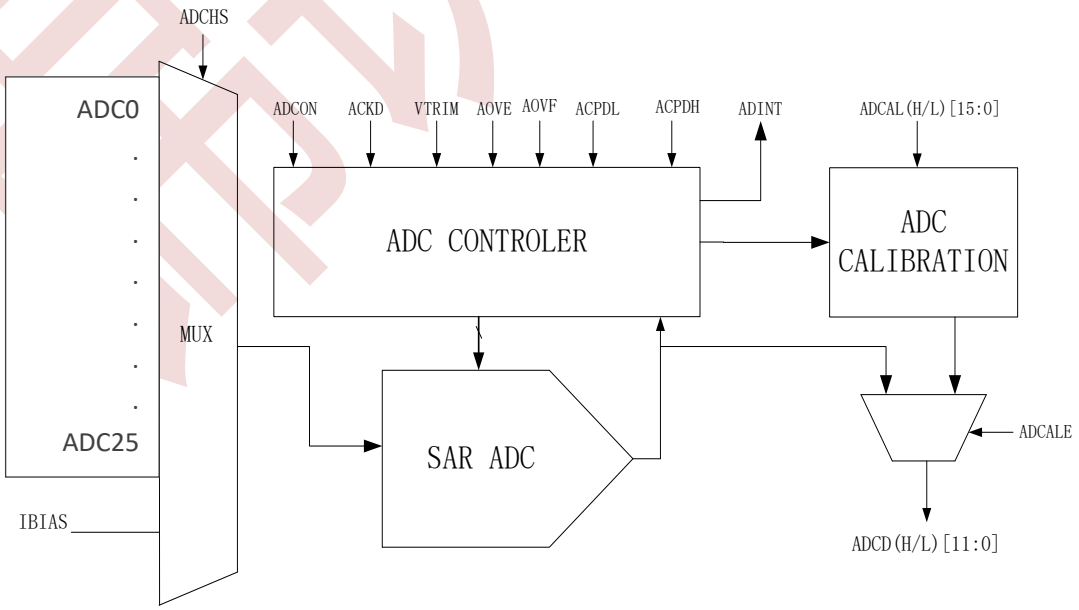


图 20-3-1 ADC 结构示意图

20.4 功能描述

ADC 的启动通过 AST 位使能，设置 AST=1 后，ADC 模块对 ADCHS 选择的输入电压源进行模/数转换。ADC 可通过 ACKD 设置时钟预分频，由系统时钟预分频后的时钟作为 ADC 转换时钟。在 ADC 时钟不变的条件下，ADC 的单次转换时间是由 HTME 设置的，转换时间为(13+2^HTME)个 ADC 时钟周期。当转换结束后，12 位的 A/D 值会被加载到寄存器 ADCDH、ADCDL，转换完后的 2.5 个时钟周期，AST 位自动清 0，同时中断标志 ADIF 位会置 1，如果 ADC 中断使能，会产生 ADC 中断。图 20-4-1 为 ADC 的转换时序图。

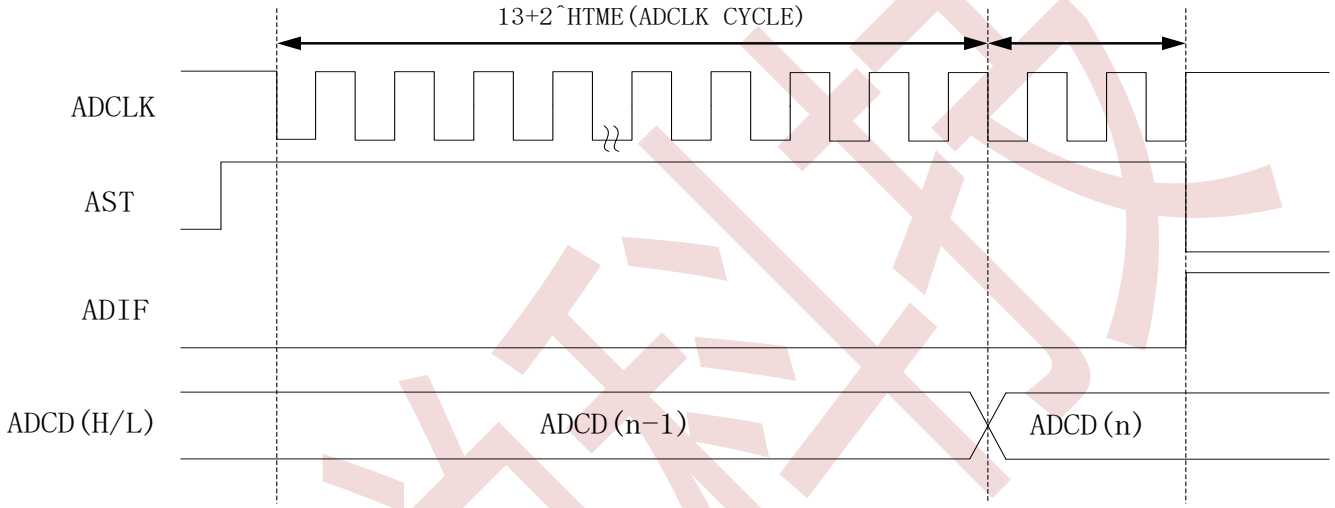


图 20-4-1 ADC 时序示意图

- **ADC 数据校正**

当选择内部 1.5V 作为参考电压时，由于芯片的离散性，每个芯片的内部电压不一定完全相同，导致每个芯片的 ADC 转换结果也有一定的偏差，所以在 ADC 转换完后，有必要对 AD 值进行校正。芯片在出厂时，会对每个芯片的内部电压进行测试，得出与内部电压成反比例的校正值，在芯片上电启动时，自动将此校正值加载到寄存器 ADCALL、ADCALH，当 ADC 转换完成后自动将 AD 值根据校正寄存器 ADCALL、ADCALH 的值进行等比例换算，得出准确的 AD 值，最终的 AD 值也是存放在寄存器 ADCD 中。此功能通过 ADCALE 使能，对于用户来说，在应用时只需要设置 ADCALE=1 即可，校正过程是自动完成的。

20.5 寄存器描述

表 20-5-1 寄存器 ADCON

8060H	7	6	5	4	3	2	1	0
ADCON	AST	ADIE	ADCIF	HTME			ADCALE	VSEL
R/W	R/W	R/W	R/W	R/W			R/W	R/W
初始值	0	0	0	0	1	0	1	0
位编号	位符号	说明						
7	AST	ADC 转换开始控制位，写 1 启动转换，转换后硬件自动清 0						

6	ADIE	ADC 中断使能位，1 有效
5	ADCIF	ADC 中断标志位，写 1 清 0
4~2	HTME	采样保持周期数为 2 的 HTME 次幂
1	ADCALE	ADC 校准使能位，1 有效 此位只有选择参考电压为内部 1.5V 时才有效，当 ADCALE=1，ADC 的转换结果将根据 ADCAL 寄存器的数值进行校准。具体参考寄存器 ADCAL 说明。
0	VSEL	ADC 参考电压选择位 0：内部 1.5V(INNER_VREF)作为参考电压 1：电源作为参考电压

表 20-5-2 寄存器 ADCFGL

8061H	7	6	5	4	3	2	1	0
ADCFGL	ACKD			ADCHS				
R/W	R/W			R/W				
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7~5	ACKD	ADC 时钟分频设置 000：不分频 001：2 分频 010：4 分频 … 111：14 分频
4~0	ADCHS	ADC 通道使能选择位域 00000：通道关闭 00001：通道 AD_CH[0](P0.0)使能 00010：通道 AD_CH[1](P0.1)使能 00011：通道 AD_CH[2](P0.2)使能 00100：通道 AD_CH[3](P0.3)使能 00101：通道 AD_CH[4](P0.4)使能 00110：通道 AD_CH[5](P0.5)使能 00111：通道 AD_CH[6](P0.6)使能 01000：通道 AD_CH[7](P0.7)使能 01001：通道 AD_CH[8](P1.0)使能 01010：通道 AD_CH[9](P1.1)使能 01011：通道 AD_CH[10](P1.2)使能 01100：通道 AD_CH[11](P1.3)使能 01101：通道 AD_CH[12](P1.4)使能 01110：通道 AD_CH[13](P1.5)使能 01111：通道 AD_CH[14](P1.6)使能 10000：通道 AD_CH[15](P1.7)使能 10001：通道 AD_CH[16](P2.0)使能 10010：通道 AD_CH[17](P2.1)使能

		10011：通道 AD_CH[18](P2.2)使能 10100：通道 AD_CH[19](P2.3)使能 10101：通道 AD_CH[20](P2.4)使能 10110：通道 AD_CH[21](P2.5)使能 10111：通道 AD_CH[22](P2.6)使能 11000：通道 AD_CH[23](P2.7)使能 11001：通道 AD_CH[24](P3.0)使能 11010：通道 AD_CH[25](P3.1)使能 11011：检测 VDD 的 1/4 使能 其他：通道关闭

表 20-5-4 寄存器 ADCAL

8064H	7	6	5	4	3	2	1	0
ADCALL	ADCAL[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
8065H	7	6	5	4	3	2	1	0
ADCALH	ADCAL[15:8]							
R/W	R/W							
初始值	0	0	0	1	0	0	0	0

位编号	位符号	说明
15~0	ADCAL	ADC 校准寄存器，只有 ADCALE=1 并且选择参考电压为内部 1.5V 才有效。有效时，ADC 的输出按照如下公式： ADCDL=（ADC 转换结果*ADCAL）/32768 ***备注：**此校准值上电后自动加载，用户不允许修改。*

表 20-5-5 寄存器 ADCD

8062H	7	6	5	4	3	2	1	0
ADCDL	ADCDL[3:0]				-	-	-	-
R/W	R				-	-	-	-
初始值	0	0	0	0	-	-	-	-
8063H	7	6	5	4	3	2	1	0
ADCDH	ADCDH[11:4]							
R/W	R							
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
11~0	ADCD	ADC 转换值

21 电容式触摸按键（Touch Key）

21.1 功能简介

CA51M151 系列芯片的触摸功能模块具有优越的抗干扰性能，可通过 EFT、CS 等测试。触摸模块最大可支持多达 26 个通道。内置触摸感应控制器，在应用时无需外接电容。针对有低功耗需求的应用，支持触摸低功耗模式，以达到产品省电功能。

21.2 主要特性

- 内置触摸感应控制器，不需外接电容
- 最大支持 26 路触摸通道
- 高抗干扰性能，符合 EMC(CS)标准
- 支持触摸中断
- 支持触摸通道并联，可实现触摸低功耗模式

21.3 功能描述

触摸功能需通过一系列寄存器设置相应参数，包括充放电时钟频率、计数时钟频率、计数位数、放电电阻、比较器电压、充电电流源电流等。参数设置完成后，通过 TKEN 使能触摸功能，设置触摸通道后，设 TKST=1 即可启动相应通道的触摸数据采集。数据采集完成后，16 位触摸数据存放在 TKDH、THDL，并产生触摸中断标志 TKIF。

触摸模块支持多个触摸通道并联，此功能用于触摸省电模式可显著降低功耗。

更多与触摸相关的内容可参考触摸库及相关文档描述，并建议在本公司提供的触摸库基础上进行二次开发。

21.4 寄存器描述

表 21-4-1 寄存器 TKST

F8H	7	6	5	4	3	2	1	0
TKST	-	-	-	-	-	-	-	TKST
R/W	-	-	-	-	-	-	-	RW
初始值	-	-	-	-	-	-	-	0

位编号	位符号	说明
7~1	-	-
0	TKST	数据采集启动使能位，1 有效，采集完后自动清 0

表 21-4-2 寄存器 TKCFG1

F9H	7	6	5	4	3	2	1	0
TKCFG1	-	TKEN	TKDIV					
R/W	-	R/W	R/W					
初始值	-	0	0	0	0	0	0	0

位编号	位符号	说明
7	-	-
6	TKEN	触摸模拟模块开启使能
5~0	TKDIV	充放电时钟选择： 0~61：充放电时钟为固定频率，时钟频率为 F = 24/(TKDIV +4) 62~63：充放电时钟跳频，最高频率 3M，最低频率 1M，中心频率为 1.5MHz

表 21-4-3 寄存器 TKCFG2

FAH	7	6	5	4	3	2	1	0
TKCFG2	-	TKPULL8	PAREN	TKCADDR				
R/W	-	R/W	R/W	R/W				
初始值	-	1	0	0				

位编号	位符号	说明
7	-	-
6	TKPULL8	充电电流源配置最高位
5	PAREN	0：单通道 1：所有使能的通道并联
4~0	TKCADDR	通道选择位： 00000: TK0 00001:TK1 …… 11001:TK25

表 21-4-4 寄存器 TKCFG3

FBH	7	6	5	4	3	2	1	0
TKCFG3	-	RESO			TKCKSEL		CHGSEL	DCHSEL
R/W	-	R/W			R/W		R/W	R/W
初始值	-	0	0	1	1	1	1	1

位编号	位符号	说明
7~5	-	-
6~4	RESO	计数器位数选择 000: 9位

		001: 10位 010: 11位 011: 12位 100: 13位 101: 14位 110: 15位 111: 16位
3~2	TKCKSEL	计数时钟选择 0: 24Mhz 3: 4Mhz 其他值：无效
1	CHGSEL	预充电时间选择 0：20 us 1：40us
0	DCHSEL	预放电时间选择 0：2 us 1：10us

表 21-4-5 寄存器 TKPULL

FEH	7	6	5	4	3	2	1	0
TKPULL[7:0]	TKPULL[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	TKPULL	充电电流源设置低 8 位						

表 21-4-6 寄存器 TKDL/TKDH

FCH	7	6	5	4	3	2	1	0
TKDL	TKD[7:0]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
FDH	7	6	5	4	3	2	1	0
TKDH	TKD[15:8]							
R/W	R/W							
初始值	0	0	0	0	0	0	0	0
位编号	位符号	说明						
7~0	TKD	触摸通道当前采样值						

表 21-4-7 寄存器 TKPULLTRIM

F2H	7	6	5	4	3	2	1	0
TKPULLTRIM	-	-	TKPULLTRIM [5:0]					
R/W	-	-	RW					
初始值	-	-	0	0	0	0	0	0

位编号	位符号	说明
7~0	TKPULLTRIM	充电电流补偿值

表 21-4-8 寄存器 TKIE

F4H	7	6	5	4	3	2	1	0
TKIE	TKIE	-	-	-	-	-	-	TKIF
R/W	RW	-	-	-	-	-	-	R
初始值	0	-	-	-	-	-	-	0

位编号	位符号	说明
7	TKIE	触摸中断使能位，1 有效
6~1	-	-
0	TKIF	触摸采集中断标志位，写 1 清 0

表 21-4-9 寄存器 TKCFG4

FFH	7	6	5	4	3	2	1	0
TKCFG4	-	-	RBSEL			VRSEL		
R/W	-	-	RW			RW		
初始值	-	-	0	0	0	0	0	0

位编号	位符号	说明
7~6	-	-
5~3	RBSEL	放电电阻大小选择 000：20K 001：30K 010：40K 011：50K 100：70K 101：90K 110：200K 111：350K
2~0	VRSEL	比较器门限电压选择

		000：1.8V 001：2.1V 010：2.5V 011：2.8V 100：3.2V 101：3.5V 110：3.9V 111：4.1V

锦锐科技

21 低电压检测（LVD）

21.1 功能简介

低电压检测（LVD）用于监控芯片自身的供电 VDD，可设置检测电压范围为 2.2V、2.5V、2.7V、3.0V、3.3V、3.6V、3.9V 和 4.2V。当 VDD 小于所设定的电压值时，可设置触发中断或复位。

***备注：**由于生产工艺的影响，芯片之间 LVD 触发电压存在一定的差异。*

LVD 结构图如图 21-1-1 所示。

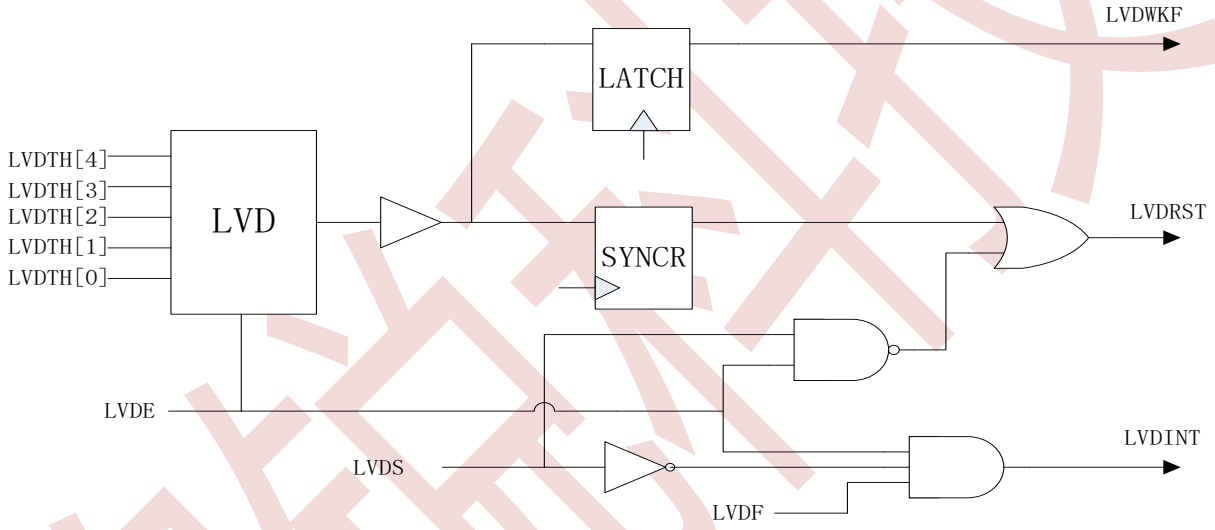


图 21-1-1 LVD 模块示意图

21.2 功能描述

LVD 功能通过 LVDE 位使能，而检测的电压则通过 LVDTH 位域设置。当芯片 VDD 小于所设置的电压时，LVD 功能产生的标志 LVDF 位将置 1，如果 LVDS=0，会产生 LVD 中断，如果 LVDS=1，会产生复位。要注意的是，LVD 复位产生之后，LVD 自身的电路并不会复位，寄存器 LVDCON 还会保持之前的状态，所以，当 LVD 复位产生之后，如果 VDD 持续低于所设定的电压，芯片将会一直处于复位状态。同样地，当 LVD 中断产生后，如果 VDD 持续低于所设定的电压，LVD 中断也会重复地产生。

21.3 寄存器描述

表 21-3-1 寄存器 LVDCON

E8H	7	6	5	4	3	2	1	0
LVDCON	LVDE	LVDS	LVDF			LVDTH[2:0]		
R/W	R/W	R/W	R/W			R/W		
初始值	0	0	0	0	0	0	0	0

位编号	位符号	说明
7	LVDE	LVD 使能位，1 有效
6	LVDS	LVD 功能选择位 0：中断 1：复位
5	LVDF	LVD 产生标志位，写 1 清 0
4~3	-	-
2~0	LVDTH	LVD 触发电平选择位域 000：2.2V 001：2.5V 010：2.7V 011：3.0V 100：3.3V 101：3.6V 110：3.9V 111：4.2V

21.4 LVD 控制例程

LVD 中断例程

例如，设置 LVD 为中断模式，检测电压为 3V，程序如下：

```
-----------------------------------------------------------------------------------------
#define LVDE(N)         (N<<7)    //N=0~1
#define LVDS_reset      (1<<6)
#define LVDS_int        (0<<6)
#define LVDF            (1<<5)
#define LVDTH_3V        3
void LVD_init(void)
{
    LVDCON = LVDE(1) | LVDS_int | LVDF |LVDTH_3V;//设置 LVD 使能，设置 LVD 为中断模式，检测电压为 3V
    INT4EN = 1;                 //INT4 中断使能
    EA = 1;                     //开启总中断
}
void INT4_ISR (void) interrupt 6
{
    if(LVDCON & LVDF)
    {
        LVDCON |= LVDF;         //清除 LVD 中断标志
    }
}
-----------------------------------------------------------------------------------------
```

LVD 复位例程

例如，设置 LVD 为复位模式，检测电压为 3V，程序如下：

```
-----------------------------------------------------------------------------------------
#define LVDE(N)         (N<<7)    //N=0~1
#define LVDS_reset      (1<<6)
#define LVDS_int        (0<<6)
#define LVDF            (1<<5)
#define LVDTH_3V        3
void LVD_init(void)
{
    LVDCON = LVDE(1) | LVDS_reset | LVDF | LVDTH_3V;//设置 LVD 使能，设置 LVD 为复位模式，检测电压为 3V
}
-----------------------------------------------------------------------------------------
```

22 程序下载和仿真

22.1 程序下载

CA51M151 系列芯片主要采用 ISP 方式下载程序，芯片通过 I2C 接口与下载工具相连接，不同型号其下载口不同，详见表 22-1-1。

表 22-1-1 下载/仿真接口表

芯片型号	下载/仿真接口
CA51M151S6A CA51M151P6A	P1.4（IIC_SDA）、P1.6（IIC_SCL）
CA51M151S6B	P3.1（IIC_SDA）、P3.0（IIC_SCL）
CA51M151N3	P0.0（IIC_SDA）、P0.1（IIC_SCL）
CA51M151S4A	P1.4（IIC_SDA）、P1.6（IIC_SCL）
CA51M151T3A	P3.0（IIC_SDA）、P3.1（IIC_SCL）
CA51M151N2	P0.0（IIC_SDA）、P0.1（IIC_SCL）
CA51M151S3A	P1.4（IIC_SDA）、P1.6（IIC_SCL）

更多关于程序下载步骤的细节请参考“CACHIP 开发下载工具使用说明”。

22.2 在线仿真

CA51M151 系列芯片支持在线仿真，芯片与仿真器之间通过 IIC 接口进行通信，出厂 IIC 接口详见表 22-1-1。要注意的是，由于芯片与仿真器间通过 IIC 通信，所以与仿真器连接的 I2C 接口引脚不能设置为其他功能，并且应用程序里不能使用 IIC 功能，否则将无法进入仿真模式。另外，由于 I2C 的通信速度是由主时钟决定，所以应用程序里不能将主时钟设置为低速时钟，也不能进入省电模式，否则都会影响芯片与仿真器间的通信。

当芯片进入仿真模式后，TSMODE 位（PCON[2]）置 1，应用程序可通过判断此位状态来决定是否切换至低速时钟或进入省电模式。

更多关于仿真功能的细节可参考仿真器的相关文档介绍。

23 电气特性

23.1 极限参数

参数	最小值	最大值	单位
直流供电电压	-0.3	6	V
I/O 引脚输入电压	-0.3	VDD+0.3	V
工作环境温度	-40	85	℃
储存温度	-55	125	℃
CPU 工作频率	-	8	MHz

*备注：超过“**极限参数**”范围有可能对芯片造成损坏，无法预期芯片在上述范围外的工作状态，若长期在标示范围外工作，可能会影响芯片的可靠性。*

23.2 直流电气特性

直流电气特性（VDD=2.2-5.5V, TA=25℃，除非其它说明）

芯片参数	符号	工作电压	最小值	典型值	最大值	单位	测试条件
工作电流	Iop1	VDD=2.7V	-	1.49	-	mA	系统时钟为 IRCH(8MHz)，其他时钟关闭，所有输出引脚无负载，所有数字输入引脚不浮动，所有外设关闭，CPU 执行 NOP 指令
		VDD=3.3V	-	1.70	-		
		VDD=5.0V	-	2.57	-		
	Iop3	VDD=2.2V	-	85	-	uA	系统时钟为 IRCL(128kHZ)，其他时钟关闭，所有输出引脚无负载，所有数字输入引脚不浮动，所有外设关闭，CPU 执行 NOP 指令
		VDD=3.3V	-	96	-		
		VDD=5.0V	-	120	-		
STOP 模式电流	Istp	VDD=2.7V	-	4.7	-	uA	所有时钟关闭，所有输出引脚无负载，所有数字输入引脚不浮动，所有外设关闭，Flash 进入睡眠模式，CPU 进入 STOP 模式。
		VDD=3.3V	-	4.9	-		
		VDD=5.0V	-	5.2	-		
IDLE 模式电流	Iidl1	VDD=2.7V	-	0.72	-	mA	系统时钟设为 IRCH(8MHz)，其他时钟关闭，所有输出引脚无负载，所有数字输入引脚不浮动，所有外设关闭，Flash 进入睡眠模式，CPU 进入 IDLE 模式。
		VDD=3.3V	-	0.86	-		
		VDD=5.0V	-	1.35	-		
	Iidl2	VDD=2.2V	-	17	-	uA	系统时钟设为 IRCL(128KHz)，其他时钟关闭，所有输出引脚无负载，所
		VDD=3.3V	-	23	-		

		VDD=5.0V	-	33	-		有数字输入引脚不浮动，所有外设关闭， CPU 进入 IDLE 模式。
IO 端口输入高电压（斯密特模式开启）	Vhi1	VDD=2.7V	1.17	-	2.7	V	-
		VDD=3.3V	1.45	-	3.3		
		VDD=5.0V	2.25	-	5		
IO 端口输入高电压（斯密特模式关闭）	Vhi2	VDD=2.7V	-	0.5*VDD	VDD	V	-
		VDD=3.3V					
		VDD=5.0V					
IO 端口输入低电压（斯密特模式开启）	Vlo1	VDD=2.7V	0	-	1.04	V	-
		VDD=3.3V	0	-	1.32		
		VDD=5.0V	0	-	1.85		
IO 端口输入低电压（斯密特模式关闭）	Vlo2	VDD=2.7V	0	0.5*VDD	-	V	-
		VDD=3.3V					
		VDD=5.0V					
IO 端口推电流	Ipu	VDD=3.3V	-	2mA @DRV=0 3mA @DRV=1 4mA @DRV=2 5mA @DRV=3	-	mA	IO 设为推挽输出模式，4 级驱动能力可选(DRV)，Voh=0.9*VDD
		VDD=5.0V	-	6mA @DRV=0 8mA @DRV=1 10mA @DRV=2 12mA @DRV=3	-		
IO 端口灌电流	Iol	VDD=3.3V	-	4mA @DRV=0/1 8mA @DRV=2/3	-	mA	IO 设为推挽输出模式，2 级驱动能力可选(DRV)，Vol=0.1*VDD
		VDD=5.0V	-	8mA @DRV=0/1 16mA @DRV=2/3	-		
IO 端口下拉电阻	Rd1	VDD=2.2~5.5V	-	30	-		-
IO 端口上拉电阻	Ru1	VDD=2.2~5.5V	-	30	-		-

说明：以上参数是随机抽取的典型芯片测试结果，仅供参考。

23.3 ESD/EFT 特性

符号	模式	条件	封装	最大值	单位
V_{ESD}	静电放电 （人体放电模式 HBM）	TA=+25℃	SOP28	VDD/VSS 8000	V
				I/O 5000	
	静电放电 （组件放电模式 CDM）			1000	V

符号	条件		封装	最大值	单位
EFT	F_{sys} = 8MHz / UART 通讯	TA=+25℃	SOP28	±4000	V

23.4 交流电气特性

交流电气特性（VDD=2.2-5.5V, TA=25℃，除非其它说明）

芯片参数	符号	最小值	典型值	最大值	单位	条件
内部低速时钟（IRCL）起振时间	Trc1	-	50	-	us	IRCL 频率为 128K
内部高速时钟（IRCH）起振时间	Trc2	-	10	-	us	IRCH 频率为 8MHz
外部时钟（32.768KHz）起振时间	Trc3		1		s	频率为 32.768KHz

备注：VDD=5V,TA=25℃,内部高速时钟出厂频率为8MHz，精度为±2%.

23.5 内部 RC 时钟温度特性

- **IRCH 温度特性**

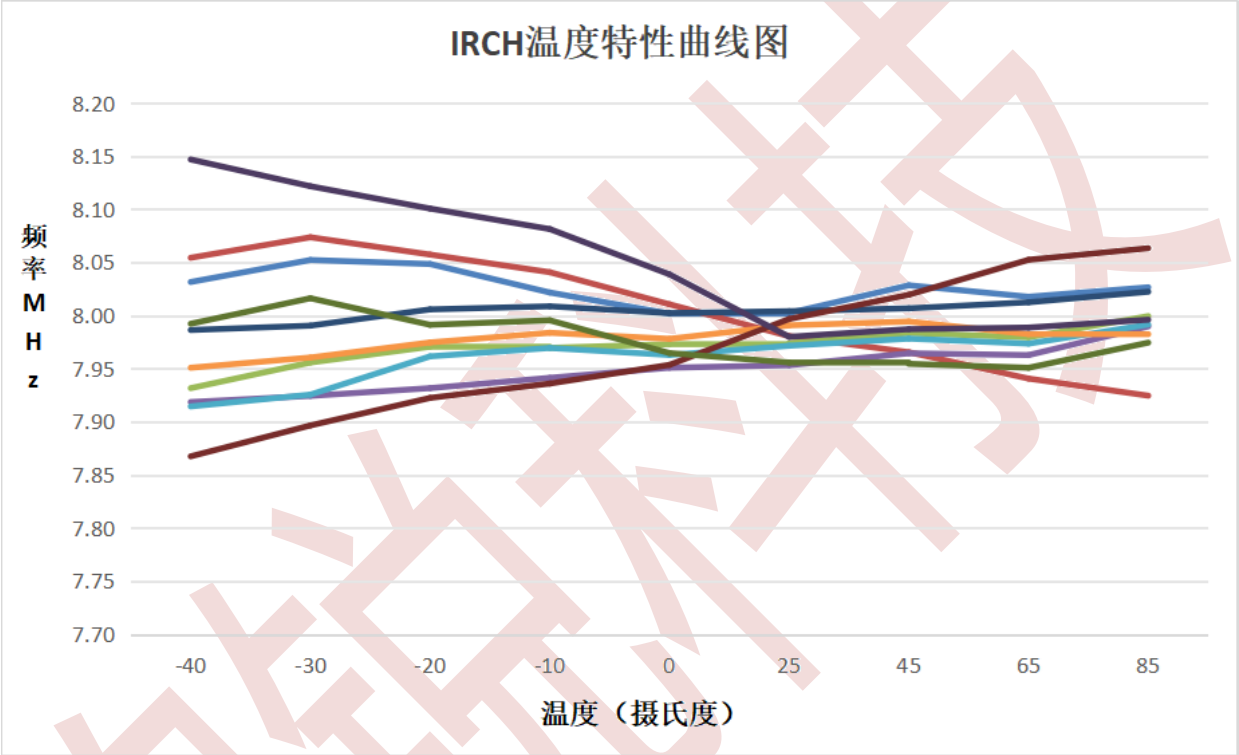
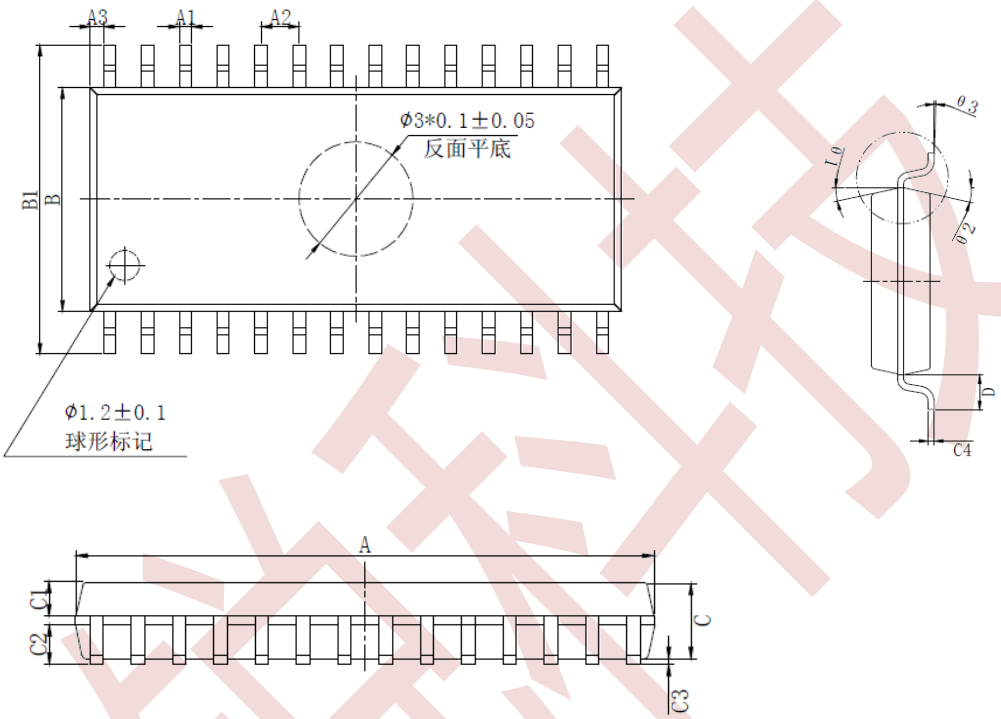


图 23-5-1 IRCH 温度特性曲线图

备注：以上图形数据为随机抽取部分芯片实测数据，数据仅供参考。

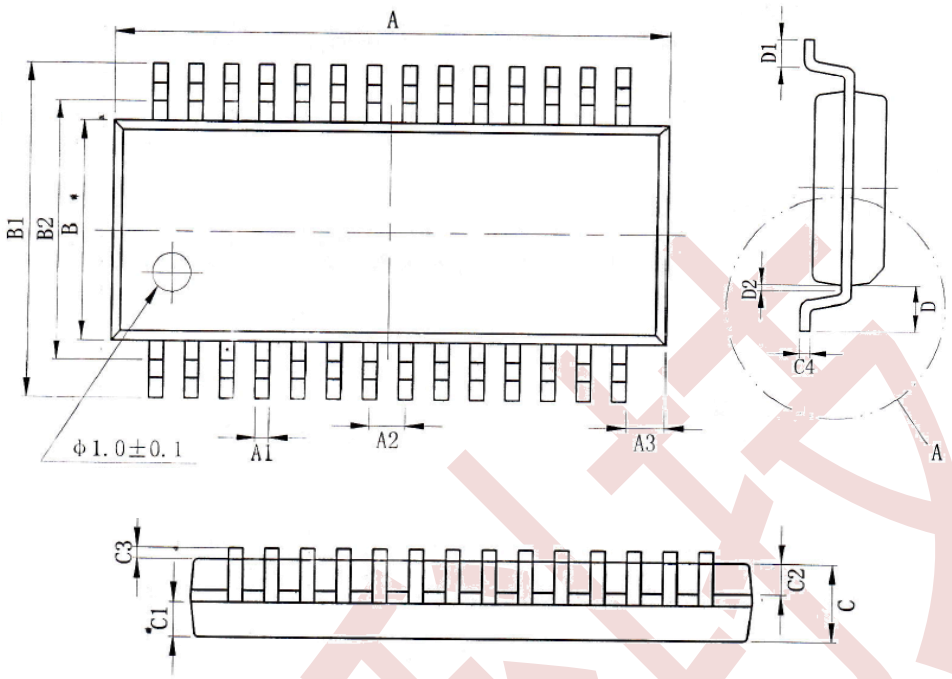
24 封装类型

封装形式：SOP28



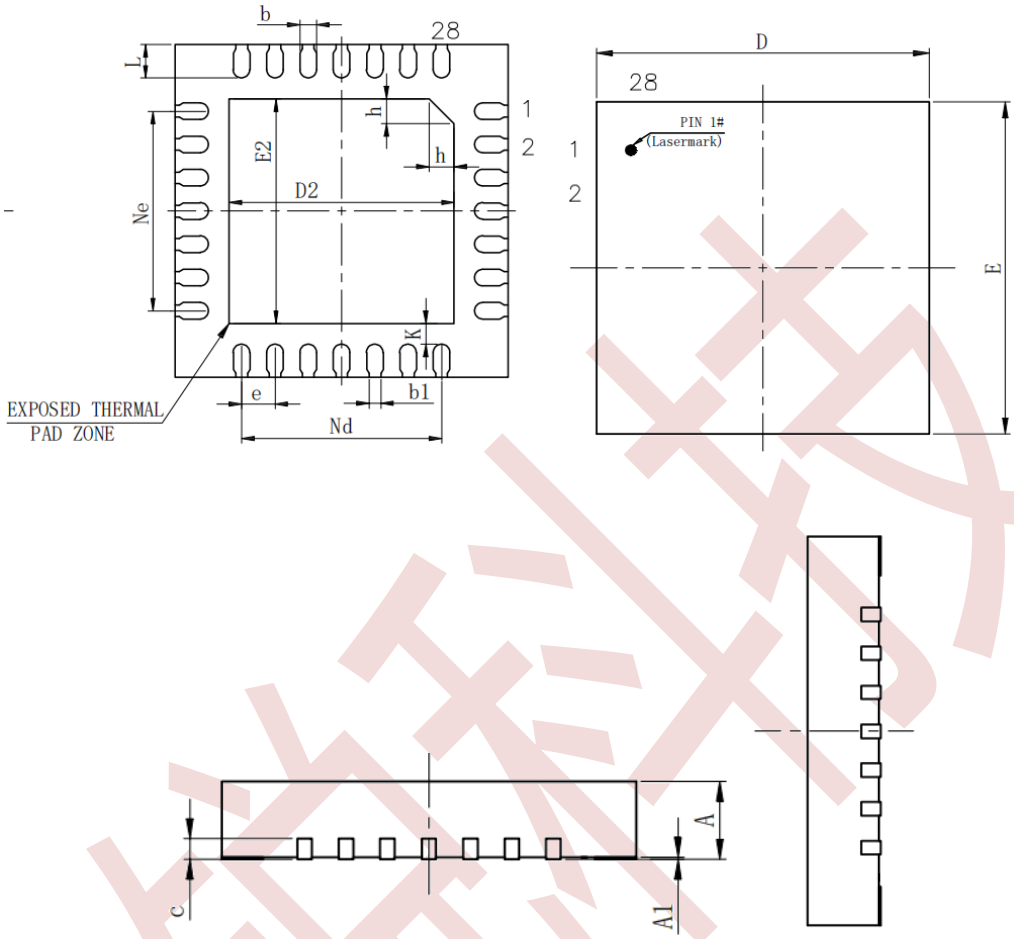
序号	最小值(mm)	标准值(mm)	最大值(mm)
A	17.90	18.00	18.10
A1	0.356	0.40	0.456
A2	1.24	1.27	1.30
A3	---	0.542 TYP	---
B	7.40	7.50	7.60
B1	10.206	10.30	10.406
C	2.18	2.23	2.28
C1	0.938	1.0	1.038
C2	0.938	1.0	1.038
C3	0.03	0.09	0.17
D	1.353	1.40	1.453
C4	0.244	0.25	0.264

封装形式： SSOP28



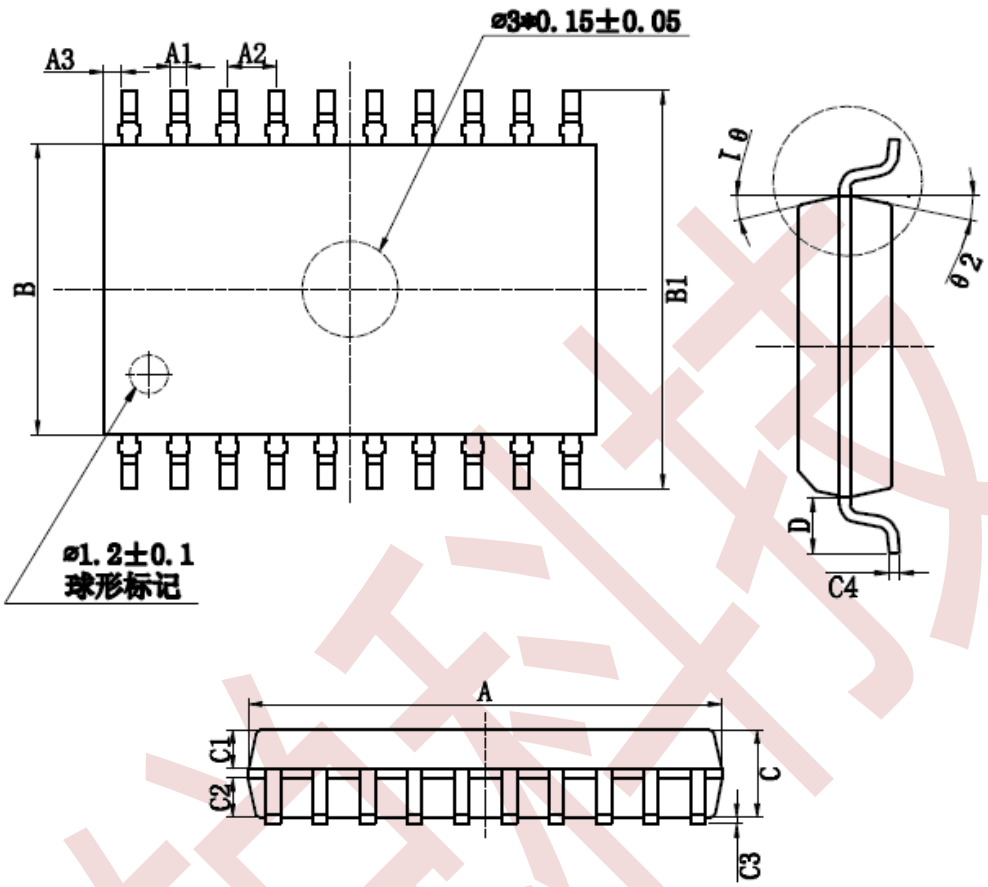
序号	最小值(mm)	标准值(mm)	最大值(mm)
A	9.80	9.90	10.00
A1	---	0.254TYP	---
A2	---	0.635TYP	---
A3	---	0.695TYP	---
B	3.85	3.90	3.95
B1	5.85	6.00	6.24
B2	---	5.00TYP	---
C	1.40	1.50	1.60
C1	0.61	0.66	0.71
C2	0.54	0.59	0.64
C3	0.05	0.15	0.25
C4	0.203	0.215	0.233
D	---	1.05TYP	---
D1	0.40	0.55	0.70
D2	0.15	0.20	0.25

封装形式： QFN28



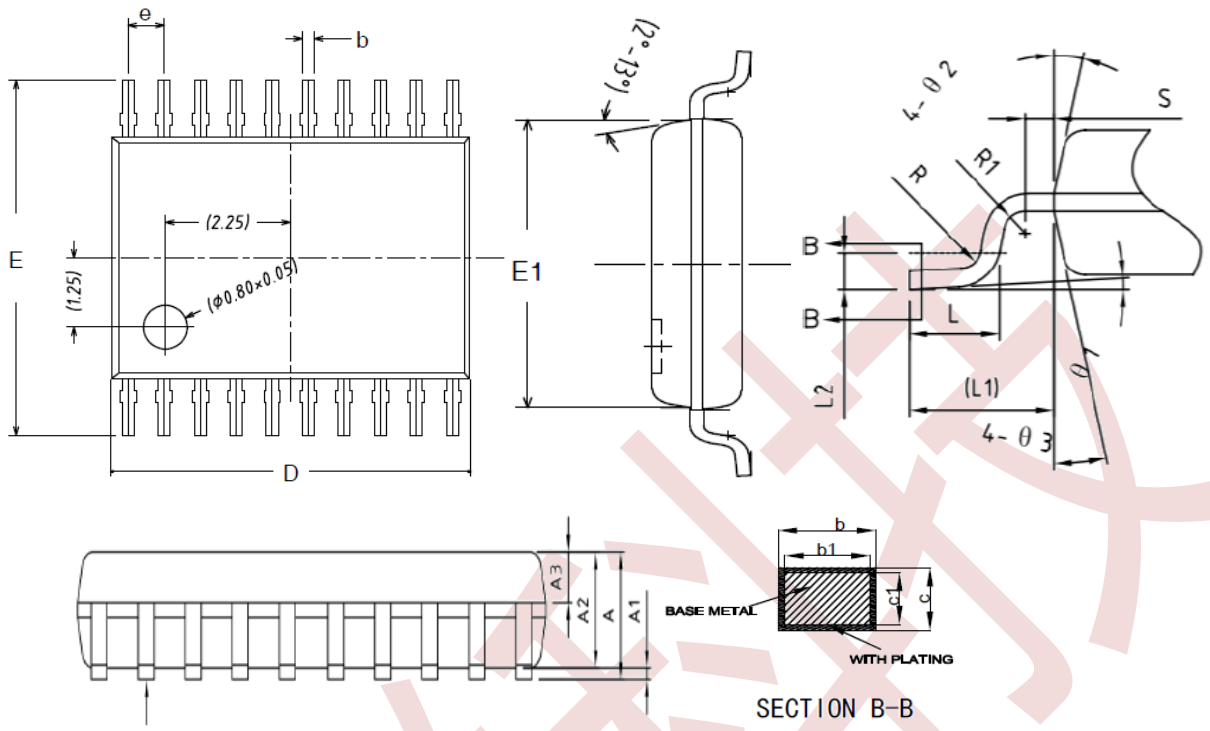
序号	最小值(mm)	标准值(mm)	最大值(mm)
A	0.70	0.75	0.80
A1	0	0.02	0.05
b	0.15	0.20	0.25
b1	0.14REF		
c	0.20REF		
D	3.90	4.0	4.1
D2	2.75	2.80	2.85
e	0.40BSC		
Ne	2.40BSC		
Nd	2.40BSC		
E	3.90	4.0	4.1
E2	2.75	2.80	2.85
L	0.30	0.35	0.40
h	0.30	0.35	0.40
K	0.25REF		

封装形式： **SOP20**



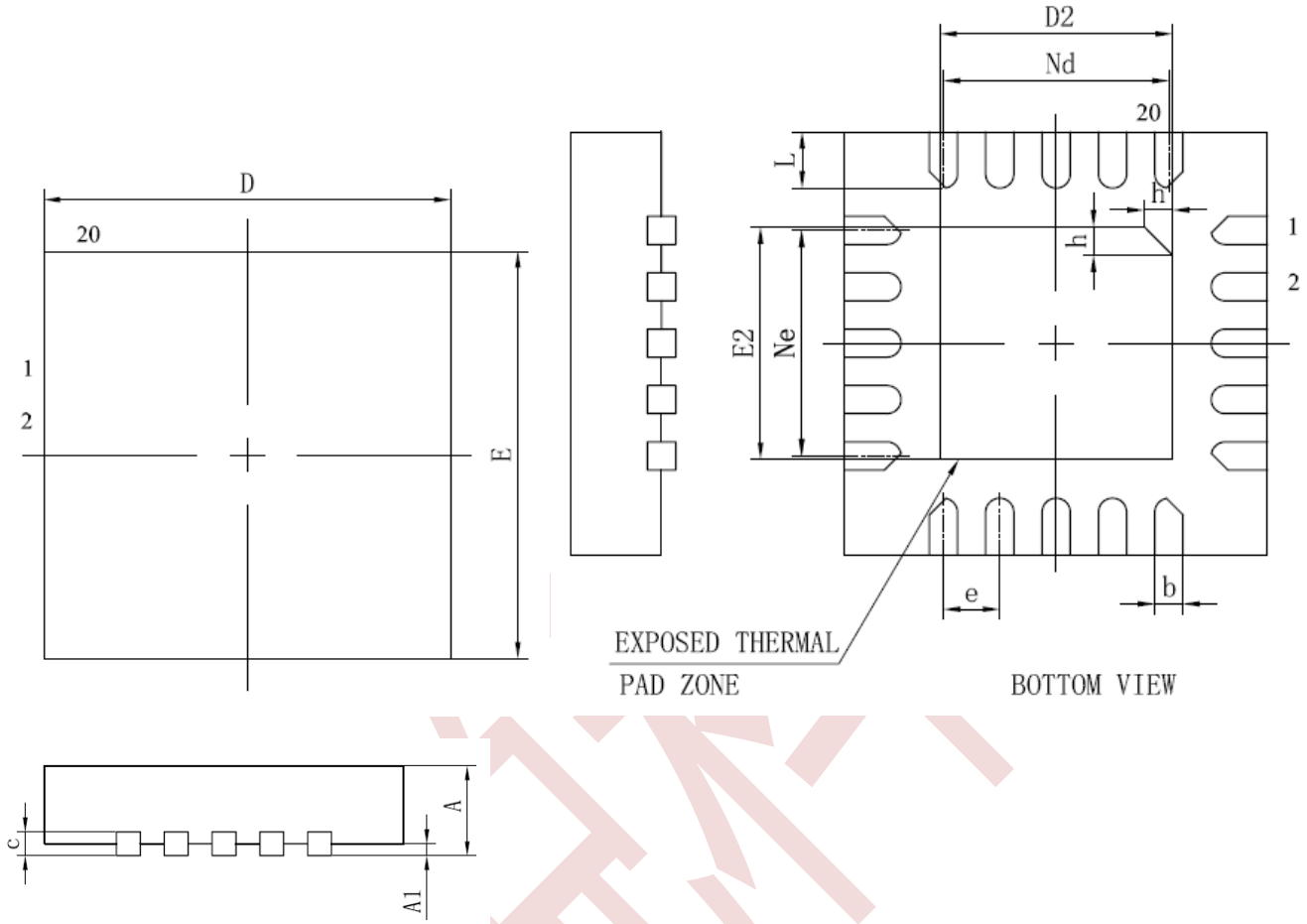
序号	最小值(mm)	标准值(mm)	最大值(mm)
A	12.65	12.70	12.80
A1	0.381	0.40	0.431
A2	1.24	1.27	1.30
A3	0.45	0.455	0.46
B	7.40	7.50	7.60
B1	10.206	10.30	10.406
C	2.18	2.23	2.28
C1	0.938	1.0	1.038
C2	0.938	1.0	1.038
C3	0.145	0.175	0.205
D	1.353	1.40	1.453
C4	0.246	0.25	0.262

封装形式： TSSOP20



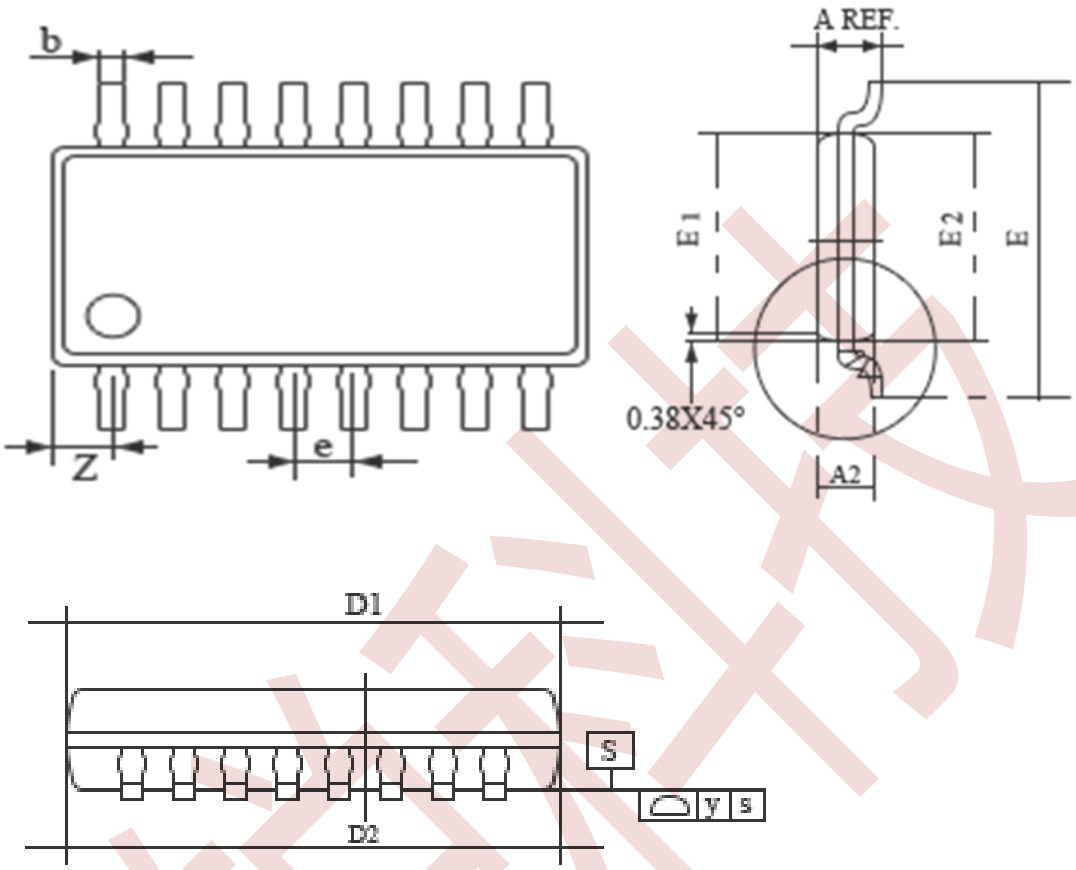
序号	最小值(mm)	标准值(mm)	最大值(mm)
A	1.0	---	1.1
A1	0.05	---	0.15
A2	---	---	0.95
A3	0.39	---	0.40
b	0.20	0.22	0.24
c	0.10	---	0.19
c1	0.10		0.15
D	6.40	6.45	6.50
E	6.25	6.40	6.55
E1	---	4.35	4.40
L	0.50	0.60	0.70
e	0.55	0.65	0.75
L2	0.25BSC		
R	0.09	---	---
L1	1.0REF		

封装形式： QFN20(3X3MM)



序号	最小值(mm)	标准值(mm)	最大值(mm)
A	0.70	0.75	0.80
A1	---	0.02	0.05
b	0.15	0.20	0.25
c	0.18	0.20	0.25
D	2.90	3.00	3.10
D2	1.55	1.65	1.75
e	0.40BSC		
Ne	1.60BSC		
Nd	1.60BSC		
E	2.90	3.00	3.10
E2	1.55	1.65	1.75
L	0.35	0.40	0.45
h	0.20	0.25	0.30

封装形式： **SOP16**



序号	最小值(mm)	标准值(mm)	最大值(mm)
A	1.500	1.600	1.700
A2	1.400	1.450	1.500
b	0.356	0.406	0.456
D1	9.70	9.90	10.10
D2	9.75	9.95	10.15
E	5.90	6.000	6.100
E1	3.800	3.900	4.000
E2	3.850	3.950	4.050
e	---	1.27	---
Z	---	0.505	---

25 附录

附录 1 指令集速查表

指令	描述	说明	周期
数据传送指令			
MOV A,Rn	寄存器内容送入累加器	(A) ← (Rn)	1
MOV A,direct	直接地址单元中的数据送入累加器	(A) ← (direct)	1
MOV A,@Ri	间接 RAM 中的数据送入累加器	(A) ← ((Ri))	1
MOV A,#data8	8 位立即数送入累加器	(A) ← #data	1
MOV Rn,A	累加器内容送入寄存器	(Rn) ← (A)	1
MOV Rn,direct	直接地址单元中的数据送入寄存器	(Rn) ← (direct)	2
MOV Rn,#data8	8 位立即数送入寄存器	(Rn) ← #data	1
MOV direct,A	累加器内容送入直接地址单元	(direct) ← (A)	1
MOV direct,Rn	寄存器内容送入直接地址单元	(direct) ← (Rn)	2
MOV direct,direct	直接地址单元中的数据送入直接地址单元	(direct) ← (direct)	2
MOV direct,@Ri	间接 RAM 中的数据送入直接地址单元	(direct) ← ((Ri))	2
MOV direct,#data8	8 位立即数送入直接地址单元	(direct) ← #data	2
MOV @Ri,A	累加器内容送入间接 RAM 单元	((Ri)) ← (A)	1
MOV @Ri,direct	直接地址单元中的数据送入间接 RAM 单元	((Ri)) ← (direct)	2
MOV @Ri,#data8	8 位立即数送入间接 RAM 单元	((Ri)) ← #data	1
MOV DPTR,#data16	16 位立即数地址送入地址寄存器	(DPTR) ← #data116	2
MOV A,@A+DPTR	以 DPTR 为基地址变址寻址单元中的数据送入累加器	(A) ← ((A)) + (DPTR)	2
MOV A,@A+PC	以 PC 为基地址变址寻址单元中的数据送入累加器	(PC) ← (PC) + 1 (A) ← ((A) + (PC))	2
MOVX A,@Ri	外部 RAM(8 位地址)送入累加器	(A) ← ((Ri))	2
MOVX A,@DPTR	外部 RAM(16 位地址)送入累加器	(A) ← ((DPTR))	2
MOVX @Ri,A	累加器送入外部 RAM(8 位地址)	((Ri)) ← (A)	2
MOVX @DPTR,A	累加器送入外部 RAM(16 位地址)	(DPTR) ← (A)	2
PUSH direct	直接地址单元中的数据压入堆栈	(SP) ← (SP) + 1 ((SP)) ← (direct)	2
POP DIRECT	堆栈中的数据弹出到直接地址单元	(direct) ← ((SP)) (SP) ← (SP) - 1	2
XCH A,Rn	寄存器与累加器交换	(A) ↔ (Rn)	1
XCH A,direct	直接地址单元与累加器交换	(A) ↔ (direct)	1
XCH A,@Ri	间接 RAM 与累加器交换	(A) ↔ ((Ri))	1

XCHD A,@Ri	间接 RAM 与累加器进行低半字节交换	(A.3,...,A.0) ↔ ((Ri).3,...,(Ri).0)	1
SWAP A	累加器半字节交换	(A.3,...,A.0) ↔ (A.7,...,A.4)	1
算术操作类指令			
ADD A, Rn	寄存器内容加到累加器	(A) ← (A) + (Rn)	1
ADD A, direct	直接地址单元加到累加器	(A) ← (A) + (direct)	1
ADD A, @Ri	间接 RAM 内容加到累加器	(A) ← (A) + ((Ri))	1
ADD A, #data8	8 位立即数加到累加器	(A) ← (A) + #data	1
ADDC A, Rn	寄存器内容带进位加到累加器	(A) ← (A) + (C) + (Rn)	1
ADDC A, direct	直接地址单元带进位加到累加器	(A) ← (A) + (C) + (direct)	1
ADDC A, @Ri	间接 RAM 内容带进位加到累加器	(A) ← (A) + (C) + ((Ri))	1
ADDC A, #data8	8 位立即数带进位加到累加器	(A) ← (A) + (C) + #data	1
SUBB A, Rn	累加器带借位减寄存器内容	(A) ← (A) - (C) - (Rn)	1
SUBB A, direct	累加器带借位减直接地址单元	(A) ← (A) - (C) - (direct)	1
SUBB A, @Ri	累加器带借位减间接 RAM 内容	(A) ← (A) - (C) - ((Ri))	1
SUBB A, #data8	累加器带借位减 8 位立即数	(A) ← (A) - (C) - #data	1
INC A	累加器加 1	(A) ← (A) + 1	1
INC Rn	寄存器加 1	(Rn) ← (Rn) + 1	1
INC direct	直接地址单元内容加 1	(direct) ← (direct) + 1	1
INC @Ri	间接 RAM 内容加 1	((Ri)) ← ((Ri)) + 1	1
INC DPTR	DPTR 加 1	(DPTR) ← (DPTR) + 1	2
DEC A	累加器减 1	(A) ← (A) – 1	1
DEC Rn	寄存器减 1	(Rn) ← (Rn) - 1	1
DEC direct	直接地址单元内容减 1	(direct) ← (direct) - 1	1
DEC @Ri	间接 RAM 内容减 1	((Ri)) ← ((Ri)) - 1	1
MUL AB	A 乘以 B	temp16 ← (A) X (B) (A)←(temp.7,temp.6,...,temp.0) (B)←(temp.15,temp.14,...,temp.8)	4

DIV AB	A 除以 B	QUO ← (A) / (B) .....REM (A) ← QUO (B) ← REM	4

DA A	累加器进行十进制转换	IF (A.3,...,A.0) > 9 || AC = 1
		THEN
		temp16 ← (A) + 0x06
		(A) ← (temp.7,...,temp.0)
		
		IF (temp16) > 0xFF
		THEN
		CY ← 1
		
		IF (A.7,...,A.4) > 9 || CY = 1
		THEN
		temp16 ← (A) + 0x60
		(A) ← (temp.7,...,temp.0)
		
		IF (temp16) > 0xFF
		THEN
		CY ← 1
			1

逻辑操作类指令			
ANL A, Rn	累加器与寄存器相“与”	(A) ← (A) & (Rn)	1
ANL A, direct	累加器与直接地址单元相“与”	(A) ← (A) & (direct)	1
ANL A, @Ri	累加器与间接 RAM 内容相“与”	(A) ← (A) & ((Ri))	1
ANL A, #data8	累加器与 8 位立即数相“与”	(A) ← (A) & #data	1
ANL direct, A	直接地址单元与累加器相“与	(direct) ← (direct) & (A)	1
ANL direct, #data8	直接地址单元与 8 位立即数相“与”	(direct) ← (direct) & #data	2
ORL A, Rn	累加器与寄存器相“或”	(A) ← (A) | (Rn)	1
ORL A, direct	累加器与直接地址单元相“或”	(A) ← (A) | (direct)	1
ORL A, @Ri	累加器与间接 RAM 内容相“或”	(A) ← (A) | ((Ri))	1
ORL A, #data8	累加器与 8 位立即数相“或”	(A) ← (A) | #data	1
ORL direct, A	直接地址单元与累加器相“或”	(direct) ← (direct) |	1

		(A)	
ORL direct, #data8	直接地址单元与 8 位立即数相“或”	(direct) ← (direct) | #data	2
XRL A, Rn	累加器与寄存器相“异或”	(A) ← (A) ^ (Rn)	1
XRL A, direct	累加器与直接地址单元相“异或”	(A) ← (A) ^ (direct)	1
XRL A, @Ri	累加器与间接 RAM 内容相“异或”	(A) ← (A) ^ ((Ri))	1
XRL A, #data8	累加器与 8 位立即数相“异或”	(A) ← (A) ^ #data	1
XRL direct, A	直接地址单元与累加器相“异或”	(direct) ← (direct) ^ (A)	1
XRL direct, #data8	直接地址单元与 8 位立即数相“异或”	(direct) ← (direct) ^ #data	2
CLR A	累加器清 0	(A) ← 0	1
CPL A	累加器求反	(A) ← /(A)	1
RL A	累加器循环左移	(A) ← (A.6,A.5,...,A.0,A.7)	1
RLC A	累加器带进位循环左移	C ← A.7 (A) ← (A.6,A.5,...,A.0,C)	1
RR A	累加器循环右移	(A) ← (A.0,A.7,...,A.2,A.1)	1
RRC A	累加器带进位循环右移	C ← A.0 (A) ← (C,A.7,...,A.2,A.1)	1
控制转移类指令			
ACALL addr11	绝对短调用子程序	(PC) ← (PC) + 2 (SP) ← (SP) + 1 ((SP)) ← (PC7-0) (SP) ← (SP) + 1 ((SP)) ← (PC15-8) (PC10-0) ← page address	2
LACLL addr16	长调用子程序	(PC) ← (PC) + 3 (SP) ← (SP) + 1 ((SP)) ← (PC7-0) ((SP)) ← (PC15-8) (PC) ←addr15-0	2
RET	子程序返回	(PC15-8) ← ((SP)) (SP) ← (SP) - 1 (PC7-0) ← ((SP)) (SP) ← (SP) - 1	2
RETI	中断返回	(PC15-8) ← ((SP)) (SP) ← (SP) - 1	2

		(PC7-0) ← ((SP)) (SP) ← (SP) - 1	
AJMP addr11	绝对短转移	(PC) ← (PC) + 2 (PC10-0) ← page address	2
LJMP addr16	长转移	(PC) ← (PC) + 3 (SP) ← (SP) + 1 ((SP)) ← (PC7-0) (SP) ← (SP) + 1 ((SP)) ← (PC15-8) (PC10-0) ←addr15-0	2
SJMP rel	相对转移	(PC) ← (PC) + 2 (PC) ← (PC) + rel	2
JMP @A+DPTR	相对于 DPTR 的间接转移	(PC) ← (A) + (DPTR)	2
JZ rel	累加器为零转移	(PC) ← (PC) + 2 IF (A) = 0 THEN (PC) ← (PC) + rel	2
JNZ rel	累加器非零转移	(PC) ← (PC) + 2 IF (A) <> 0 THEN (PC) ← (PC) + rel	2

CJNE A, direct, rel	累加器与直接地址单元比较，不等则转移	(PC) ← (PC) + 3
		IF (A) <> (direct)
		THEN
		(PC) ← (PC) + relative offset
		IF (A) < (direct)
		THEN
		(C) ← 1
		ELSE
		(C) ← 0
			2

CJNE A, #data8, rel	累加器与 8 位立即数比较，不等则转移	(PC) ← (PC) + 3
		IF (A) <> data
		THEN
		(PC) ← (PC) + relative offset
		IF (A) < data
		THEN
		(C) ← 1
		ELSE
		(C) ← 0
			2

CJNE Rn, #data8, rel	寄存器与 8 位立即数比较，不等则转移	(PC) ← (PC) + 3
		IF (Rn) <> data
		THEN
		(PC) ← (PC) +
		relative offset
		IF (Rn) < data
		THEN
		(C) ← 1
		ELSE
		(C) ← 0
			2

CJNE @Ri, #data8, rel	间接 RAM 单元，不等则转移	(PC) ← (PC) + 3
		IF ((Ri)) <> data
		THEN
		(PC) ← (PC) +
		relative offset
		IF ((Ri)) < data
		THEN
		(C) ← 1
		ELSE
		(C) ← 0
			2

DJNZ Rn, rel	寄存器减 1，非零转移	(PC) ← (PC) + 2 (Rn) ← (Rn) - 1 IF (Rn) <> 0 THEN (PC) ← (PC) + rel	2
DJNZ direct, rel	直接地址单元减 1，非零转移	(PC) ← (PC) + 2 (direct) ← (direct) - 1 IF (direct) <> 0 THEN (PC) ← (PC) + rel	2
NOP	空操作	(PC) ← (PC) + 1	1
布尔变量操作类指令			
CLR C	清进位位	(C) ← 0	1
CLR bit	清直接地址位	(bit) ← 0	1
SETB C	置进位位	(C) ← 1	1
SETB bit	置直接地址位	(bit) ← 1	1
CPL C	进位位求反	(C) ← /(C)	1
CPL bit	直接地址位求反	(bit) ← /(bit)	1
ANL C, bit	进位位和直接地址位相“与”	(C) ← (C) & (bit)	2
ANL C, /bit	进位位和直接地址位的反码相“与”	(C) ← (C) & /(bit)	2
ORL C, bit	进位位和直接地址位相“或”	(C) ← (C) | (bit)	2
ORL C, /bit	进位位和直接地址位的反码相“或”	(C) ← (C) | /(bit)	2
MOV C, bit	直接地址位送入进位位	(C) ← (bit)	1

MOV bit, C	进位位送入直接地址位	(bit) ← (C)	2
JC rel	进位位为 1 则转移(CY=O 不转移，=1 转移)	(PC) ← (PC) + 2 IF (C) = 1 THEN (PC) ← (PC) + rel	2
JNC rel	进位位为 0 则转移	(PC) ← (PC) + 2 IF (C) = 0 THEN (PC) ← (PC) + rel	2
JB bit, rel	直接地址位为 1 则转移	(PC) ← (PC) + 3 IF (bit) = 1 THEN (PC) ← (PC) + rel	2
JNB bit, rel	直接地址位为 0 则转移	(PC) ← (PC) + 3 IF (bit) = 0 THEN (PC) ← (PC) + rel	2
JBC bit, rel	直接地址位为 1 则转移，该位清零	(PC) ← (PC) + 3 IF (bit) = 1 THEN (bit) ← 0 (PC) ← (PC) + rel	2
伪指令			
ORG	设置程序起始地址		
END	标志源代码结束		
EQU	定义常数		
SET	定义整型数		
DATA	给数据地址定值		
BYTE	给字节类型符号定值		
WROD	给字类型符号定值		
BIT	给位地址取名		
ALTNAME	用自定义名取代保留字		
DB	给一块连续的存储区装载字节型数据		
DW	给一块连续的存储区装载字型数据		
DS	预留一个连续的存储区或装入指定字节		
INCLUDE	将一个源文件插入程序中		
TITLE	列表文件中加入标题行		
NOLIST	汇编时不产生列表文件		
NOCODE	条件汇编时，条件为假的不产生清单		