

C8051F336/7/8/9
混合信号 ISP FLASH 微控制器
数 据 手 册

潘 琢 金 译

Rev 0.2 2008.01

版权所有

版 权 声 明

本手册中文版版权归译者和新华龙电子有限公司所有。研究和开发人员可以自由使用本手册。任何单位和个人未经版权所有者授权不得在任何形式的出版物中摘抄本手册内容。

原文中比较明显的错误已经在译文中更正。译者将在本手册英文版更新后及时更新中文版内容。译文中一定存在不少错误和不准确之处，望各位同仁不吝赐教，以便在新版本中更正。

译者联系方式：

沈阳航空工业学院 计算机学院 潘琢金

电话：024-89723476，13066535936

Email: panzhuojin@sina.com 或 panzhj@syiae.edu.cn

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

模拟外设

- **10 位 ADC (仅 F336/8)**
 - 转换速率可达 200ksps
 - 可多达 20 个外部单端或差分输入
 - VREF 取自内部 VREF、外部引脚或 VDD
 - 内部或外部转换启动源
 - 内置温度传感器
- **10 位电流输出 DAC (仅 F336/338)**
- **比较器**
 - 可编程回差电压和响应时间
 - 可配置为中断或复位源

在片调试逻辑

- 片内调试电路提供全速、非侵入式的在系统调试（不需仿真器）
- 支持断点、单步、观察/修改存储器和寄存器
- 比使用仿真芯片、目标仿真头和仿真插座的仿真系统有更优越的性能
- 廉价而完整的开发套件

供电电压 2.7V - 3.6V

- 内建电源电压监视器

高速 8051 微控制器内核

- 流水线指令结构；70% 的指令的执行时间为一个或两个系统时钟周期
- 速度可达 25MIPS（时钟频率为 25MHz 时）
- 扩展的中断系统

温度范围：-40°C - +85°C

存储器

- 768 字节内部数据 RAM (256+512)
- 16KB FLASH；可在系统编程，扇区大小为 512 字节（有 512 字节被保留）

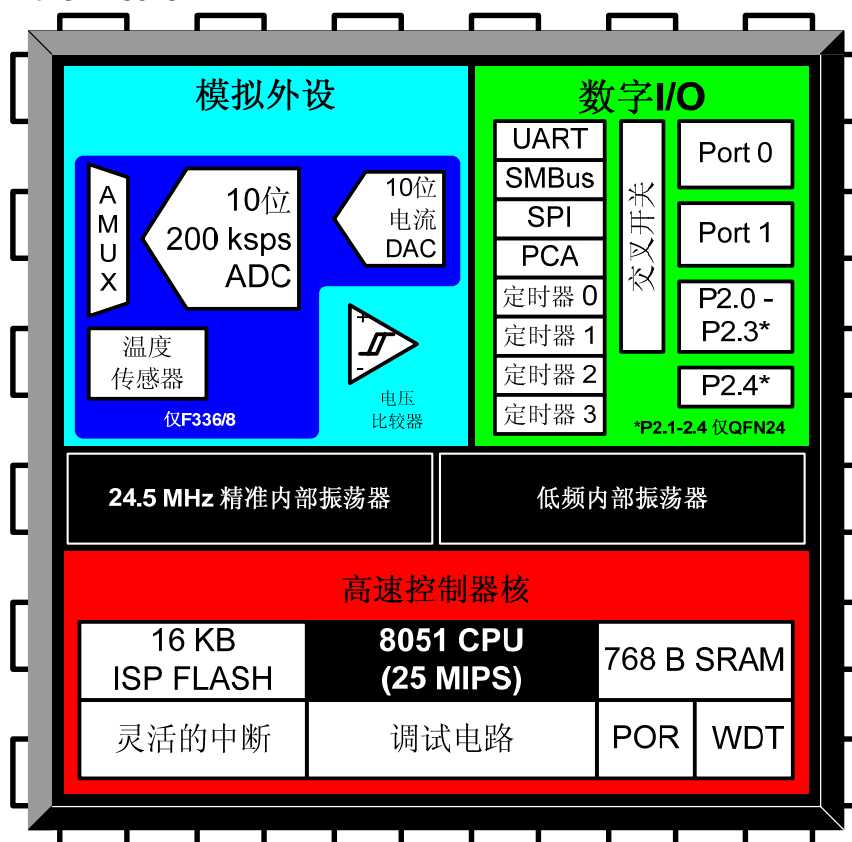
数字外设

- 21 或 17 个端口 I/O；均耐 5V 电压，大灌电流
- **引脚与 C8051F330 系列 MCU 兼容。**
- 硬件增强型 UART、SMBus (I²C 兼容) 和增强型 SPI 串口
- 4 个通用 16 位计数器/定时器
- 16 位可编程计数器/定时器阵列 (PCA)，有 3 个捕捉/比较模块和增强的 PWM 功能
- 使用定时器和晶体的实时时钟方式

时钟源

- **24.5MHz±2%振荡器：**
 - 支持无晶体 UART 操作
 - 具有快速唤醒时间的低功耗挂起方式
- 80/40/20/10 kHz 低频率、低功耗振荡器
- 外部振荡器：晶体、RC、C、或外部时钟（1 或 2 脚方式）
- 可在运行中切换时钟源，适用于节电方式

封装：20 或 24 脚 QFN (4x4 mm)



C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

1. 系统概述	7
1.1 CIP-51™ 微控制器核	9
1.1.1 与 8051 完全兼容	9
1.1.2 速度提高	9
1.1.3 增加的功能	9
1.2 片内存储器	11
1.3 片内调试电路	12
1.4 可编程数字 I/O 和交叉开关	12
1.5 串行端口	13
1.6 可编程计数器阵列	13
1.7 10 位模/数转换器	15
1.8 比较器	16
1.9 10 位电流输出 DAC	17
2. 定货信息	18
3. 引脚定义	19
4. QFN-20 封装规格	22
5. QFN-24 封装规格	23
6. 电气特性	24
6.1 极限参数	24
6.2 总体电气特性	25
7. 10 位 ADC (ADC0, 仅 C8051F336/8)	32
7.1 输出码格式	33
7.2 工作方式	33
7.2.1 启动转换	33
7.2.2 跟踪方式	34
7.2.3 建立时间要求	36
7.3 可编程窗口检测器	39
7.3.1 单端方式下的窗口检测器	40
7.3.2 差分方式下的窗口检测器	41
7.4 ADC0 模拟多路选择器 (仅 C8051F336/8)	42
8. 温度传感器 (仅 C8051F336/8)	45
9. 10 位电流模式 DAC (IDA0, 仅 C8051F336/8)	46

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

9.1 IDA0 输出更新	46
9.1.1 On-Demand 输出更新.....	46
9.1.2 基于定时器溢出的输出更新模式.....	47
9.1.3 基于CNVSTR 边沿的输出更新模式.....	47
9.2 IDAC 输出字格式.....	47
10. 电压基准（仅 C8051FF336/8）	49
11. 比较器 0.....	52
11.1 比较器多路器.....	56
12. CIP-51 微控制器	59
12.1 指令集.....	60
12.1.1 指令和CPU 时序.....	60
12.2 CIP-51 寄存器说明	64
13. 存储器组织	67
13.1 程序存储器	68
13.1.1 MOVX 指令和程序存储器.....	68
13.2 数据存储器	68
13.2.1 内部RAM	68
13.2.2 外部RAM	69
14. 特殊功能寄存器	71
15. 中断	75
15.1 MCU 中断源和中断向量	76
15.1.1 中断优先级.....	76
15.1.2 中断响应时间.....	76
15.2 中断寄存器说明.....	78
15.3 外部中断.....	82
16. FLASH 存储器	84
16.1 FLASH 存储器编程.....	84
16.1.1 FLASH 锁定和关键码功能.....	84
16.1.2 FLASH 擦除.....	84
16.1.3 FLASH 写.....	85
16.2 非易失性数据存储	85
16.3 安全选项	86
16.4 FLASH 写和擦除指南.....	88
16.4.1 VDD 维护和VDD 监视器.....	88
16.4.2 PSWE 维护.....	88

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

16.4.3 系统时钟.....	89
17. 电源管理方式	91
17.1 空闲方式.....	91
17.2 停机方式.....	93
17.3 挂起方式.....	93
18. 复位源	94
18.1 上电复位.....	95
18.2 掉电复位和 VDD 监视器	96
18.3 外部复位.....	97
18.4 时钟丢失检测器复位.....	97
18.5 比较器 0 复位.....	97
18.6 PCA 看门狗定时器复位.....	97
18.7 FLASH 错误复位.....	97
18.8 软件复位.....	98
19. 振荡器和时钟选择	99
19.1 系统时钟选择.....	100
19.1 可编程内部高频 (H-F) 振荡器.....	101
19.2.1 内部振荡器挂起方式.....	101
19.3 可编程内部低频 (L-F) 振荡器.....	103
19.3.1 内部 L-F 振荡器标定.....	103
19.4 外部振荡器驱动电路.....	104
19.4.1 外部晶体示例.....	106
19.4.2 外部 RC 示例.....	107
19.4.3 外部电容示例.....	107
20. 端口输入/输出	109
20.1 端口 I/O 的工作方式.....	110
20.1.1 端口引脚配置为模拟 I/O	110
20.1.2 端口引脚配置为数字 I/O	110
20.1.3 端口 I/O 与 5V 逻辑的接口.....	111
20.2 端口 I/O 引脚分配给模拟和数字功能.....	111
20.2.1 端口引脚分配给模拟功能.....	111
20.2.2 端口引脚分配给数字功能.....	112
20.2.3 端口引脚分配给外部数字事件捕捉功能.....	112
20.3 优先权交叉开关译码器.....	113
20.4 端口 I/O 初始化.....	116
20.5 端口匹配.....	119
20.6 访问和配置端口 I/O 的特殊功能寄存器.....	120

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

21. SMBUS	126
21.1 支持文档	127
21.2 SMBus 配置	127
21.3 SMBus 操作	128
21.3.1 发送器和接收器的概念	128
21.3.2 总线仲裁	129
21.3.3 时钟低电平扩展	129
21.3.4 SCL 低电平超时	129
21.3.5 SCL 高电平 (SMBus 空闲) 超时	129
21.4 SMBus 的使用	129
21.4.1 SMBus 配置寄存器	130
21.4.2 SMBus 控制寄存器	133
21.4.3 硬件从地址识别	136
21.4.4 数据寄存器	138
21.5 SMBus 传输方式	139
21.5.1 写序列 (主方式)	139
21.5.2 读序列 (主方式)	140
21.5.3 写序列 (从方式)	141
21.5.4 读序列 (从方式)	142
21.6 SMBus 状态译码	143
22. UART0	147
22.1 增强的波特率发生器	148
22.2 工作方式	149
22.2.1 8 位 UART	149
22.2.2 9 位 UART	150
22.3 多机通信	151
23. 增强型串行外设接口 (SPI0)	155
23.1 信号说明	156
23.1.1 主输出、从输入 (MOSI)	156
23.1.2 主输入、从输出 (MISO)	156
23.1.3 串行时钟 (SCK)	156
23.1.4 从选择 (NSS)	156
23.2 SPI0 主方式	157
23.3 SPI0 从方式	159
23.4 SPI0 中断源	159
23.5 串行时钟相位和极性	160
23.6 SPI 特殊功能寄存器	162
24. 定时器	168

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

24.1 定时器 0 和定时器 1.....	170
24.1.1 方式 0 — 13 位计数器/定时器.....	170
24.1.2 方式 1 — 16 位计数器/定时器.....	171
24.1.3 方式 2 — 自动重载的 8 位计数器/定时器.....	172
24.1.4 方式 3 — 两个 8 位计数器/定时器（仅定时器 0）.....	173
24.2 定时器 2.....	177
24.2.1 16 位自动重载定时器.....	177
24.2.2 8 位自动重载定时器.....	178
24.2.3 低频振荡器（LFO）捕捉方式.....	179
24.3 定时器 3.....	182
24.3.1 16 位自动重载定时器.....	182
24.3.2 8 位自动重载定时器.....	183
24.3.3 低频振荡器（LFO）捕捉方式.....	184
25. 可编程计数器阵列	187
25.1 PCA 计数器/定时器	188
25.2 PCA0 中断源	190
25.3 捕捉/比较模块.....	191
25.3.1 边沿触发的捕捉方式.....	192
25.3.2 软件定时器（比较）方式.....	193
25.3.3 高速输出方式.....	194
25.3.4 频率输出方式.....	195
25.3.5 8 位、9 位、10 位和 11 位脉宽调制器方式.....	196
25.3.6 16 位脉宽调制器方式.....	198
25.4 看门狗定时器方式	199
25.4.1 看门狗定时器操作.....	199
25.4.2 看门狗定时器的使用.....	200
25.5 PCA0 寄存器说明.....	201
26. C2 接口.....	207
26.1 C2 接口寄存器.....	207
26.2 C2 引脚共享.....	209

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

1. 系统概述

C8051F336/7/8/9 器件是完全集成的混合信号片上系统型 MCU。下面列出了一些主要特性，有关某一产品的具体特性参见表 2.1。

- 高速、流水线结构的 8051 兼容的 CIP-51 内核（可达 25MIPS）
- 全速、非侵入式的在系统调试接口（片内）
- 真正 10 位 200 ksps 的 20 通道单端/差分 ADC，带模拟多路器
- 10 位电流输出 DAC
- 高精度可编程的 24.5MHz 内部振荡器
- 16KB 的 FLASH 存储器—512 字节被保留
- 768 字节片内 RAM
- 硬件实现的 SMBus/ I²C、增强型 UART 和增强型 SPI 串行接口
- 4 个通用的 16 位定时器
- 具有 3 个捕捉/比较模块和看门狗定时器功能的可编程计数器/定时器阵列（PCA）
- 片内上电复位、VDD 监视器和温度传感器
- 片内电压比较器
- 21 或 17 个端口 I/O（容许 5V 输入）

具有片内上电复位、VDD 监视器、看门狗定时器和时钟振荡器的 C8051F336/7/8/9 是真正能独立工作的片上系统。FLASH 存储器还具有在系统重新编程能力，可用于非易失性数据存储，并允许现场更新 8051 固件。用户软件对所有外设具有完全的控制，可以关断任何一个或所有外设以节省功耗。

片内 Silicon Labs 二线（C2）开发接口允许使用安装在最终应用系统上的产品 MCU 进行非侵入式（不占用片内资源）、全速、在系统调试。调试逻辑支持观察和修改存储器和寄存器，支持断点、单步、运行和停机命令。在使用 C2 进行调试时，所有的模拟和数字外设都可全功能运行。两个 C2 接口引脚可以与用户功能共享，使在系统调试功能不占用封装引脚。

每种器件都可在工业温度范围（-45℃到+85℃）内用 2.7V-3.6V 的电压工作。端口 I/O 和 /RST 引脚都容许 5V 的输入信号电压。C8051F336/7 采用 20 脚 QFN 封装，C8051F338/9 采用 24 脚 QFN 封装。图 1.1 和图 1.2 给出了这些器件的原理框图。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

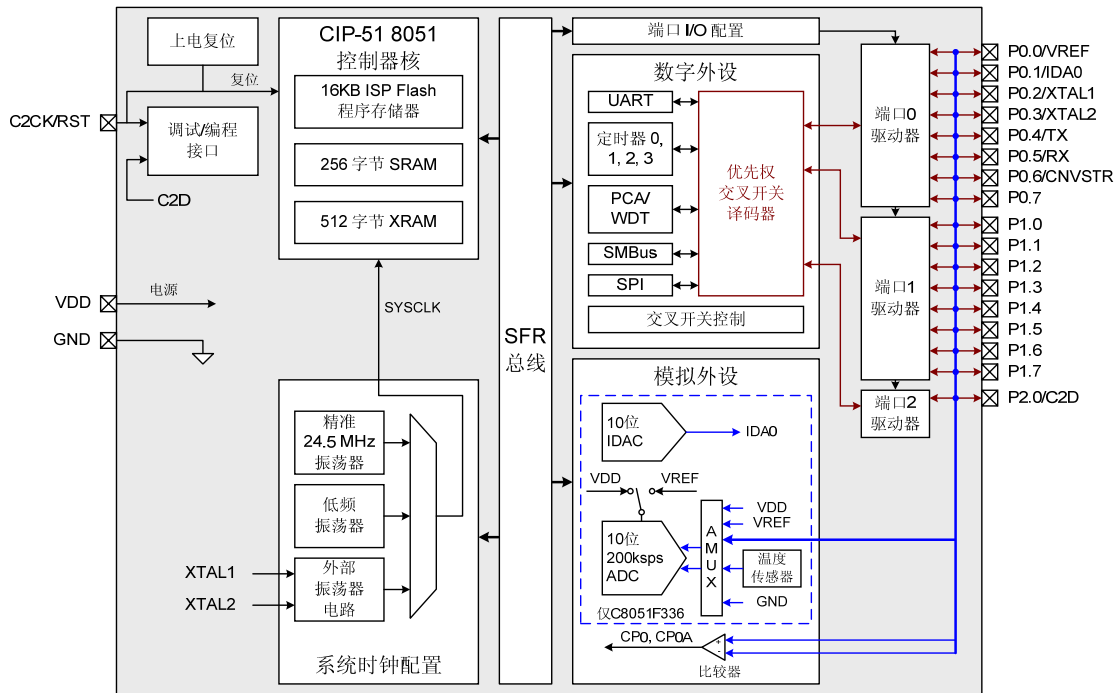


图 1.1 C8051F336/7 原理框图

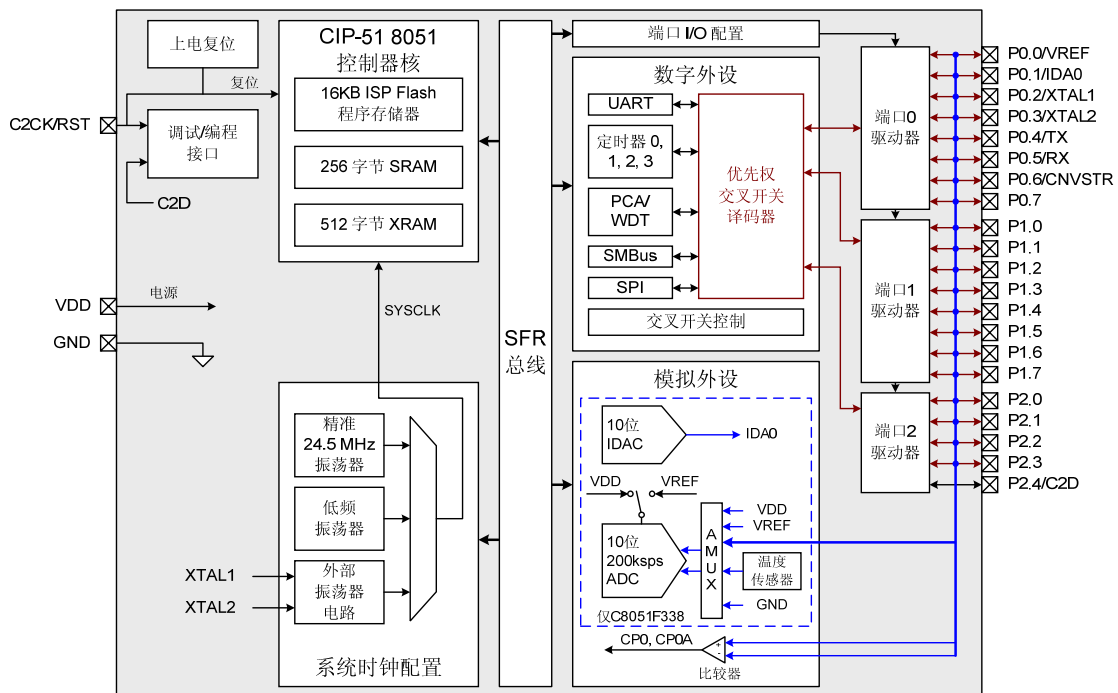


图 1.2 C8051F338/9 原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

1.1 CIP-51™ 微控制器核

1.1.1 与 8051 完全兼容

C8051F336/7/8/9 器件使用 Silicon Labs 的专利 CIP-51 微控制器内核。CIP-51 与 MCS-51™ 指令集完全兼容，可以使用标准 803x/805x 的汇编器和编译器进行软件开发。CIP-51 内核具有标准 8052 的所有外设部件，包括 4 个 16 位计数器/定时器、一个具有增强波特率配置的全双工 UART、一个增强型 SPI 端口、768 字节内部 RAM、128 字节特殊功能寄存器（SFR）地址空间及 21 或 17 个 I/O 端口。

1.1.2 速度提高

CIP-51 采用流水线结构，与标准的 8051 结构相比指令执行速度有很大的提高。在一个标准的 8051 中，除 MUL 和 DIV 以外所有指令都需要 12 或 24 个系统时钟周期，最大系统时钟频率为 12-24MHz。而对于 CIP-51 内核，70% 的指令的执行时间为 1 或 2 个系统时钟周期，只有 4 条指令的执行时间大于 4 个系统时钟周期。

CIP-51 共有 111 条指令。下表列出了各种指令执行时间（指令执行时所需的系统时钟周期数）所对应的指令条数。

执行周期数	1	2	2/3	3	3/4	4	4/5	5	8
指令数	26	50	5	16	7	3	1	2	1

CIP-51 工作在最大系统时钟频率 25MHz 时，它的峰值速度达到 25MIPS。

1.1.3 增加的功能

C8051F336/7/8/9 系列 MCU 在 CIP-51 内核和外设方面有几项关键性的改进，提高了整体性能，更易于在最终应用中使用。

扩展的中断系统向 CIP-51 提供 14 个中断源（标准 8051 只有 7 个中断源），允许大量的模拟和数字外设中断微控制器。一个中断驱动的系统需要较少的 MCU 干预，因而有更高的执行效率。在设计一个多任务实时系统时，这些增加的中断源是非常有用的。

MCU 有 8 个复位源：上电复位电路（POR）、一个片内 VDD 监视器（当电源电压低于 V_{RST} 时强制复位）、一个看门狗定时器、一个时钟丢失检测器、一个由比较器 0 提供的电压检测器、一个软件强制复位、外部复位输入引脚和 FLASH 读/写错误保护电路复位。除了 POR、复位输入引脚及 FLASH 操作错误这三个复位源之外，其他复位源都可以被软件禁止。在一次上电复位之后的 MCU 初始化期间，WDT 可以被永久性使能。

内部振荡器在出厂时已经被校准为 24.5MHz，在整个温度范围和电源电压范围内精度 $\pm 2\%$ 。内部振荡器的周期可以由用户用固件调整。片内还集成了一个低频振荡器，更适合于低功耗操作。器件内集成了外部振荡器驱动电路，允许使用外部晶体、陶瓷谐振器、电容、RC 或 CMOS 时钟源产生系统时钟。如果需要，时钟源可以在运行时切换到外部振荡器。外部振荡器在低功耗系统中是非常有用的，它允许 MCU 从一个低频率（节电）时钟源运行，当需要时再周期性地切换到高速（可达 25MHz）的内部振荡器。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

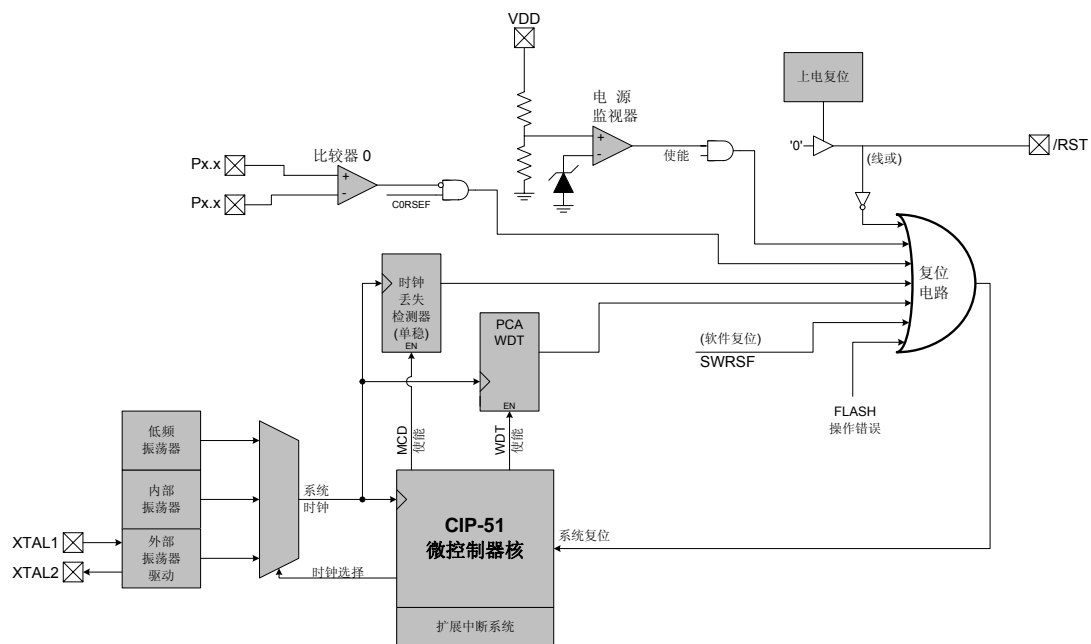


图 1.3 片内时钟和复位电路

1.2 片内存储器

CIP-51 有标准 8051 的程序和数据地址配置。它包括 256 字节的数据 RAM，其中高 128 字节为双映射。用间接寻址访问通用 RAM 的高 128 字节，用直接寻址访问 128 字节的 SFR 地址空间。数据 RAM 的低 128 字节可用直接或间接寻址方式访问。前 32 个字节为 4 个通用寄存器区，接下来的 16 字节既可以按字节寻址也可以按位寻址。

程序存储器包含 16KB 的 FLASH。该存储器以 512 字节为一个扇区，可在系统重编程，且不需特别的片外编程电压。图 1.4 给出了 MCU 系统的存储器结构。

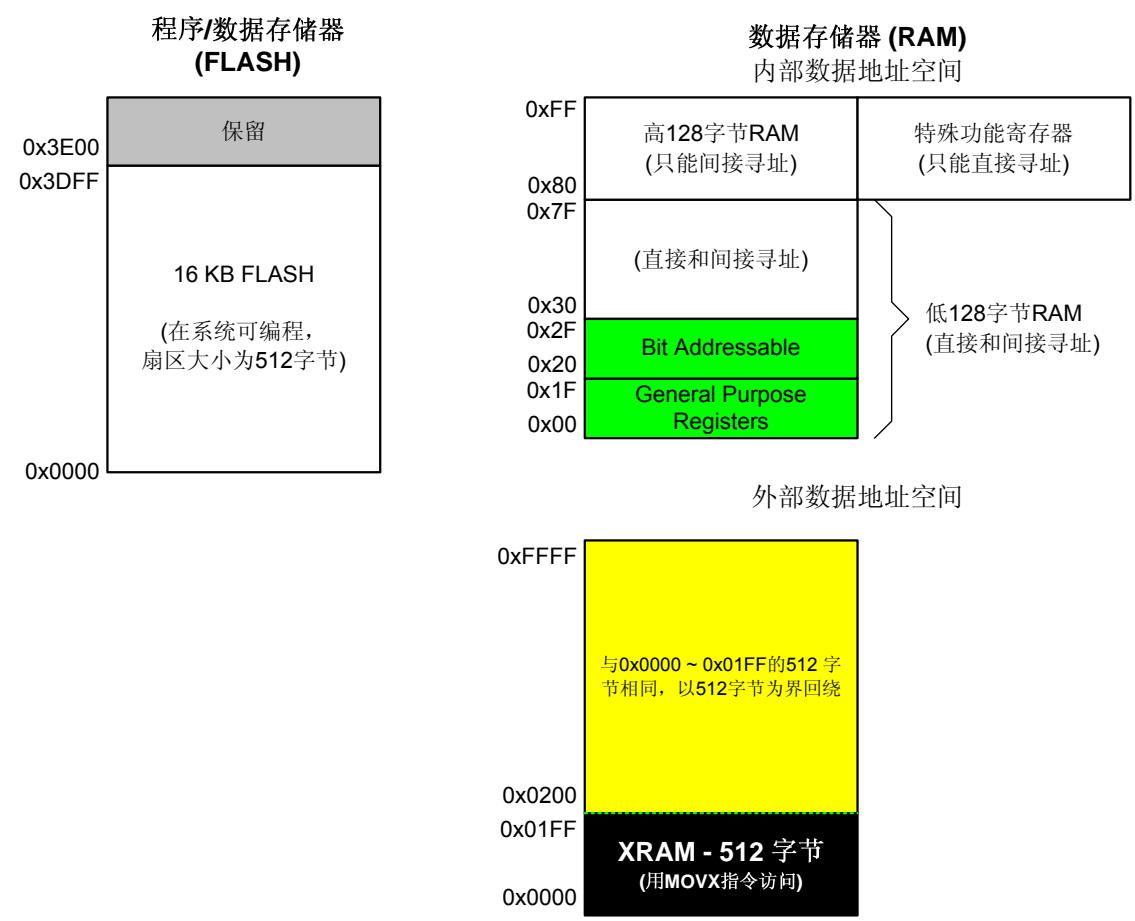


图 1.4 片内存储器组织

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

1.3 片内调试电路

C8051F336/7/8/9器件具有片内Silicon Labs 2线（C2）接口调试电路，支持使用安装在最终应用系统中的产品器件进行非侵入式、全速的在系统调试。

Silicon Labs的调试系统支持观察和修改存储器和寄存器，支持断点和单步执行。不需要额外的目标RAM、程序存储器、定时器或通信通道。在调试时所有的模拟和数字外设都正常工作。当MCU单步执行或遇到断点而停止运行时，所有的外设（ADC和SMBus除外）都停止运行，以保持与指令执行同步。

开发套件C8051F336DK具有开发应用代码和对C8051F33x MCU进行在系统调试所需要的全部硬件和软件。开发套件中包括开发者工作室软件和调试器、一个集成的8051汇编器和一个调试适配器。套件中还有一个目标应用板，上面有对应的MCU和一大块样机区域。套件中还包括所必需的电缆及墙装电源。IDE软件需要一个运行Windows操作系统的PC机。

对于开发和调试来说，Silicon Labs IDE接口比采用标准MCU仿真器要优越得多。标准的MCU仿真器要使用在板仿真芯片和目标电缆，还需要在应用板上有MCU的插座。Silicon Labs的调试环境既便于使用又能保持高精度模拟外设的性能。

1.4 可编程数字 I/O 和交叉开关

C8051F338/9器件有21个I/O引脚（两个8位口和一个5位口）。C8051F336/7器件有17个I/O引脚（两个8位口和一个1位口）。C8051F336/7/8/9端口的工作情况与标准8051相似，但有一些改进。每个端口引脚都可以被配置为模拟输入或数字I/O。被选择作为数字I/O的引脚还可以被配置为推挽或漏极开路输出。在标准8051中固定的“弱上拉”可以被总体禁止，这为低功耗应用提供了进一步节电的能力。

数字交叉开关允许将内部数字系统资源映射到端口I/O引脚（见图1.5）。可通过设置交叉开关控制寄存器将片内的计数器/定时器、串行总线、硬件中断、比较器输出以及微控制器内部的其它数字信号配置为出现在端口I/O引脚。这一特性允许用户根据自己的特定应用选择通用端口I/O和所需数字资源的组合。

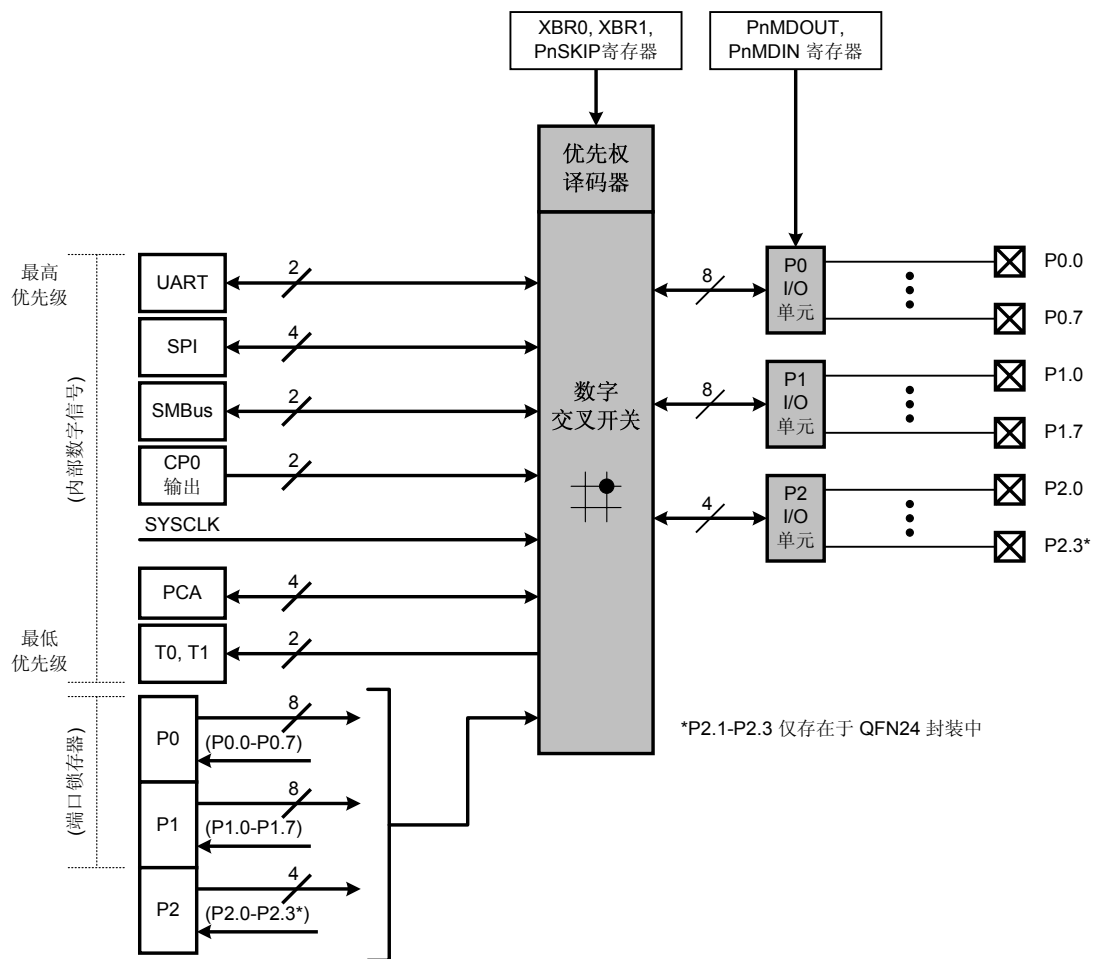


图1.5 数字交叉开关原理框图

1.5 串行端口

C8051F336/7/8/9系列MCU内部有一个SMBus/I²C接口、一个具有增强型波特率配置的全双工UART和一个增强型SPI接口。每种串行总线都完全用硬件实现，都能向CIP-51产生中断，因此需要很少的CPU干预。

1.6 可编程计数器阵列

除了4个16位的通用计数器/定时器之外，MCU中还有一个片内可编程计数器/定时器阵列（PCA）。PCA包括一个专用的16位计数器/定时器时间基准和3个可编程的捕捉/比较模块。PCA的时钟可以是下面的六个时钟源之一：系统时钟/12、系统时钟/4、定时器0溢出、外部时钟输入（ECI）、系统时钟和外部振荡源频率/8。外部时钟源对于实时时钟功能是非常有用的，可以在使用内部振荡器驱动系统时钟的同时由外部振荡器给PCA提供时钟。

每个捕捉/比较模块都有多种工作方式：边沿触发捕捉、软件定时器、高速输出、脉冲宽度调制器（8、9、10、11或16位）、频率输出。此外，捕捉/比较模块2还提供看门狗定时器（WDT）

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

功能。在系统复位后，捕捉/比较模块2被配置并被使能为WDT方式。PCA捕捉/比较模块的I/O和外部时钟输入可以通过数字交叉开关连到端口I/O。

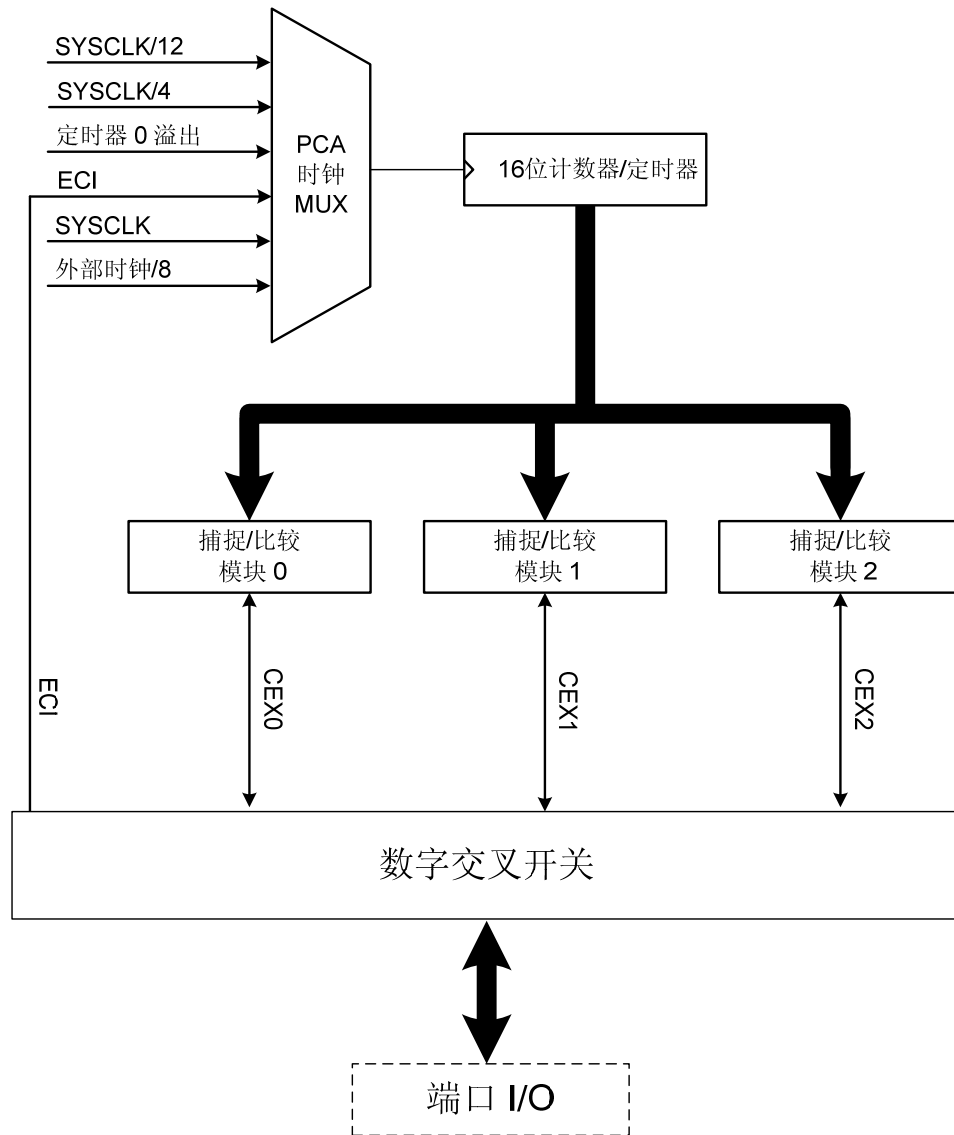


图1.7 PCA原理框图

1.7 10 位模/数转换器

C8051F336/8内部有一个10位SAR ADC和一个差分输入多路选择器。该ADC工作在200ksps的最大采样速率时可提供真正10位的线性度，INL和DNL为 $\pm 1\text{LSB}$ 。ADC系统包含一个可编程的模拟多路选择器，用于选择ADC的正输入和负输入。多达20个端口I/O引脚可用作ADC的输入；另外，片内温度传感器的输出和电源电压（VDD）也可以作为ADC的输入。用户固件可以将ADC置于关断状态以节省功耗。

A/D转换可以有6种启动方式：软件命令、定时器0溢出、定时器1溢出、定时器2溢出、定时器3溢出和外部转换启动信号。这种灵活性允许用软件事件、外部硬件信号或周期性信号（定时器溢出）触发转换。转换结束可以用一个状态位或中断（如果被允许）来指示。转换结束后10位结果数据字被锁存到ADC数据寄存器中。

窗口比较寄存器可被配置为当ADC数据位于一个规定的范围之内或之外时向控制器申请中断。ADC可以用后台方式监视一个关键电压，当转换数据位于规定的范围之内/外时才向控制器申请中断。

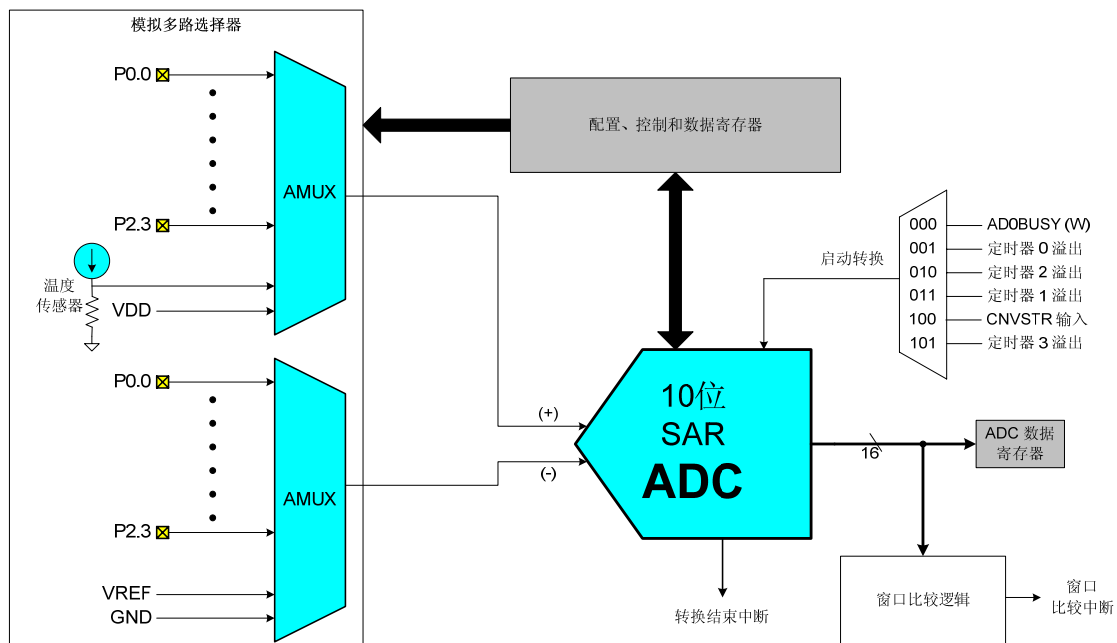


图1.8 10位ADC原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

1.8 比较器

C8051F336/7/8/9器件内部有一个电压比较器，可以由用户软件使能/禁止和配置。端口I/O引脚可以通过多路选择器被配置为比较器输入。如果需要，可以将两个比较器输出连到端口引脚：一个锁存输出和/或一个未锁存的输出（异步）。比较器的响应时间是可编程的，允许用户在高速和低功耗方式之间选择。比较器的正向和负向回差电压也是可配置的。

比较器能在上升沿、下降沿产生中断，或在两个边沿都产生中断。当MCU工作在空闲方式时，这些中断可用作“唤醒”源。比较器0还可以被配置为复位源。图1.9给出了比较器0的原理框图。

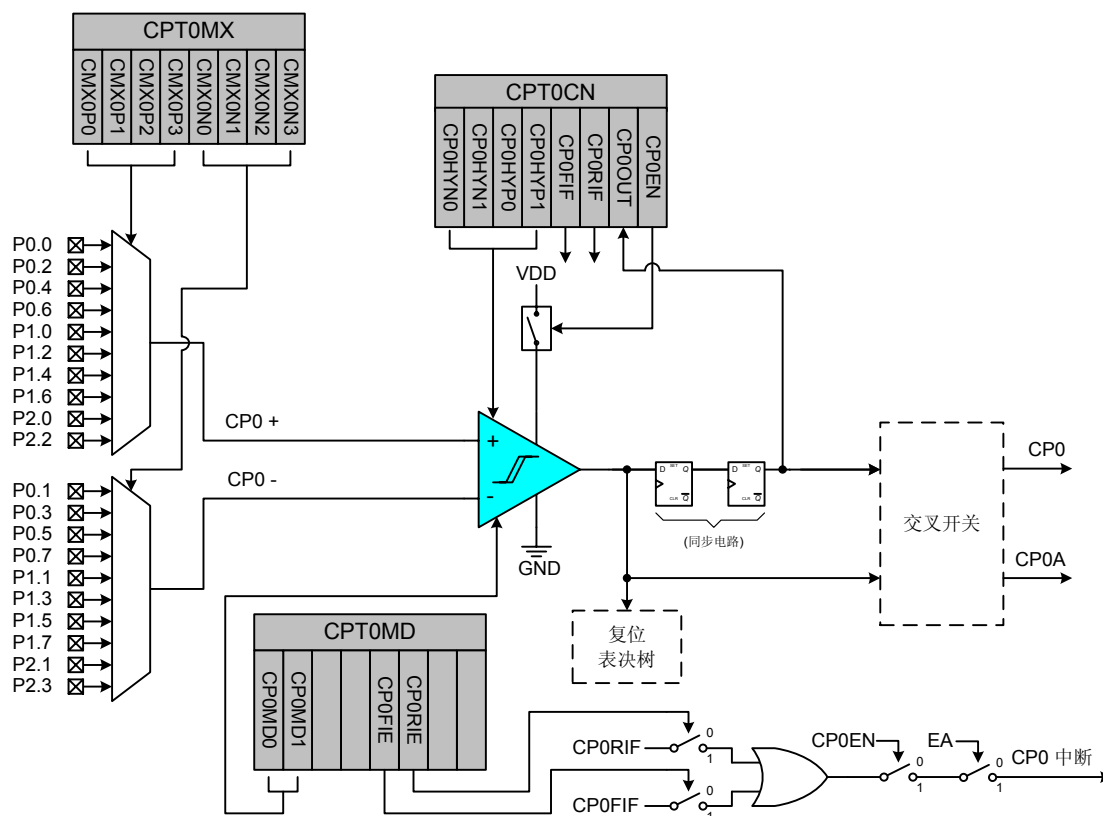


图1.9 比较器0原理框图

1.9 10 位电流输出 DAC

C8051F336/8内部有一个10位电流方式数/模转换器（IDA0）。IDA0的最大输出电流可以有三种不同的设置：0.5mA、1mA和2mA。IDA0具有灵活的输出更新机制，允许无缝满度变化，支持无抖动波形更新。IDA0有三种更新方式：写IDA0H、定时器溢出或外部引脚边沿。

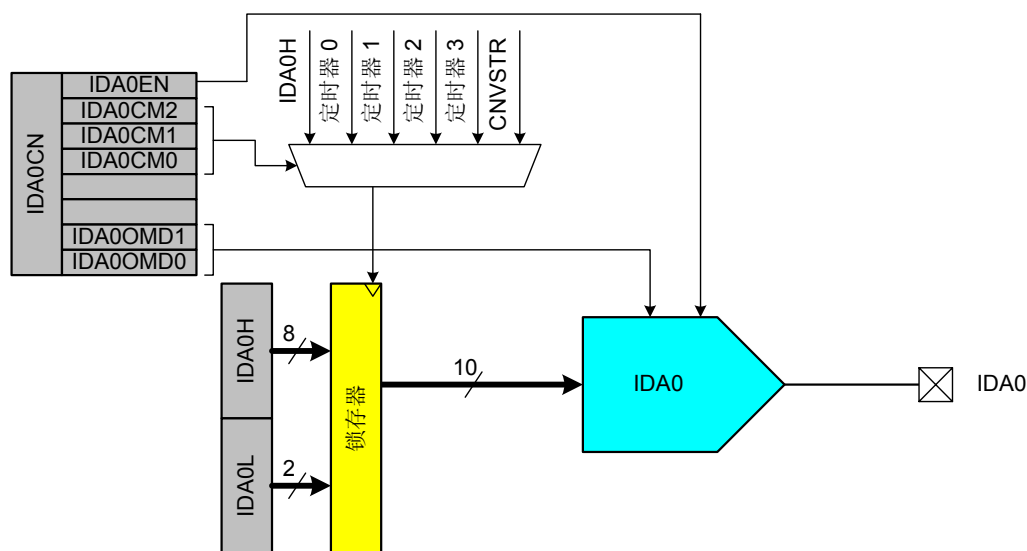


图1.10 IDA0原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

2. 定货信息

表 2.1 产品选择指南

型号	MIPS (峰值)	FLASH 存储器 (KB)	RAM (字节)	校准的内部 24.5MHz 振荡器	内部 80KHz 振荡器	SMBus/I ² C	增强型 SPI	UART	定时器(16 位)	可编程计数器阵列	数字端口 I/O	10 位 200kps ADC	10 位电流输出 DAC	内部电压基准	温度传感器	模拟比较器	封装
C8051F336-GM	25	16	768	√	√	√	√	√	4	√	17	√	√	√	√	√	QFN-20
C8051F337-GM	25	16	768	√	√	√	√	√	4	√	17	—	—	—	—	√	QFN-20
C8051F338-GM	25	16	768	√	√	√	√	√	4	√	21	√	√	√	√	√	QFN-24
C8051F339-GM	25	16	768	√	√	√	√	√	4	√	21	—	—	—	—	√	QFN-24

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

3. 引脚定义

表 3.1 C8051F336/7/8/9 引脚定义

引脚名称	引脚号 F336/7	引脚号 F338/9	引脚类型	说 明
VDD	3	4		电源
GND	2	3		地。注：中央焊盘也可以选择连接到地。
/RST	4	5	数字 I/O	器件复位。内部上电复位或 VDD 监视器的漏极开路输出。一个外部源可以通过将该引脚驱动为低电平（至少 10μs）来启动一次系统复位。
C2CK			数字 I/O	C2 调试接口的时钟信号。
P2.0	5		数字 I/O	端口 P2.0
C2D			数字 I/O	C2 调试接口的双向数据信号。
P2.4		6	数字 I/O	端口 P2.4
C2D			数字 I/O	C2 调试接口的双向数据信号。
P0.0	1	2	数字 I/O 或模拟输入	端口 P0.0
VREF			模拟输入	外部 VREF 输入
P0.1	20	1	数字 I/O 或模拟输入	端口 P0.1
IDA0			模拟输出	IDA0 输出
P0.2	19	24	数字 I/O 或模拟输入	端口 P0.2
XTAL1			模拟输入	外部时钟输入。对于晶体或陶瓷谐振器，该引脚是外部振荡器电路的反馈输入。
P0.3	18	23	数字 I/O 或模拟输入	端口 P0.3
XTAL2			模拟 I/O 或数字输入	外部时钟输出。该引脚是晶体或陶瓷谐振器的激励驱动器。对于 CMOS 时钟、电容或 RC 振荡器配置，该引脚是外部时钟输入。
P0.4	17	22	数字 I/O 或模拟输入	端口 P0.4
P0.5	16	21	数字 I/O 或模拟输入	端口 P0.5
P0.6	15	20	数字 I/O 或模拟输入	端口 P0.6
CNVSTR			数字输入	ADC0 外部转换启动输入或 IDA0 更新源输入。
P0.7	14	19	数字 I/O 或模拟输入	端口 P0.7
P1.0	13	18	数字 I/O 或模拟输入	端口 P1.0
P1.1	12	17	数字 I/O 或模拟输入	端口 P1.1
P1.2	11	16	数字 I/O 或模拟输入	端口 P1.2
P1.3	10	15	数字 I/O 或模拟输入	端口 P1.3
P1.4	9	14	数字 I/O 或模拟输入	端口 P1.4
P1.5	8	13	数字 I/O 或模拟输入	端口 P1.5
P1.6	7	12	数字 I/O 或模拟输入	端口 P1.6
P1.7	6	11	数字 I/O 或模拟输入	端口 P1.7
P2.0	5	10	数字 I/O 或模拟输入	端口 P2.0
P2.1		9	数字 I/O 或模拟输入	端口 P2.1
P2.2		8	数字 I/O 或模拟输入	端口 P2.2
P2.3		7	数字 I/O 或模拟输入	端口 P2.3

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

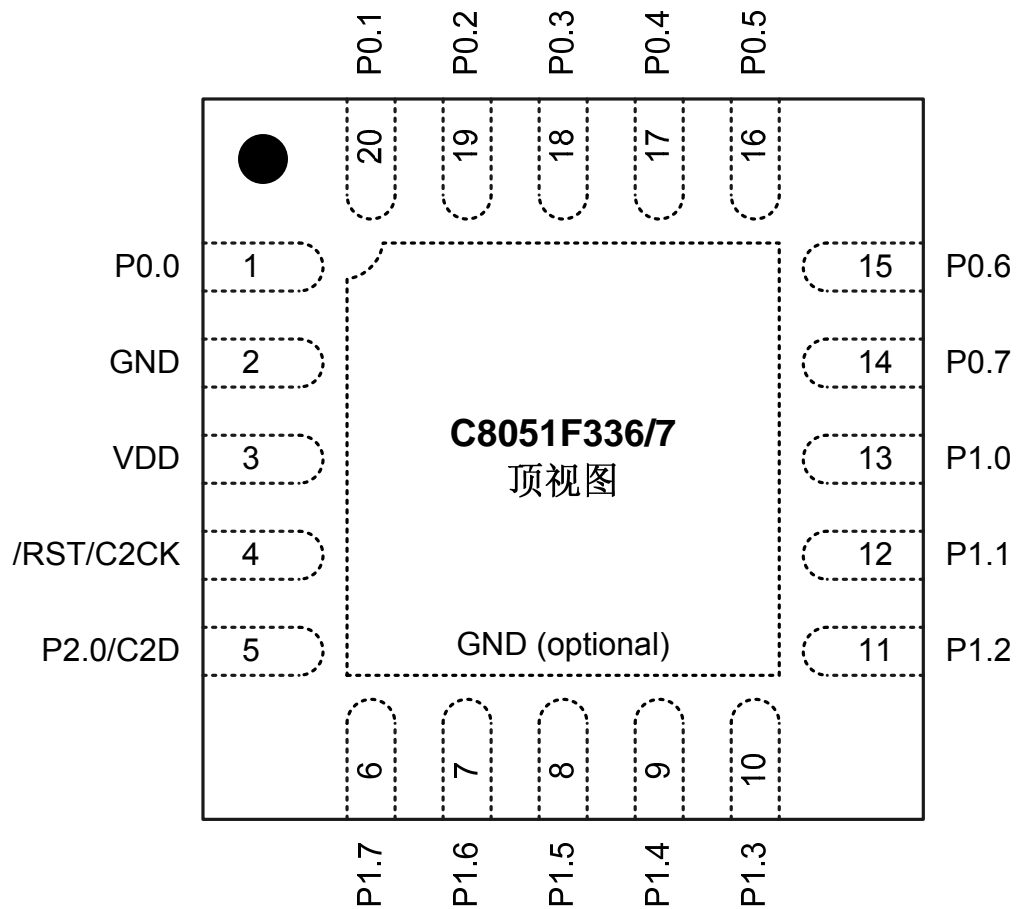


图 3.1 QFN-20 引脚图（顶视图）

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

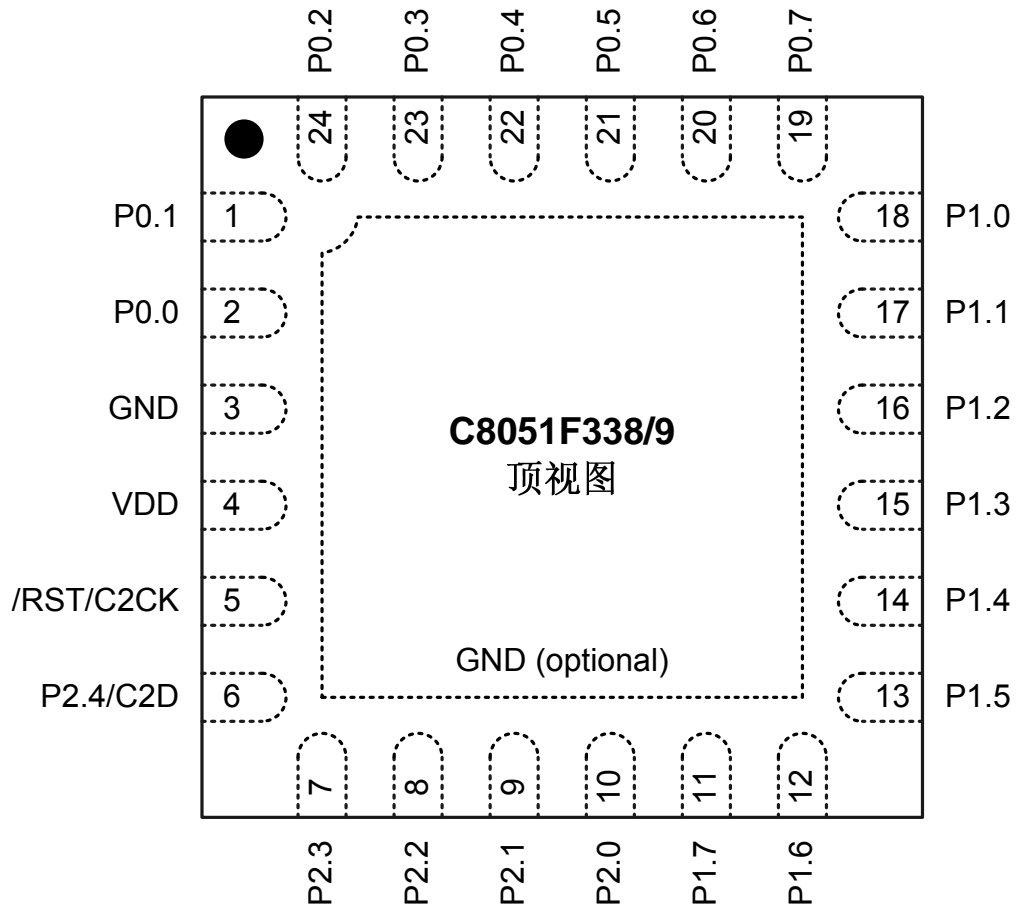


图 3.2 QFN-24 引脚图（顶视图）

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

4. QFN-20 封装规格

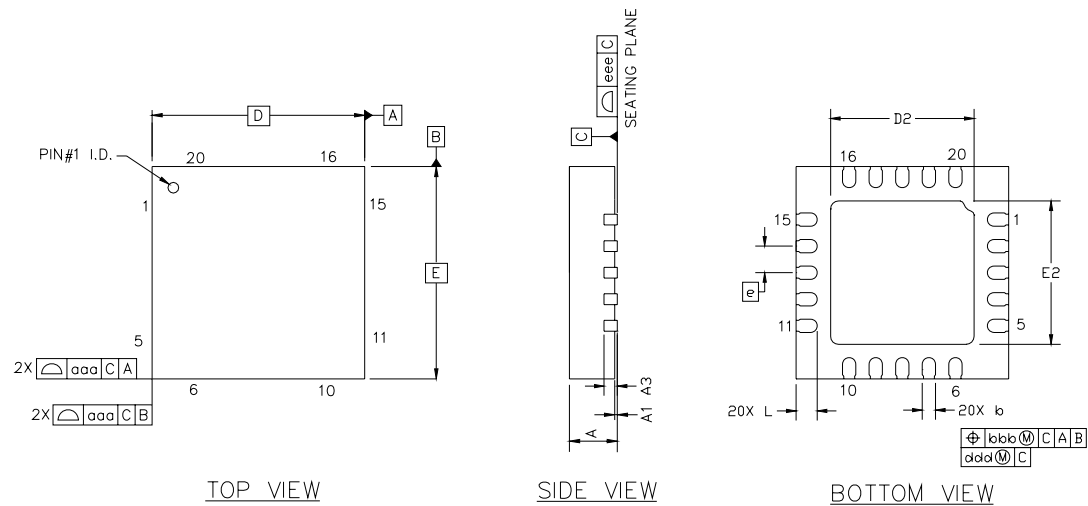


图 4.1 QFN-20 封装图

表 4.1 QFN-20 封装尺寸

尺寸	最小值	典型值	最大值
A	0.80	0.90	1.00
A1	0.00	0.02	0.05
B	0.18	0.25	0.30
D	4.0 BSC		
D2	2.05	2.15	2.25
e	0.50 BSC		
E	4.0 BSC		

尺寸	最小值	典型值	最大值
E2	2.05	2.15	2.25
L	0.30	0.40	0.50
L1	0.00	—	0.15
aaa	—	—	0.15
bbb	—	—	0.10
ddd	—	—	0.05
eee	—	—	0.08

5. QFN-24 封装规格

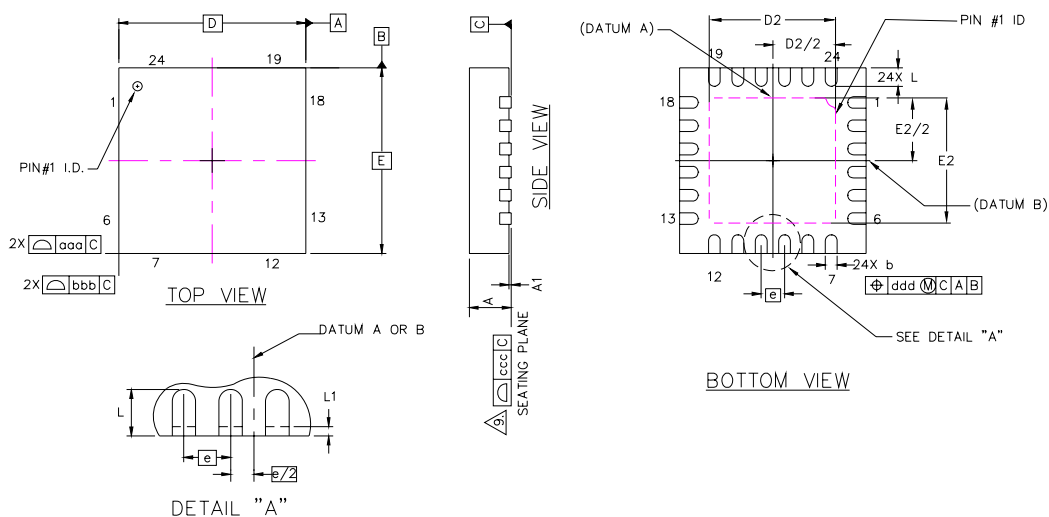


图 5.1 QFN-24 封装图

表 4.2 QFN-24 封装尺寸

尺寸	最小值	典型值	最大值
A	0.70	0.75	0.80
A1	0.00	0.02	0.05
B	0.18	0.25	0.30
D	4.0 BSC		
D2	2.05	2.15	2.25
e	0.50 BSC		
E	4.0 BSC		

尺寸	最小值	典型值	最大值
E2	2.60	2.70	2.80
L	0.35	0.40	0.45
L1	0.00	—	0.15
aaa	—	—	0.15
bbb	—	—	0.10
ccc	—	—	0.05
ddd	—	—	0.08

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

6. 电气特性

6.1 极限参数

表 6.1 极限参数*

参 数	条 件	最小值	典型值	最大值	单位
环境温度（通电情况下）		-55	—	125	℃
储存温度		-65	—	150	℃
任何端口I/O引脚或/RST相对GND的电压		-0.3	—	5.8	V
VDD引脚相对GND的电压		-0.3	—	4.2	V
通过VDD和GND的最大总电流		—	—	500	mA
/RST或任何端口引脚的最大输出灌电流		—	—	100	mA

*注：超过这些列出的“极限参数”可能导致器件永久性损坏。长时间在最大允许值或超过最大允许值的条件下工作可能影响器件的可靠性。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

6.2 总体电气特性

表 6.2 总体电气特性

-40℃到+85℃，25MHz 系统时钟（除非特别说明）。

参 数	条 件	最小值	典型值	最大值	单 位
数字电源电压	正常操作	V_{RST}^1	3.0	3.6	V
	写或擦除FLASH存储器	2.7	3.0	3.6	V
RAM数据保持电源电压		—	1.5	—	V
SYSCLK（系统时钟）（注2）		0	—	25	MHz
T_{SYSH} （SYSCLK高电平时间）		18	—	—	ns
T_{SYSL} （SYSCLK低电平时间）		18	—	—	ns
额定工作温度范围		-40	—	+85	℃
数字电源电流 — CPU 活动（正常方式，从 FLASH 中取指令）					
I_{DD} （注3）	$V_{DD} = 3.6V, F = 25\text{ MHz}$	—	12.3	TBD	mA
	$V_{DD} = 3.0V, F = 25\text{ MHz}$	—	8.9	TBD	mA
	$V_{DD} = 3.0V, F = 1\text{ MHz}$	—	0.46	—	mA
	$V_{DD} = 3.0V, F = 80\text{ kHz}$	—	40	—	μA
I_{DD} 电源敏感度（注3）	$F = 25\text{ MHz}$	—	TBD	—	%/V
	$F = 1\text{ MHz}$	—	TBD	—	%/V
I_{DD} 频率敏感度（注3和注4）	$V_{DD} = 3.0V, F \leq 15\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
	$V_{DD} = 3.0V, F > 15\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
	$V_{DD} = 3.6V, F \leq 15\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
	$V_{DD} = 3.6V, F > 15\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
数字电源电流 — CPU 不活动（空闲方式，不从 FLASH 中取指令）					
I_{DD} （注3）	$V_{DD} = 3.6V, F = 25\text{ MHz}$	—	6.0	TBD	mA
	$V_{DD} = 3.0V, F = 25\text{ MHz}$	—	4.4	TBD	mA
	$V_{DD} = 3.0V, F = 1\text{ MHz}$	—	0.2	—	mA
	$V_{DD} = 3.0V, F = 80\text{ kHz}$	—	16	—	μA
I_{DD} 电源敏感度（注3）	$F = 25\text{ MHz}$	—	TBD	—	%/V
	$F = 1\text{ MHz}$	—	TBD	—	%/V
I_{DD} 频率敏感度（注3和注5）	$V_{DD} = 3.0V, F \leq 1\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
	$V_{DD} = 3.0V, F > 1\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
	$V_{DD} = 3.6V, F \leq 1\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
	$V_{DD} = 3.6V, F > 1\text{ MHz}, T = 25^\circ\text{C}$	—	TBD	—	mA/MHz
数字电源电流（停机或挂起方式）	振荡器停止运行 VDD 监视器禁止	—	<0.1	—	μA
注： 1. 见表 6.4。 2. 为能使用调试功能，SYSCLK 至少应为 32kHz。 3. 基于器件特性数据，未经产品测试。 4. 在频率 $\leq 15\text{ MHz}$ 时，可以通过简单地将感兴趣的频率乘以该范围的频率敏感度来估算 I_{DD} 。当使用这些数字估算频率 $> 15\text{ MHz}$ 时的 I_{DD} 时，估算值应为 25 MHz 时的电流减去由频率敏感度计算出的电流差。例如： $V_{DD} = 3.0V, F = 20\text{ MHz}$ ，则 $I_{DD} = 7.8\text{ mA} - (25\text{ MHz} - 20\text{ MHz}) \times 0.21\text{ mA/MHz} = 6.75\text{ mA}$ 。 5. 在频率 $\leq 1\text{ MHz}$ 时，可以通过简单地将感兴趣的频率乘以该范围的频率敏感度来估算空闲方式 I_{DD} 。当使用这些数字估算频率 $> 1\text{ MHz}$ 时的空闲方式 I_{DD} 时，估算值应为 25 MHz 时的电流减去由频率敏感度计算出的电流差。例如： $V_{DD} = 3.0V, F = 5\text{ MHz}$ ，则空闲方式 $I_{DD} = 4.8\text{ mA} - (25\text{ MHz} - 5\text{ MHz}) \times 0.15\text{ mA/MHz} = 1.8\text{ mA}$ 。					

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 6.3 端口 I/O 直流电气特性

VDD = 2.7V – 3.6V, -40℃到+85℃（除非特别说明）。

参 数	条 件	最小值	典型值	最大值	单 位
输出高电压	I _{OH} = -10μA, 端口 I/O 为推挽方式	VDD-0.1	—	—	V
	I _{OH} = -3mA, 端口 I/O 为推挽方式	VDD-0.7	—	—	
	I _{OH} = -10mA, 端口 I/O 为推挽方式	—	VDD-0.8	—	
输出低电压	I _{OL} = 10μA	—	—	0.1	V
	I _{OL} = 8.5mA	—	—	0.6	
	I _{OL} = 25mA	—	1.0	—	
输入高电压		2.0	—	—	V
输入低电压		—	—	0.8	V
输入漏电流	弱上拉禁止	—	—	±1	μA
	弱上拉使能, V _{IN} = 0 V	—	50	100	

表 6.4 复位电气特性

-40℃到+85℃（除非特别说明）

参 数	条 件	最小值	典型值	最大值	单位
/RST 输出低电平	I _{OL} = 8.5 mA, VDD = 2.7 ~ 3.6V	—	—	0.6	V
/RST 输入低电平		—	—	0.6	V
/RST 输入上拉电流	/RST = 0.0V	—	50	100	μA
V _{DD} POR 门限(V _{RST})		2.40	2.55	2.70	V
时钟丢失检测器超时	从最后一个系统时钟上升沿到产生复位	100	220	600	μs
复位时间延迟	从退出复位到开始执行位于 0x0000 地址的代码之间的延时	—	—	40	μs
产生系统复位的最小 /RST 低电平时间		15	—	—	μs
V _{DD} 监视器启动时间		100	—	—	μs
V _{DD} 监视器电源电流		—	20	40	μA

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 6.5 FLASH 电气特性

VDD = 2.7V ~ 3.6V, -40°C 到 +85°C (除非特别说明)。

参 数	条 件	最小值	典型值	最大值	单 位
FLASH 大小		16384*	—	—	字节
擦写寿命		20k	100k	—	擦/写
擦除时间	25 MHz 系统时钟	10	15	20	ms
写入时间	25 MHz 系统时钟	40	55	70	μs
*注：位于地址 3E00~3FFF 的 512 字节保留。					

表 6.6 内部高频振荡器电气特性

VDD = 2.7V~3.6V; -40°C 到 +85°C (除非特别说明); 使用工厂校准的设置。

参 数	条 件	最小值	典型值	最大值	单位
振荡器频率	IFCN = 11b	24	24.5	25	MHz
振荡器电源电流 (自 VDD)	25°C, VDD = 3.0V OSCICN.7 = 1 OCSICN.5 = 0	—	450	600	μA
电源敏感度	恒温	—	0.12	—	%/V
温度敏感度	恒压	—	60	—	ppm/°C

表 6.7 内部低频振荡器电气特性

VDD = 2.7V~3.6V; -40°C 到 +85°C (除非特别说明); 使用工厂校准的设置。

参 数	条 件	最小值	典型值	最大值	单位
振荡器频率	OSCLD = 11b	72	80	88	KHz
振荡器电源电流 (自 VDD)	25°C, VDD = 3.0V OSCLCN.7 = 1	—	5.5	10	μA
电源敏感度	恒温	—	2.4	—	%/V
温度敏感度	恒压	—	30	—	ppm/°C

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 6.8 ADC0 电气特性

VDD=3.0V, VREF=2.40V(REFSL=0), -40°C 到+85°C (除非特别说明)

参 数	条 件	最小值	典型值	最大值	单 位
直流精度					
分辨率		10			位
积分非线性		—	±0.5	±1	LSB
微分非线性	保证单调	—	±0.5	±1	LSB
偏移误差		-12	3	12	LSB
满度误差		-5	1	5	LSB
偏移温度系数		—	3	—	ppm/°C
动态性能 (10kHz 正弦波单端输入, 满度值之下 1dB, 200ksps)					
信号与噪声加失真比		53	58	—	dB
总谐波失真	到 5 次谐波	—	-75	—	dB
无失真动态范围		—	75	—	dB
转换速率					
SAR 转换时钟		—	—	3.125	MHz
转换时间 (SAR 时钟数)		13	—	—	时钟
跟踪/保持捕获时间		300	—	—	ns
转换速率		—	—	200	ksps
模拟输入					
ADC 输入电压范围	单端方式 (AIN+-GND)	0	—	VREF	V
	差分方式 (AIN+-AIN-)	-VREF		VREF	V
引脚输入电压 (相对于 GND)	单端或差分方式	0	—	VDD	V
采样电容		—	5	—	pF
输入多路器阻抗		—	5	—	KΩ
电源指标					
电 源 电 流 (VDD 给 ADC0 供电)	工作方式, 200ksps	—	500	900	μA
电源抑制比		—	3	—	mV/V
*注: 代表偏离平均值一个标准差。					

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 6.9 温度传感器电气特性

VDD=3.0V, -40°C 到+85°C (除非特别说明)

参 数	条 件	最小值	典型值	最大值	单 位
线性度		—	±0.2	—	°C
相对精度		—	TBD	—	°C
增益		—	TBD	—	mV/°C
增益误差*		—	TBD	—	μV/°C
偏移	温度=0°C	—	TBD	—	mV
偏移误差*	温度=0°C	—	TBD	—	mV
电源电流		—	TBD	—	μA
*注：代表偏离平均值一个标准差。					

表 6.10 电压基准电气特性

VDD=3.0V, -40°C 到+85°C (除非特别说明)

参 数	条 件	最小值	典型值	最大值	单 位
内部基准 (REFBE=1)					
输出电压	环境温度 25°C	2.35	2.42	2.50	V
VREF 短路电流		—	—	10	mA
VREF 温度系数		—	30	—	ppm/°C
负载调整	负载 = 0-200μA 到 GND	—	3	—	μV/μA
VREF 开启时间 1	4.7μF 钽电容, 0.1μF 陶瓷旁路电容	—	7.5	—	ms
VREF 开启时间 2	0.1μF 陶瓷旁路电容	—	200	—	μs
电源抑制比		—	-0.6	—	mV/V
外部基准 (REFBE=0)					
输入电压范围		0	—	V _{DD}	V
输入电流	采样频率 = 200ksps, VREF=3.0V	—	TBD	—	μA
电源指标					
基准偏压发生器	REFBE = 1 或 TEMPE = 1 或 IDA0EN = 1	—	30	50	μA

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 6.11 IDAC 电气特性

-40°C 到+85°C，VDD=3.0V，满度输出电流设置为 2 mA（除非特别说明）。

参 数	条 件	最小值	典型值	最大值	单 位
静态性能					
分辨率		10			位
积分非线性		—	±0.5	±2	LSB
微分非线性	保证单调	—	±0.5	±1	LSB
输出范围		—	—	V _{DD} -1.2	V
偏移误差		—	0	—	μA
满度误差	2 mA 满度输出电流	—	0	±30	μA
满度误差温度系数		—	30	—	ppm/°C
VDD 电源抑制比		—	6	—	μA/V
动态性能					
输出建立时间 (到 1/2LSB)	IDAH:IDAL = 0x3FF~0x000	—	5	—	μs
启动时间		—	5	—	μs
增益变化	1 mA 满度输出电流	—	±1	—	%
	0.5 mA 满度输出电流	—	±1	—	%
功耗					
电源电流（VDD 给 IDAC 供电）	2 mA 满度输出电流	—	2100	—	μA
	1 mA 满度输出电流	—	1100	—	μA
	0.5 mA 满度输出电流	—	600	—	μA

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 6.12 比较器电气特性

VDD=3.0V, -40℃到+85℃（除非特别说明）。

参 数	条 件	最小值	典型值	最大值	单位
响应时间： 方式 0, $V_{cm}^*=1.5V$	(CP0+) - (CP0-) = 100mV	—	100	—	ns
	(CP0+) - (CP0-) = -100mV	—	200	—	ns
响应时间： 方式 1, $V_{cm}^*=1.5V$	(CP0+) - (CP0-) = 100mV	—	250	—	ns
	(CP0+) - (CP0-) = -100mV	—	350	—	ns
响应时间： 方式 2, $V_{cm}^*=1.5V$	(CP0+) - (CP0-) = 100mV	—	400	—	ns
	(CP0+) - (CP0-) = -100mV	—	800	—	ns
响应时间： 方式 3, $V_{cm}^*=1.5V$	(CP0+) - (CP0-) = 100mV	—	1100	—	ns
	(CP0+) - (CP0-) = -100mV	—	5000	—	ns
共模抑制比		—	1.25	5	mV/V
正向回差电压 1	CP0HYP1-0 = 00	—	0	1	mV
正向回差电压 2	CP0HYP1-0 = 01	1	5	10	mV
正向回差电压 3	CP0HYP1-0 = 10	6	10	20	mV
正向回差电压 4	CP0HYP1-0 = 11	12	20	30	mV
负向回差电压 1	CP0HYN1-0 = 00	—	0	1	mV
负向回差电压 2	CP0HYN1-0 = 01	1	5	10	mV
负向回差电压 3	CP0HYN1-0 = 10	6	10	20	mV
负向回差电压 4	CP0HYN1-0 = 11	12	20	30	mV
反相或同相输入电压 范围		-0.25	—	VDD+0.25	V
输入电容		—	4	—	pF
输入偏置电流		—	0.001	—	nA
输入偏移电压		-5	—	+5	mV
电源					
电源抑制比		—	0.1	—	mV/V
上电时间		—	10	—	μS
电源电流（DC）	方式 0	—	10	20	μA
	方式 1	—	4	10	μA
	方式 2	—	2	5	μA
	方式 3	—	0.4	2.5	μA
*注：Vcm 是 CP0+和 CP0-上的共模电压。					

7. 10 位 ADC (ADC0, 仅 C8051F336/8)

C8051F336/8 的 ADC0 是一个 200ksps 的 10 位逐次逼近寄存器型 ADC, ADC 中集成了跟踪保持电路和可编程窗口检测器。ADC0 可完全用软件通过特殊功能寄存器来配置。ADC0 可以工作在单端方式或差分方式, 可以被配置为测量由模拟多路器选择的不同信号 (模拟多路器的说明见“7.4 ADC0 模拟多路器 (仅 C8051F336/8)”)。ADC 的电压基准选择见“10. 电压基准 (仅 C8051F336/8)”)。只有当 ADC 控制寄存器 (ADC0CN) 中的 AD0EN 位被置 1 时 ADC0 子系统才被使能。当 AD0EN 位为 0 时, ADC0 子系统处于低功耗关断方式。

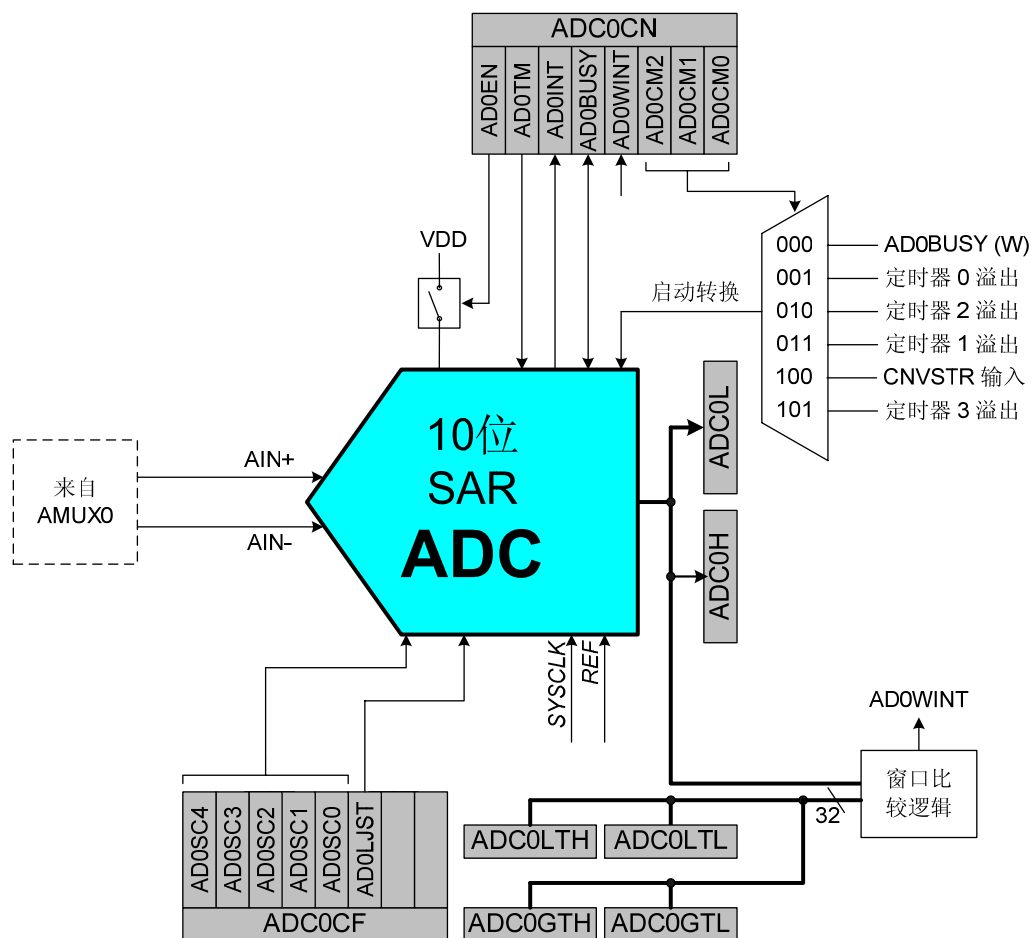


图 7.1 ADC0 功能框图

7.1 输出码格式

当负输入连接到 GND 时, ADC0 工作在单端方式; 当负输入连接到任何其它选项时, ADC0 工作在差分方式。输出码的格式在单端方式和差分方式下是不同的。每次转换结束后, 寄存器 ADC0H 和 ADC0L 中保存 ADC 转换结果的高字节和低字节。转换数据在寄存器对 ADC0H:ADC0L 中的存储方式可以是左对齐或右对齐, 由 AD0LJST 位的设置决定。当工作在单端方式时, 转化码为 10 位无符号整数, 所测量的输入范围为 $0 \sim V_{REF} \cdot 1023/1024$ 。下面是单端方式下数据左对齐和右对齐的例子。ADC0H 和 ADC0L 寄存器中未使用的位被清 0。

输入电压	右对齐的 ADC0H:ADC0L (AD0LJST = 0)	左对齐的 ADC0H:ADC0L (AD0LJST = 1)
$V_{REF} \cdot 1023/1024$	0x03FF	0xFFC0
$V_{REF} \cdot 512/1024$	0x0200	0x8000
$V_{REF} \cdot 256/1024$	0x0100	0x4000
0	0x0000	0x0000

当工作在差分方式时, 转化码为 10 位有符号整数(2 的补码), 所测量的输入范围为 $-V_{REF} \sim V_{REF} \cdot 511/512$ 。下面是差分方式下数据左对齐和右对齐的例子。对于右对齐数据, ADC0H 寄存器中未用的高位填充数据字的符号扩展位。对于左对齐数据, ADC0L 寄存器中未用的低位被清 0。

输入电压	右对齐的 ADC0H:ADC0L (AD0LJST = 0)	左对齐的 ADC0H:ADC0L (AD0LJST = 1)
$V_{REF} \cdot 511/512$	0x01FF	0x7FC0
$V_{REF} \cdot 256/512$	0x0100	0x4000
0	0x0000	0x0000
$-V_{REF} \cdot 256/512$	0xFF00	0xC000
$-V_{REF}$	0xFE00	0x8000

7.2 工作方式

ADC0 的最高转换速度为 200ksps。ADC0 的转换时钟由系统时钟分频得到, 分频数由 ADC0CF 寄存器的 AD0SC 位决定。

7.2.1 启动转换

有 6 种 ADC 转换启动方式, 由 ADC0CN 中的 ADC0 转换启动方式位 (AD0CM2-0) 的状态决定采用哪一种方式。转换触发源有:

1. 写 1 到 ADC0CN 的 AD0BUSY 位;
2. 定时器 0 溢出 (即定时的连续转换);
3. 定时器 2 溢出;
4. 定时器 1 溢出;
5. CNVSTR 输入信号 (P0.6) 的上升沿;
6. 定时器 3 溢出。

向 AD0BUSY 写 1 方式提供了用软件控制 ADC0 转换的能力。AD0BUSY 位在转换期间

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

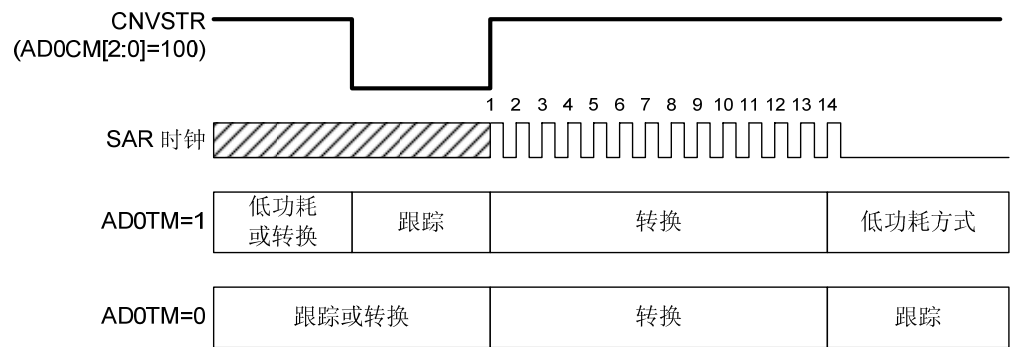
被置 1，转换结束后被清 0。AD0BUSY 位的下降沿触发中断（当被允许时）并置位 ADC0CN 中的中断标志（AD0INT）。注意：当查询 ADC 转换是否结束时，应使用 ADC0 中断标志（AD0INT）。当 AD0INT 位为逻辑 1 时，ADC0 数据寄存器（ADC0H:ADC0L）中的转换结果有效。注意：当转换源是定时器 2 溢出或定时器 3 溢出时，如果定时器 2 或定时器 3 工作在 8 位方式，使用定时器 2/3 的低字节溢出；如果定时器 2/3 工作在 16 位方式，则使用定时器 2/3 的高字节溢出。有关定时器配置方面的信息见“24. 定时器”。

需要注意的是，CNVSTR 输入引脚还是端口引脚 P0.6。当使用 CNVSTR 输入作为转换启动源时，端口引脚 P0.6 应被数字交叉开关跳过。为使交叉开关跳过 P0.6，应将寄存器 P0SKIP 中的位 6 置 1。有关端口 I/O 配置的详细信息，见“20. 端口输入/输出”。

7.2.2 跟踪方式

每次 ADC0 转换之前都必须有一个最小的跟踪时间，以保证转换结果准确。最小跟踪时间由表 6.8 给出。寄存器 ADC0CN 中的 AD0TM 位控制 ADC0 的跟踪保持方式。在缺省状态，ADC0 输入被连续跟踪（转换期间除外）。当 AD0TM 位为逻辑 1 时，ADC0 工作在低功耗跟踪保持方式。在该方式，每次转换前有 3 个 SAR 时钟的跟踪时间（跟踪发生在转换启动信号有效之后）。在低功耗跟踪保持方式下使用 CNVSTR 信号启动转换时，ADC0 只在 CNVSTR 输入为低电平时跟踪；从 CNVSTR 的上升沿开始转换（见图 7.2）。当器件处于低功耗停机或休眠方式时，可以禁止跟踪。低功耗跟踪和保持方式在 AMUX 的设置经常改变时也是很有用的，因为 ADC 有建立时间要求（见“7.3.3 建立时间要求”）。

A. 使用外部触发源的 ADC0 时序



B. 使用内部触发源的 ADC0 时序

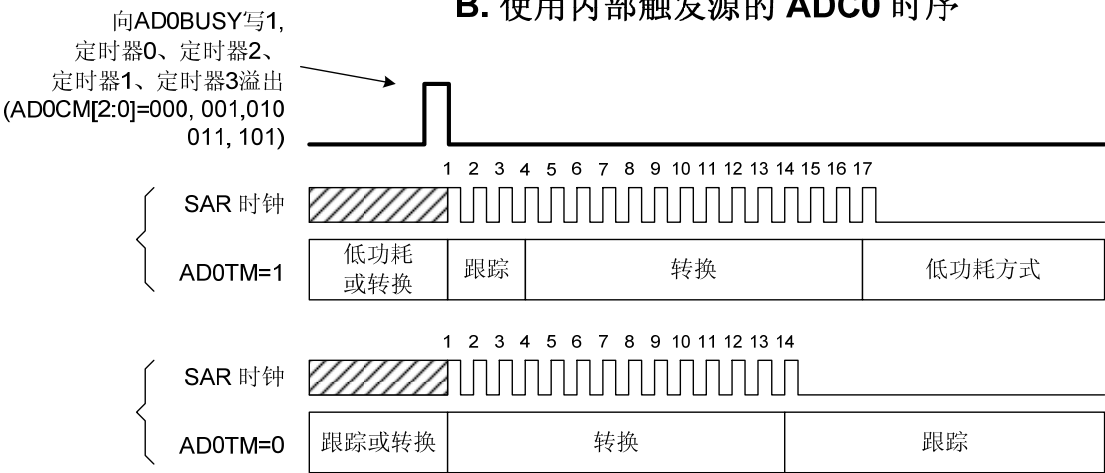


图 7.2 10 位 ADC0 跟踪和转换时序示例

7.2.3 建立时间要求

在进行一次精确的转换之前需要有一个最小的跟踪时间，以保证转换结果正确。该跟踪时间由任何串联阻抗（包括 AMUX0 的电阻）、ADC0 采样电容以及所要求的转换精度决定。注意：在低功耗跟踪方式，每次转换需要用三个 SAR 时钟跟踪。对于很多应用来说，三个 SAR 时钟可以满足最小跟踪时间的要求。

图 7.3 给出了单端和差分方式下等效的 ADC0 输入电路，这两种电路的时间常数相等。对于一个给定的建立精度（SA），所需要的 ADC0 建立时间可以用方程 7.1 估算。当测量温度传感器的输出或 VDD（相对于 GND）时， R_{TOTAL} 减小到 R_{MUX} 。表 6.8 给出了 ADC0 的最小建立时间要求以及多路器阻抗和采样电容值。

$$t = \ln\left(\frac{2^n}{SA}\right) \times R_{TOTAL} C_{SAMPLE}$$

方程 7.1 ADC0 建立时间要求

其中：

SA 是建立精度，用一个 LSB 的分数表示（例如，建立精度 0.25 对应 1/4 LSB）；

t 为所需要的建立时间，以秒为单位；

R_{TOTAL} 为 AMUX0 电阻与任何外部源的电阻之和；

n 为 ADC 的分辨率，用比特表示（10）。

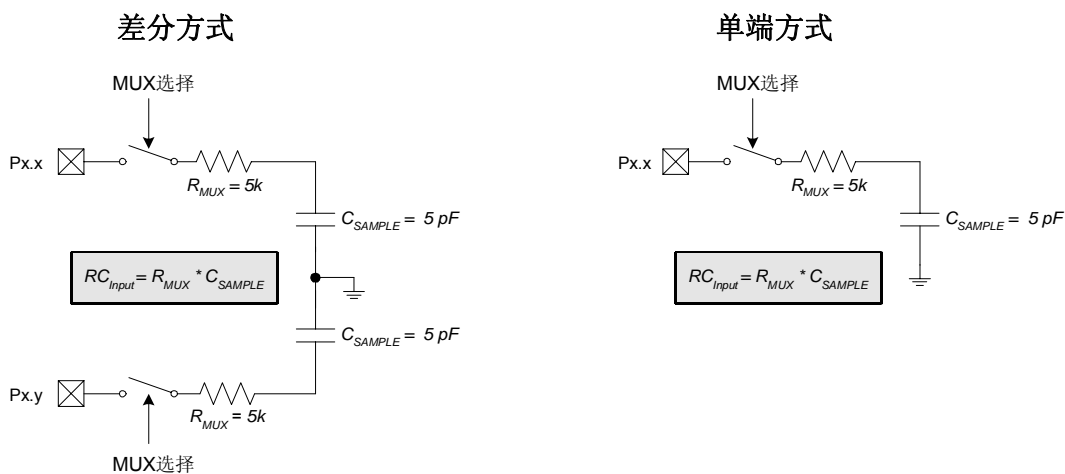


图 7.3 ADC0 等效输入电路

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 7.1 ADC0CF: ADC0 配置寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R	R	复位值
AD0SC4	AD0SC3	AD0SC2	AD0SC1	AD0SC0	AD0LJST	-	-	11111000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xBC
<p>位 7-3: AD0SC4-0: ADC0 SAR 转换时钟周期控制位 SAR 转换时钟来源于系统时钟, 由下面的方程给出, 其中 <i>AD0SC</i> 表示 AD0SC4-0 中保存的 5 位数值。对 SAR 转换时钟频率的要求见表 6.8。</p> $AD0SC = \frac{SYSCLK}{CLK_{SAR}} - 1$ <p>位 2 AD0LJST: ADC0 左对齐选择位。 0: ADC0H:ADC0L 寄存器中的数据为右对齐。 1: ADC0H:ADC0L 寄存器中的数据为左对齐。</p> <p>位 1-0: 未使用。读=00b, 写=忽略。</p>								

SFR 定义 7.2 ADC0H: ADC0 数据字高字节寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xBE
<p>位 7-0: ADC0 数据字高字节。 AD0LJST = 0 时: 位 7~2 为位 1 的符号扩展位, 位 1~0 为 10 位 ADC0 数据的高 2 位。 AD0LJST = 1 时: 位 7~0 是 10 位 ADC0 数据的高 8 位。</p>								

SFR 定义 7.3 ADC0L: ADC0 数据字低字节寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xBD
<p>位 7-0: ADC0 数据字低字节。 AD0LJST = 0 时: 位 7~0 是 10 位 ADC0 数据的低 8 位。 AD0LJST = 1 时: 位 7~6 是 10 位 ADC0 数据的低 2 位, 位 5~0 的读出值总是为 0。</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 7.4 ADC0CN: ADC0 控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM2	AD0CM1	AD0CM0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: (可位寻址) 0xE8
位 7:	AD0EN: ADC0 使能位 0: ADC0 禁止。ADC0 处于低耗停机状态。 1: ADC0 使能。ADC0 处于活动状态, 可以进行转换数据。							
位 6:	AD0TM: ADC0 跟踪方式位 0: 正常方式: 当 ADC0 被使能时, 除了转换期间之外一直处于跟踪方式。ADC 转换在启动转换事件 (由 AD0CM[2:0]定义) 到来后立即开始。 1: 低功耗跟踪方式: 对于 AD0CM[2:0] = 100, ADC 在 CNVSTR 为低电平时跟踪, 在 CNVSTR 输入信号的上升沿立即开始转换。对于 AD0CM[2:0]的其他值, 跟踪在启动转换事件到来后开始, 持续 3 个 SAR 时钟后开始转换。							
位 5:	AD0INT: ADC0 转换结束中断标志 0: 从最后一次将该位清 0 后, ADC0 还没有完成一次数据转换。 1: ADC0 完成了一次数据转换。							
位 4:	AD0BUSY: ADC0 忙标志位 读 0: ADC0 不在进行转换。 1: ADC0 正在进行转换。 写 0: 无作用。 1: 若 AD0CM2-0=000b 则启动 ADC0 转换。							
位 3:	AD0WINT: ADC0 窗口比较中断标志 0: 未发生 ADC0 窗口比较匹配。 1: 发生 ADC0 窗口比较匹配。							
位 2-0:	AD0CM2-0: ADC0 转换启动方式选择 000: 向 AD0BUSY 写 1 时启动 ADC0 转换。 001: 定时器 0 溢出启动 ADC0 转换。 010: 定时器 2 溢出启动 ADC0 转换。 011: 定时器 1 溢出启动 ADC0 转换。 100: 外部 CNVSTR 输入信号的上升沿启动 ADC0 转换。 101: 定时器 3 溢出启动 ADC0 转换。 11x: 保留。							

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

7.3 可编程窗口检测器

ADC 可编程窗口检测器不停地将 ADC0 输出与用户编程的极限值进行比较，并在检测到越限条件时通知系统控制器。这在一个中断驱动的系统尤其有效，既可以节省代码空间和 CPU 带宽又能提供快速响应时间。窗口检测器中断标志（ADC0CN 中的 AD0WINT）也可被用于查询方式。ADC0 下限（大于）寄存器（ADC0GTH:ADC0GTL）和 ADC0 上限（小于）寄存器（ADC0LTH:ADC0LTL）中保持比较值。注意，窗口检测器标志可以被编程为指示测量数据位于用户编程的极限值以内还是以外，这取决于 ADC0GT 和 ADC0LT 寄存器的编程值。

SFR 定义 7.5 ADC0GTH: ADC0 下限（大于）数据字高字节寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xC4
位 7-0: ADC0 下限数据字高字节。								

SFR 定义 7.6 ADC0GTL: ADC0 下限（大于）数据字低字节寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xC3
位 7-0: ADC0 下限数据字低字节。								

SFR 定义 7.7 ADC0LTH: ADC0 上限（小于）数据字高字节寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xC6
位 7-0: ADC0 上限数据字高字节。								

SFR 定义 7.8 ADC0LTL: ADC0 上限（小于）数据字低字节寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xC5
位 7-0: ADC0 上限数据字低字节。								

7.3.1 单端方式下的窗口检测器

图 7.4 给出了单端方式下数据右对齐窗口比较的两个例子。左边的例子所使用的极限值为：ADC0LTH:ADC0LTL = 0x0080 和 ADC0GTH:ADC0GTL = 0x0040；右边的例子所使用的极限值为：ADC0LTH:ADC0LTL = 0x0040 和 ADC0GTH:ADC0GTL = 0x0080。在单端方式，转换码是 10 位无符号整数，对应的电压范围为 0 ~ VREF*1023/1024。在左边的例子中，如果 ADC0 转换结果数据字位于由 ADC0GTH:ADC0GTL 和 ADC0LTH:ADC0LTL 定义的范围之内（即 $0x0040 < \text{ADC0H:ADC0L} < 0x0080$ ），则会产生一个 AD0WINT 中断。在右边的例子中，如果 ADC0 转换结果数据字位于由 ADC0GT 和 ADC0LT 定义的范围之外（即 $\text{ADC0H:ADC0L} < 0x0040$ 或 $\text{ADC0H:ADC0L} > 0x0080$ ），则会产生一个 AD0WINT 中断。图 7.5 给出了单端方式下数据左对齐窗口比较的例子。

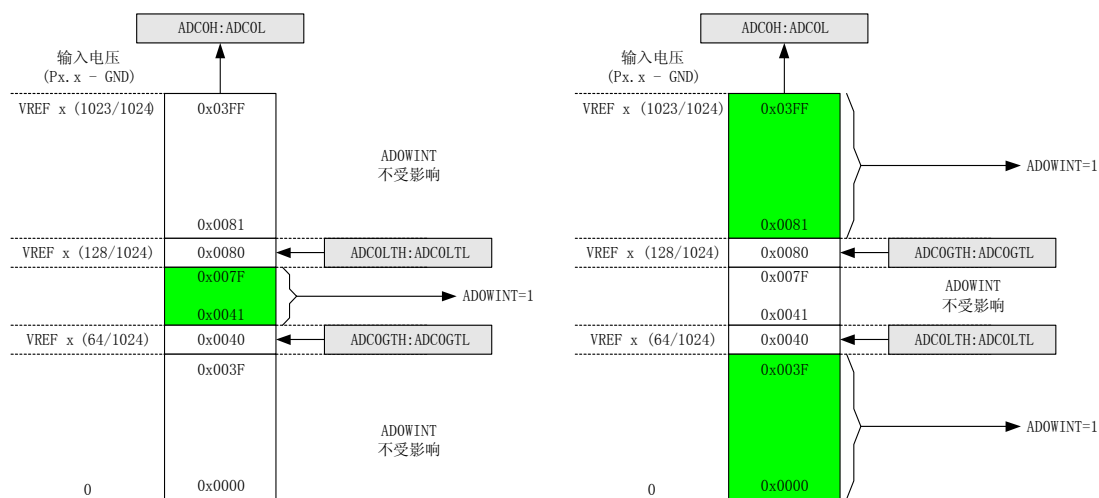


图 7.4 ADC 窗口比较示例（单端方式右对齐数据）

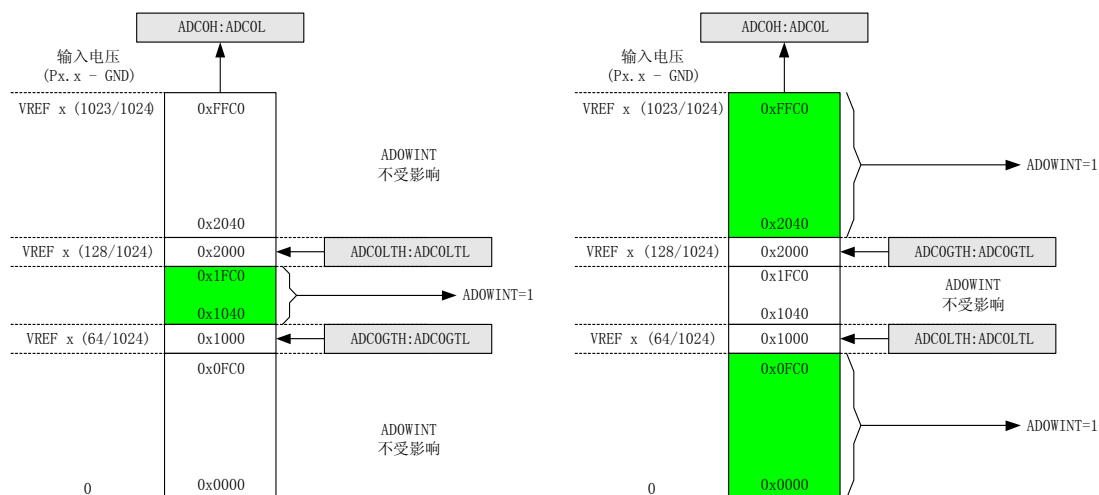


图 7.5 ADC 窗口比较示例（单端方式左对齐数据）

7.3.2 差分方式下的窗口检测器

图 7.6 给出了差分方式下数据右对齐窗口比较的两个例子。左边的例子所使用的极限值为：ADC0LTH:ADC0LTL = 0x0040 (+64d) 和 ADC0GTH:ADC0GTL = 0xFFFF (-1d)；右边的例子所使用的极限值为：ADC0LTH:ADC0LTL = 0xFFFF (-1d) 和 ADC0GTH:ADC0GTL = 0x0040 (+64d)。在差分方式，转换码是 10 位有符号整数(2 的补码)，对应的电压范围为 -VREF ~ VREF*511/512。在左边的例子中，如果 ADC0 转换结果数据字 (ADC0H:ADC0L) 位于由 ADC0GTH:ADC0GTL 和 ADC0LTH:ADC0LTL 定义的范围之内 (即 $0xFFFF (-1d) < ADC0H:ADC0L < 0x0040 (+64d)$)，则会产生一个 AD0WINT 中断。在右边的例子中，如果 ADC0 转换结果数据字 (ADC0H:ADC0L) 位于由 ADC0GT 和 ADC0LT 定义的范围之外 (即 $ADC0H:ADC0L < 0xFFFF (-1d)$ 或 $ADC0H:ADC0L > 0x0040 (+64d)$)，则会产生一个 AD0WINT 中断。图 7.6 给出了差分方式下数据左对齐窗口比较的两个例子。

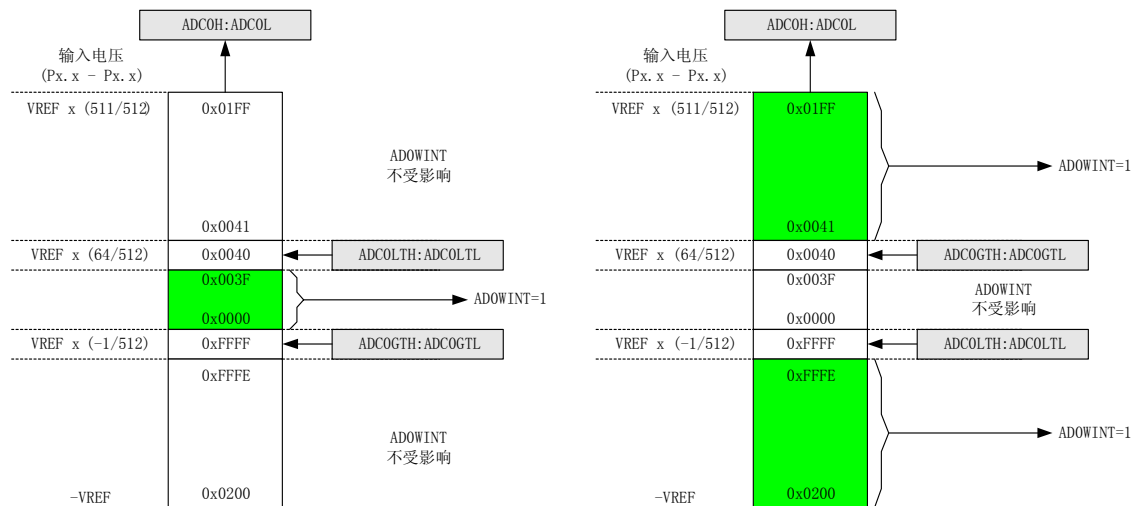


图 7.6 ADC 窗口比较示例 (差分方式右对齐数据)

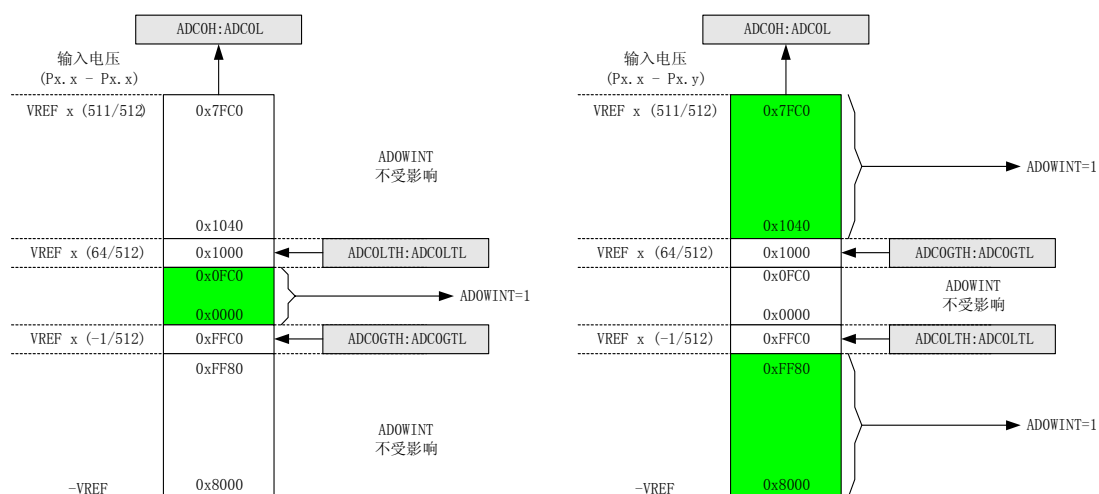


图 7.7 ADC 窗口比较示例 (差分方式左对齐数据)

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

7.4 ADC0 模拟多路选择器（仅 C8051F336/8）

C8051F336/8 中的 ADC0 有两个模拟多路选择器，总体称为 AMUX0。

AMUX0 选择去 ADC 的正输入和负输入。端口 I/O 引脚、片内温度传感器输出和正电源（VDD）中的任何一个都可以被选择为正输入；端口 I/O 引脚、VREF 和 GND 中的任何一个都可以被选择为负输入。当 GND 被选择为负输入时，ADC0 工作在单端方式；在所有其它时间，ADC0 工作在差分方式。ADC0 的输入通道由 AMX0P 和 AMX0N 寄存器选择（见 SFR 定义 7.9 和 SFR 定义 7.10）。

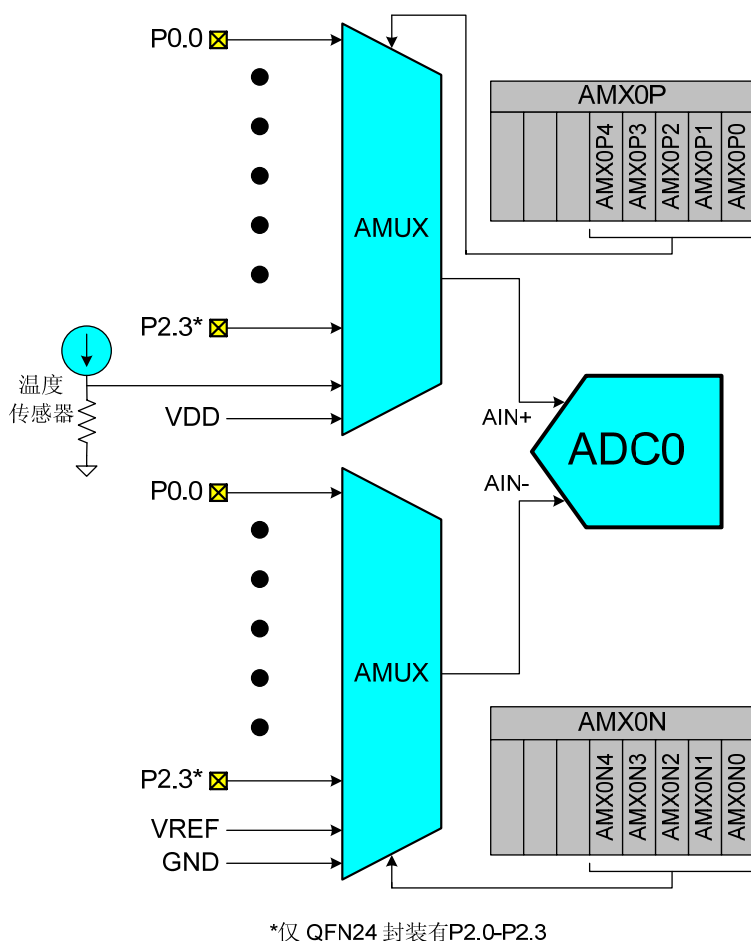


图 7.8 ADC0 多路选择器原理框图

关于 ADC0 输入配置的重要注意事项：被选择为 ADC0 输入的端口引脚应被配置为模拟输入，并且应被数字交叉开关跳过。要将一个端口引脚配置为模拟输入，应将 PnMDIN 寄存器中的对应位清 0。为了使交叉开关跳过一个端口引脚，应将 PnSKIP 寄存器中的对应位置 1。有关端口 I/O 配置的详细信息见“20. 端口输入/输出”。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 7.9 AMX0P: AMUX0 正输入通道选择寄存器

R	R	R	R/W	R/W	R/W	R/W	R/W	复位值
-	-	-	AMX0P4	AMX0P3	AMX0P2	AMX0P1	AMX0P0	00011111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xBB
位 7-5: 未使用。读=000b, 写=忽略。								
位 4-0: AMX0P4-0: AMUX0 正输入选择								
AMX0P4-0			ADC0 正输入					
00000			P0.0					
00001			P0.1					
00010			P0.2					
00011			P0.3					
00100			P0.4					
00101			P0.5					
00110			P0.6					
00111			P0.7					
01000			P1.0					
01001			P1.1					
01010			P1.2					
01011			P1.3					
01100			P1.4					
01101			P1.5					
01110			P1.6					
01111			P1.7					
10000			温度传感器					
10001			VDD					
10010			P2.0 (仅 C8051F338/9)					
10011			P2.1 (仅 C8051F338/9)					
10100			P2.2 (仅 C8051F338/9)					
10101			P2.3 (仅 C8051F338/9)					
10110~11111			不选择任何输入					

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 7.10 AMX0N: AMUX0 负输入通道选择寄存器

R	R	R	R/W	R/W	R/W	R/W	R/W	复位值
-	-	-	AMX0N4	AMX0N3	AMX0N2	AMX0N1	AMX0N0	00011111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xBA

位 7-5: 未使用。读=000b，写=忽略。

位 4-0: AMX0N4-0: AMUX0 负输入选择

当 GND 被选择为负输入时，ADC0 工作在单端方式。对于所有其它负输入选择，ADC0 工作在差分方式。

AMX0N4-0	ADC0 负输入
00000	P0.0
00001	P0.1
00010	P0.2
00011	P0.3
00100	P0.4
00101	P0.5
00110	P0.6
00111	P0.7
01000	P1.0
01001	P1.1
01010	P1.2
01011	P1.3
01100	P1.4
01101	P1.5
01110	P1.6
01111	P1.7
10000	VREF
10001	GND（ADC 工作在单端方式）
10010	P2.0（仅 C8051F338/9）
10011	P2.1（仅 C8051F338/9）
10100	P2.2（仅 C8051F338/9）
10101	P2.3（仅 C8051F338/9）
10110~11111	不选择任何输入

8. 温度传感器（仅 C8051F336/8）

C8051F336/8 内部包含一个温度传感器，可以通过单端方式的 ADC 多路器访问。为了使用 ADC 测量温度传感器，ADC MUX 的正通道应被配置为连接到温度传感器，ADC MUX 的负通道应被配置为连接到 GND。温度传感器的传输函数示于图 8.1。当 ADC 多路器被正确设置时，输出电压 (V_{TEMP}) 为 ADC 的正输入。寄存器 REF0CN 中的 TEMPE 位用于使能/禁止温度传感器（见 SFR 定义 10.1）。当被禁止时，温度传感器为缺省的高阻状态，此时对温度传感器的任何 ADC0 测量结果都是无意义的。温度传感器的增益和偏移参数请参见表 6.9。

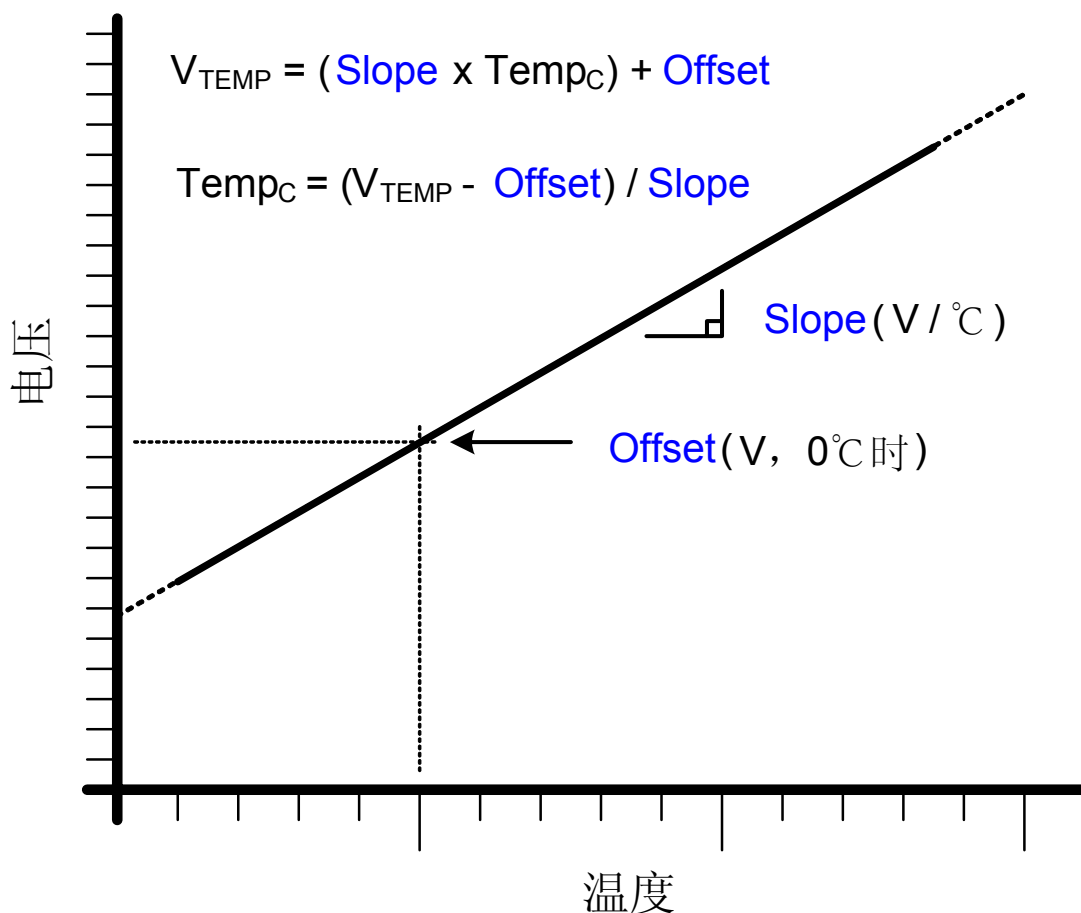


图 8.1 温度传感器传输函数

9. 10 位电流模式 DAC (IDA0, 仅 C8051F336/8)

C8051F336/8 器件内部有一个 10 位的电流模式数/模转换器 (IDAC)。IDAC 的最大输出电流可以有三种不同的设置: 0.5mA、1mA 和 2mA。用 IDA0 控制寄存器中的 IDA0EN 位来使能或禁止 IDAC (见 SFR 定义 9.1)。当 IDA0EN 被设置为 0 时, IDAC 引脚 (P0.1) 作为 GPIO 引脚使用; 当 IDA0EN 被置 1 时, IDAC 引脚的数字输出驱动器和弱上拉被自动禁止, 该引脚被连到 IDAC 的输出。当 IDAC 被使能时, 内部的带隙偏置发生器为其生成基准电流。当使用 IDAC 时, P0SKIP 寄存器中的位 1 应被置 1, 以使交叉开关跳过 IDAC 引脚。

9.1 IDA0 输出更新

IDA0 具有灵活的输出更新机制, 允许无缝满度变化, 支持无抖动波形更新。IDA0 有三种更新模式: 写 IDA0H、定时器溢出或外部引脚边沿。

9.1.1 On-Demand 输出更新

IDA0 的缺省更新模式 (IDA0CN.[6:4] = '111') 为 “On-Demand” 模式, 更新发生在写 IDA0 数据寄存器高字节 (IDA0H) 时。在该模式下, 写 IDA0L 时数据被保持, 在写 IDA0H 之前 IDA0 的输出不会发生变化。如果要向 IDAC 的数据寄存器写 10 位的数据字, 则 10 位数据字要写入低字节 (IDA0L) 和高字节 (IDA0H) 数据寄存器。在执行完对 IDA0H 的写操作后, 数据被锁存到 IDA0, 因此, 在需要 10 位分辨率的情况下, 应先写 IDA0L, 再写 IDA0H。IDAC 可以用于 8 位方式, 此时要将 IDA0L 初始化为一个所希望的数值 (通常为 0x00), 只对 IDA0H 写入 (有关 10 位 IDAC 数据字在 16 位 SFR 空间内的存储格式信息见 9.2 节)。

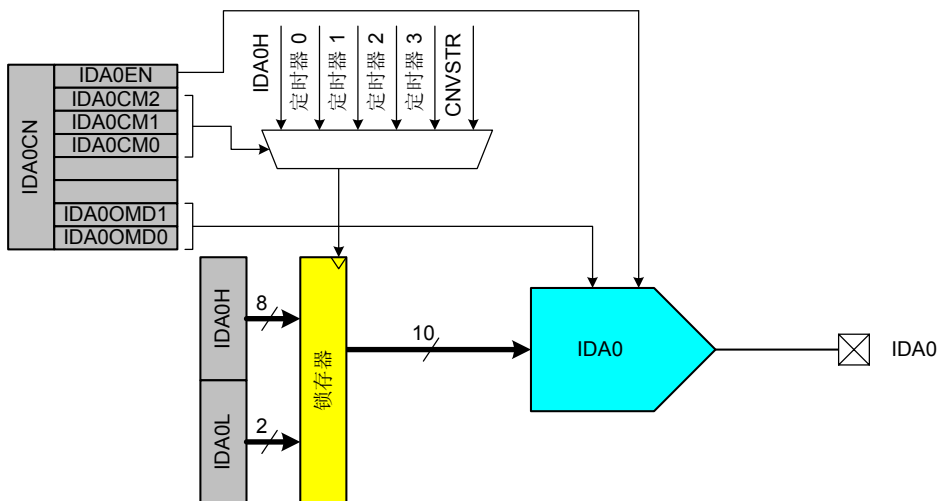


图9.1 IDA0原理框图

9.1.2 基于定时器溢出的输出更新模式

在用定时器溢出启动 ADC 转换时，转换发生时间与处理器无关。与此类似，IDAC 的输出也可以用定时器溢出事件触发更新。这一特性在以给定采样频率产生输出波形的系统中非常有用，可以避免中断延迟时间和指令执行时间变化对 IDAC 输出时序的影响。当 IDA0CM 位（IDA0CN.[6:4]）被设置为 ‘000’、‘001’、‘010’ 或 ‘011’ 时，写入到两个 IDAC 数据寄存器（IDA0L 和 IDA0H）的数据被保持，直到相应的定时器溢出事件（分别为定时器 0、定时器 1、定时器 2 或定时器 3）发生时，IDA0H:IDA0L 的内容才被复制到 IDAC 输入锁存器，允许 IDAC 输出变为新值。

9.1.3 基于 CNVSTR 边沿的输出更新模式

IDAC 还可以被配置为在外部 CNVSTR 信号的上升沿、下降沿或两个边沿进行输出更新。当 IDA0CM 位（IDA0CN.[6:4]）被设置为 ‘100’、‘101’ 或 ‘110’ 时，写入到两个 IDAC 数据寄存器（IDA0L 和 IDA0H）的数据被保持，直到 CNVSTR 输入引脚的边沿发生。IDA0CM 位的具体设置决定 IDAC 输出更新发生在 CNVSTR 的上升沿、下降沿或在两个边沿都发生更新。当相应的边沿发生时，IDA0H:IDA0L 的内容被复制到 IDAC 输入锁存器，允许 IDAC 输出变为所希望的新值。

9.2 IDAC 输出字格式

IDAC 数据寄存器（IDA0H 和 IDA0L）中的数据是左对齐的，这意味着 IDAC 输出数据字的高 8 位被映射到 IDA0H 的位 7-0，而 IDAC 输出数据字的低 2 位被映射到 IDA0L 的位 7 和位 6。图 9.2 示出了 IDAC 数据字的格式。

IDA0H								IDA0L							
IDA09	IDA08	IDA07	IDA06	IDA05	IDA04	IDA03	IDA02	IDA01	IDA00						

输入数据字 (IDA09 – IDA00)	输出电流 IDA0OMD[1:0]='1x'	输出电流 IDA0OMD[1:0]='01'	输出电流 IDA0OMD[1:0]='00'
0x000	0 mA	0 mA	0 mA
0x001	1/1024 × 2 mA	1/1024 × 1 mA	1/1024 × 0.5 mA
0x200	512/1024 × 2 mA	512/1024 × 1 mA	512/1024 × 0.5 mA
0x3FF	1023/1024 × 2 mA	1023/1024 × 1 mA	1023/1024 × 0.5 mA

图 9.2 IDA0 数据字格式

IDAC 的满度输出电流由 IDA0OMD 位（IDA0CN[1:0]）选择。缺省情况下，IDAC 的满度输出电流被设置为 2 mA。通过配置 IDA0OMD 位还可以将满度输出电流设置为 0.5mA 或 1mA。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 9.1 IDA0CN: IDA0 控制寄存器

R/W	R/W	R/W	R/W	R	R	R/W	R/W	复位值
IDA0EN	IDA0CM[2:0]			-	-	IDA0OMD[1:0]		01110010
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xB9
<p>位 7: IDA0EN: IDA0 使能位 0: IDA0 禁止。 1: IDA0 使能。</p> <p>位 6-4: IDA0CM[2:0]: IDA0 输出更新源选择位 000: 定时器 0 溢出触发 DAC 输出更新。 001: 定时器 1 溢出触发 DAC 输出更新。 010: 定时器 2 溢出触发 DAC 输出更新。 011: 定时器 3 溢出触发 DAC 输出更新。 100: CNVSTR 的上升沿触发 DAC 输出更新。 101: CNVSTR 的下降沿触发 DAC 输出更新。 110: CNVSTR 的任一边沿触发 DAC 输出更新。 111: 写 IDA0H 触发 DAC 输出更新。</p> <p>位 3-2: 未使用。读 = 00b。写 = 忽略。</p> <p>位 1-0: IDA0OMD[1:0]: IDA0 输出方式选择位 00: 0.5 mA 满度输出电流。 01: 1.0 mA 满度输出电流。 1x: 2.0 mA 满度输出电流。</p>								

SFR 定义 9.2 IDA0H: IDA0 数据字高字节寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x97
<p>位 7-0: IDA0[9:2]: IDA0 数据字高字节。 位 7~0 是 10 位 IDA0 数据字的高 8 位。</p>								

SFR 定义 9.3 IDA0L: IDA0 数据字低字节寄存器

R/W	R/W	R	R	R	R	R	R	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x96
<p>位 7-6: IDA0[1:0]: 10 位 IDA0 数据字的低 2 位。 位 5-0: 未使用。读 = 000000b。写 = 忽略。</p>								

10. 电压基准（仅 C8051FF336/8）

C8051F336/8 的电压基准 MUX 可以被配置为连接到外部电压基准、内部电压基准或电源电压 VDD（见图 10.1）。基准控制寄存器 REF0CN（见 SFR 定义 10.1）中的 REFSL 位选择 ADC 的基准源。选择使用外部或内部基准时，REFSL 位应被设置为 0，以选择 VREF 引脚；选择 VDD 作为基准源时，REFSL 应被置 1。

REF0CN 中的 BIASE 位控制内部偏置电压发生器，片内很多模拟外设都要用到它。当任何一个需要它的外设被使能时，偏压发生器被自动使能。也可以通过向 REF0CN 中的 BIASE 位写 1 来使能偏置电压发生器，表 6.10 给出了电压基准电路的电气特性。

内部电压基准电路包含一个 1.2V、温度稳定性好的带隙电压基准发生器和一个两倍增益的输出缓冲放大器。内部电压基准可以被驱动输出到 VREF 引脚，这可通过将 REF0CN 寄存器中的 REFBE 位置 1 来实现。VREF 引脚对地的负载电流应小于 200 μ A。当使用内部电压基准时，建议在 VREF 和 GND 之间跨接 0.1 μ F 和 4.7 μ F 的旁路电容。如果不使用内部基准，REFBE 位应被清 0。表 6.10 给出了内部电压基准的电气特性。

注意：当使用外部电压基准和内部基准中的任何一个时，VREF 引脚应被配置为模拟输入并被数字交叉开关跳过。有关 VREF 引脚的位置以及如何将该引脚配置为模拟输入和使交叉开关跳过该引脚的详细信息，见“20. 端口输入/输出”。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

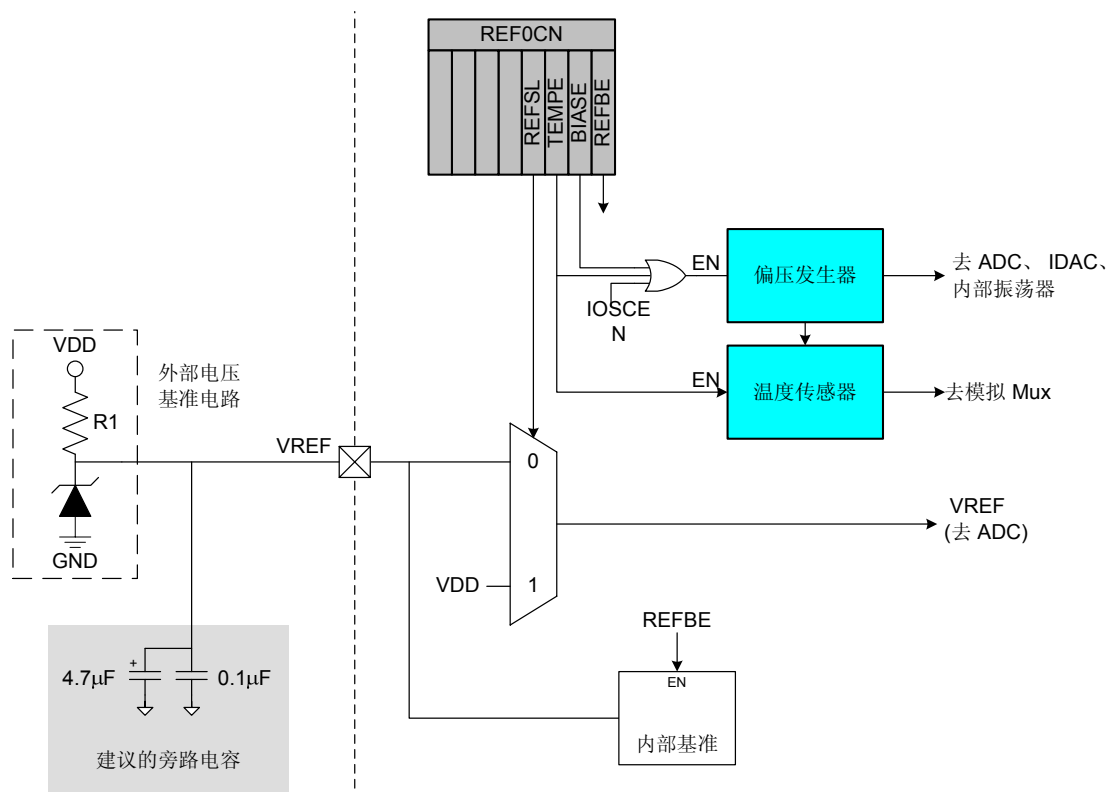


图 10.1 电压基准功能框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 10.1 REF0CN: 电压基准控制寄存器

R	R	R	R	R/W	R/W	R/W	R/W	复位值
-	-	-	-	REFSL	TEMPE	BIASE	REFBE	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xD1
位 7-4: 未用。读 = 0000b, 写 = 忽略。								
位 3: REFSL: 电压基准选择 该位选择电压基准源。 0: VREF 引脚作为电压基准。 1: VDD 作为电压基准。								
位 2: TEMPE: 温度传感器使能位 0: 内部温度传感器关闭。 1: 内部温度传感器工作。								
位 1: BIASE: 内部模拟偏压发生器使能位 0: 内部偏压发生器关闭。 1: 内部偏压发生器工作。								
位 0: REFBE: 内部基准缓冲器使能位 0: 内部基准缓冲器被禁止。 1: 内部基准缓冲器被使能。内部电压基准被驱动到 VREF 引脚。								

11. 比较器 0

C8051F336/7/8/9 器件内部有一个可编程电压比较器，即比较器 0，如图 11.1 所示。

比较器提供可编程的响应时间和回差电压，有一个模拟输入多路器和两个可以通过交叉开关接到外部引脚的输出：一个同步“锁存”输出（CP0）和一个异步“直接”输出（CP0A）。即使在系统时钟停止时，CP0A 信号仍然可用，这就允许比较器在器件处于停机方式时工作并产生输出。当被分配了端口引脚时，比较器的输出可以被配置为漏极开路或推挽方式（见 20.4 端口 I/O 初始化）。比较器 0 可以被用作复位源（见 18.5 比较器 0 复位）。

比较器 0 的输入由比较器输入多路器来选择，详见“11.1 比较器多路器”。

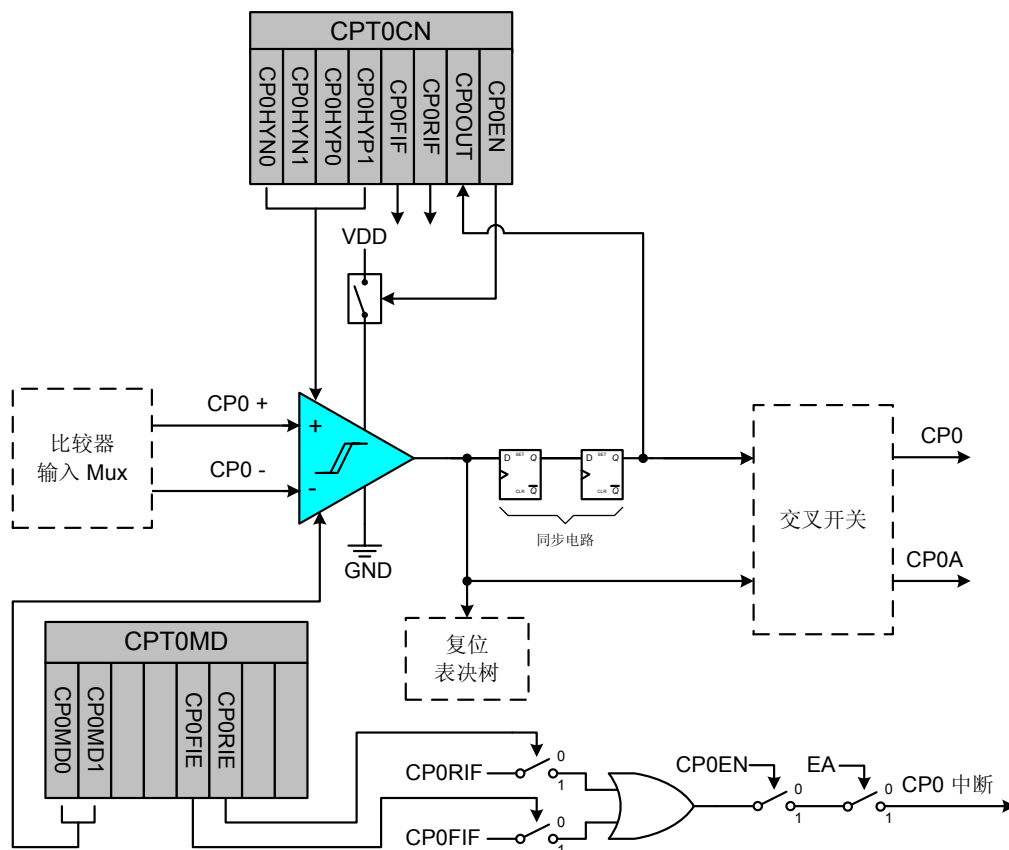


图 11.1 比较器 0 功能框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

比较器的输出可以被软件查询，可以作为中断源，也可以被连到端口引脚。当被连到端口引脚时，比较器的输出可以是与系统时钟同步的或者是不同步的。即使在停机方式（系统时钟停止），异步输出信号仍然可用。当被禁止时，比较器输出（如果已通过交叉开关分配了端口 I/O 引脚）的缺省值为逻辑低电平，供给比较器的电源被断开。有关通过交叉开关配置比较器输出的详细信息见“20.3 优先权交叉开关译码器”。比较器的输入可以承受-0.25V 到(VDD) + 0.25V 的外部驱动电压而不至损坏或发生工作错误。表 6.12 给出了比较器的电气特性。

可以通过对寄存器 CPT0MD（见 SFR 定义 11.2）编程来设置比较器的响应时间。选择较长的响应时间可以减小比较器的电源电流。

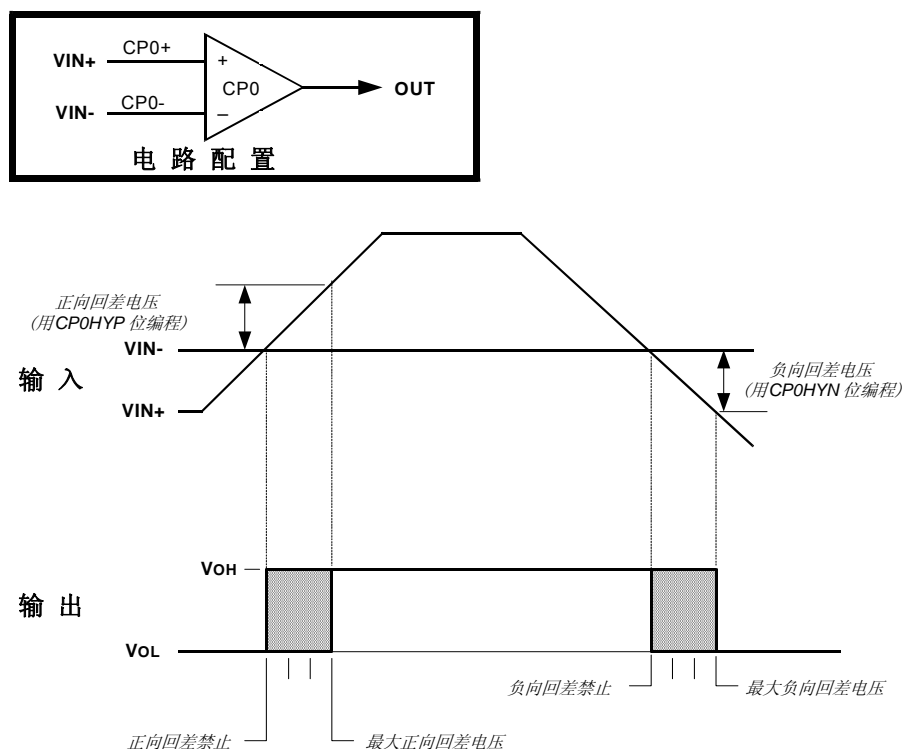


图 11.2 比较器回差电压曲线

比较器的回差电压可以通过比较器控制寄存器（CPT0CN）用软件编程。用户既可以对回差电压值（相对于输入电压）编程，也可以对门限电压两侧的正向和负向回差对称度编程。

使用比较器控制寄存器 CPT0CN（SFR 定义 11.1）中的位 3-0 对比较器的回差值进行编程。负向回差电压值由 CP0HYN 位的设置决定。如图 11.2 所示，可以设置 20mV、10mV 或 5mV 的负向回差电压值，或者禁止负向回差电压。类似地，通过编程 CP0HYP 位设置正向回差电压值。

在比较器输出的上升沿和下降沿都可以产生中断。CP0FIF 中断标志在比较器的下降沿置 1，CP0RIF 中断标志在比较器的上升沿置 1。这些位一旦被置 1，将一直保持 1 状态直到被软件清除。通过将 CP0RIE 设置为逻辑 1 来允许比较器上升沿中断，通过将 CP0FIE 设置为逻辑 1 来允许比较器下降沿中断。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

可以在任意时刻通过读取 CP0OUT 位得到比较器的输出状态。通过置位 CP0EN 位来使能比较器，通过将该位清 0 来禁止比较器。

注意：在对比较器上电或改变比较器的回差电压或响应时间控制位时，可能产生假上升沿或下降沿。建议在比较器被使能或方式位发生变化后经过一段延时将上升沿和下降沿标志清 0。

SFR 定义 11.1 CPT0CN: 比较器 0 控制寄存器

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	复位值
CP0EN	CP0OUT	CP0RIF	CP0FIF	CP0HYP1	CP0HYP0	CP0HYN1	CP0HYN0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x9B
位 7:	CP0EN: 比较器 0 使能位 0: 比较器 0 禁止。 1: 比较器 0 使能。							
位 6:	CP0OUT: 比较器 0 输出状态标志 0: 电压值 CP0+ < CP0-。 1: 电压值 CP0+ > CP0-。							
位 5:	CP0RIF: 比较器 0 上升沿中断标志，必须用软件清 0。 0: 自该标志位被清除后，没有发生过比较器 0 上升沿中断。 1: 自该标志位被清除后，发生了比较器 0 上升沿中断。							
位 4:	CP0FIF: 比较器 0 下降沿中断标志，必须用软件清 0 0: 自该标志位被清除后，没有发生过比较器 0 下降沿中断。 1: 自该标志位被清除后，发生了比较器 0 下降沿中断。							
位 3-2:	CP0HYP[1:0]: 比较器 0 正向回差电压控制位 00: 禁止正向回差电压。 01: 正向回差电压= 5mV。 10: 正向回差电压= 10mV。 11: 正向回差电压= 20mV。							
位 1-0:	CP0HYN[1:0]: 比较器 0 负向回差电压控制位 00: 禁止负向回差电压。 01: 负向回差电压= 5mV。 10: 负向回差电压= 10mV。 11: 负向回差电压= 20mV。							

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 11.2 CPT0MD: 比较器方式选择寄存器

R	R	R/W	R/W	R	R	R/W	R/W	复位值
-	-	CP0RIE	CP0FIE	-	-	CP0MD1	CP0MD0	00000010
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x9D
位 7-6: 未用。读 = 00b, 写 = 忽略。								
位 5: CP0RIE: 比较器 0 上升沿中断允许								
0: 比较器 0 上升沿中断禁止								
1: 比较器 0 上升沿中断允许								
位 4: CP0FIE: 比较器 0 下降沿中断允许								
0: 比较器 0 下降沿中断禁止								
1: 比较器 0 下降沿中断允许								
位 3-2: 未用。读 = 00b, 写 = 忽略。								
位 1-0: CP0MD[1:0]: 比较器 0 方式选择								
这两位选择比较器 0 的响应时间和功耗。								
00: 方式 0 (最快响应时间, 最高功耗)								
01: 方式 1								
10: 方式 2								
11: 方式 3 (最慢响应时间, 最低功耗)								

11.1 比较器多路器

C8051F336/7/8/9 器件内部包含一个模拟输入多路器，将端口 I/O 连接到比较器输入。比较器 0 的输入用 CPT0MX 寄存器（SFR 定义 11.3）来选择。CMX0P1 和 CMX0P0 位选择比较器 0 的正输入；CMX0N1 和 CMX0N0 位选择比较器 0 的负输入。**注意：**被选择为比较器输入的引脚应被配置为模拟输入（通过相应的端口配置寄存器），交叉开关应被配置为跳过这些端口引脚（有关端口配置的详细信息见“20.6 访问和配置端口 I/O 的特殊功能寄存器”）。

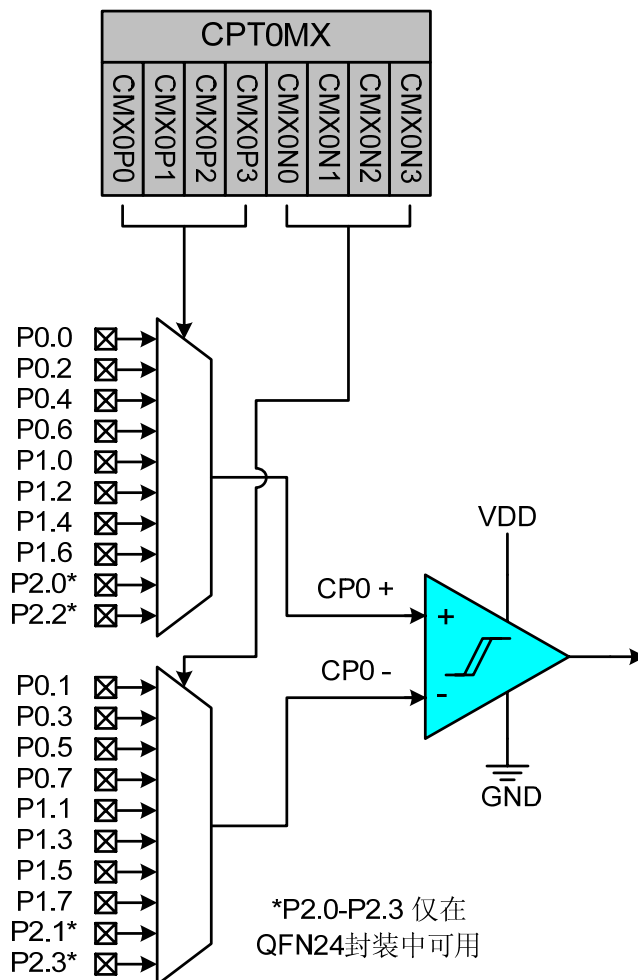


图 11.3 比较器输入多路器原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 11.3 CPT0MX: 比较器 0 MUX 选择寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
CMX0N3	CMX0N2	CMX0N1	CMX0N0	CMX0P3	CMX0P2	CMX0P1	CMX0P0	11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x9F
位 7-4: CMX0N[3:0]: 比较器 0 负输入 MUX 选择								
这些位选择作为比较器 0 负输入的端口引脚。								
0000		P0.1						
0001		P0.3						
0010		P0.5						
0011		P0.7						
0100		P1.1						
0101		P1.3						
0110		P1.5						
0111		P1.7						
1000		P2.1 (仅 C8051F368/9)						
1001		P2.3 (仅 C8051F368/9)						
1010-1111		无						
位 3-0: CMX0P[3:0]: 比较器 0 正输入 MUX 选择								
这些位选择作为比较器 0 正输入的端口引脚。								
0000		P0.0						
0001		P0.2						
0010		P0.4						
0011		P0.6						
0100		P1.0						
0101		P1.2						
0110		P1.4						
0111		P1.6						
1000		P2.0 (仅 C8051F368/9)						
1001		P2.2 (仅 C8051F368/9)						
1010-1111		无						

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

比较器 0 的输入用 CPT0MX 寄存器 (SFR 定义 11.2) 来选择。CMX0P1 和 CMX0P0 位选择比较器 0 的正输入；CMX0N1 和 CMX0N0 位选择比较器 0 的负输入。注意：被选择为比较器输入的引脚应被配置为模拟输入（通过端口配置寄存器），交叉开关应被配置为跳过这些引脚，有关端口配置的详细信息见“14.3 通用端口 I/O”。

12. CIP-51 微控制器

MCU 系统控制器的内核是 CIP-51 微控制器。CIP-51 与 MCS-51TM 指令集完全兼容，可以使用标准 803x/805x 的汇编器和编译器进行软件开发。该系列 MCU 具有标准 8051 的所有外设的一个超集。CIP-51 还包含片内调试硬件（详见第 26 章），以及与模拟和数字子系统的直接接口，在一个集成电路内提供了完整的数据采集或控制系统解决方案。

CIP-51 微控制器内核除了具有标准 8051 的组织结构和外设以外，另有增加的定制外设和功能，大大增强了它的处理能力（见图 12.1 的原理框图）。CIP-51 具有下列特点：

- 与 MCS-51 指令集完全兼容
- 在 25MHz 时钟时最大速度为 25MIPS
- 0~25MHz 的时钟频率
- 256 字节内部 RAM
- 扩展的中断处理系统
- 复位输入
- 电源管理方式
- 片内调试逻辑
- 程序和数据存储器安全

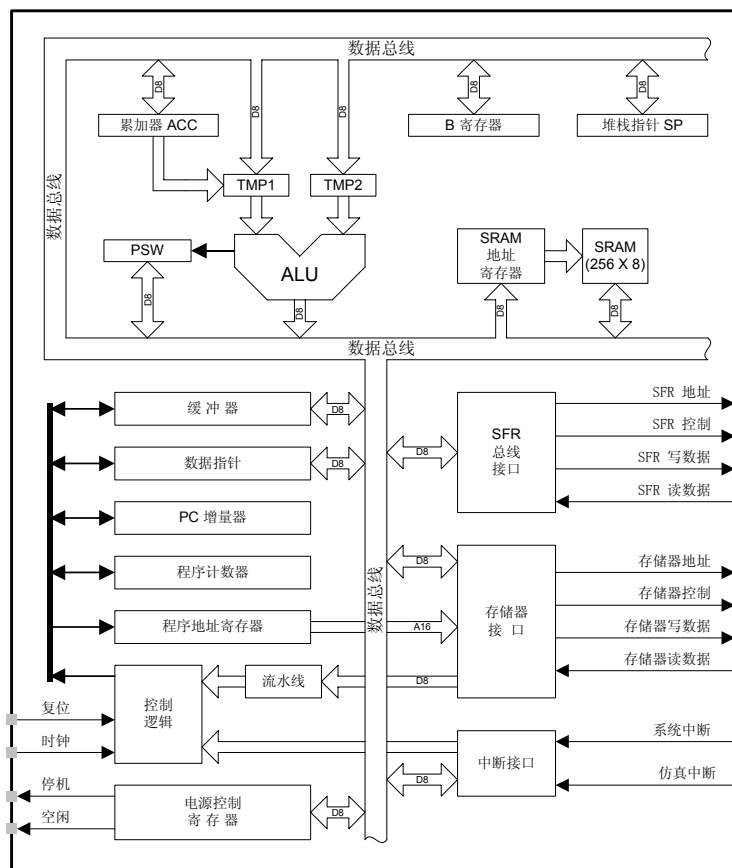


图 12.1 CIP-51 原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

性能

CIP-51采用流水线结构，与标准的8051结构相比指令执行速度有很大的提高。在一个标准的8051中，除MUL和DIV以外所有指令都需要12或24个系统时钟周期，并且通常最大系统时钟频率为12 MHz。而对于CIP-51内核，70%的指令的执行时间为1或2个系统时钟周期，没有执行时间超过8个系统时钟周期的指令。

CIP-51工作在最大系统时钟频率 25MHz 时，它的峰值速度达到 25MIPS。CIP-51 共有 111 条指令。下表列出了指令条数与执行时所需的系统时钟周期数的关系。

执行周期数	1	2	2/3	3	3/4	4	4/5	5	8
指令数	26	50	5	16	7	3	1	2	1

编程和调试支持

对 FLASH 程序存储器的在系统编程和与片内调试支持逻辑的通信是通过 Silicon Labs 2 线开发接口（C2）实现的。

片内调试逻辑支持全速的在系统调试，允许设置硬件断点，支持开始、停止和单步执行（包括中断服务程序）命令，支持检查程序调用堆栈及读/写寄存器和存储器。这种片内调试方法完全是非侵入式的，不需要占用 RAM、堆栈、定时器或其它片内资源。有关 C2 接口的详细信息见“26. C2 接口”。

CIP-51 有 Silicon Labs 和第三方供应商的开发工具支持。Silicon Labs 提供一个集成开发环境（IDE），包括编辑器、宏汇编器、调试器和编程器。IDE 的调试器和编程器与 CIP-51 之间通过 C2 接口连接，提供快速和有效的在系统编程和调试。也有第三方的宏汇编器和 C 编译器可用。

12.1 指令集

CIP-51 系统控制器的指令集与标准 MCS-51TM 指令集完全兼容，可以使用标准 8051 的开发工具开发 CIP-51 的软件。所有的 CIP-51 指令在二进制码和功能上与同类的 MCS-51TM 产品完全等价，包括操作码、寻址方式和对 PSW 标志的影响，但是指令时序与标准 8051 不同。

12.1.1 指令和 CPU 时序

在很多的 8051 实现中，机器周期和时钟周期是不同的，机器周期的长度在 2 到 12 个时钟周期之间。但是 CIP-51 只基于时钟周期，所有指令时序都以时钟周期计算。

由于 CIP-51 采用了流水线结构，大多数指令执行所需的时钟周期数与指令的字节数一致。条件转移指令在不发生转移时的执行周期数比发生转移时少一个。表 12.1 给出了所有 CIP-51 指令一览表，包括每条指令的助记符、字节数和时钟周期数。

表 12.1 CIP-51 指令集

助记符	功能说明	字节数	时钟周期数
算术操作类指令			
ADD A,Rn	寄存器加到累加器	1	1
ADD A,direct	直接寻址字节加到累加器	2	2
ADD A,@Ri	间址 RAM 内容加到累加器	1	2
ADD A,#data	立即数加到累加器	2	2
ADDC A,Rn	寄存器加到累加器(带进位)	1	1
ADDC A,direct	直接寻址字节加到累加器(带进位)	2	2
ADDC A,@Ri	间址 RAM 加到累加器(带进位)	1	2
ADDC A,#data	立即数加到累加器(带进位)	2	2
SUBB A,Rn	累加器减去寄存器(带借位)	1	1
SUBB A,direct	累加器减去直接寻址字节(带借位)	2	2
SUBB A,@Ri	累加器减去间址 RAM(带借位)	1	2
SUBB A,#data	累加器减去立即数(带借位)	2	2
INC A	累加器加 1	1	1
INC Rn	寄存器加 1	1	1
INC direct	直接寻址字节加 1	2	2
INC @Ri	间址 RAM 加 1	1	2
DEC A	累加器减 1	1	1
DEC Rn	寄存器减 1	1	1
DEC direct	直接寻址字节减 1	2	2
DEC @Ri	间址 RAM 减 1	1	2
INC DPTR	数据指针加 1	1	1
MUL AB	累加器与寄存器 B 相乘	1	4
DIV AB	累加器除以寄存器 B	1	8
DA A	累加器十进制调整	1	1
逻辑操作类指令			
ANL A,Rn	寄存器“与”到累加器	1	1
ANL A,direct	直接寻址字节“与”到累加器	2	2
ANL A,@Ri	间址 RAM “与”到累加器	1	2
ANL A,#data	立即数“与”到累加器	2	2
ANL direct,A	累加器“与”到直接寻址字节	2	2
ANL direct,#data	立即数“与”到直接寻址字节	3	3
ORL A,Rn	寄存器“或”到累加器	1	1
ORL A,direct	直接寻址字节“或”到累加器	2	2
ORL A,@Ri	间址 RAM “或”到累加器	1	2
ORL A,#data	立即数“或”到累加器	2	2
ORL direct,A	累加器“或”到直接寻址字节	2	2
ORL direct,#data	立即数“或”到直接寻址字节	3	3
XRL A,Rn	寄存器“异或”到累加器	1	1
XRL A,direct	直接寻址字节“异或”到累加器	2	2
XRL A,@Ri	间址 RAM “异或”到累加器	1	2
XRL A,#data	立即数“异或”到累加器	2	2
XRL direct,A	累加器“异或”到直接寻址字节	2	2
XRL direct,#data	立即数“异或”到直接寻址字节	3	3
CLR A	累加器清零	1	1
CPL A	累加器求反	1	1
RL A	累加器循环左移	1	1
RLC A	带进位的累加器循环左移	1	1

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

助记符	功能说明	字节数	时钟周期数
RR A	累加器循环右移	1	1
RRC A	带进位的累加器循环右移	1	1
SWAP A	累加器内高低半字节交换	1	1
数据传送类指令			
MOV A,Rn	寄存器传送到累加器	1	1
MOV A,direct	直接寻址字节传送到累加器	2	2
MOV A,@Ri	间址 RAM 传送到累加器	1	2
MOV A,#data	立即数传送到累加器	2	2
MOV Rn,A	累加器传送到寄存器	1	1
MOV Rn,direct	直接寻址字节传送到寄存器	2	2
MOV Rn,#data	立即数传送到寄存器	2	2
MOV direct,A	累加器传送到直接寻址字节	2	2
MOV direct,Rn	寄存器传送到直接寻址字节	2	2
MOV direct,direct	直接寻址字节传送到直接寻址字节	3	3
MOV direct,@Ri	间址 RAM 传送到直接寻址字节	2	2
MOV direct,#data	立即数传送到直接寻址字节	3	3
MOV @Ri,A	累加器传送到间址 RAM	1	2
MOV @Ri,direct	直接寻址字节传送到间址 RAM	2	2
MOV @Ri,#data	立即数传送到间址 RAM	2	2
MOV DPTR,#data16	16 位常数装入 DPTR	3	3
MOVC A,@A+DPTR	相对于 DPTR 的代码字节传送到累加器	1	3
MOVC A,@A+PC	相对于 PC 的代码字节传送到累加器	1	3
MOVX A,@Ri	外部 RAM(8 位地址)传送到累加器	1	3
MOVX @Ri,A	累加器传到外部 RAM (8 位地址)	1	3
MOVX A,@DPTR	外部 RAM(16 位地址)传送到累加器	1	3
MOVX @DPTR,A	累加器传到外部 RAM (16 位地址)	1	3
PUSH direct	直接寻址字节压入栈顶	2	2
POP direct	栈顶数据弹出到直接寻址字节	2	2
XCH A,Rn	寄存器和累加器交换	1	1
XCH A,direct	直接寻址字节与累加器交换	2	2
XCH A,@Ri	间址 RAM 与累加器交换	1	2
XCHD A,@Ri	间址 RAM 和累加器交换低半字节	1	2
位操作类指令			
CLR C	清进位位	1	1
CLR bit	清直接寻址位	2	2
SETB C	进位位置 1	1	1
SETB bit	直接寻址位置位	2	2
CPL C	进位位取反	1	1
CPL bit	直接寻址位取反	2	2
ANL C,bit	直接寻址位“与”到进位位	2	2
ANL C,/bit	直接寻址位的反码“与”到进位位	2	2
ORL C,bit	直接寻址位“或”到进位位	2	2
ORL C,/bit	直接寻址位的反码“或”到进位位	2	2
MOV C,bit	直接寻址位传送到进位位	2	2
MOV bit,C	进位位传送到直接寻址位	2	2
JC rel	若进位位为 1 则跳转	2	2/3
JNC rel	若进位位为零则跳转	2	2/3
JB bit,rel	若直接寻址位为 1 则跳转	3	3/4
JNB bit,rel	若直接寻址位为零则跳转	3	3/4
JBC bit,rel	若直接寻址位为 1 则跳转，并清除该位	3	3/4

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

助记符	功能说明	字节数	时钟 周期数
控制转移类指令			
ACALL addr11	绝对调用子程序	2	3
LCALL addr16	长调用子程序	3	4
RET	从子程序返回	1	5
RETI	从中断返回	1	5
AJMP addr11	绝对转移	2	3
LJMP addr16	长转移	3	4
SJMP rel	短转移（相对地址）	2	3
JMP @A+DPTR	相对 DPTR 的间接转移	1	3
JZ rel	累加器为 0 则转移	2	2/3
JNZ rel	累加器为非 0 则转移	2	2/3
CJNE A,direct,rel	比较直接寻址字节与累加器，不相等则转移	3	3/4
CJNE A,#data,rel	比较立即数与累加器，不相等则转移	3	3/4
CJNE Rn,#data,rel	比较立即数与寄存器，不相等则转移	3	3/4
CJNE @Ri,#data,rel	比较立即数与间接寻址 RAM，不相等则转移	3	4/5
DJNZ Rn,rel	寄存器减 1，不为零则转移	2	2/3
DJNZ direct,rel	直接寻址字节减 1，不为零则转移	3	3/4
NOP	空操作	1	1

寄存器、操作数和寻址方式说明：

Rn – 当前选择的寄存器区的寄存器 R0-R7。

@Ri – 通过寄存器 R0 或 R1 间接寻址的数据 RAM 地址。

rel – 相对于下一条指令第一个字节的 8 位有符号（2 的补码）偏移量。SJMP 和所有条件转移指令使用。

direct – 8 位内部数据存储器地址。可以是直接访问数据 RAM 地址（0x00-0x7F）或一个 SFR 地址（0x80-0xFF）。

#data – 8 位常数

#data16 – 16 位常数

bit – 数据 RAM 或 SFR 中的直接寻址位

addr11 – ACALL 和 AJMP 使用的 11 位目的地址。目的地址必须与下一条指令第一个字节处于同一个 2K 字节的程序存储器页。

addr16 – LCALL 和 LJMP 使用的 16 位目的地址。目的地址可以是 8K 程序存储器空间内的任何位置。

有一个未使用的操作码（0xA5），它执行与 NOP 指令相同的功能。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

12.2 CIP-51 寄存器说明

下面对与 CIP-51 系统控制器操作有关的 SFR 加以说明。保留位不应被置为逻辑 1。将来的产品版本可能会使用这些位实现新功能，在这种情况下各位的复位值将是逻辑 0 以选择缺省状态。有关其它 SFR 的详细说明见本数据表中与它们对应的系统功能相关的章节。

SFR 定义 12.1 DPL：数据指针低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0x82
<p>位 7-0: DPL：数据指针低字节 DPL 为 16 位数据指针（DPTR）的低字节。DPTR 用于访问间接寻址 FLASH 存储器或 XRAM。</p>								

SFR 定义 12.2 DPH：数据指针高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0x83
<p>位 7-0: DPH：数据指针高字节 DPH 为 16 位数据指针（DPTR）的高字节。DPTR 用于访问间接寻址的 FLASH 存储器或 XRAM。</p>								

SFR 定义 12.3 SP：堆栈指针

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0x81
<p>位 7-0: SP：堆栈指针 堆栈指针保持栈顶位置。在每次执行 PUSH 操作前，堆栈指针加 1。SP 寄存器复位后的默认值为 0x07。</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 12.4 ACC: 累加器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
ACC.7	ACC.6	ACC.5	ACC.4	ACC.3	ACC.2	ACC.1	ACC.0	00000000
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xE0
<p>位 7-0: ACC: 累加器 该寄存器为算术操作的累加器。</p>								

SFR 定义 12.5 B: B 寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
B.7	B.6	B.5	B.4	B.3	B.2	B.1	B.0	00000000
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xF0
<p>位 7-0: B: B 寄存器 该寄存器为某些算术操作的第二累加器。</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 12.7 PSW：程序状态字

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	复位值
CY	AC	F0	RS1	RS0	OV	F1	PARITY	00000000
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xD0

位 7: CY: 进位标志。
当最后一次算术操作产生进位（加法）或借位（减法）时，该位置 1。其它算术操作将其清 0。

位 6: AC: 辅助进位标志。
当最后一次算术操作向高半字节有进位（加法）或借位（减法）时，该位置 1。其它算术操作将其清 0。

位 5: F0: 用户标志 0。
这是一个可位寻址、受软件控制的通用标志位。

位 4-3: RS1-RS0: 寄存器区选择。
该两位在寄存器访问时用于选择寄存器区。

RS1	RS0	寄存器区	地址
0	0	0	0x00-0x07
0	1	1	0x08-0x0F
1	0	2	0x10-0x17
1	1	3	0x18-0x1F

位 2: OV: 溢出标志。
该位在下列情况下被置 1:

- ADD、ADDC 或 SUBB 指令引起符号位变化溢出。
- MUL 指令引起溢出（结果大于 255）。
- DIV 指令的除数为 0。

ADD、ADDC、SUBB、MUL 和 DIV 指令的其它情况使该位清 0。

位 1: F1: 用户标志 1。
这是一个可位寻址、受软件控制的通用标志位。

位 0: PARITY: 奇偶标志。
若累加器中 8 个位的和为奇数时该位置 1，为偶数时清 0。

13. 存储器组织

CIP-51 系统控制器的存储器组织与标准 8051 的存储器组织类似。有两个独立的存储器空间：程序存储器和数据存储器。程序和数据存储器共享同一个地址空间，但用不同的指令类型访问。CIP-51 的存储器组织如图 13.1 所示。

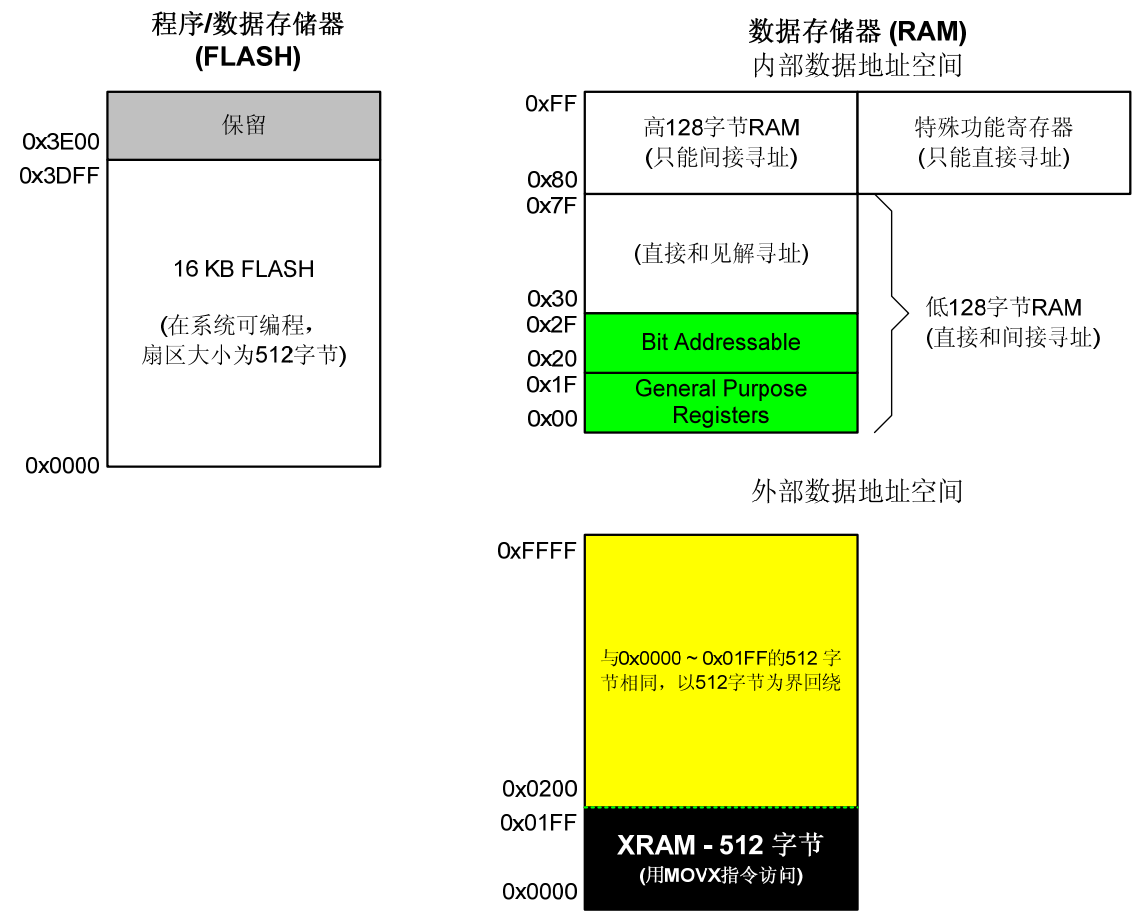


图 13.1 C8051F336/7/8/9 存储器组织

13.1 程序存储器

CIP-51 有 64KB 的程序存储器空间。C8051F336/7/8/9 在这个程序存储器空间中实现了 16KB 的可在系统编程的 FLASH 存储器，组织在一个连续的存储块内（0x0000 ~ 0x3DFF）。地址 0x3DFF 为器件的锁定字节，0x3DFF 以上的地址被保留。

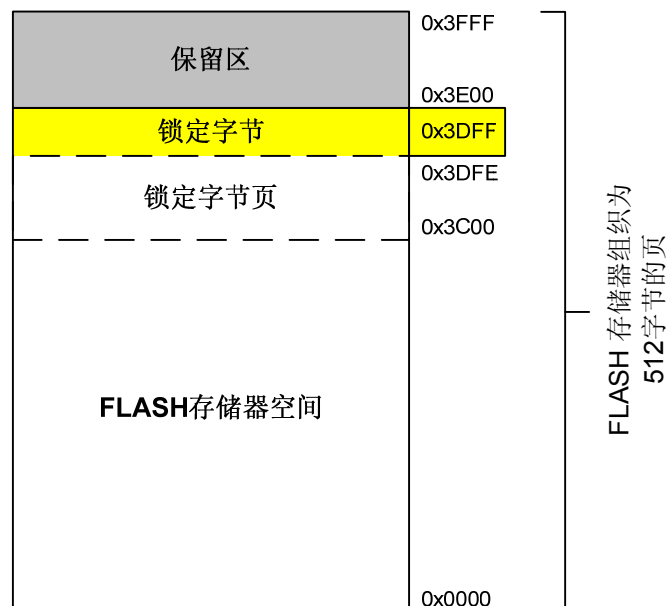


图 13.2 FLASH 程序存储器结构图

13.1.1 MOVX 指令和程序存储器

8051 器件的 MOVX 指令通常用于访问外部数据存储器。在 C8051F336/7/8/9 器件中，MOVX 指令一般用于读和写片内 XRAM，但也可以被配置为写和擦除片内 FLASH 存储器空间。MOVC 指令总是用于读 FLASH 存储器，MOVX 指令用于擦除和写 FLASH 存储器。这一 FLASH 访问特性为 C8051F336/7/8/9 提供了由用户程序更新程序代码和将程序存储器空间用于非易失性数据存储的机制，详见“16. FLASH 存储器”。

13.2 数据存储器

C8051F336/7/8/9 内部有 768 字节的 RAM 数据存储器。该存储器中有 256 字节被映射到 8051 的内部 RAM 空间，另外 512 字节是片内“外部”存储器（XRAM）。图 13.1 给出了数据存储器组织图。

13.2.1 内部 RAM

CIP-51 的数据存储器空间中有 256 字节的内部 RAM，位于地址 0x00 到 0xFF 的地址空间。数据存储器中的低 128 字节用于通用寄存器和临时存储器。可以用直接或间接寻址方式访问数据存储器中的低 128 字节。从 0x00 到 0x1F 为 4 个通用寄存器区，每个区有 8 个 8 位寄存器。接下来的 16 字节，从地址 0x20 到 0x2F，既可以按字节寻址又可以作为 128 个位地址用直接寻址方式访问。

数据存储器中的高 128 字节只能用间接寻址访问。该存储区与特殊功能寄存器（SFR）占据相同的地址空间，但物理上与 SFR 空间是分开的。当寻址高于 0x7F 的地址时，指令所用的寻址方式决定了 CPU 是访问数据存储器的高 128 字节还是访问 SFR。使用直接寻址方式的指令将访问 SFR 空间，间接寻址高于 0x7F 地址的指令将访问数据存储器的高 128 字节。图 13.1 给出了 C8051F336/7/8/9 数据存储器组织的示意图。

13.2.1.1 通用寄存器

数据存储器的低 32 字节，从地址 0x00 到 0x1F，可以作为 4 个通用寄存器区访问。每个区有 8 个 8 位寄存器，称为 R0 - R7。在某一时刻只能选择一个寄存器区。程序状态字中的 RS0（PSW.3）和 RS1（PSW.4）位用于选择当前的寄存器区（见 SFR 定义 12.6 中 PSW 的说明）。这允许在进入子程序或中断服务程序时进行快速现场切换。间接寻址方式使用 R0 和 R1 作为间址寄存器。

13.2.1.2 位寻址空间

除了直接访问按字节组织的数据存储器外，从 0x20 到 0x2F 的 16 个数据存储器单元还可以作为 128 个独立寻址位访问。每个位有一个位地址，从 0x00 到 0x7F。位于地址 0x20 的数据字节的位 0 具有位地址 0x00，位于 0x20 的数据字节的位 7 具有位地址 0x07。位于 0x2F 的数据字节的位 7 具有位地址 0x7F。由所用指令的类型来区分是位寻址还是字节寻址。

MCS-51TM 汇编语言允许用 XX.B 的形式替代位地址，XX 为字节地址，B 为寻址位在字节中的位置。例如，指令：

```
MOV    C,  22.3h
```

将 0x13 中的布尔值（字节地址 0x22 中的位 3）传送到进位标志。

13.2.1.3 堆栈

程序的堆栈可以位于 256 字节数据存储器中的任何位置。堆栈区域用堆栈指针（SP，0x81）SFR 指定。SP 指向最后使用的位置。下一个压入堆栈的数据将被存放在 SP+1，然后 SP 加 1。复位后堆栈指针被初始化为地址 0x07，因此第一个被压入堆栈的数据将被存放在地址 0x08，这也是寄存器区 1 的第一个寄存器（R0）。如果使用不止一个寄存器区，SP 应被初始化为数据存储器中不用于数据存储的位置。堆栈深度最大可达 256 字节。

13.2.2 外部 RAM

C8051F336/7/8/9 器件内部有映射到外部数据存储器空间的 512 字节 RAM。可以用外部传送指令（MOVX）和数据指针（DPTR）访问这些地址单元，或者通过使用 R0 或 R1 用间接寻址方式访问。如果 MOVX 指令使用一个 8 位地址操作数（例如 @R1），则 16 位地址的高字节由外部存储器接口控制寄存器（EMI0CN，如 SFR 定义 13.1 所示）提供。**注：MOVX 指令还用于写 FLASH 存储器，详见第 11 章。缺省情况下 MOVX 指令访问 XRAM。**

对于 16 位 MOVX 操作（@DPTR），16 位外部数据存储器地址的高 7 位是被“忽略”的。因此，这个 512 字节的 RAM 以取模的方式映射到整个 64KB 的外部数据存储器地址范围。例如，位于地址 0x0000 的 XRAM 字节也位于 0x0200、0x0400、0x0600、0x0800 等地址。在进行线性存储器填充时这是一个很有用的特性，因为在达到 RAM 块的边界时不必对地址指针复位。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 13.1 EMI0CN: 外部存储器接口控制

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-	-	-	-	-	-	-	PGSEL	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xAA
<p>位 7-1: 未用—读为 0000000b, 写 = 忽略。</p> <p>位 0: PGSEL: XRAM 页选择位</p> <p>当使用 8 位的 MOVX 命令时, XRAM 页选择位提供 16 位外部数据存储器地址的高字节, 实际上是选择一个 256 字节的 RAM 页。由于该寄存器的高位(未使用)总是为 0, 所以 PGSEL 决定要访问的 XRAM 页。</p> <p>例如: 如果 EMI0CN = 0x01, 则访问的地址范围是 0x0100 ~ 0x01FF。</p>								

14. 特殊功能寄存器

从 0x80 到 0xFF 的直接寻址存储器空间为特殊功能寄存器（SFR）。SFR 提供对 C8051F336/7/8/9 的资源和外设的控制及 CIP-51 与这些资源和外设之间的数据交换。CIP-51 具有标准 8051 中的全部 SFR，还增加了一些用于配置和访问 C8051F336/7/8/9 专有子系统的 SFR。这就允许在保证与 MCS-51™ 指令集兼容的前提下增加新的功能。表 14.1 列出了 C8051F336/7/8/9 器件中的全部 SFR。

任何时刻用直接寻址方式访问 0x80~0xFF 的存储器空间将访问特殊功能寄存器（SFR）。地址以 0x0 或 0x8 结尾的 SFR（例如 P0、TCON、P1、SCON0、IE 等）既可以按字节寻址也可以按位寻址，所有其它 SFR 只能按字节寻址。SFR 空间中未使用的地址保留为将来使用，访问这些地址会产生不确定的结果，应予避免。有关每个寄存器的详细说明请参见本数据表的相关部分（表 14.2 中已标明）。

表 14.1 特殊功能寄存器存储器映象

F8	SPI0CN	PCA0L	PCA0H	PCA0CPL0	PCA0CPH0	P0MAT	P0MASK	VDM0CN
F0	B	P0MDIN	P1MDIN	P2MDIN			EIP1	PCA0PWM
E8	ADC0CN	PCA0CPL1	PCA0CPH1	PCA0CPL2	PCA0CPH2	P1MAT	P1MASK	RSTSRC
E0	ACC	XBR0	XBR1	OSCLCN	IT01CF		EIE1	SMB0ADM
D8	PCA0CN	PCA0MD	PCA0CPM0	PCA0CPM1	PCA0CPM2			
D0	PSW	REF0CN			P0SKIP	P1SKIP	P2SKIP	SMB0ADR
C8	TMR2CN		TMR2RLL	TMR2RLH	TMR2L	TMR2H		
C0	SMB0CN	SMB0CF	SMB0DAT	ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH	
B8	IP	IDA0CN	AMX0N	AMX0P	ADC0CF	ADC0L	ADC0H	
B0		OSCXCN	OSCICN	OSCICL			FLSCL	FLKEY
A8	IE	CLKSEL	EMI0CN					
A0	P2	SPI0CFG	SPI0CKR	SPI0DAT	P0MDOUT	P1MDOUT	P2MDOUT	
98	SCON0	SBUF0		CPT0CN		CPT0MD		CPT0MX
90	P1	TMR3CN	TMR3RLL	TMR3RLH	TMR3L	TMR3H	IDA0L	IDA0H
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH				PCON
	0(8) 可位寻址	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 14.2 特殊功能寄存器

SFR 以字母顺序排列，所有未定义的 SFR 位置保留。

寄存器	地址	说 明	页码
ACC	0xE0	累加器	
ADC0CF	0xBC	ADC0 配置寄存器	
ADC0CN	0xE8	ADC0 控制寄存器	
ADC0GTH	0xC4	ADC0 下限（大于）比较字高字节	
ADC0GTL	0xC3	ADC0 下限（大于）比较字低字节	
ADC0H	0xBE	ADC0 数据字高字节	
ADC0L	0xBD	ADC0 数据字低字节	
ADC0LTH	0xC6	ADC0 上限（小于）比较字高字节	
ADC0LTL	0xC5	ADC0 上限（小于）比较字低字节	
AMX0N	0xBA	AMUX0 负通道选择寄存器	
AMX0P	0xBB	AMUX0 正通道选择寄存器	
B	0xF0	B 寄存器	
CKCON	0x8E	时钟控制寄存器	
CLKSEL	0xA9	时钟选择寄存器	
CPT0CN	0x9B	比较器 0 控制寄存器	
CPT0MD	0x9D	比较器 0 方式选择寄存器	
CPT0MX	0x9F	比较器 0 MUX 选择寄存器	
DPH	0x83	数据指针高字节	
DPL	0x82	数据指针低字节	
EIE1	0xE6	扩展中断允许寄存器 1	
EIP1	0xF6	扩展中断优先级寄存器 1	
EMI0CN	0xAA	外部存储器接口控制寄存器	
FLKEY	0xB7	FLASH 锁定和关键码寄存器	
FLSCL	0xB6	FLASH 存储器时序预分频器	
IDA0CN	0xB9	电流模式 DAC0 控制寄存器	
IDA0H	0x97	电流模式 DAC0 数据字高字节	
IDA0L	0x96	电流模式 DAC0 数据字低字节	
IE	0xA8	中断允许寄存器	
IP	0xB8	中断优先级寄存器	
IT01CF	0xE4	INT0/INT1 配置寄存器	
OSCICL	0xB3	内部振荡器校准寄存器	
OSICN	0xB2	内部振荡器控制寄存器	
OSCLCN	0xE3	低频振荡器控制寄存器	
OSCXCN	0xB1	外部振荡器控制寄存器	
P0	0x80	端口 0 锁存器	
P0MASK	0xFE	端口 0 屏蔽配置寄存器	
P0MAT	0xFD	端口 0 匹配配置寄存器	
P0MDIN	0xF1	端口 0 输入方式配置寄存器	
P0MDOUT	0xA4	端口 0 输出方式配置寄存器	
P0SKIP	0xD4	端口 0 跳过寄存器	

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 12.3 特殊功能寄存器（续）

SFR 以字母顺序排列，所有未定义的 SFR 位置保留。

寄存器	地址	说 明	页码
P1	0x90	端口 1 锁存器	
P1MASK	0xEE	端口 1 屏蔽配置寄存器	
P1MAT	0xED	端口 1 匹配配置寄存器	
P1MDIN	0xF2	端口 1 输入方式配置寄存器	
P1MDOUT	0xA5	端口 1 输出方式配置寄存器	
P1SKIP	0xD5	端口 1 跳过寄存器	
P2	0xA0	端口 2 锁存器	
P2MDIN	0xF3	端口 2 输入方式配置寄存器	
P2MDOUT	0xA6	端口 2 输出方式配置寄存器	
P2SKIP	0xD6	端口 2 跳过寄存器	
PCA0CN	0xD8	PCA 控制寄存器	
PCA0CPH0	0xFC	PCA 捕捉模块 0 高字节	
PCA0CPH1	0xEA	PCA 捕捉模块 1 高字节	
PCA0CPH2	0xEC	PCA 捕捉模块 2 高字节	
PCA0CPL0	0xFB	PCA 捕捉模块 0 低字节	
PCA0CPL1	0xE9	PCA 捕捉模块 1 低字节	
PCA0CPL2	0xEB	PCA 捕捉模块 2 低字节	
PCA0CPM0	0xDA	PCA 模块 0 方式寄存器	
PCA0CPM1	0xDB	PCA 模块 1 方式寄存器	
PCA0CPM2	0xDC	PCA 模块 2 方式寄存器	
PCA0H	0xFA	PCA 计数器高字节	
PCA0L	0xF9	PCA 计数器低字节	
PCA0MD	0xD9	PCA 方式寄存器	
PCA0PWN	0xF7	PCA PWM 配置寄存器	
PCON	0x87	电源控制寄存器	
PSCTL	0x8F	程序存储读/写控制寄存器	
PSW	0xD0	程序状态字	
REF0CN	0xD1	电压基准控制寄存器	
RSTSRC	0xEF	复位源寄存器	
SBUF0	0x99	UART0 数据缓冲器	
SCON0	0x98	UART0 控制寄存器	
SMB0ADM	0xE7	SMBus 从地址掩码寄存器	
SMB0ADR	0xD7	SMBus 从地址寄存器	
SMB0CF	0xC1	SMBus 配置寄存器	
SMB0CN	0xC0	SMBus 控制寄存器	
SMB0DAT	0xC2	SMBus 数据寄存器	
SP	0x81	堆栈指针	
SPI0CFG	0xA1	SPI 配置寄存器	
SPI0CKR	0xA2	SPI 时钟频率控制寄存器	
SPI0CN	0xF8	SPI 控制寄存器	
SPI0DAT	0xA3	SPI 数据寄存器	

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 12.3 特殊功能寄存器（续）

SFR 以字母顺序排列，所有未定义的 SFR 位置保留。

寄存器	地址	说 明	页码
TCON	0x88	计数器/定时器控制寄存器	
TH0	0x8C	计数器/定时器 0 高字节	
TH1	0x8D	计数器/定时器 1 高字节	
TL0	0x8A	计数器/定时器 0 低字节	
TL1	0x8B	计数器/定时器 1 低字节	
TMOD	0x89	计数器/定时器方式寄存器	
TMR2CN	0xC8	计数器/定时器 2 控制寄存器	
TMR2H	0xCD	计数器/定时器 2 高字节	
TMR2L	0xCC	计数器/定时器 2 低字节	
TMR2RLH	0xCB	计数器/定时器 2 重载值高字节	
TMR2RLL	0xCA	计数器/定时器 2 重载值低字节	
TMR3CN	0x91	计数器/定时器 3 控制寄存器	
TMR3H	0x95	计数器/定时器 3 高字节	
TMR3L	0x94	计数器/定时器 3 低字节	
TMR3RLH	0x93	计数器/定时器 3 重载值高字节	
TMR3RLL	0x92	计数器/定时器 3 重载值低字节	
VDM0CN	0xFF	VDD 监视器控制寄存器	
XBR0	0xE1	端口 I/O 交叉开关控制 0	
XBR1	0xE2	端口 I/O 交叉开关控制 1	

15. 中断

C8051F336/7/8/9 包含一个扩展的中断系统,支持 14 个中断源,每个中断源有两个优先级。中断源在片内外设与外部输入引脚之间的分配随器件的不同而变化。每个中断源可以在一个 SFR 中有一个或多个中断标志。当一个外设或外部源满足有效的中断条件时,相应的中断标志被置为逻辑 1。

如果一个中断源被允许,则在中断标志被置位时将产生一个中断请求。一旦当前指令执行完,CPU 产生一个 LCALL 到预定地址,开始执行中断服务程序 (ISR)。每个 ISR 必须以 RETI 指令结束,使程序回到中断前执行的那条指令的下一条指令。如果中断未被允许,中断标志将被硬件忽略,程序继续正常执行。中断标志置 1 与否不受中断允许/禁止状态的影响。

每个中断源都可以用一个 SFR (IE – EIE1) 中的相关中断允许位来允许或禁止,但是必须首先将 EA 位 (IE.7) 置 1,以保证每个单独的中断允许位有效。不管每个中断允许位的设置如何,清 0 EA 位将禁止所有中断。在 EA 位被清 0 期间所发生的中断请求被挂起,直到 EA 位被置 1 后才能得到服务。

注意: 任何使 EA 位清 0 的指令的后面都应跟随一条具有两个或更多操作码字节的指令。
例如:

```
// 用 C 语言编程
EA = 0; // 清 '0' EA 位
EA = 0; // 这是一条两字节操作码的假指令

; 用汇编语言编程
CLR    EA    ; 清 '0' EA 位
CLR    EA    ; 这是一条两字节操作码的假指令
```

如果一个中断在“CLR EA”指令(或任何使 EA 清 0 的指令)的执行阶段有效,并且该指令的后面是一条单周期的指令,则该中断可能被响应,但在中断服务程序中读 EA 位将返回 0 值。如果“CLR EA”指令后面是一条多周期的指令,则中断不会被响应。

某些中断标志在 CPU 进入 ISR 时被硬件自动清除,但大多数中断标志不是由硬件清除的,必须在 ISR 返回前用软件清除。如果一个中断标志在 CPU 执行完中断返回 (RETI) 指令后仍然保持置位状态,则会立即产生一个新的中断请求,CPU 将在执行完下一条指令后再次进入该 ISR。

15.1 MCU 中断源和中断向量

C8051F336/7/8/9 MCU 支持 14 个中断源。软件可以通过将任何一个中断标志设置为逻辑 1 来模拟一个中断。如果中断标志被允许，系统将产生一个中断请求，CPU 将转向与该中断标志对应的 ISR 地址。表 15.1 给出了 MCU 中断源、对应的向量地址、优先级和控制位一览表。关于外设有有效中断条件和中断标志位工作状态方面的详细信息，请见与特定外设相关的章节。

15.1.1 中断优先级

每个中断源都可以被独立地被编程为两个优先级中的一个：低优先级或高优先级。一个低优先级的中断服务程序可以被高优先级的中断所中断，但高优先级的中断不能被中断。每个中断在 SFR（IP 或 EIP1）中都有一个配置其优先级的中断优先级设置位，缺省值为低优先级。如果两个中断同时发生，具有高优先级的中断先得到服务。如果这两个中断的优先级相同，则由固定的优先级顺序决定哪一个中断先得到服务（见表 15.1）。

15.1.2 中断响应时间

中断响应时间取决于中断发生时 CPU 的状态。中断系统在每个系统时钟周期对中断标志采样并对优先级译码。最快的响应时间为 5 个系统时钟周期：一个周期用于检测中断，4 个周期完成对 ISR 的长调用（LCALL）。如果中断标志有效时 CPU 正在执行 RETI 指令，则需要再执行一条指令才能进入中断服务程序。因此，最长的中断响应时间（没有其它中断正被服务或新中断具有较高优先级）发生在 CPU 正在执行 RETI 指令，而下一条指令是 DIV 的情况。在这种情况下，响应时间为 18 个系统时钟周期：1 个时钟周期检测中断，5 个时钟周期执行 RETI，8 个时钟周期完成 DIV 指令，4 个时钟周期执行对 ISR 的长调用（LCALL）。如果 CPU 正在执行一个具有相同或更高优先级的中断的 ISR，则新中断要等到当前 ISR 执行完（包括 RETI 和下一条指令）才能得到服务。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 15.1 中断一览表

中断源	中断向量	优先级	中断标志	位寻址	硬件清除	中断允许	优先级控制
复位	0x0000	最高	无	N/A	N/A	始终允许	总是最高
外部中断 0 (/INT0)	0x0003	0	IE0 (TCON.1)	Y	Y	EX0 (IE.0)	PX0 (IP.0)
定时器 0 溢出	0x000B	1	TF0 (TCON.5)	Y	Y	ET0 (IE.1)	PT0 (IP.1)
外部中断 1 (/INT1)	0x0013	2	IE1 (TCON.3)	Y	Y	EX1 (IE.2)	PX1 (IP.2)
定时器 1 溢出	0x001B	3	TF1 (TCON.7)	Y	Y	ET1 (IE.3)	PT1 (IP.3)
UART0	0x0023	4	RI0 (SCON0.0) TI0 (SCON0.1)	Y	N	ES0 (IE.4)	PS0 (IP.4)
定时器 2 溢出	0x002B	5	TF2H (TMR2CN.7) TF2L (TMR2CN.6)	Y	N	ET2 (IE.5)	PT2 (IP.5)
SPI0	0x0033	6	SPIF (SPI0CN.7) WCOL (SPI0CN.6) MODF (SPI0CN.5) RXOVRN(SPI0CN.4)	Y	N	ESPI0 (IE.6)	PSPI0 (IP.6)
SMB0	0x003B	7	SI (SMB0CN.0)	Y	N	ESMB0 (EIE1.0)	PSMB0 (EIP1.0)
端口匹配	0x0043	8	无	N/A	N/A	EMAT (EIE1.1)	PMAT (EIP1.1)
ADC0 窗口比较	0x004B	9	AD0WINT (ADC0CN.3)	Y	N	EWADC0 (EIE1.2)	PWADC0 (EIP1.2)
ADC0 转换结束	0x0053	10	AD0INT (ADC0CN.5)	Y	N	EADC0C (EIE1.3)	PADC0 (EIP1.3)
可编程计数器阵列	0x005B	11	CF (PCA0CN.7) CCFn (PCA0CN.n)	Y	N	EPCA0 (EIE1.4)	PPCA0 (EIP1.4)
比较器 0	0x0063	12	CP0FIF(CPT0CN.4) CP0RIF(CPT0CN.5)	N	N	ECP0 (EIE1.5)	PCP0 (EIP1.5)
保留	0x006B	13	N/A	N/A	N/A	N/A	N/A
定时器 3 溢出	0x0073	14	TF3H(TMR3CN.7) TF3L(TMR3CN.6)	N	N	ET3 (EIE1.7)	PT3 (EIP1.7)

15.2 中断寄存器说明

下面介绍用于允许中断源和设置中断优先级的特殊功能寄存器。关于外设有效中断条件和中断标志位工作状态方面的详细信息，请见与特定片内外设相关的章节。

SFR 定义 15.1 IE: 中断允许寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
EA	ESPI0	ET2	ES0	ET1	EX1	ET0	EX0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xA8
(可位寻址)								
<p>位 7: EA: 允许所有中断。 该位允许 / 禁止所有中断。它超越所有的单个中断屏蔽设置。 0: 禁止所有中断源。 1: 开放中断。每个中断由它对应的中断屏蔽设置决定。</p> <p>位 6: ESPI0: 串行外设接口 (SPI0) 中断允许位。 该位用于设置 SPI0 的中断屏蔽。 0: 禁止 SPI0 中断。 1: 允许 SPI0 的中断请求。</p> <p>位 5: ET2: 定时器 2 中断允许位。 该位用于设置定时器 2 的中断屏蔽。 0: 禁止定时器 2 中断。 1: 允许 TF2L 或 TF2H 标志的中断请求。</p> <p>位 4: ES0: UART0 中断允许位。 该位设置 UART0 的中断屏蔽。 0: 禁止 UART0 中断。 1: 允许 UART0 中断。</p> <p>位 3: ET1: 定时器 1 中断允许位。 该位用于设置定时器 1 的中断屏蔽。 0: 禁止定时器 1 中断。 1: 允许 TF1 标志位的中断请求。</p> <p>位 2: EX1: 外部中断 1 允许位。 该位用于设置外部中断 1 的中断屏蔽。 0: 禁止外部中断 1。 1: 允许/INT1 引脚的中断请求</p> <p>位 1: ET0: 定时器 0 中断允许位。 该位用于设置定时器 0 的中断屏蔽。 0: 禁止定时器 0 中断。 1: 允许 TF0 标志位的中断请求。</p> <p>位 0: EX0: 外部中断 0 允许位。 该位用于设置外部中断 0 的中断屏蔽。 0: 禁止外部中断 0。 1: 允许/INT0 引脚的中断请求</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 15.2 IP: 中断优先级寄存器

R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-	PSPI0	PT2	PS0	PT1	PX1	PT0	PX0	10000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xB8
(可位寻址)								
位 7: 未用。读=1b, 写=忽略。								
位 6: PSPI0: 串行外设接口 (SPI0) 中断优先级控制 该位设置 SPI0 中断的优先级。 0: SPI0 为低优先级。 1: SPI0 为高优先级。								
位 5: PT2: 定时器 2 中断优先级控制 该位设置定时器 2 中断的优先级。 0: 定时器 2 为低优先级。 1: 定时器 2 为高优先级。								
位 4: PS0: UART0 中断优先级控制。 该位设置 UART0 中断的优先级。 0: UART0 为低优先级。 1: UART0 为高优先级。								
位 3: PT1: 定时器 1 中断优先级控制 该位设置定时器 1 中断的优先级。 0: 定时器 1 为低优先级。 1: 定时器 1 为高优先级。								
位 2: PX1: 外部中断 1 优先级控制 该位设置外部中断 1 的优先级。 0: 外部中断 1 为低优先级。 1: 外部中断 1 为高优先级。								
位 1: PT0: 定时器 0 中断优先级控制 该位设置定时器 0 中断的优先级。 0: 定时器 0 为低优先级。 1: 定时器 0 为高优先级。								
位 0: PX0: 外部中断 0 优先级控制 该位设置外部中断 0 的优先级。 0: 外部中断 0 为低优先级。 1: 外部中断 0 为高优先级。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 15.3 EIE1: 扩展中断允许 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
ET3	保留	ECP0	EPCA0	EADC0	EWADC0	EMAT	ESMB0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xE6
位 7:	ET3: 定时器 3 中断允许位 该位设置定时器 3 的中断屏蔽。 0: 禁止定时器 3 中断。 1: 允许 TF3L 或 TF3H 标志的中断请求。							
位 6:	保留。读=0b, 必须写 0。							
位 5:	ECP0: 比较器 0 (CP0) 中断允许位 该位设置 CP0 的中断屏蔽。 0: 禁止 CP0 中断。 1: 允许 CP0RIF 或 CP0FIF 标志的中断请求。							
位 4:	EPCA0: 可编程计数器阵列 (PCA0) 中断允许位 该位设置 PCA0 的中断屏蔽。 0: 禁止所有 PCA0 中断。 1: 允许 PCA0 的中断请求。							
位 3:	EADC0: ADC0 转换结束中断允许位 该位设置 ADC0 转换结束中断屏蔽。 0: 禁止 ADC0 转换结束中断。 1: 允许 AD0INT 标志的中断请求。							
位 2:	EWADC0: ADC0 窗口比较中断允许位 该位设置 ADC0 窗口比较中断屏蔽。 0: 禁止 ADC0 窗口比较中断。 1: 允许 ADC0 窗口比较标志 (AD0WINT) 的中断请求。							
位 1:	EMAT: 端口匹配中断允许位 该位设置端口匹配事件的中断屏蔽。 0: 禁止所有端口匹配中断。 1: 允许由端口匹配产生的中断请求。							
位 0:	ESMB0: SMBus 中断允许位 该位设置 SMBus (SMB0) 的中断屏蔽。 0: 禁止 SMB0 中断。 1: 允许 SMB0 的中断请求。							

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 15.4 EIP1: 扩展中断优先级 1

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
PT3	保留	PCP0	PPCA0	PADC0	PWADC0	PMAT	PSMB0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xF6
位 7:	PT3: 定时器 3 中断优先级控制 该位设置定时器 3 中断的优先级。 0: 定时器 3 中断为低优先级。 1: 定时器 3 中断为高优先级。							
位 6:	保留。读=0b, 必须写 0。							
位 5:	PCP0: 比较器 0 (CP0) 中断优先级控制 该位设置 CP0 中断的优先级。 0: CP0 中断为低优先级。 1: CP0 中断为高优先级。							
位 4:	PPCA0: 可编程计数器阵列 (PCA0) 中断优先级控制 该位设置 PCA0 中断的优先级。 0: PCA0 中断为低优先级。 1: PCA0 中断为高优先级。							
位 3:	PADC0: ADC0 转换结束中断优先级控制 该位设置 ADC0 转换结束中断的优先级。 0: ADC0 转换结束中断为低优先级。 1: ADC0 转换结束中断为高优先级。							
位 2:	PWADC0: ADC0 窗口比较器中断优先级控制 该位设置 ADC0 窗口中断的优先级。 0: ADC0 窗口中断为低优先级。 1: ADC0 窗口中断为高优先级。							
位 1:	PMAT: 端口匹配中断优先级控制 该位设置端口匹配中断的优先级。 0: 端口匹配中断为低优先级。 1: 端口匹配中断为高优先级。							
位 0:	PSMB0: SMBus (SMB0) 中断优先级控制 该位设置 SMB0 中断的优先级。 0: SMB0 中断为低优先级。 1: SMB0 中断为高优先级。							

15.3 外部中断

两个外部中断源/INT0 和/INT1 可被配置为低电平有效或高电平有效，边沿触发或电平触发。IT0ICF 寄存器中的 IN0PL (/INT0 极性) 和 IN1PL (/INT1 极性) 位用于选择高电平有效还是低电平有效；TCON 中的 IT0 和 IT1 用于选择电平或边沿触发（见“24.1 定时器 0 和定时器 1”）。下面的表列出了可能的配置组合。

IT0	IN0PL	/INT0 中断	IT1	IN1PL	/INT1 中断
1	0	低电平有效，边沿触发	1	0	低电平有效，边沿触发
1	1	高电平有效，边沿触发	1	1	高电平有效，边沿触发
0	0	低电平有效，电平触发	0	0	低电平有效，电平触发
0	1	高电平有效，电平触发	0	1	高电平有效，电平触发

/INT0 和/INT1 所使用的端口引脚在 IT0ICF 寄存器中定义(见 SFR 定义 15.5)。注意,/INT0 和/INT0 端口引脚分配与交叉开关的设置无关。/INT0 和/INT1 监视分配给它们的端口引脚，不影响被交叉开关分配了相同引脚的外设。如果要将一个端口引脚只分配给/INT0 或/INT1，则应使交叉开关跳过这个引脚。这可以通过设置寄存器 XBR0 中的相应位来实现（有关配置交叉开关的详细信息见“20.3 优先权交叉开关译码器”）。

IE0 (TCON.1) 和 IE1 (TCON.3) 分别为外部中断/INT0 和/INT1 的中断标志。如果/INT0 或/INT1 外部中断被配置为边沿触发，CPU 在转向 ISR 时用硬件自动清除相应的中断标志。当被配置为电平触发时，在输入有效期间（根据极性控制位 IN0PL 或 IN1PL 的定义）中断标志将保持在逻辑 1 状态；在输入无效期间该标志保持逻辑 0 状态。电平触发的外部中断源必须一直保持输入有效直到中断请求被响应，在 ISR 返回前必须使该中断请求无效，否则将产生另一个中断请求。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 15.5 IT01CF: INT0/INT1 配置寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
IN1PL	IN1SL2	IN1SL1	IN1SL0	IN0PL	IN0SL2	IN0SL1	IN0SL0	00000001
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xE4

位 7:

IN1PL: /INT1 极性
0: /INT1 为低电平有效。
1: /INT1 为高电平有效。

位 6-4:

IN1SL2-0: /INT1 端口引脚选择位
这些位用于选择分配给/INT1 的端口引脚。注意，该引脚分配与交叉开关无关；/INT1 将监视分配给它的端口引脚，但不影响被交叉开关分配了相同引脚的外设。如果将交叉开关配置为跳过这个引脚（通过将寄存器 POSKIP 中的对应位置 1 来实现），则该引脚将不会被分配给外设。

IN1SL2-0	/INT1 端口引脚
000	P0.0
001	P0.1
010	P0.2
011	P0.3
100	P0.4
101	P0.5
110	P0.6
111	P0.7

位 3:

IN0PL: /INT0 极性
0: /INT0 为低电平有效。
1: /INT0 为高电平有效。

位 2-0:

IN0SL2-0: /INT0 端口引脚选择位
这些位用于选择分配给/INT0 的端口引脚。注意，该引脚分配与交叉开关无关；/INT0 将监视分配给它的端口引脚，但不影响被交叉开关分配了相同引脚的外设。如果将交叉开关配置为跳过这个引脚（通过将寄存器 POSKIP 中的对应位置 1 来实现），则该引脚将不会被分配给外设。

IN0SL2-0	/INT0 端口引脚
000	P0.0
001	P0.1
010	P0.2
011	P0.3
100	P0.4
101	P0.5
110	P0.6
111	P0.7

16. FLASH 存储器

C8051F336/7/8/9 内部有可编程的 FLASH 存储器，用于程序代码和非易失性数据存储。可以通过 C2 接口或由软件使用 MOVX 指令对 FLASH 存储器进行在系统编程，每次一个字节。一个 FLASH 位一旦被清 0，必须经过擦除才能再回到 1 状态。在进行重新编程之前，一般要将数据字节擦除（置为 0xFF）。为了保证操作正确，写和擦除操作由硬件自动定时，不需要进行数据查询来判断写/擦除操作何时结束。在 FLASH 写/擦除操作期间，程序停止执行。参见表 6.5 的 FLASH 存储器电气特性。

16.1 FLASH 存储器编程

对 FLASH 存储器编程的最简单的方法是使用由 Silicon Labs 公司或第三方供应商提供的编程工具，通过 C2 接口编程，这是对未被初始化过的器件的唯一编程方法。有关对 FLASH 程序存储器编程的 C2 命令的详细信息见“26. C2 接口”。

为了保证 FLASH 内容的正确性，强烈建议在使用用户软件对 FLASH 存储器进行写和/或擦除操作的系统中使能片内 VDD 监视器。详见 16.4 节。

16.1.1 FLASH 锁定和关键码功能

从用户软件写和擦除 FLASH 受 FLASH 锁定和关键码功能的保护。在进行 FLASH 操作之前，必须按顺序向 FLASH 锁定和关键码寄存器（FLKEY）写入正确的关键码。关键码为：0xA5，0xF1。写关键码的时序并不重要，但必须按顺序写。如果写关键码的顺序不对或写入了错误的键码，FLASH 写和擦除操作将被禁止，直到下一次系统复位。如果在正确写入关键码之前进行了 FLASH 写或擦除操作，FLASH 写和擦除也将被禁止。每次 FLASH 写和擦除操作之后，FLASH 锁定功能复位；在进行下一次 FLASH 写或擦除操作之前，必须重新写关键码。FLKEY 寄存器的详细说明见 SFR 定义 16.2。

16.1.2 FLASH 擦除

可以用软件使用 MOVX 指令对 FLASH 存储器编程，象一般的操作数一样为 MOVX 指令提供待编程的地址和数据字节。在使用 MOVX 指令对 FLASH 存储器写入之前，必须先允许 FLASH 写操作。允许 FLASH 写操作的过程是：1）将程序存储写允许位 PSWE（PSCTL.0）设置为逻辑 1（这将使 MOVX 操作指向目标 FLASH 存储器）；2）按顺序向 FLASH 锁定寄存器（FLKEY）写入 FLASH 关键码。PSWE 位将保持置位状态，直到被软件清除。

写 FLASH 存储器可以清除数据位，但不能使数据位置 1，只有擦除操作能将 FLASH 中的数据位置 1。所以在写入新值之前，必须先擦除待编程的地址。FLASH 存储器是以 512 字节的扇区为单位组织的，一次擦除操作将擦除整个扇区（将扇区内的所有字节置为 0xFF）。擦除一个页的步骤如下：

1. 禁止中断（建议这样做）。
2. 置 1 程序存储器擦除允许位（PSCTL 中的 PSEE），以允许 FLASH 扇区擦除。
3. 置 1 程序存储器写允许位（PSCTL 中的 PSWE），以允许 FLASH 写入。
4. 向 FLKEY 写第一个关键码：0xA5。
5. 向 FLKEY 写第二个关键码：0xF1。
6. 用 MOVX 指令向待擦除页内的任何一个地址写入一个数据字节。

7. 清除 PSWE 和 PSEE 位。

16.1.3 FLASH 写

用软件对 FLASH 字节编程的步骤如下：

1. 禁止中断（建议这样做）。
2. 擦除包含目标地址的 512 字节 FLASH 页。
3. 置位 PSCTL 中的 PSWE 位。
4. 清除 PSCTL 中的 PSEE 位。
5. 向 FLKEY 写第一个关键码：0xA5。
6. 向 FLKEY 写第二个关键码：0xF1。
7. 用 MOVX 指令向扇区内的目标地址写入一个数据字节。
8. 清除 PSWE 位。
9. 重新允许中断。

重复步骤 5-7，直到写完每个字节。在完成了对 FLASH 的数据写入后，PSWE 位应被清 0，以使 MOVX 指令不再指向程序存储器。

16.2 非易失性数据存储

FLASH 存储器除了用于存储程序代码之外还可以用于非易失性数据存储。这就允许在程序运行时计算并存储类似标定系数这样的数据。数据写入时用 MOVX 指令，读出时用 MOVC 指令。注意：MOVX 读指令总是指向 XRAM。

16.3 安全选项

CIP-51 提供了安全选项以保护 FLASH 存储器不会被软件意外修改，以及防止产程序代码和常数被读取。程序存储器写允许（PSCTL 寄存器中的 PSWE）和程序存储器擦除允许（PSCTL 寄存器中的 PSEE）位保护 FLASH 存储器不会被软件意外修改。在用软件修改 FLASH 存储器的内容之前，PSWE 必须被置为逻辑 1；在用软件擦除 FLASH 存储器之前，PSWE 位和 PSEE 位都必须被置为逻辑 1。此外，CIP-51 还提供了可以防止通过 C2 接口读取或修改产程序代码和常数这一安全功能。

保存在 FLASH 用户空间的最后一个字节中的安全锁定字节保护 FLASH 存储器，使其不能被非保护代码或通过 C2 接口读、写或擦除。FLASH 安全机制允许用户从 0 页（地址 0x0000 ~ 0x01FF）开始锁定 n 个 512 字节的 FLASH 页，其中 n 是安全锁定字节的反码。**注意：在没有其它 FLASH 页被锁定时（锁定字节的所有位均为 1），包含 FLASH 安全锁定字节的页被解锁。当任何一个其他 FLASH 页被锁定时（锁定字节有任何一位为 0），包含 FLASH 安全锁定字节的页也被锁定。**见图 16.1 中的例子。

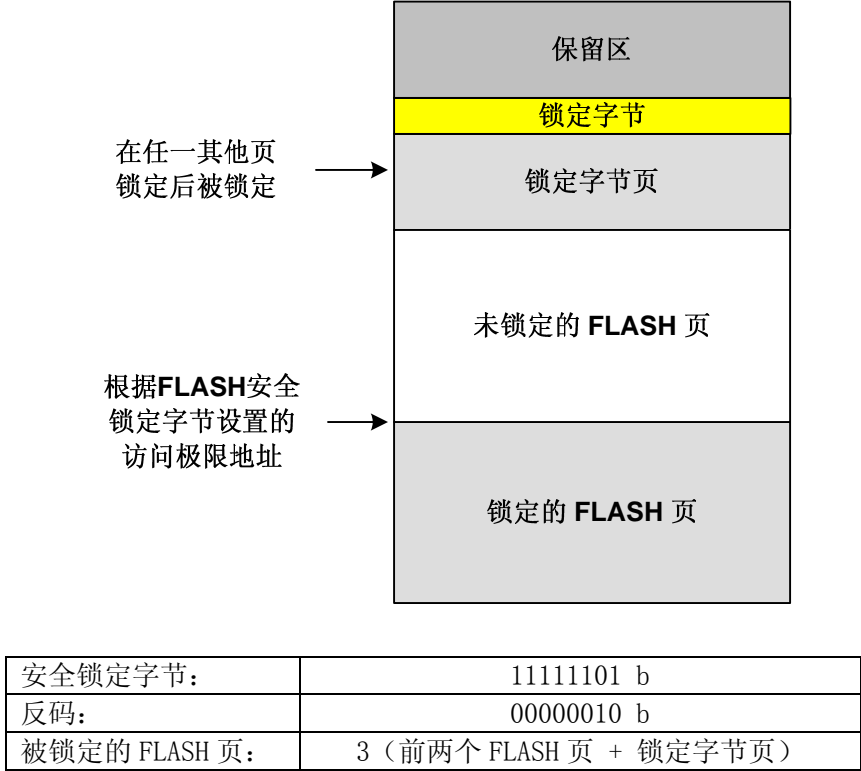


图 16.1 FLASH 程序存储器组织

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

FLASH 安全级别取决于对 FLASH 访问的方式。有 3 种可被限制的访问方式：经 C2 调试接口的读、写和擦除，在非锁定页执行的用户固件，在锁定页执行的用户固件。表 16.1 概述了 C8051F336/7/8/9 器件的 FLASH 安全特性。

表 16.1 FLASH 安全一览表

操作	C2 调试接口	用户固件所在执行区域：	
		未锁定页	被锁定页
读、写或擦除未锁定页 (锁定字节所在页除外)	允许	允许	允许
读、写或擦除被锁定页 (锁定字节所在页除外)	不允许	FLASH 错误复位	允许
读或写锁定字节所在页 (如果没有被锁定的页)	允许	允许	允许
读或写锁定字节所在页 (如果有任何页被锁定)	不允许	FLASH 错误复位	允许
读锁定字节的内容 (如果没有被锁定的页)	允许	允许	允许
读锁定字节的内容 (如果有任何页被锁定)	不允许	FLASH 错误复位	允许
擦除锁定字节所在页 (如果没有被锁定的页)	允许	FLASH 错误复位	FLASH 错误复位
擦除锁定字节所在页—解锁所有 页 (如果有任何页被锁定)	只能进行 C2 器件擦除	FLASH 错误复位	FLASH 错误复位
锁定附加页 (将锁定字节中的 1 变成 0)	不允许	FLASH 错误复位	FLASH 错误复位
解锁单个页 (将锁定字节中的 0 变成 1)	不允许	FLASH 错误复位	FLASH 错误复位
读、写或擦除保留区	不允许	FLASH 错误复位	FLASH 错误复位
<p>C2 器件擦除——擦除所有 FLASH 页，包括锁定字节所在页。</p> <p>FLASH 错误复位——不允许相应的操作；导致 FLASH 错误器件复位（复位后寄存器 RSTSRC 中的 FERROR 位为 1）。</p> <ul style="list-style-type: none">● 所有被禁止的经由 C2 接口的操作都被忽略（不导致器件复位）。● 锁定任何一个 FLASH 页导致包含锁定字节的页也被锁定。● 一旦被写入，锁定字节不能被修改，除非执行一次 C2 器件擦除。● 如果用户代码写锁定字节，则在下一次器件复位前锁定功能不会生效。			

16.4 FLASH 写和擦除指南

如果 CPU 工作在 V_{DD} 、系统时钟频率或温度的额定范围之外，任何包含写或擦除 FLASH 代码的系统都存在这样的危险，即意外执行写或擦除 FLASH 代码。修改 FLASH 内容的代码之意外执行会导致 FLASH 存储器内容的改变，所引发的系统故障只能通过重新烧写 FLASH 来解决。

建议在任何包含写或擦除 FLASH 代码的系统中遵循下述指南。

16.4.1 VDD 维护和 VDD 监视器

1. 如果系统电源易受电压或电流尖峰的干扰，应在电源部分增加瞬变保护器件，以保证电源电压不超过极限值。
2. 保证满足 1ms 的最小 VDD 上升时间。如果系统不满足这个最小上升时间指标，则要在器件的复位引脚加一个外部 VDD 欠压检测电路，以使器件在 VDD 达到 2.7V 之前保持复位状态和在 VDD 下降到低于 2.7V 时使复位引脚有效。
3. 在代码中尽可能早地使能片内 VDD 监视器并将其使能为复位源。这应该是复位向量之后最先被执行的指令。对于用 C 语言开发的系统，要做到这一点需修改随 C 编译器提供的启动代码。有关这方面的详细信息，请参见所用编译器的文档。要保证在使能 VDD 监视器和将其使能为复位源之间的软件没有延时。在 Silicon Laboratories 网站上提供的“AN201：从固件写 FLASH”中给出了示例代码。
4. 可以增加一层预防措施，即在写和擦除 FLASH 存储器的函数中显式地使能 VDD 监视器并将其使能为复位源。使能 VDD 监视器的指令应紧接在将 PSWE 置 1 的指令之后，但位于 FLASH 写或擦除操作指令之前。
5. 保证所有写 RSTSRC（复位源）寄存器的指令都使用直接赋值操作符显式赋值，不要使用位操作（如 AND 或 OR）。例如，“RSTSRC = 0x02”是正确的，而“RSTSRC |= 0x02”是不正确的。
6. 保证所有写 RSTSRC 寄存器的指令都显式地将 PORSF 位置 1，要仔细检查使能其它复位源的初始化代码（例如时钟丢失检测器或比较器）和强制软件复位的指令。通过全局搜索“RSTSRC”可以很快完成检查。

16.4.2 PSWE 维护

7. 在代码中尽量减少将 PSWE 位（PSCTL 的位 0）置 1 的位置数。在代码中应只使用一个将 PSWE 置 1 的例程（写 FLASH 字节）和一个将 PSWE 及 PSEE 置 1 的例程（擦除 FLASH 页）。
8. 在 PSWE 被置 1 期间，尽量减少变量访问次数。在“PSWE = 1; ... PSWE = 0;”的区域之外处理指针地址更新和改变循环变量。在 Silicon Laboratories 网站上提供的“AN201：从固件写 FLASH”中给出了示例代码。
9. 在将 PSWE 置 1 之前禁止中断，并保持中断的禁止状态直到 PSWE 被清 0。在 FLASH 写或擦除操作期间所产生的任何中断都会在 FLASH 操作完成和中断被软件重新使能之后按优先级顺序得到服务。
10. 保证 FLASH 写和擦除指针变量不位于 XRAM 空间。有关如何显式地将变量定位到不同存储器区域的说明请参见您所使用的编译器的文档。

11. 在写或擦除 FLASH 存储器的例程中增加地址边界检查，以保证在使用非法地址调用一个例程时不会修改 FLASH。

16.4.3 系统时钟

12. 如果 CPU 使用外部晶体工作，应注意晶体的性能容易受到电气干扰的影响和对布局布线及温度变化敏感。如果系统工作在有强电气噪声的环境，应使用内部振荡器或外部 CMOS 时钟。
13. 如果 CPU 使用外部振荡器工作，在 FLASH 写或擦除操作期间将系统时钟切换到内部振荡器。外部振荡器可以继续运行，CPU 可以在 FLASH 操作结束后切换回外部振荡器。

另外一些 FLASH 操作建议及示例代码请见 Silicon Laboratories 网站上提供的“AN201：从固件写 FLASH”。

SFR 定义 16.1 PSCTL：程序存储读写控制

R	R	R	R	R	R	R/W	R/W	复位值
-	-	-	-	-		PSEE	PSWE	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0x8F
位 7-2： 未使用。读 = 000000b，写 = 忽略。								
位 1： PSEE：程序存储擦除允许								
将该位置 1 后允许擦除 FLASH 存储器中的一个页(前提是 PSWE 位也被置 1)。								
在将该位置 1 后，用 MOVX 指令进行一次写操作将擦除包含 MOVX 指令寻址地址的那个 FLASH 页。用于写操作的数据可以是任意值。								
0：禁止擦除 FLASH 存储器。								
1：允许擦除 FLASH 存储器。								
位 0： PSWE：程序存储写允许								
将该位置 1 后允许用 MOVX 指令向 FLASH 存储器写一个字节。在写数据之前必须先进行擦除。								
0：禁止写 FLASH 存储器。								
1：允许写 FLASH 存储器； MOVX 写指令寻址 FLASH 存储器。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 16.2 FLKEY: FLASH 锁定和关键码寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xB7
<p>位 7-0: FLKEY: FLASH 锁定和关键码寄存器</p> <p>写:</p> <p>该寄存器为 FLASH 擦除和写操作提供锁定和关键码功能。通过向该寄存器按顺序写入下面的关键码 0xA5 和 0xF1 来使能 FLASH 写和擦除。在执行完一次写或擦除操作后, 后续的 FLASH 写或擦除操作被自动禁止。如果写 FLKEY 操作不正确或在正确写入关键码之前进行了 FLASH 操作, 则 FLASH 将被锁定, 直到下一次系统复位。如果应用固件从不写 FLASH, 可以用软件向 FLKEY 写入一个非 0xA5 的值, 以锁定 FLASH。</p> <p>读:</p> <p>位 1-0 指示当前的 FLASH 锁定状态</p> <p>00: FLASH 写/擦除被锁定。</p> <p>01: 第一个关键码已被写入 (0xA5)。</p> <p>10: FLASH 处于解锁状态 (允许写/擦除)</p> <p>11: FLASH 写/擦除操作被禁止, 直到下一次复位。</p>								

SFR 定义 16.3 FLSC: FLASH 定时预分频

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
FOSE	保留	保留	保留	保留	保留	保留	保留	10000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xB6
<p>位 7: FOSE: FLASH 单稳态定时器使能</p> <p>该位用于使能 FLASH 读操作单稳态定时器 (建议如此)。当 FLASH 单稳态定时器被禁止时, FLASH 读出放大器在整个 FLASH 读周期内被使能, 导致器件功耗增加。</p> <p>0: 禁止 FLASH 单稳态定时器。</p> <p>1: 使能 FLASH 单稳态定时器。</p> <p>位 6-0: 保留。必须写 0000000b。</p>								

17. 电源管理方式

C8051F336/7/8/9 有三种可软件编程的电源管理方式：空闲、停机和挂起。空闲和挂起方式是标准 8051 体系结构的组成部分，而挂起方式是用高速振荡器外设实现的一种增强节电方式。

在空闲方式，CPU 停止运行，而外设和时钟处于活动状态。在停机方式，CPU 停止运行，所有的中断和定时器（时钟丢失检测器除外）都处于非活动状态，内部振荡器停止（模拟外设保持在所选择的状态；外部振荡器不受影响）。挂起方式与停机方式相似，内部振荡器和 CPU 停止运行，但器件可被端口不匹配、比较器低电平输出或定时器 3 溢出这些唤醒事件唤醒。由于在空闲方式下时钟仍然运行，所以功耗与进入空闲方式之前的系统时钟频率和处于活动状态的外设数目有关。停机方式和挂起方式消耗最少的功率，因为器件的主要部分因无活动时钟而停止运行。SFR 定义 17.12 对用于控制 CIP-51 电源管理方式的电源控制寄存器作出了说明。挂起方式由 OSCICN 寄存器中的 SUSPEND 位控制（见 SFR 定义 19.3）。

虽然 CIP-51 具有空闲、停机和挂起方式，但根据需要独立地使能和禁止每个外设，可以使整个 MCU 的功耗最小。每个模拟外设在不使用时都可以被禁止，使其进入低功耗方式。像定时器、串行总线这样的数字外设在不使用时消耗很少的功率。关闭振荡器可以大大降低消耗功率，但以降低功能为代价。

17.1 空闲方式

将空闲方式选择位（PCON.0）置 1 导致 CIP-51 停止 CPU 运行并进入空闲方式，在执行完对该位置 1 的指令后 MCU 立即进入空闲方式。所有内部寄存器和存储器都保持原来的数据不变。所有模拟和数字外设空闲方式期间都可以保持活动状态。

有被允许的中断发生或复位有效将结束空闲方式。当有一个被允许的中断发生时，空闲方式选择位（PCON.0）被清 0，CPU 将继续工作。该中断将得到服务，中断返回（RETI）后将开始执行设置空闲方式选择位的那条指令的下一条指令。如果空闲方式因一个内部或外部复位而结束，则 CIP-51 进行正常的复位过程并从地址 0x0000 开始执行程序。

注意：如果在写 IDLE 位的指令之后是一条单字节指令，并且在将 IDLE 位置 1 的指令的执行阶段发生了中断，则将来发生中断时 CPU 可能不会被唤醒。因此，将 IDLE 位置 1 的指令的后面应跟随一条具有 2 或多字节操作码的指令。例如：

// 用 ‘C’ 语言：

```
PCON |= 0x01 ; //      将 IDLE 位置 1
PCON = PCON ; //      ... 跟随一条 3 字节操作码的指令
```

；用汇编语言：

```
ORL  PCON, #01h      ; 将 IDLE 位置 1
MOV  PCON, PCON      ; ... 跟随一条 3 字节操作码的指令
```

如果被允许，WDT 将产生一个内部看门狗复位，从而结束空闲方式。这一功能可以保护

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

系统不会因为对 PCON 寄存器的意外写入而导致永久性停机。如果不需要这一功能，可以在进入空闲方式之前禁止 WDT。这将进一步节省功耗，允许系统一直保持在空闲状态，等待一个外部激励唤醒系统。有关使用和配置 WDT 的详细信息，请参见“18.6 PCA 看门狗定时器复位”。

17.2 停机方式

将停机方式选择位（PCON.1）置 1 导致 CIP-51 进入停机方式，在执行完对该位置 1 的指令后 MCU 立即进入停机方式。在停机方式，内部振荡器、CPU 和所有的数字外设都停止工作，但外部振荡器电路的状态不受影响。在进入停机方式之前，每个模拟外设（包括外部振荡器电路）都可以被单独关断。只有内部或外部复位能结束停机方式。复位时，CIP-51 进行正常的复位过程并从地址 0x0000 开始执行程序。

如果被使能，时钟丢失检测器将产生一个内部复位，从而结束停机方式。如果想要使 CPU 的休眠时间长于 100 微秒的 MCD 超时时间，则应禁止时钟丢失检测器。

17.3 挂起方式

向 SUSPEND 位（OSCICN.5）写 1 将导致硬件停止 CPU 和高频内部振荡器。在执行完对该位置 1 的指令后 MCU 立即进入挂起方式。所有内部寄存器和存储器都保持原来的数据不变。大多数数字外设处于挂起方式不活动。例外情况是端口匹配功能和定时器 3 使用外部振荡器或内部低频振荡器运行。

挂起方式可以被四类事件终止：端口匹配（见“20.5 端口匹配”）、定时器 3 溢出（见“24.3 定时器 3”）、比较器低电平（如果被使能）输出和器件复位。注意：为了使定时器 3 在挂起方式运行，该定时器应被配置为使用外部时钟源或内部低频振荡器作为其时钟。当挂起方式终止时，器件将从置位 SUSPEND 位的那条指令的下一条指令继续执行。如果唤醒事件被配置为能产生中断，则该中断将在器件被唤醒后得到服务。如果挂起方式被一个内部或外部复位终止，则 CIP-51 执行正常的复位过程并从地址 0x0000 开始执行程序。

SFR 定义 17.1 PCON：电源控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
GF5	GF4	GF3	GF2	GF1	GF0	STOP	IDLE	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0x87
位 7-2: GF[5:0]: 通用标志位 5-0。 这些位是用作软件控制的通用标志位。								
位 1: STOP: 停机方式选择。 将该位置 1 使 CIP-51 进入停机方式。该位读出值总是为 0。 1: 进入掉电方式（内部振荡器停止运行）								
位 0: IDLE: 空闲方式选择。 将该位置 1 使 CIP-51 进入空闲方式。该位读出值总是为 0。 1: CPU 进入空闲方式。（关闭供给 CPU 的时钟信号，但供给定时器、中断、 串口和模拟外设的时钟仍然保持活动状态。）								

18. 复位源

复位电路允许很容易地将控制器置于一个预定的缺省状态。在进入复位状态时，将发生以下过程：

- CIP-51 停止程序执行
- 特殊功能寄存器（SFR）被初始化为所定义的复位值
- 外部端口引脚被置于一个已知状态
- 中断和定时器被禁止。

所有的 SFR 都被初始化为预定值，SFR 中各位的复位值在 SFR 的详细说明中定义。在复位期间内部数据存储器的内容不发生改变，复位前存储的数据保持不变。但由于堆栈指针 SFR 被复位，堆栈实际上已丢失，尽管堆栈中的数据未发生变化。

端口 I/O 锁存器的复位值为 0xFF（全部为逻辑‘1’），处于漏极开路方式。在复位期间和复位之后弱上拉被使能。对于 VDD 监视器和上电复位，/RST 引脚被驱动为低电平，直到器件退出复位状态。

在退出复位状态时，程序计数器（PC）被复位，MCU 使用内部振荡器作为默认的系统时钟。有关选择和配置系统时钟源的详细说明见“13. 振荡器”。看门狗定时器被使能，使用系统时钟的 12 分频作为其时钟源。程序从地址 0x0000 开始执行。

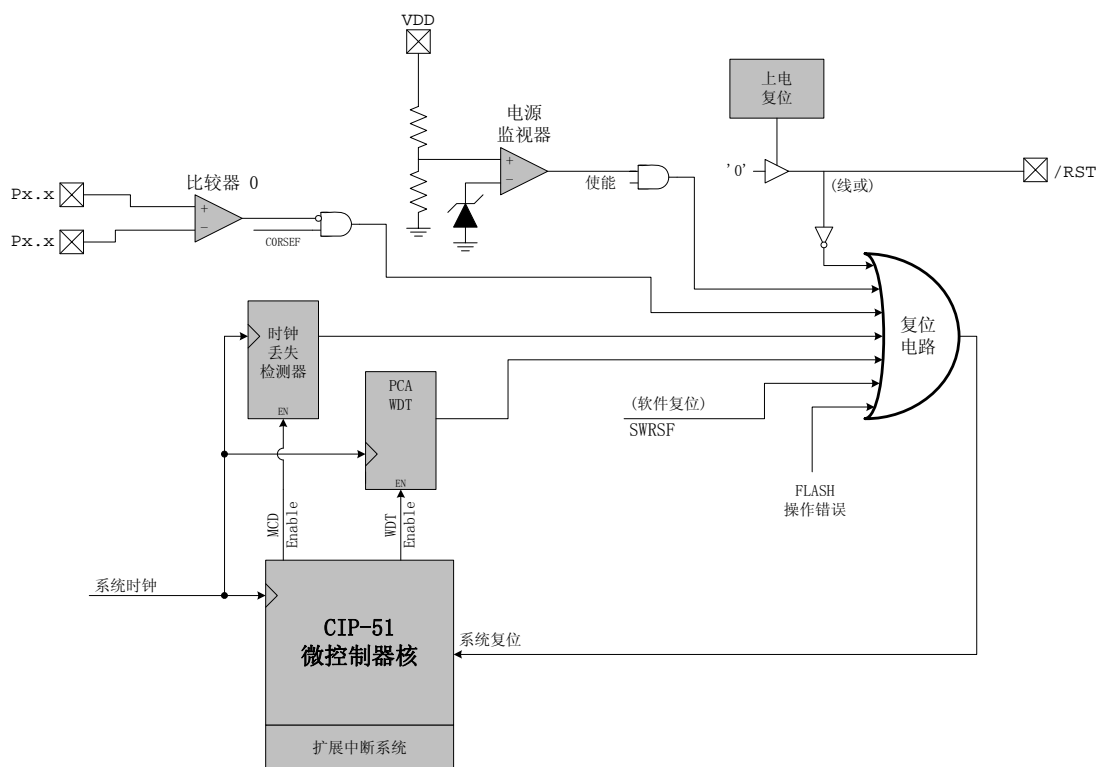


图 18.1 复位源框图

18.2 掉电复位和 VDD 监视器

当发生掉电或因电源波动导致 VDD 降到 V_{RST} 以下时，电源监视器将 \overline{RST} 引脚驱动为低电平并使 CIP-51 保持复位状态（见图 18.2）。当 VDD 又回到高于 V_{RST} 的电平时，CIP-51 将退出复位状态。注意，尽管内部数据存储器的内容可能没有因掉电复位而发生改变，但无法确定 VDD 是否降到了数据保持所要求的最低电平以下。如果 PORSF 标志的读出值为 1，则内部 RAM 的数据可能不再有效。在上电复位后 VDD 监视器被使能，但它的状态（使能/禁止）不受任何其它复位源的影响。例如，在 VDD 监视器被禁止后执行一次软件复位，复位后 VDD 监视器仍然为禁止状态。

注意：如果 VDD 监视器从禁止状态被使能，则它必须在被选择作为复位源之前使能。在 VDD 监视器被使能或稳定之前选其为复位源可能导致系统复位。在有些应用中，这种复位可能是不希望产生的。如果在应用中不希望产生这样的复位，则在使 VDD 监视器和选择它作为复位源之间应引入延时。从禁止状态使能 VDD 监视器和将其配置为复位源的步骤如下：

- 1. 使能 VDD 监视器（VDM0CN 中的 VDMEN 位 = 1）；
- 2. 如果需要，等待 VDD 监视器稳定（见表 6.4 的 VDD 监视器启动时间）；
- 3. 选择 VDD 监视器作为复位源（RSTSRC 中的 PORSF 位 = 1）。

图 18.2 给出了 VDD 监视器的时序。注意，在 VDD 监视器复位后没有复位延时。VDD 监视器的完全电气特性见表 6.4。

SFR 定义 18.1 VDM0CN: VDD 监视器控制寄存器

R/W	R	R	R	R	R	R	R	复位值
VDMEN	VDDSTAT	保留	保留	保留	保留	保留	保留	可变
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xFF
<p>位 7: VDMEN: VDD 监视器使能位。 该位控制 VDD 监视器电源的通断。VDD 监视器在被选择为复位源之前不可能产生系统复位。在被选择为复位源之前，VDD 监视器必须稳定。在 VDD 监视器稳定之前选其为复位源可能导致系统复位。在不希望产生这种复位的系统中，在使 VDD 监视器和选择它作为复位源之间应引入延时。VDD 监视器的最小启动时间见表 6.4。 0: 禁止 VDD 监视器。 1: 使能 VDD 监视器。</p> <p>位 6: VDDSTAT: VDD 状态。 该位指示当前电源状态（VDD 监视器输出）。 0: VDD 等于或低于 VDD 监视器阈值。 1: VDD 高于 VDD 监视器阈值</p> <p>位 5-0: 保留。读 = 00000b，写 = 忽略。</p>								

18.3 外部复位

外部/RST 引脚提供了使用外部电路强制 MCU 进入复位状态的手段。在/RST 引脚上加一个低电平有效信号将导致 MCU 进入复位状态。尽管在内部有弱上拉，但最好能提供一个外部上拉和/或对/RST 引脚去耦以防止强噪声引起复位。从外部复位状态退出后，PINRSF 标志（RSTSRC.0）被置 1。

18.4 时钟丢失检测器复位

时钟丢失检测器（MCD）实际上是由系统时钟触发的单稳态电路。如果系统时钟保持在高电平或低电平的时间大于 100 微秒，单稳态电路将超时并产生复位。在发生时钟丢失检测器复位后，MCDRSF 标志（RSTSRC.2）的读出值为 1，表示本次复位源为 MCD；否则该位读出值为 0。向 MCDRSF 位写 1 将使能时钟丢失检测器；写 0 将禁止时钟丢失检测器。/RST 引脚的状态不受该复位的影响。

18.5 比较器 0 复位

向 CORSEF 标志（RSTSRC.5）写 1 可以将比较器 0 配置为复位源。应在写 CORSEF 之前使能比较器 0 并等待输出稳定，以防止通电瞬间在输出端产生抖动，从而导致不希望的复位。比较器 0 复位为低电平有效：如果同相端输入电压（CP0+）小于反相端输入电压（CP0-），则 MCU 被置于复位状态。在发生比较器 0 复位后，CORSEF 标志（RSTSRC.5）的读出值为 1，表示本次复位源为比较器 0；否则该位读出值为 0。/RST 引脚的状态不受该复位的影响。

18.6 PCA 看门狗定时器复位

可编程计数器阵列（PCA）的可编程看门狗定时器（WDT）功能可用于在系统出现错误的情况下防止软件运行失控。可以通过软件使能或禁止 PCA 的 WDT 功能。在每次复位后，WDT 被使能并使用 SYSCLK/12 作为计数时钟。如果因系统出错使用户软件不能更新 WDT，则 WDT 将产生一次复位，WDTRSF 位（RSTSRC.5）被置 1。/RST 引脚的状态不受该复位的影响。

18.7 FLASH 错误复位

如果 FLASH 读/写/擦除操作的地址或程序读地址为非法地址，将发生系统复位。下述的任何一种情况都会导致 FLASH 操作错误：

- FLASH 写或擦除地址超出了用户代码空间。这种情况发生在 PSWE 被置 1，并且 MOVX 操作的地址大于 0x3DFF 时。
- FLASH 读地址超出了用户代码空间，即 MOVC 操作的地址大于 0x3DFF。
- 程序读超出了用户代码空间。这种情况发生在用户代码试图转移到大于 0x3DFF 的地址时。
- 当 FLASH 读、写或擦除被安全设置禁止时（见 16.3 安全选项）。

在发生 FLASH 错误复位后，FERROR 位（RSTSRC.6）被置位。/RST 引脚的状态不受该复位的影响。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

18.8 软件复位

软件可以通过向 SWRSF 位 (RSTSRC.4) 写 1 强制产生一次系统复位。在发生软件强制复位后, SWRSF 位的读出值为 1。/RST 引脚的状态不受该复位的影响。

SFR 定义 18.2 RSTSRC: 复位源寄存器

R	R	R/W	R/W	R	R/W	R/W	R	复位值
-	FERROR	CORSEF	SWRSF	WDTRSF	MCDRSF	PORSF	PINRSF	可变
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xEF
位 7:	未用。读 = 0, 写 = 忽略。							
位 6:	FERROR: FLASH 错误标志。 0: 最后一次复位不是来自 FLASH 读/写/擦除错误。 1: 最后一次复位是由于 FLASH 读/写/擦除错误。							
位 5:	CORSEF: 比较器 0 复位使能和标志 写 0: 比较器 0 不是复位源。 1: 比较器 0 是复位源 (低电平有效)。 读 0: 最后一次复位不是来自比较器 0。 1: 最后一次复位来自比较器 0。							
位 4:	SWRSF: 软件强制复位和标志 写 0: 无作用 1: 强制产生一次系统复位。 读 0: 最后一次复位不是来自写 SWRSF 位。 1: 最后一次复位来自写 SWRSF 位。							
位 3:	WDTRSF: 看门狗定时器复位标志 0: 最后一次复位不是来自 WDT 超时。 1: 最后一次复位来自 WDT 超时。							
位 2:	MCDRSF: 时钟丢失检测器标志 写 0: 禁止时钟丢失检测器。 1: 使能时钟丢失检测器; 检测到时钟丢失条件时触发复位。 读 0: 最后一次复位不是来自时钟丢失检测器超时。 1: 最后一次复位来自时钟丢失检测器超时。							
位 1:	PORSF: 上电/VDD 监视器复位标志和 VDD 监视器复位使能位 该位在上电复位后被置位。对该位写入可以使能/禁止 VDD 监视器作为复位源。 在 VDD 监视器被使能和稳定之前向该位写 1 可能导致系统复位。见寄存器 VDM0CN (图 18.3)。 写 0: 禁止 VDD 监视器为复位源。 1: 使能 VDD 监视器为复位源。 读 0: 最后一次复位不是上电复位或 VDD 监视器复位。 1: 最后一次复位是上电或 VDD 监视器复位, 所有其它复位标志是不确定的。							
位 0:	PINRSF: 硬件引脚复位标志 0: 最后一次复位不是来自/RST 引脚。 1: 最后一次复位来自/RST 引脚。							

19. 振荡器和时钟选择

C8051F336/7/8/9 器件有一个可编程内部高频振荡器、一个可编程内部低频振荡器和一个外部振荡器驱动电路。可以通过对 OSCICN 和 OSCICL 寄存器编程来使能/禁止内部高频振荡器和调节其输出频率（如图 19.1 所示）；可以通过对 OSCLCN 寄存器编程来使能/禁止内部低频振荡器和调节其输出频率。系统时钟可以由外部振荡器电路或任何一个内部振荡器提供。两个内部振荡器都提供了分频选择功能。

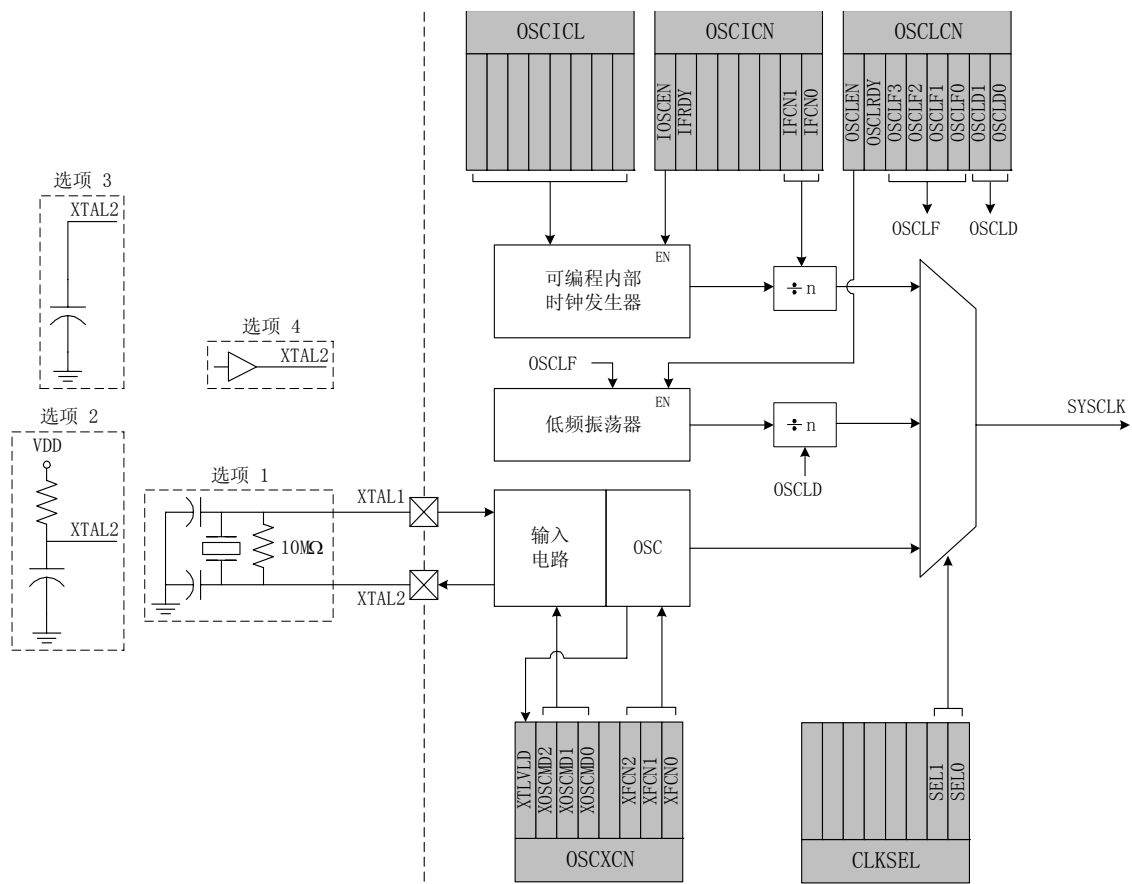


图 19.1 振荡器框图

19.1 系统时钟选择

寄存器 CLKSEL 中的 CLKSL[1:0]位选择用作系统时钟的振荡源。要选择外部振荡器作为系统时钟时，CLKSL[1:0]位必须被设置为 01b。当选择内部振荡器作为系统时钟时，外部振荡器仍然可以给某些外设（定时器、PCA）提供时钟。系统时钟可以在内部高频振荡器、外部振荡器及内部低频振荡器之间自由切换，只要所选择的时钟源被使能并稳定运行。

内部高频振荡器和低频振荡器的启动时间很短，因此可以在使能内部振荡器的 OSCICN 写操作之后立即选择内部振荡器为系统时钟。外部 RC 和 C 方式通常不需要起动时间。

外部晶体和陶瓷谐振器通常需要较长的起动时间，应待其稳定后方可用作系统时钟。当外部振荡器稳定后，晶体有效标志（寄存器 OSCXCN 中的 XTLVLD）被硬件置 1。在晶体方式，为了防止读到假 XTLVLD 标志，软件在使能外部振荡器和检查 XTLVLD 之间至少应延时 1ms。

SFR 定义 19.1 CLKSEL：时钟选择寄存器

R	R	R	R	R	R	R/W	R/W	复位值
-	-	-	-	-	-	CLKSL1	CLKSL0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0xA9
位 7-2: 未用。读 = 000000b，写 = 忽略。								
位 1-0: CLKSL[1:0]: 系统时钟源选择位								
00: 系统时钟取自内部高频振荡器，分频数由 OSCICN 寄存器中的 IFCN 位决定。								
01: 系统时钟取自外部振荡器。								
10: 系统时钟取自内部低频振荡器，分频数由 OSCLCN 寄存器中的 OSCLD 位决定。								
11: 保留。								

19.1 可编程内部高频（H-F）振荡器

所有 C8051F336/7/8/9 器件包含一个可编程内部高频振荡器，该振荡器在系统复位后被默认为系统时钟。内部振荡器的周期可以通过 OSCICL 寄存器编程，见 SFR 定义 19.2。

C8051F336/7/8/9 器件的 OSCICL 已经过工厂校准，对应的基频为 24.5MHz。

注意：系统时钟可以从内部振荡器分频得到，分频数由寄存器 OSCICN 中的 IFCN 位设定，可为 1、2、4 或 8。复位后的缺省分频数为 8。

19.2.1 内部振荡器挂起方式

当软件向 SUSPEND 位（OSCICN.5）写 1 时，内部振荡器被挂起。如果系统时钟源自内部振荡器，则外设和 CIP-51 的输入时钟被停止，直到有下面的事件之一发生：

- 端口 0 匹配事件。
- 端口 1 匹配事件。
- 比较器 0 被使能且输出为逻辑 0。
- 定时器 3 溢出。

当有一个内部振荡器唤醒事件发生时，不论该事件是否产生中断，内部振荡器、CIP-51 和受其影响的外设都恢复正常操作。CPU 从写 SUSPEND 那条指令的下一条指令恢复执行。

SFR 定义 19.2 OSCICL：内部 H-F 振荡器校准寄存器

R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-								可变
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0xB3
位 7： 未用。读=0b，写 = 忽略。								
位 6-0： OSCICL： 内部振荡器校准寄存器。								
该寄存器决定内部振荡器的周期。当其值被设置为 0000000b 时，H-F 振荡器工作在最高频率；当被设置为 1111111b 时，H-F 振荡器工作在最低频率。对于 C8051F33x 器件，复位值已经过工厂校准，产生的内部振荡器频率为 24.5MHz。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 19.3 OSCICN: 内部 H-F 振荡器控制寄存器

R/W	R	R/W	R	R	R	R/W	R/W	复位值
IOSCEN	IFRDY	SUSPEND	STSYNC	-	-	IFCN1	IFCN0	11000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xB2
位 7:	IOSCEN: 内部 H-F 振荡器使能位 0: 内部 H-F 高频振荡器禁止。 1: 内部 H-F 振荡器使能。							
位 6:	IFRDY: 内部 H-F 振荡器频率准备好标志 0: 内部 H-F 振荡器未运行在编程频率。 1: 内部 H-F 振荡器按编程频率运行。							
位 5:	SUSPEND: 内部振荡器挂起使能位 向该位写 1 将内部振荡器置于 SUSPEND 模式。当有一个 SUSPEND 模式唤醒事件发生时，内部振荡器恢复运行。							
位 4:	挂起定时器同步位 该位用于指示何时读和写与挂起唤醒定时器相关的寄存器是安全的（有关挂起唤醒的详细信息见 17.3 节）。如果一个非挂起唤醒定时器的挂起唤醒源使振荡器退出挂起方式，可能要过三个定时器时钟之后才能读或写该定时器。当 STSYNC 的读出值为 1 时，不能读和写该定时器的寄存器。当 STSYNC 的读出值为 0 时，可以安全地读和写该定时器的寄存器。							
位 3-2:	未用。读 = 00b，写 = 忽略。							
位 1-0:	IFCN1-0: 内部 H-F 振荡器频率控制位 00: SYSCLK 为内部 H-F 振荡器 8 分频。 01: SYSCLK 为内部 H-F 振荡器 4 分频。 10: SYSCLK 为内部 H-F 振荡器 2 分频。 11: SYSCLK 为内部 H-F 振荡器输出（不分频）。							

19.3 可编程内部低频（L-F）振荡器

C8051F336/7/8/9 器件包含一个可编程低频内部振荡器，该振荡器被校准到标称频率 80KHz。该低频振荡器电路包含一个分频器，分频数由寄存器 OSCLCN 中的 OSCLD 位设定，可为 1、2、4 或 8（见 SFR 定义 19.4）。此外，可用 OSCLF[3:0]位调节该振荡器的输出频率。

19.3.1 内部 L-F 振荡器标定

定时器 2 和定时器 3 具有捕捉功能，当驱动它们的时基已知时，可以用该功能获取内部 L-F 振荡器的频率。当定时器 2 或定时器 3 被配置为 L-F 振荡器捕捉方式时，低频振荡器输出的下降沿（定时器 2）或上升沿（定时器 3）将导致相应定时器发生捕捉事件。当捕捉事件发生时，定时器的当前计数值（TMRnH:TMRnL）被复制到定时器的重载寄存器（TMRnRLH:TMRnRLL）。根据两次相邻定时器捕捉值的差值，可以计算低频振荡器的周期。然后可以调整 OSCLF 位以产生期望的振荡器频率。

SFR 定义 19.4 OSCLCN: 内部 L-F 振荡器控制寄存器

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	复位值
OSCLCN	OSCLRDY	OSCLF3	OSCLF2	OSCLF1	OSCLF0	OSCLD1	OSCLD0	00xxxx00
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xE3
<p>位 7: OSCLCN: 内部 L-F 振荡器使能位 0: 内部 L-F 振荡器禁止。 1: 内部 L-F 振荡器使能。</p> <p>位 6: OSCLRDY: 内部 L-F 振荡器频率准备好标志 0: 内部 L-F 振荡器频率未稳定。 1: 内部 L-F 振荡器频率已稳定。</p> <p>位 5-2: OSCLF[3:0]: 内部 L-F 振荡器频率控制位。 内部 L-F 振荡器频率的微调控制位。当这些位被设置为 0000b 时，L-F 振荡器工作在最高频率；被设置为 1111b 时，L-F 振荡器工作在最低频率。</p> <p>位 1-0: OSCLD[1:0]: 内部 L-F 振荡器分频位 00: 选择 8 分频。 01: 选择 4 分频。 10: 选择 2 分频。 11: 选择不分频。</p>								

19.4 外部振荡器驱动电路

外部振荡器电路可以驱动外部晶体、陶瓷谐振器、电容或 RC 网络。也可以使用一个外部 CMOS 时钟提供系统时钟。对于晶体和陶瓷谐振器配置，晶体/陶瓷谐振器必须并接到 XTAL1 和 XTAL2 引脚（见图 19.1，选项 1），还必须在 XTAL1 和 XTAL2 引脚之间并接一个 $10M\Omega$ 的电阻。对于 RC、电容或 CMOS 时钟配置，时钟源应接到 XTAL2 引脚（见图 19.1，选项 2、3、4）。必须在 OSCXCN 寄存器中选择外部振荡器类型，还必须正确选择频率控制位 XFCN（见 SFR 定义 19.5）。

注意：当使用外部振荡器电路时，必须对所用端口引脚进行配置。当外部振荡器电路被配置为晶体/谐振器方式时，端口引脚 P0.2 和 P0.3 分别被用作 XTAL1 和 XTAL2。当外部振荡器电路被配置为 RC、电容或 CMOS 时钟方式时，端口引脚 P0.3 被用作 XTAL2。端口 I/O 交叉开关应被配置为跳过被振荡器占用的引脚。有关交叉开关配置的详细信息见“20.3 优先权交叉开关译码器”。另外，当在晶体/陶瓷谐振器、电容或 RC 方式使用外部振荡器电路时，应将所用的端口引脚配置为**模拟输入**。在 CMOS 时钟方式，应将所用的端口引脚配置为**数字输入**。有关端口输入方式选择的详细信息见“20.4 端口 I/O 初始化”。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 19.5 OSCXCN: 外部振荡器控制寄存器

R	R/W	R/W	R/W	R	R/W	R/W	R/W	复位值
XTLVLD	XOSCMD2	XOSCMD1	XOSCMD0	-	XFCN2	XFCN1	XFCN0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xB1

位 7: XTLVLD: 晶体振荡器有效标志
 (只读, 在 XOSCMD=11x 时)
 0: 晶体振荡器未用或未稳定。
 1: 晶体振荡器正在运行并且工作稳定。

位 6-4: XOSCMD[2:0]: 外部振荡器方式位
 00x: 外部振荡器电路关闭。
 010: 外部 CMOS 时钟方式。
 011: 外部 CMOS 时钟方式二分频。
 100: RC 振荡器方式。
 101: 电容振荡器方式。
 110: 晶体振荡器方式。
 111: 晶体振荡器方式二分频。

位 3: 保留。读 = 0, 写 = 忽略。

位 2-0: XFCN[2:0]: 外部振荡器频率控制位。
 000-111: 见下表

XFCN	晶体 (XOSCMD=11x)	RC(XOSCMD=10x)	C(XOSCMD=10x)
000	$F \leq 32\text{kHz}$	$F \leq 25\text{kHz}$	K 因子= 0.87
001	$32\text{kHz} < f \leq 84\text{kHz}$	$25\text{kHz} < f \leq 50\text{kHz}$	K 因子= 2.6
010	$84\text{kHz} < f \leq 225\text{kHz}$	$50\text{kHz} < f \leq 100\text{kHz}$	K 因子= 7.7
011	$225\text{kHz} < f \leq 590\text{kHz}$	$100\text{kHz} < f \leq 200\text{kHz}$	K 因子= 22
100	$590\text{kHz} < f \leq 1.5\text{MHz}$	$200\text{kHz} < f \leq 400\text{kHz}$	K 因子= 65
101	$1.5\text{MHz} < f \leq 4\text{MHz}$	$400\text{kHz} < f \leq 800\text{kHz}$	K 因子= 180
110	$4\text{MHz} < f \leq 10\text{MHz}$	$800\text{kHz} < f \leq 1.6\text{MHz}$	K 因子= 664
111	$10\text{MHz} < f \leq 30\text{MHz}$	$1.6\text{MHz} < f \leq 3.2\text{MHz}$	K 因子= 1590

晶体方式 (电路见图 19.1, 选项 1; XOSCMD=11x)
 选择与晶体振荡器频率匹配的 XFCN 值。

RC 方式 (电路见图 19.1, 选项 2; XOSCMD=10x)
 选择与频率范围匹配的 XFCN 值:
 $f = 1.23(10^3)/(R \times C)$, 其中:
 f = 振荡器频率 (MHz)
 C = 电容值 (pF)
 R = 上拉电阻值 (kΩ)

C 方式 (电路见图 19.1, 选项 3; XOSCMD=10x)
 根据所期望的振荡器频率选择 K 因子 (KF):
 $f = KF/(C \times VDD)$, 其中:
 f = 振荡器频率 (MHz)
 C = XTAL2 引脚的电容值 (pF)
 VDD = MCU 的电源电压值 (伏)

19.4.1 外部晶体示例

如果使用晶体或陶瓷谐振器作为 MCU 的外部振荡源，则电路应被配制为图 19.1 中的选项 1。应从 SFR 定义 19.5 (OSCXCN 寄存器) 中的晶体列选择外部振荡器频率控制值 (XFCN)。例如，一个 11.0592MHz 的晶体要求 XFCN 设置值为 111b，而一个 32.768 KHz 的钟表晶体要求 XFCN 设置值为 001b。在外部的 32.768 KHz 振荡器稳定后，XFCN 设置值可以被切换到 000 以节省功耗。建议在将系统时钟切换到任何一个外部振荡源之前使能时钟丢失检测器。

当晶体振荡器第一次被使能时，振荡器幅度检测电路需要一个建立时间来达到合适的偏置。在使能晶体振荡器和检查 XTLVLD 位之间引入 1ms 的延时可以防止提前将系统时钟切换到外部振荡器。在晶体振荡器稳定之前就切换到外部晶体振荡器可能产生不可预料的后果。建议的步骤如下：

1. 强制 XTAL1 和 XTAL2 引脚为低电平，这可通过使能交叉开关并向与 XTAL1 和 XTAL2 相关的引脚写 0 来实现。
2. 用寄存器 P0MDIN 将 XTAL1 和 XTAL2 配置为模拟输入。
3. 使能外部振荡器。
4. 等待至少 1ms。
5. 查询 XTLVLD => 1。
6. 使能时钟丢失检测器。
7. 将系统时钟切换到外部振荡器。

注意，晶体振荡器电路对 PCB 布局非常敏感。应将晶体尽可能地靠近器件的 XTAL 引脚，布线应尽可能地短并用地平面屏蔽，以防止其它引线引入噪声或干扰。

外部晶体配置电路中的电容提供晶体正确振荡所需要的负载电容。从晶体来看，这两个电容是“串联”的，与 XTAL1 和 XTAL2 引脚的寄生电容“并联”。

注：负载电容的大小取决于晶体的振荡频率和生产厂家。计算负载电容值时请参考晶体的数据手册。

例如，使用 32.768 KHz 的音叉晶体，建议的负载电容为 12.5 pF，应采用图 19.1 中的配置选项 1。负载电容与 XTAL 引脚的寄生电容的总值应为 25 pF。如果每个 XTAL 引脚的寄生电容值为 3 pF，则采用 22 pF 的负载电容后跨接在晶体两端的等效电容为 12.5 pF，如图 19.2 所示。

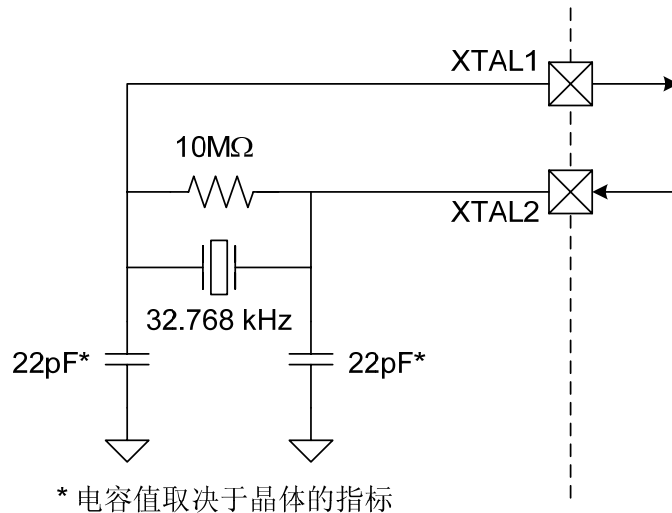


图 19.2 32.768 KHz 外部晶体振荡器连接图

19.4.2 外部 RC 示例

如果使用外部 RC 网络作为 MCU 的外部振荡源，则配置电路为图 19.1 中的选项 2。电容不应大于 100pF；但当电容值很小时，PCB 的寄生电容将在总电容中占支配地位，使频率偏差很大。为了确定 OSCXCN 寄存器所需要的外部振荡器频率控制值 (XFCN)，首先根据方程 1 选择能产生所要求的振荡频率的 RC 网络值。方程 1 中： f = 振荡器频率 (MHz)， C = 电容值 (pF)， R = 上拉电阻值 (kΩ)。

方程 1. RC 方式振荡器频率

$$f = 1.23 \times 10^3 / (R \times C)$$

例如：如果所希望的频率是 100kHz，选 $R = 246\text{k}\Omega$ 和 $C = 50\text{pF}$ ：

$$f = 1.23 \times 10^3 / RC = 1.23 \times 10^3 / [246 \times 50] = 0.1\text{MHz} = 100\text{kHz}$$

查 SFR 定义 19.5 中的表，得到所需要的 XFCN 值为 010b。

19.4.3 外部电容示例

如使用外部电容作为 MCU 的外部振荡源，则电路为图 19.1 中的选项 3。电容不应大于 100pF；但当电容值很小时，PCB 的寄生电容将在总电容中占支配地位，使频率偏差很大。为了确定 OSCXCN 寄存器所需要的外部振荡器频率控制值 (XFCN)，选择要用的电容并根据方程 2 计算振荡频率。

方程 2. C 方式振荡器频率

$$f = KF / (C \times VDD)$$

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

例如：假设 $VDD = 3.0V$ 和 $f = 150\text{ KHz}$ ：

$$f = KF / (C \times VDD)$$

$$0.150\text{ MHz} = KF / (C \times 3.0)$$

所需要的频率大约为 150kHz，从 SFR 定义 19.5 (OSCXCN) 的表中选择 K 因子，得到 $KF = 22$ ：

$$0.150\text{ MHz} = 22 / (C \times 3.0)$$

$$C \times 3.0 = 22 / 0.150\text{ MHz}$$

$$C = 146.6 / 3.0\text{ pF} = 48.8\text{ pF}$$

因此，本例中要用的 XFCN 值为 011b， $C = 50\text{ pF}$ 。

20. 端口输入/输出

数字和模拟资源可以通过 17 (C8051F336/337) 或 21 (C8051F338/339) 个 I/O 引脚使用。P0.0 ~ P2.3 可以被定义为通用 I/O (GPIO)，可以被分配给内部数字资源，可以被分配给模拟功能，如图 20.3 所示。C8051F338/339 的端口引脚 P2.4 和 C8051F336/337 的端口引脚 P2.0 可以被用作 GPIO 并且与 C2 接口数据信号 (C2D) 共享。设计者完全控制数字功能的引脚分配，只受 I/O 引脚数的限制。这种资源分配的灵活性是通过使用优先权交叉开关译码器实现的。注意，不论交叉开关的设置如何，端口 I/O 引脚的状态总是可以被读到相应的端口锁存器。

交叉开关根据优先权译码表 (图 20.3 和图 20.4) 为所选择的内部数字资源分配 I/O 引脚。寄存器 XBR0 和 XBR1 (见 SFR 定义 20.1 和 SFR 定义 20.2) 用于选择内部数字功能。

所有端口 I/O 都耐 5V 电压 (端口 I/O 单元示于图 20.2)。端口 I/O 单元可以被配置为漏极开路或推挽方式 (在端口输出方式寄存器 PnMDOUT 中设置，n = 0,1,2)。表 6.3 给出了端口 I/O 的电气特性。

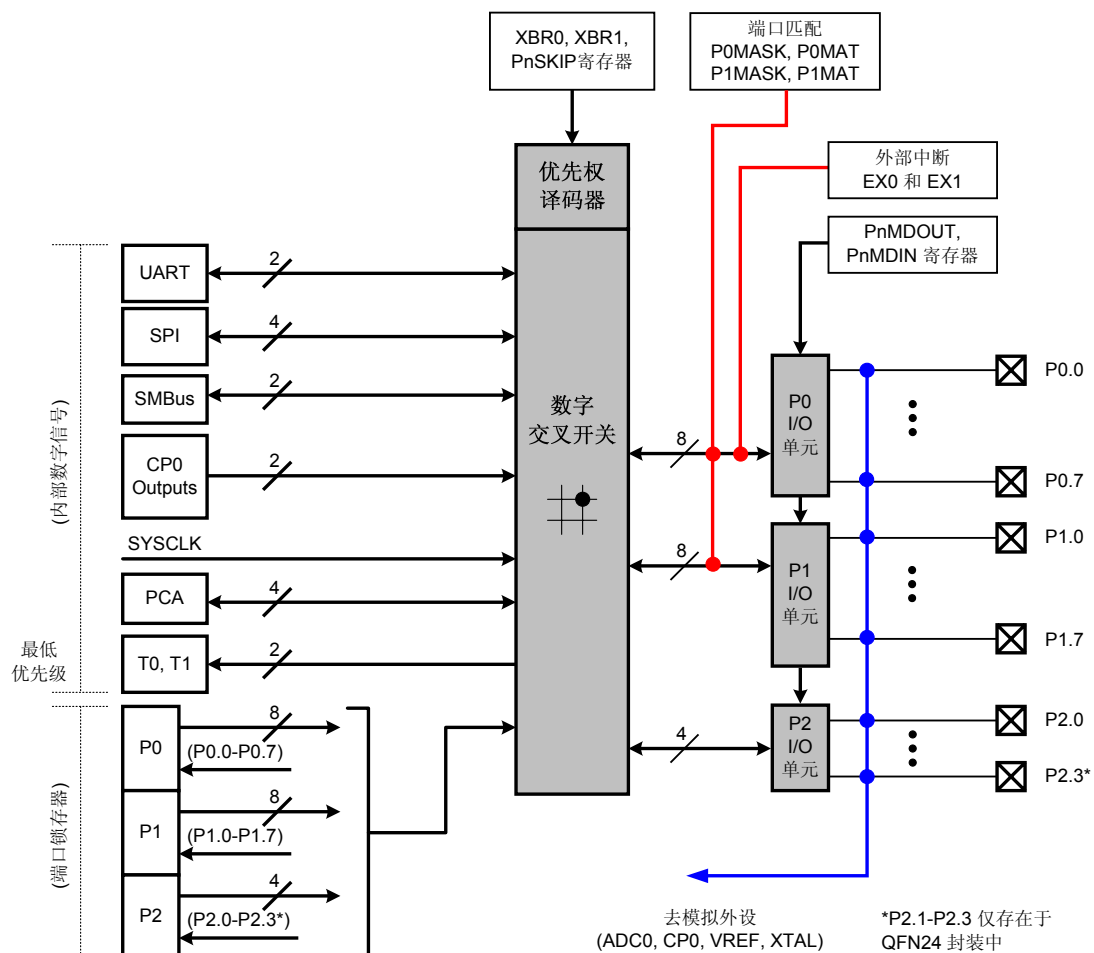


图 20.1 端口 I/O 功能框图

20.1 端口 I/O 的工作方式

P0.0 ~ P2.3 使用如图 20.2 所示的端口 I/O 单元。每个端口 I/O 单元都可以由软件用 PnMDIN 寄存器配置为模拟 I/O 或数字 I/O。在复位后，所有端口 I/O 单元均默认为高阻状态且弱上拉使能，直到交叉开关被使能（XBARE = 1）。

20.1.1 端口引脚配置为模拟 I/O

任何用作比较器或 ADC 输入、外部振荡器输入/输出、VREF 或 IDAC 输出的引脚都应被配置为模拟 I/O（PnMDIN.n = 0）。当一个引脚被配置为模拟 I/O 时，其弱上拉、数字驱动器和数字接收器均被禁止。读被配置为模拟 I/O 的引脚将总是返回 0 值。

将引脚配置为模拟 I/O 可以节省功耗并将其与数字干扰隔离。被配置为数字输入的引脚仍可被模拟外设使用，但不建议这样做，因为这样可能导致测量误差。

20.1.2 端口引脚配置为数字 I/O

任何被数字外设（UART、SPI、SMBus 等）、外部数字事件捕捉功能使用的引脚和作为 GPIO 使用的引脚都应被配置为数字 I/O（PnMDIN.n = 1）。对于数字 I/O 引脚，必须用 PnMDOUT 寄存器选择其输出方式为漏极开路或推挽方式。

推挽输出（PnMDOUT.n = 1）根据端口引脚的输出逻辑值将端口引脚驱动到 VDD/DC+ 或 GND。漏极开路输出时高端驱动器被禁止，因此当输出逻辑值为 0 时，端口引脚只被驱动到 GND；而当输出逻辑值为 1 时，端口引脚变成高阻输入（高端和低端驱动器均被断开）。

当一个数字 I/O 单元被置于高阻状态时，一个弱上拉晶体管将端口引脚拉向 VDD 电源电压，以保证数字输入处于一个有定义的逻辑状态。个数字 I/O 单元被驱动到 GND 时，弱上拉被禁止以节省功耗。通过将 WEAKPUD 置 1 可以总体禁止弱上拉。用户应保证数字 I/O 总是被内部或外部拉向或驱动到一个有效的逻辑状态，以使功耗最小。被配置为数字 I/O 的端口引脚总是读回端口引脚的逻辑状态，而与该端口引脚的输出逻辑值无关。

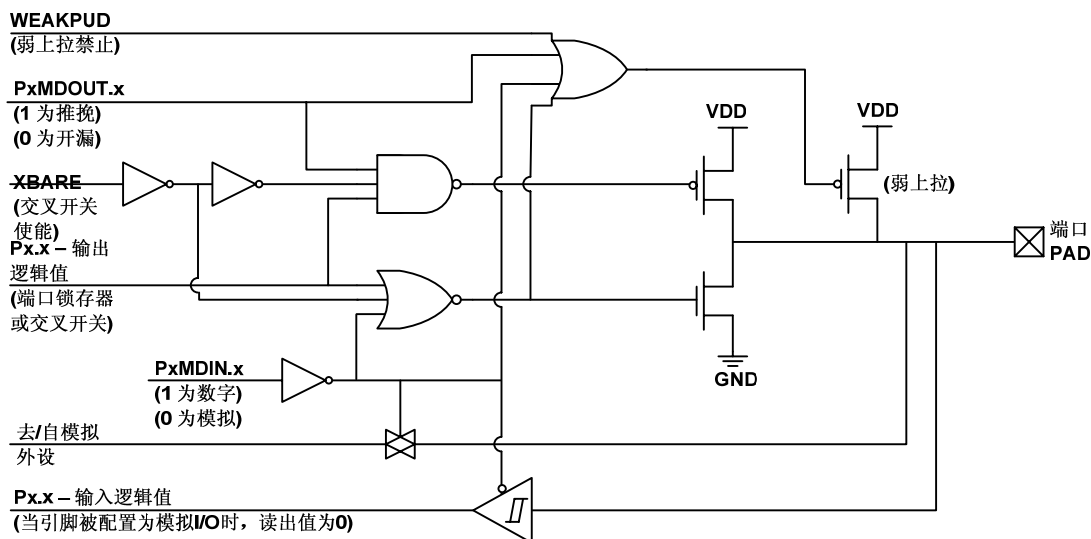


图 20.2 端口 I/O 单元框图

20.1.3 端口 I/O 与 5V 逻辑的接口

被配置为数字、漏极开路操作的所有端口 I/O 都能与工作在高于 VDD 但小于 5.25V 的电源电压的数字逻辑接口。对于大多数系统来说，通常需要一个连接到较高电源电压的上拉电阻。

注意：在一个多电压接口中，外部上拉电阻的选择应满足：当电源电压位于（VDD+0.6V）与（VDD+1.0V）之间时，应允许至少 150 μ A 流入端口引脚。一旦端口引脚的电压增加到高于这一范围，流入端口引脚的电流为最小。

20.2 端口 I/O 引脚分配给模拟和数字功能

端口 I/O 引脚 P0.0 ~ P2.3 可以被分配给不同的模拟、数字和外部中断功能。分配给模拟功能的端口引脚应被配置为模拟 I/O，分配给数字或外部中断功能的端口引脚应被配置为数字 I/O。

20.2.1 端口引脚分配给模拟功能

表 20.1 列出了所有需要分配端口 I/O 的模拟功能。为这些模拟功能选择的引脚在 PnSKIP 寄存器中的对应位应被置 1。这样将保留这些引脚为模拟功能所用，而不允许这些引脚被交叉开关分配。表 20.1 列出了端口 I/O 到每个模拟功能的潜在映射。

表 20.1 模拟功能的端口 I/O 分配

模拟功能	潜在的可分配端口引脚	用于引脚分配的SFR
ADC输入	P0.0 ~ P2.3	AMX0P, AMX0N, PnSKIP
比较器0输入	P0.0 ~ P2.3	CPT0MX, PnSKIP
电压基准（VREF0）	P0.0	REF0CN, PnSKIP
电流DAC输出（IDA0）	P0.1	IDA0CN, PnSKIP
晶体模式的外部振荡器（XTAL1）	P0.2	OSCXCN, PnSKIP
RC、C或晶体模式的外部振荡器（XTAL2）	P0.3	OSCXCN, PnSKIP

20.2.2 端口引脚分配给数字功能

任何未分配给模拟功能的端口 I/O 都可以被分配给数字功能或或用作 GPIO。大多数数字功能依赖交叉开关分配引脚；但是，有些数字功能以与前面列出的模拟功能类似的方式旁路了交叉开关。这些数字功能使用的端口引脚和任何被选择作为 GPIO 使用的端口引脚在 PnSKIP 寄存器中的对应位应被置 1。表 20.2 列出了所有可用的数字功能和端口 I/O 到每个数字功能的潜在映射。

表 20.2 数字功能的端口 I/O 分配

数字功能	潜在的可分配端口引脚	用于引脚分配的SFR
UART0, SPI0, SMBus, CP0, CP0A, SYSCLK, PCA0 (CEX0-2 和 ECI), T0或T1。	任何可供交叉开关分配的端口引脚。这包括PnSKIP位为0的P0.0 ~ P2.3引脚。 注：交叉开关总是将UART0引脚分配到P0.4和P0.5。	XBAR0, XBAR1
用作GPIO的任何引脚	P0.0 ~ P2.4	P0SKIP, P1SKIP, P2SKIP

20.2.3 端口引脚分配给外部数字事件捕捉功能

外部数字事件捕捉功能可用于触发中断或将器件从一个低功耗方式唤醒（当一个数字 I/O 引脚上发生状态改变时）。数字事件捕捉功能不需要专用的引脚，它既可以使用 GPIO 引脚工作（PnSKIP = 1），也可以使用交叉开关所使用的引脚工作（PnSKIP = 0）。外部数字事件捕捉功能不能在那些被配置为模拟 I/O 的引脚上使用。表 20.3 列出了所有可用的外部数字事件捕捉功能。

表 20.3 外部数字事件捕捉功能的端口 I/O 分配

数字功能	潜在的可分配端口引脚	用于引脚分配的SFR
外部中断0	P0.0 ~ P0.7	IT01CF
外部中断1	P0.0 ~ P0.7	IT01CF
端口匹配	P0.0 ~ P1.7	P0MASK, P0MAT P1MASK, P1MAT

20.3 优先权交叉开关译码器

优先权交叉开关译码器（图 20.3）为每个 I/O 功能分配优先权，从优先权最高的 UART0 开始。当一个数字资源被选择时，尚未分配的端口引脚中的最低位被分配给该资源（UART0 例外，它总是被分配到引脚 P0.4 和 P0.5）。如果一个端口引脚已经被分配，则交叉开关在为下一个被选择的资源分配引脚时将跳过该引脚。此外，交叉开关还将跳过在 PnSKIP 寄存器中被置 1 的那些位所对应的引脚。PnSKIP 寄存器允许软件跳过那些被用作模拟输入、特殊功能或 GPIO 的引脚。

注意：如果一个端口引脚被一个外设使用而不经交叉开关，则该引脚在 PnSKIP 寄存器中的对应位应被置 1。这种情况适用于 P0.0（如果使用 VREF）、P0.3 和/或 P0.2（如果外部振荡器电路被使能）、P0.6（如果 ADC 或 IDAC 被配置为使用外部转换启动信号 CNVSTR）和任何被选择的 ADC 或比较器输入。交叉开关跳过那些被选择的引脚，就像它们已经被分配了一样，移向下一个未被分配的引脚。图 20.3 示出了没有引脚被跳过（P0SKIP, P1SKIP = 0x00）的优先权交叉开关译码表；图 20.4 给出了 XTAL1(P0.2)脚和 XTAL2(P0.3)脚被跳过情况下（P0SKIP = 0x0C）的交叉开关优先权译码表。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

	P0								P1								P2				
特殊功能信号	VREF	IDA	x1	x2	CNVSTR																
PIN I/O	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1 ²	2 ²	3 ²	4 ²
TX0																					
RX0																					
SCK																					
MISO																					
MOSI																					
NSS ¹																					
SDA																					
SCL																					
CP0																					
CP0A																					
SYSCCLK																					
CEX0																					
CEX1																					
CEX2																					
ECI																					
T0																					
T1																					
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	P0SKIP[0:7]								P1SKIP[0:7]								P2SKIP[0:3]				
	可供外设使用的端口引脚								注:												
特殊功能信号	特殊功能信号不由交叉开关分配引脚。当这些信号被使能时，交叉开关必须被手动配置为跳过对应的端口引脚。								1. NSS 只在4线SPI方式引出								2. 引脚 P2.1-P2.4 仅存在于 QFN24 封装中				

图 20.3 没有引脚被跳过的交叉开关优先级译码表

20.4 端口 I/O 初始化

端口 I/O 初始化包括以下步骤：

1. 用端口输入方式寄存器（PnMDIN）选择所有端口引脚的输入方式（模拟或数字）。
2. 用端口输出方式寄存器（PnMDOUT）选择所有端口引脚的输出方式（漏极开路或推挽）。
3. 用端口跳过寄存器（PnSKIP）选择应被交叉开关跳过的那些引脚。
4. 将引脚分配给要使用的外设。
5. 使能交叉开关（XBARE = 1）。

所有端口引脚都必须被配置为模拟或数字输入。被用作比较器或 ADC 输入的任何引脚都应被配置为模拟输入。当一个引脚被配置为模拟输入时，它的弱上拉、数字驱动器和数字接收器都被禁止，这可以节省功耗并减小模拟输入的噪声。被配置为数字输入的引脚可以被模拟外设使用，但不建议这样做。

此外，应将交叉开关配置为跳过所有被用作模拟输入的引脚（通过将 PnSKIP 寄存器中的对应位置 1 来实现）。端口输入方式在 PnMDIN 寄存器中设置，其中 1 表示数字输入，0 表示模拟输入。复位后所有引脚的缺省设置都是数字输入。对 PnMDIN 寄存器的详细说明见 SFR 定义 20.8。

I/O 引脚的输出驱动器特性由端口输出方式寄存器 PnMDOUT 中的对应位决定，每个端口输出驱动器都可被配置为漏极开路或推挽方式。不管交叉开关是否将端口引脚分配给某个数字外设，都需要对端口驱动器的输出方式进行设置。唯一的例外是 SMBus 引脚（SDA，SCL），不管 PnMDOUT 的设置如何，这两个引脚总是被配置为漏极开路。

当 XBR1 寄存器中的 WEAKPUD 位为 0 时，输出方式为漏极开路的那些引脚的弱上拉被使能。WEAKPUD 不影响推挽端口 I/O。当漏极开路输出被驱动到逻辑 0 时，弱上拉被自动关断（禁止）以避免不必要的功率消耗。

寄存器 XBR0 和 XBR1 必须被装入正确的值以选择所需要的数字 I/O 功能。置位 XBR1 中的 XBARE 位将使能交叉开关。不管 XBRn 寄存器的设置如何，在交叉开关被使能之前，外部引脚保持标准端口 I/O 方式（输入）。对于给定的 XBRn 设置，可以使用优先权译码表确定 I/O 引脚分配；另一种方法是使用 Silicon Labs IDE 软件的配置向导功能来确定基于 XBRn 寄存器设置的端口 I/O 引脚分配。

注意：为使端口引脚工作在标准端口 I/O 输出方式，交叉开关必须被使能。当交叉开关被禁止时，端口输出驱动器被禁止。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.1 XBR0: 端口 I/O 交叉开关寄存器 0

R	R	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-	-	CP0AE	CP0E	SYSCKE	SMB0E	SPI0E	URT0E	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xE1
位 7-6: 未用。读 = 00b, 写 = 忽略。								
位 5: CP0AE: 比较器 0 异步输出使能位								
0: CP0A 不连到端口引脚。								
1: CP0A 连到端口引脚。								
位 4: CP0E: 比较器 0 输出使能位								
0: CP0 不连到端口引脚。								
1: CP0 连到端口引脚。								
位 3: SYSCKE: /SYSCLK 输出使能位								
0: /SYSCLK 不连到端口引脚;								
1: /SYSCLK 连到端口引脚。								
位 2: SMB0E: SMBus I/O 使能位								
0: SMBus I/O 不连到端口引脚。								
1: SMBus I/O 连到端口引脚。								
位 1: SPI0E: SPI I/O 使能位								
0: SPI I/O 不连到端口引脚。								
1: SPI I/O 连到端口引脚。注意: SPI 可以被分配 3 个或 4 个 GPIO 引脚。								
位 0: URT0E: UART I/O 使能位								
0: UART I/O 不连到端口引脚。								
1: UART TX0, RX0 连到端口引脚 P0.4 和 P0.5。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.2 XBRI: 端口 I/O 交叉开关寄存器 1

R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	复位值
WEAKPUD	XBARE	T1E	T0E	ECIE	-	PCA0ME		00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xE2
位 7:	WEAKPUD: 端口 I/O 弱上拉禁止位。 0: 弱上拉使能 (被配置为模拟方式的端口 I/O 除外)。 1: 弱上拉禁止。							
位 6:	XBARE: 交叉开关使能位 0: 交叉开关禁止。 1: 交叉开关使能。							
位 5:	T1E: T1 使能位 0: T1 不连到端口引脚。 1: T1 连到端口引脚。							
位 4:	T0E: T0 使能位 0: T0 不连到端口引脚。 1: T0 连到端口引脚。							
位 3:	ECIE: PCA0 外部计数输入使能位 0: ECI 不连到端口引脚。 1: ECI 连到端口引脚。							
位 2:	未用。读 = 0b, 写 = 忽略。							
位 1-0:	PCA0ME: PCA 模块 I/O 使能位 00: 所有的 PCA I/O 都不连到端口引脚。 01: CEX0 连到端口引脚。 10: CEX0、CEX1 连到端口引脚。 11: CEX0、CEX1、CEX2 连到端口引脚。							

20.5 端口匹配

端口匹配功能允许 P0 或 P1 口的逻辑值变化触发系统事件。存储在 PnMATCH 寄存器中的软件控制值定义了 P0 和 P1 期望或正常逻辑值。如果端口输入引脚上的逻辑电平与该软件控制值不匹配，则会产生一个端口不匹配事件。该功能允许在 P0 或 P1 输入引脚发生某种变化时软件会得到通知，与 XBRn 的设置无关。

PnMASK 寄存器用于选择 P0 和 P1 的哪个引脚应与 PnMATCH 寄存器比较。如果 (P0 & P0MASK) 不等于 (P0MATCH & P0MASK) 或如果 (P1 & P1MASK) 不等于 (P1MATCH & P1MASK)，则会产生一个端口不匹配事件。

端口不匹配事件可以用于产生中断或将器件从低功耗方式唤醒 (IDLE 和 SUSPEND)。有关中断和唤醒源的详细信息见中断和电源选项章节。

SFR 定义 20.3 P0MASK：端口 0 屏蔽寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0xFE
位 7-0: P0MASK[7:0]: 端口 0 屏蔽值 选择要与 P0MAT 中的对应位比较的端口引脚。 0: 对应的 P0.n 引脚逻辑值被忽略，不能产生端口不匹配事件。 1: 对应的 P0.n 引脚逻辑值与 P0MAT.n 比较。								

SFR 定义 20.4 P0MAT：端口 0 匹配寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0xFD
位 7-0: P0MAT[7:0]: 端口 0 匹配值 用于端口 0 的匹配比较值 (对于 P0MAT 中被置 1 的那些位)。 0: P0.n 引脚逻辑值与逻辑低电平比较。 1: P0.n 引脚逻辑值与逻辑高电平比较。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.5 P1MASK: 端口 1 屏蔽寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xEE
位 7-0: P1MASK[7:0]: 端口 1 屏蔽值 选择要与 P1MAT 中的对应位比较的端口引脚。 0: 对应的 P1.n 引脚逻辑值被忽略, 不能产生端口不匹配事件。 1: 对应的 P1.n 引脚逻辑值与 P1MAT.n 比较。								

SFR 定义 20.4 P1MAT: 端口 1 匹配寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xED
位 7-0: P1MAT[7:0]: 端口 1 匹配值 用于端口 1 的匹配比较值 (对于 P1MAT 中被置 1 的那些位)。 0: P1.n 引脚逻辑值与逻辑低电平比较。 1: P1.n 引脚逻辑值与逻辑高电平比较。								

20.6 访问和配置端口 I/O 的特殊功能寄存器

所有端口 I/O 都通过对应的特殊功能寄存器访问, 这些寄存器既可以按位寻址也可以按字节寻址。向端口写入时, 数值被锁存到 SFR 中, 以保持每个引脚上的输出数据值不变。读端口时总是返回端口输入引脚本身的逻辑状态, 而与 XBRn 的设置值无关 (即使在引脚被交叉开关分配给另一信号时, 端口寄存器总是读其对应的端口 I/O 引脚)。例外情况是执行目的操作数为端口锁存器的读-修改-写指令。对端口锁存器执行的读-修改-写指令如下: ANL、ORL、XRL、JBC、CPL、INC、DEC、DJNZ 和对端口 SFR 中的单个位执行的 MOV、CLR、SETB。这些指令读端口寄存器 (而不是引脚) 的值, 修改后再写回端口 SFR。

每个端口都有一个对应的 PnSKIP 寄存器, 该寄存器允许单独的端口引脚被分配给数字功能或被交叉开关跳过。应将所有用于模拟功能、GPIO 或专用数字功能 (如 EMIF) 的端口引脚的对应 PnSKIP 位置 1。

用端口输入方式寄存器 (PnMDIN) 定义端口引脚的输入方式。每个端口单元都可以被配置为模拟或数字 I/O。即使对于在 XBRn 中被选择的数字资源, 也需要进行这种选择 (不是自动的)。唯一的例外是 P2.4, 它只能被用作数字 I/O。

I/O 引脚的输出驱动器特性由端口输出方式寄存器 PnMDOUT 定义, 每个端口输出驱动器都可被配置为漏极开路或推挽方式。即使对于在 XBRn 中被选择的数字资源, 也需要进行这种选择 (不是自动的)。唯一的例外是 SMBus 引脚 (SDA, SCL), 不管 PnMDOUT 的设置如何, 这两个引脚总是被配置为漏极开路。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.7 P0：端口 0 寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0	11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： (可位寻址) 0x80
位 7-0: P0.[7:0]								
写 — 设置端口锁存器逻辑值。								
0: 设置输出锁存器为逻辑低电平。								
1: 设置输出锁存器为逻辑高电平。								
读 — 读那些被配置为数字 I/O 的端口引脚的逻辑状态。								
0: P0.n 端口引脚为逻辑低电平。								
1: P0.n 端口引脚为逻辑高电平。								

SFR 定义 20.8 P0MDIN：端口 0 输入方式寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0xF1
位 7-0: P0.7 – P0.0 的模拟配置位（分别对应）								
当端口引脚被配置为模拟输入时，其弱上拉、数字驱动器和数字接收器都被禁止。								
0: 对应的 P0.n 引脚被配置为模拟输入。								
1: 对应的 P0.n 引脚不配置为模拟输入。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.9 P0MDOUT: 端口 0 输出方式寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xA4
位 7-0: P0.7 – P0.0 的输出配置位（分别对应）。如果 P0MDIN 寄存器中的对应位为逻辑 0，则这些位被忽略。								
0: 对应的 P0.n 输出为漏极开路。								
1: 对应的 P0.n 输出为推挽方式。								

SFR 定义 20.10 P0SKIP: 端口 0 跳过寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xD4
位 7-0: P0SKIP[7:0]: 端口 0 交叉开关跳过使能位。								
这些位选择被交叉开关译码器跳过的端口 0 引脚。被用作模拟 I/O、特殊功能或 GPIO 的引脚应被交叉开关跳过。								
0: 对应的 P0.n 不被交叉开关跳过。								
1: 对应的 P0.n 被交叉开关跳过。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.11 P1: 端口 1 寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0	11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: (可位寻址) 0x90

位 7-0: P1[7:0]

写 — 设置端口锁存器逻辑值。

0: 设置输出锁存器为逻辑低电平。

1: 设置输出锁存器为逻辑高电平。

读 — 读那些被配置为数字 I/O 的端口引脚的逻辑状态。

0: P1.n 端口引脚为逻辑低电平。

1: P1.n 端口引脚为逻辑高电平。

SFR 定义 20.12 P1MDIN: 端口 1 输入方式寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xF2

位 7-0: P1MDIN[7:0]: P1.7 – P1.0 的模拟配置位（分别对应）

当端口引脚被配置为模拟输入时，其弱上拉、数字驱动器和数字接收器都被禁止。

0: 对应的 P1.n 引脚被配置为模拟输入。

1: 对应的 P1.n 引脚不配置为模拟输入。

SFR 定义 20.13 P1MDOUT: 端口 1 输出方式寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xA5

位 7-0: P1MDOUT[7:0]: P1.7 – P1.0 的输出配置位（分别对应）。如果 P1MDIN 寄存器中的对应位为逻辑 0，这些位被忽略。

0: 对应的 P1.n 输出为漏极开路。

1: 对应的 P1.n 输出为推挽方式。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.14 P1SKIP: 端口 1 跳过寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xD5

位 7-0: P1SKIP[7:0]: 端口 1 交叉开关跳过使能位。
这些位选择被交叉开关译码器跳过的端口 1 引脚。被用作模拟 I/O、特殊功能或 GPIO 的引脚应被交叉开关跳过。
0: 对应的 P1.n 不被交叉开关跳过。
1: 对应的 P1.n 被交叉开关跳过。

SFR 定义 20.15 P2: 端口 2 寄存器

R	R	R	R	R	R	R	R/W	复位值
-	-	-	P2.4	P2.3	P2.2	P2.1	P2.0	00011111
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xA0

位 7-5: 未使用。读 = 000b, 写 = 忽略。
位 4-0: P2[4:0]
写 — 设置端口锁存器逻辑值。
0: 设置输出锁存器为逻辑低电平。
1: 设置输出锁存器为逻辑高电平。
读 — 读那些被配置为数字 I/O 的端口引脚的逻辑状态。
0: P2.n 端口引脚为逻辑低电平。
1: P2.n 端口引脚为逻辑高电平。

注: 引脚 P2.1 ~ P2.4 仅在 QFN24 封装的器件中可用。

SFR 定义 20.16 P2MDIN: 端口 2 输入方式寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00011111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xF3

位 7-5: 未使用。读 = 000b, 写 = 忽略。
位 4-0: P2MDIN[4:0]: P2.4 – P2.0 的模拟配置位 (分别对应)
当端口引脚被配置为模拟输入时, 其弱上拉、数字驱动器和数字接收器都被禁止。
0: 对应的 P2.n 引脚被配置为模拟输入。
1: 对应的 P2.n 引脚不配置为模拟输入。

注: 引脚 P2.1 ~ P2.4 仅在 QFN24 封装的器件中可用。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 20.17 P2MDOUT: 端口 2 输出方式寄存器

R	R	R	R	R	R	R	R/W	复位值
-	-	-						00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xA6

位 7-5: 未使用。读 = 000b, 写 = 忽略。

位 4-0: P2MDOUT[4:0]: P2.4 – P2.0 的输出配置位 (分别对应)。如果 P2MDIN 寄存器中的对应位为逻辑 0, 这些位被忽略。

0: 对应的 P2.n 输出为漏极开路。

1: 对应的 P2.n 输出为推挽方式。

注: 引脚 P2.1 ~ P2.4 仅在 QFN24 封装的器件中可用。

SFR 定义 20.18 P2SKIP: 端口 2 跳过寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-	-	-						00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xD6

位 7-5: 未使用。读 = 000b, 写 = 忽略。

位 4-0: P2SKIP[4:0]: 端口 2 交叉开关跳过使能位。

这些位选择被交叉开关译码器跳过的端口 2 引脚。被用作模拟 I/O、特殊功能或 GPIO 的引脚应被交叉开关跳过。

0: 对应的 P2.n 不被交叉开关跳过。

1: 对应的 P2.n 被交叉开关跳过。

注: 引脚 P2.1 ~ P2.4 仅在 QFN24 封装的器件中可用。

21. SMBus

SMBus I/O 接口是一个二线的双向串行总线。SMBus 完全符合系统管理总线规范 1.1 版，与 I²C 串行总线兼容。系统控制器对接口的读写操作都是以字节为单位的，由 SMBus 接口自动控制数据的串行传输。在作为主或从器件时，数据传输的最大速率可达系统时钟频率的二十分之一（这可能比 SMBus 的规定速度要快，取决于所使用的系统时钟）。可以采用延长低电平时间的方法协调同一总线上不同速度的器件。

SMBus 可以工作在主和/或从方式，一个总线上可以有多个主器件。SMBus 提供了 SDA（串行数据）控制、SCL（串行时钟）产生和同步、仲裁逻辑以及起始/停止的控制和产生电路。SMBus 外设可以完全由软件驱动（即软件接受/拒绝从地址和产生 ACK），或者被使能为由硬件识别从地址和自动产生 ACK 以减小软件开销。SMBus 外设的原理框图和相关 SFR 示于图 21.1。

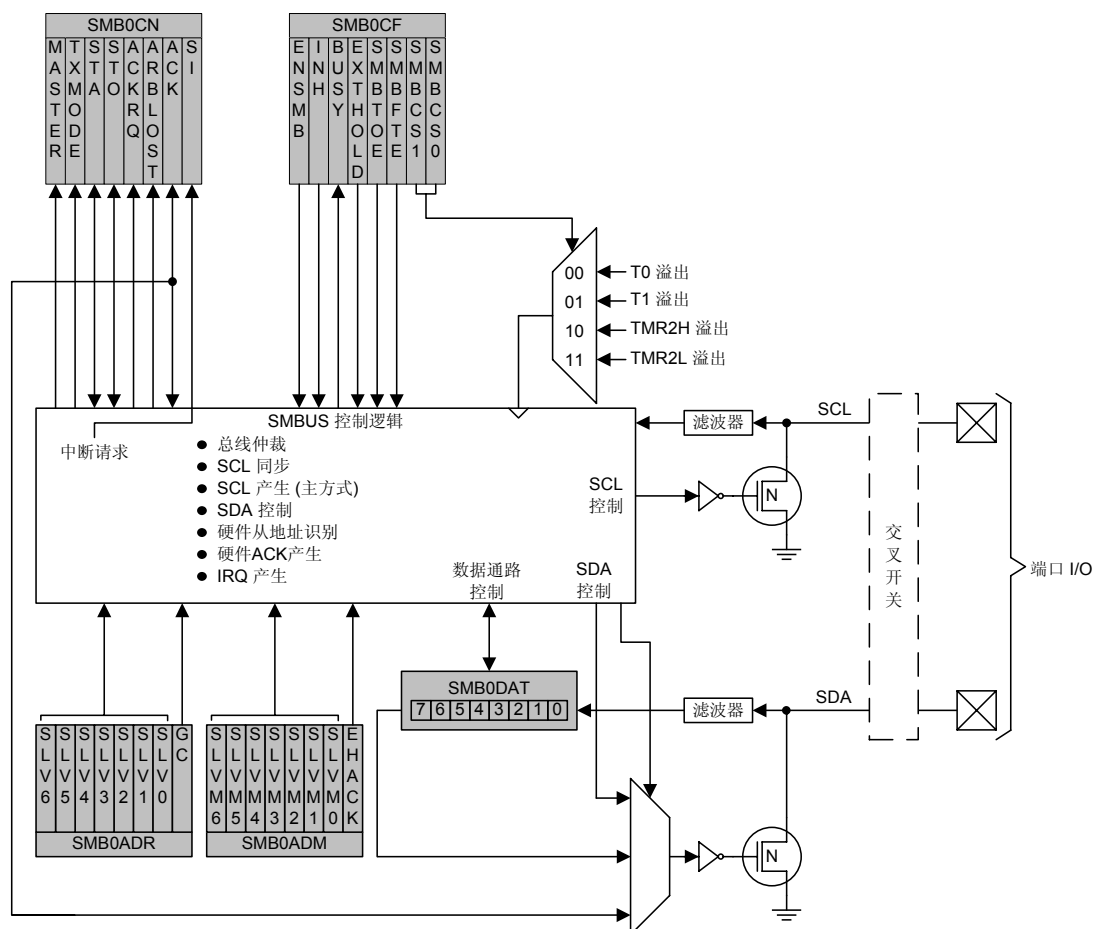


图 21.1 SMBus 原理框图

21.1 支持文档

假设读者熟悉或有条件阅读下列支持文档：

1. I²C 总线及使用（包含规范），菲利浦半导体。
2. I²C 总线规范—2.0 版，菲利浦半导体。
3. 系统管理总线规范—1.1 版。

21.2 SMBus 配置

图 21.2 给出了一个典型的 SMBus 配置。SMBus 接口的隐性状态电压可以在 3.0V 和 5.0V 之间，总线上不同器件的工作电压可以不同。SCL（串行时钟）和 SDA（串行数据）线是双向的，必须通过一个上拉电阻或等效电路将它们连到电源电压。连接在总线上的每个器件的 SCL 和 SDA 都必须是漏极开路或集电极开路的，因此当总线空闲时，这两条线都被拉到高电平（隐性状态）。总线上的最大器件数只受规定的上升和下降时间的限制，上升和下降时间分别不能超过 300ns 和 1000ns。

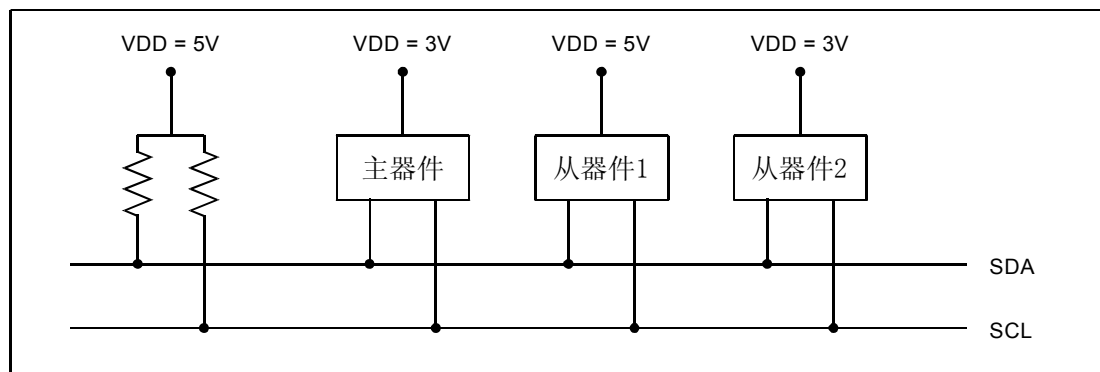


图 21.2 典型 SMBus 配置

21.3 SMBus 操作

有两种可能的数据传输类型：从主发送器到所寻址的从接收器（写）和从被寻址的从发送器到主接收器（读）。这两种数据传输都由主器件启动，主器件还在 SCL 上提供串行时钟。SMBus 接口可以工作在主方式或从方式，总线上可以有多个主器件。如果两个或多个主器件同时启动数据传输，仲裁机制将保证有一个主器件会赢得总线。注意：没有必要在一个系统中指定某个器件作为主器件；任何一个发送起始条件（START）和从器件地址的器件就成为该次数据传输的主器件。

一次典型的 SMBus 数据传输包括一个起始条件（START）、一个地址字节（位 7-1：7 位从地址；位 0：R/W 方向位）、一个或多个字节的数据和一个停止条件（STOP）。每个接收的字节（由一个主器件或从器件）都必须用 SCL 高电平期间的 SDA 低电平（见图 21.3）来确认（ACK）。如果接收器件不确认（ACK），则发送器件将读到一个“非确认”（NACK），这用 SCL 高电平期间的 SDA 高电平表示。

方向位(R/W)占据地址字节的最低位。方向位被设置为逻辑 1 表示这是一个“读”（READ）操作，方向位为逻辑 0 表示这是一个“写”（WRITE）操作。

所有的数据传输都由主器件启动，可以寻址一个或多个目标从器件。主器件产生一个起始条件，然后发送从地址和方向位。如果本次数据传输是一个从主器件到从器件的写操作，则主器件每发送一个数据字节后等待来自从器件的确认。如果是一个读操作，则由从器件发送数据并在每个字节结束后等待主器件的确认。在数据传输结束时，主器件产生一个停止条件，结束数据交换并释放总线。图 21.3 示出了一次典型的 SMBus 数据传输过程。

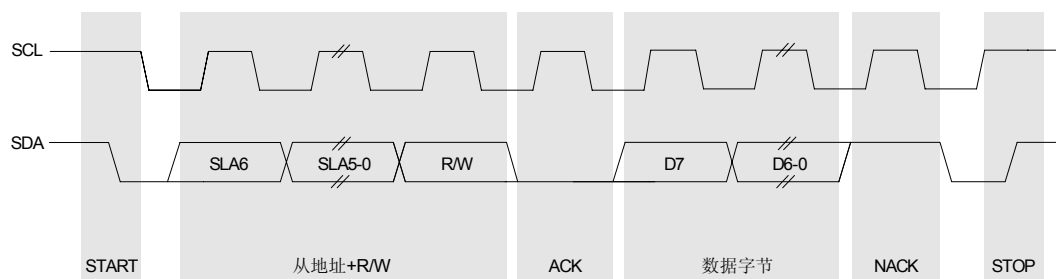


图 21.3 SMBus 数据传输

21.3.1 发送器和接收器的概念

在 SMBus 通信接口中，发送器是指在正在总线上发送一个地址或数据字节到其他器件的器件，而接收器是指总线上的另一器件正在向它发送一个地址或数据字节。发送器在传输地址或数据字节期间控制 SDA 线。在发送器每发送完一个地址或数据信息字节之后，接收器在本次传送的 ACK 阶段发送一个 ACK 或 NACK 位，在此期间接收器控制 SDA 线。

21.3.2 总线仲裁

一个主器件只能在总线空闲时启动一次传输。在一个停止条件之后或 SCL 和 SDA 保持高电平已经超过了规定时间（见“21.3.5 SCL 高电平超时”），则总线是空闲的。在两个或多个器件试图同时开始传送的情况下，使用仲裁机制迫使一个主器件放弃总线。这些主器件继续发送起始条件，直到其中一个主器件发送高电平而其它主器件在 SDA 上发送低电平。由于总线是漏极开路的，因此被拉为低电平。试图发送高电平的主器件将检测到 SDA 上的低电平而退出竞争。赢得总线的主器件继续其数据传输过程；而未赢得总线的器件成为从器件，如果被寻址，则在剩下的传输过程中接收数据。该仲裁机制是非破坏性的：总会有一个器件赢得总线，不会发生数据丢失。

21.3.3 时钟低电平扩展

SMBus 提供一种与 I²C 类似的时钟同步机制，允许不同速度的器件共存于同一个总线上。为了使低速从器件能与高速主器件通信，在传输期间采取低电平扩展。从器件可以临时保持 SCL 线为低电平以扩展时钟低电平时间，这实际上相当于降低了串行时钟频率。

21.3.4 SCL 低电平超时

如果 SCL 线被总线上的从器件保持为低电平，则不能再进行通信，并且主器件也不能强制 SCL 为高电平来纠正这种错误情况。为了解决这一问题，SMBus 协议规定：参加一次数据传输的器件必须检查时钟低电平时间，若超过 25ms 则认为是“超时”。检测到超时条件的器件必须在 10ms 以内对通信复位。

当 SMB0CF 中的 SMBTOE 位被置位时，定时器 3 被用于检测 SCL 低电平超时。定时器 3 在 SCL 为高电平时被强制重载，在 SCL 为低电平时开始计数。如果定时器 3 被使能并且溢出周期被配置为 25ms（且 SMBTOE 被置 1），则可在发生 SCL 低电平超时事件时用定时器 3 中断服务程序对 SMBus 复位（禁止后重新使能）。

21.3.5 SCL 高电平（SMBus 空闲）超时

SMBus 规范规定：如果一个器件保持 SCL 和 SDA 线为高电平的时间超过 50 微秒，则认为总线处于空闲状态。当 SMB0CF 中的 SMBFTE 位被置 1 时，如果 SCL 和 SDA 保持高电平的时间超过 10 个 SMBus 时钟周期，总线将被视为空闲。如果一个 SMBus 器件正等待产生一个主起始条件，则该起始条件将在总线空闲超时之后立即产生。注意：总线空闲超时检测需要一个时钟源，即使对从器件方式也不例外。

21.4 SMBus 的使用

SMBus 可以工作在主方式或从方式。接口电路提供串行传输的时序和移位控制；更高层的协议由用户软件实现。SMBus 接口提供下述与应用无关的特性：

- 以字节为单位的串行数据传输；
- SCL 时钟信号产生（只限于主方式）及 SDA 数据同步；
- 超时/总线错误识别（在配置寄存器 SMB0CF 中定义）；
- START/STOP 定时、检测和产生；
- 总线仲裁；

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

- 中断产生；
- 状态信息；
- 可选的硬件从地址识别和地址/数据的自动确认。

每次数据字节或从地址传输都产生 SMBus 中断。当硬件确认被禁止时，中断产生的时刻取决于硬件正在作为数据发送者还是作为接收者。当作为发送者时（即发送地址/数据，接收 ACK），该中断产生在 ACK 周期之后，以使软件能读取收到的 ACK 值；当接收数据时（即接收地址/数据，发送 ACK），中断产生在 ACK 周期之前，使软件能确定要发出的 ACK 值。如果硬件确认被使能，这些中断总是产生在 ACK 周期之后。有关传输时序的详细信息见 21.5 节。

主器件产生起始条件时也会产生一个中断，指示数据传输开始；从器件在检测到停止条件时产生一个中断，指示数据传输结束。软件应通过读 SMB0CN（SMBus 控制寄存器）来确定 SMBus 中断的原因。SMB0CN 寄存器的说明见 21.4.2 节

21.4.1 SMBus 配置寄存器

SMBus 配置寄存器（SMB0CF）用于使能 SMBus 主和/或从方式，选择 SMBus 时钟源和设置 SMBus 时序和超时选项。当 ENSMB 位被置 1 时，SMBus 的所有主和从事件都被允许。可以通过将 INH 位置 1 来禁止从事件。在从事件被禁止的情况下，SMBus 接口仍然监视 SCL 和 SDA 引脚；但在接收到地址时会发出 NACK（非确认）信号，并且不会产生任何从中断。当 INH 被置位时，在下一个起始条件（START）后所有的从事件都将被禁止（当前传输过程的中断将继续）。

SMBCS1-0 位选择 SMBus 时钟源（见表 21.1），时钟源只在主方式或空闲超时检测被使能时使用。当 SMBus 接口工作在主方式时，所选择的时钟源的溢出周期决定 SCL 低电平和高电平的最小时间，该最小时间由下面的方程 21.1 给出。注意，SMBus 可以与其它外设共享该时钟源，前提是时钟源定时器一直保持运行状态。例如，定时器 1 溢出可以同时用于产生 SMBus 和 UART 波特率。有关定时器配置见“24. 定时器”。

表 21.1 SMBus 时钟源选择

SMBCS1	SMBCS0	SMBus 时钟源
0	0	定时器 0 溢出
0	1	定时器 1 溢出
1	0	定时器 2 高字节溢出
1	1	定时器 2 低字节溢出

$$T_{HighMin} = T_{LowMin} = \frac{1}{f_{ClockSourceOverflow}}$$

方程 21.1 最小 SCL 高和低电平时间

其中 $T_{HighMin}$ 为最小 SCL 高电平时间， T_{LowMin} 为最小 SCL 低电平时间， $f_{ClockSourceOverflow}$ 为时钟源的溢出频率。

所选择的时钟源应被配置为能产生由方程 21.1 所定义的最小 SCL 高电平和低电平时间。当接口工作在主方式时（并且 SCL 不被总线上的任何其它器件驱动或扩展），典型的 SMBus

位速率可由下面的方程 21.2 估算：

$$\text{位速率} = \frac{f_{\text{ClockSourceOverflow}}}{3}$$

方程 21.2 典型 SMBus 位速率

图 21.4 给出了由方程 21.2 定义的典型 SCL 波形。注意， T_{HIGH} 通常为 T_{LOW} 的两倍。实际的 SCL 输出波形可能会因总线上有其它器件而发生改变（SCL 可能被低速从器件扩展低电平，或被其它参与竞争的主器件驱动为低电平）。当工作在主方式时，位速率不能超过由方程 21.1 定义的极限值。

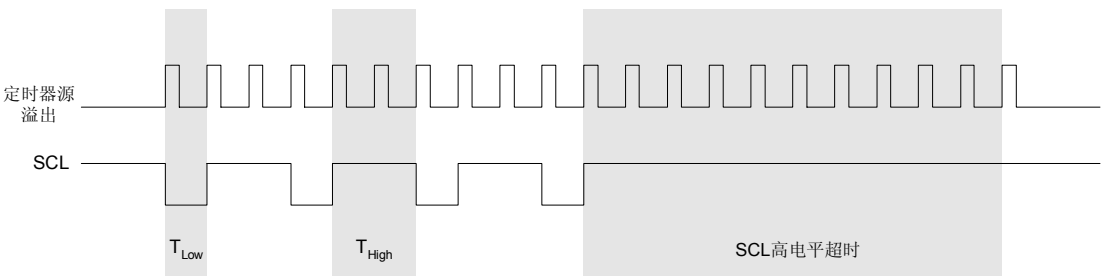


图 21.4 典型的 SMBus SCL 波形

设置 EXTHOLD 位为逻辑 1 将扩展 SDA 线的最小建立时间和保持时间。最小 SDA 建立时间定义了 SCL 上升沿到来之前 SDA 的最小稳定时间。最小 SDA 保持时间定义了 SCL 下降沿过去之后当前 SDA 值继续保持稳定的最小时间。SMBus 规定的最小建立和保持时间分别为 250ns 和 300ns。必要时应将 EXTHOLD 位置 1，以保证最小建立和保持时间符合 SMBus 规范。表 21.2 列出了对应两种 EXTHOLD 设置情况的最小建立和保持时间。当 SYSCLK 大于 10MHz 时，通常需要扩展建立和保持时间。

表 21.2 最小 SDA 建立和保持时间

EXTHOLD	最小 SDA 建立时间	最小 SDA 保持时间
0	T_{low} – 4 个系统时钟 或 1 个系统时钟 + 软件延时*	3 个系统时钟
1	11 个系统时钟	12 个系统时钟
*注：发送 ACK 位和所有数据传输中 MSB 的建立时间。当使用软件确认时，软件延时发生在写 SMB0DAT 或 ACK 到 SI 被清除之间。注意，如果写 ACK 和清除 SI 发生在同一个写操作，则软件延时为 0。		

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

在 SMBTOE 位被置 1 的情况下，定时器 3 应被配置为以 25ms 为周期溢出，以检测 SCL 低电平超时（见“21.3.4 SCL 低电平超时”）。SMBus 接口在 SCL 为高电平时强制重装载定时器 3，并允许定时器 3 在 SCL 为低电平时开始计数。应使用定时器 3 中断服务程序对 SMBus 通信复位，这可通过先禁止然后再重新使能 SMBus 接口来实现。

通过将 SMBFTE 位置 1 来使能 SMBus 空闲超时检测。当该位被置 1 时，如果 SCL 和 SDA 保持高电平的时间超过 10 个 SMBus 时钟周期（见图 21.4），总线将被视为空闲。

SFR 定义 21.1 SMB0CF: SMBus 配置寄存器

R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	复位值															
ENSMB	INH	BUSY	EXTHOLD	SMBTOE	SMBFTE	SMBCS1	SMBCS0	00000000															
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xC1															
<p>位 7: ENSMB: SMBus 使能 该位使能/禁止 SMBus 接口。当被使能时，接口一直监视 SDA 和 SCL 引脚。 0: 禁止 SMBus 接口。 1: 使能 SMBus 接口。</p> <p>位 6: INH: SMBus 从禁止 当该位被设置为逻辑 1 时，SMBus 接口在有从事件发生时不产生从事件中断。这实际上相当于将 SMBus 从器件移出总线。主方式中断不受影响。 0: SMBus 从方式使能。 1: SMBus 从方式禁止。</p> <p>位 5: BUSY: SMBus 忙状态标志 当正在进行一次传输时，该位由硬件置 1。当检测到停止条件或空闲超时，该位被清 0。</p> <p>位 4: EXTHOLD: SMBus 建立和保持时间扩展允许 该位控制 SDA 的建立和保持时间（根据表 21.2）。 0: 禁止 SDA 建立和保持时间扩展。 1: 允许 SDA 建立和保持时间扩展。</p> <p>位 3: SMBTOE: SMBus SCL 超时检测允许位 该位使能 SCL 低电平超时检测。当被置 1 时，SMBus 接口在 SCL 为高电平时强制重装载定时器 3，并允许定时器 3 在 SCL 为低电平时开始计数。如果定时器 3 被配置为分割方式（两个 8 位计数器，T3SPLIT 被置 1），则当 SCL 为高电平时，只有定时器 3 的高字节被重装载。应将定时器 3 编程为每 25ms 产生一次中断，并使用定时器 3 中断服务程序对 SMBus 通信复位。</p> <p>位 2: SMBFTE: SMBus 空闲超时检测允许位 当该位被置 1 时，如果 SCL 和 SDA 保持高电平的时间超过 10 个 SMBus 时钟周期，总线将被视为空闲。</p> <p>位 1-0: SMBCS1-SMBCS0: SMBus 时钟源选择位 这两位选择用于产生 SMBus 位速率的时钟源。应根据方程 21.1 配置所选时钟源。</p>																							
<table><tr><th>SMBCS1</th><th>SMBCS0</th><th>SMBus 时钟源</th></tr><tr><td>0</td><td>0</td><td>定时器 0 溢出</td></tr><tr><td>0</td><td>1</td><td>定时器 1 溢出</td></tr><tr><td>1</td><td>0</td><td>定时器 2 高字节溢出</td></tr><tr><td>1</td><td>1</td><td>定时器 2 低字节溢出</td></tr></table>									SMBCS1	SMBCS0	SMBus 时钟源	0	0	定时器 0 溢出	0	1	定时器 1 溢出	1	0	定时器 2 高字节溢出	1	1	定时器 2 低字节溢出
SMBCS1	SMBCS0	SMBus 时钟源																					
0	0	定时器 0 溢出																					
0	1	定时器 1 溢出																					
1	0	定时器 2 高字节溢出																					
1	1	定时器 2 低字节溢出																					

21.4.2 SMBus 控制寄存器

SMBus 控制寄存器(SMB0CN)用于控制 SMBus 接口和提供状态信息(见 SFR 定义 21.2)。SMB0CN 中的高 4 位(MASTER、TXMODE、STA 和 STO)组成一个状态向量,可利用该状态向量转移到中断服务程序。MASTER 指示在当前的传送期间器件工作在主方式还是从方式。TXMODE 指示对于当前字节器件是在发送数据还是接收数据。

STA 和 STO 指示自上次 SMBus 中断以来检测到或产生了一个起始(START)和/或停止条件(STOP)。当 SMBus 工作在主方式时,STA 和 STO 还用于产生起始和停止条件。当总线变成空闲时(在起始条件产生后,STA 未被硬件清 0),向 STA 写 1 将使 SMBus 接口进入主方式并产生一个起始条件。在主方式,向 STO 写 1 将使 SMBus 接口产生一个停止条件,并在下一个 ACK 周期后结束当前的数据传输。如果 STA 和 STO 都被置位(在主方式),则发送一个停止条件后再发送一个起始条件。

ARBLOST 位指示 SMBus 接口是否在一次总线竞争中失败。当接口正在发送时(主或从方式),可能会出现这种情况。当工作在从方式时,出现这种情况表示发生了总线错误条件。在每次 SI 被清除后,ARBLOST 被硬件清除。

在每次传输的开始和结束、每个字节帧之后或竞争失败时,SI 位(SMBus 中断标志)被硬件置 1,详见表 21.3。

有关 SI 位的重要注意事项:当 SI 标志被置 1 时,SMBus 接口被冻结;SCL 线被保持为低电平,总线状态被冻结到 SI 被清 0 为止。

21.4.2.1 软件 ACK 产生

当寄存器 SMB0ADM 中的 EHACK 位被清 0 时,器件的固件必须检查接收的从地址,并对该从地址和接收的数据字节发 ACK 或 NACK。当 SMBus 接口作为接收器时,写 ACK 位定义要发出的 ACK 值;当作为发送器时,读 ACK 位将返回最后一个 ACK 周期的接收值。ACKRQ 在每接收到一个字节后置位,表示需要写待发出的 ACK 值。当 ACKRQ 置位时,软件应在清除 SI 之前向 ACK 位写入要发出的 ACK 值。如果在清除 SI 之前软件未写 ACK 位,接口电路将产生一个 NACK。在向 ACK 位写入后,SDA 线将立即出现所定义的 ACK 值;但 SCL 将保持低电平,直到 SI 被清除。如果接收的从地址未被确认,则以后的从事件将被忽略,直到检测到下一个起始条件。

21.4.2.2 硬件 ACK 产生

当寄存器 SMB0ADM 中的 EHACK 位被置 1 时,自动从地址识别和 ACK 产生被使能。有关自动从地址识别的更详细的信息可以在 21.4.3 找到。作为接收器时,ACK 位的当前值会在输入数据字节的 ACK 周期被自动发送;当作为发送器时,读 ACK 位将返回最后一个 ACK 周期的接收值。当硬件 ACK 产生被使能时,不使用 ACKRQ 位。如果接收的从地址被硬件否认(NACK),则以后的从事件将被忽略,直到检测到下一个起始条件,并且不产生中断。

表 21.3 列出了影响 SMB0CN 寄存器中各个位的硬件源。有关使用 SMB0CN 寄存器进行 SMBus 的状态译码的详细信息见表 21.5。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 21.2 SMB0CN: SMBus 控制寄存器

R	R	R/W	R/W	R	R	R/W	R/W	复位值
MASTER	TXMODE	STA	STO	ACKRQ	ARBLOST	ACK	SI	00000000
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xC0
位 7:	MASTER: SMBus 主/从标志 该只读位指示 SMBus 是否工作在主方式。 0: SMBus 工作在从方式。 1: SMBus 工作在主方式。							
位 6:	TXMODE: SMBus 发送方式标志 该只读位指示 SMBus 是否工作在发送器方式。 0: SMBus 工作在接收器方式。 1: SMBus 工作在发送器方式。							
位 5:	STA: SMBus 起始标志 写: 0: 不产生起始条件。 1: 当配置为主方式时, 发送出一个起始条件或重复起始条件。 读: 0: 未检测到起始条件或重复起始条件。1: 检测到起始条件或重复起始条件。							
位 4:	STO: SMBus 停止标志 写: 0: 不发送停止条件。 1: 当配置为主方式时, 导致发送一个停止条件 (在下一个 ACK 周期之后)。 由硬件清除。 读: 0: 未检测到停止条件。1: 检测到停止条件 (在从方式) 或挂起 (在主方式)。							
位 3:	ACKRQ: SMBus 确认请求 0: 无 ACK 请求。 1: 有 ACK 请求。							
位 2:	ARBLOST: SMBus 竞争失败标志 0: 无仲裁错误。 1: 竞争失败。							
位 1:	ACK: SMBus 确认位 写: 0: 发送 NACK。 1: 发送 ACK。 读: 0: 收到 NACK。1: 收到 ACK。							
位 0:	SI: SMBus 中断标志 当出现表 21.3 列出的条件时该位被硬件置 1。SI 必须用软件清除。当 SI 被置 1 时, SCL 被保持低电平, SMBus 被冻结。 写: 0: 清除中断并指示下一个状态机事件。1: 强制中断。 读: 0: 无中断请求。1: 有中断请求。							

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 21.3 影响 SMB0CN 的硬件源

位	在下述情况被硬件置 1:	在下述情况被硬件清 0:
MASTER	<ul style="list-style-type: none">产生了起始条件。	<ul style="list-style-type: none">产生了停止条件。竞争失败。
TXMODE	<ul style="list-style-type: none">产生了起始条件。在一个 SMBus 帧开始之前写了 SMB0DAT。	<ul style="list-style-type: none">检测到起始条件。竞争失败。在一个 SMBus 帧开始之前未写 SMB0DAT。
STA	<ul style="list-style-type: none">在起始条件后接收到一个地址字节。	<ul style="list-style-type: none">必须用软件清除。
STO	<ul style="list-style-type: none">在作为从器件被寻址时检测到一个停止条件。因检测到停止条件而导致竞争失败。	<ul style="list-style-type: none">产生了一个挂起的停止条件。
ACKRQ	<ul style="list-style-type: none">接收到一个字节并需要一个 ACK 响应值。 (仅在硬件 ACK 未被使能时)。	<ul style="list-style-type: none">每个 ACK 周期之后。
ARBLOST	<ul style="list-style-type: none">当 STA 为低时, 主器件检测到一个重复起始条件 (不希望的重复起始条件)。在试图产生一个停止条件或重复起始条件时检测到 SCL 为低电平。在试图发送 1 时检测到 SDA 为低电平 (ACK 位除外)。	<ul style="list-style-type: none">每次 SI 被清除时。
ACK	<ul style="list-style-type: none">输入的 ACK 值为低 (确认)。	<ul style="list-style-type: none">输入的 ACK 值为高 (非确认)。
SI	<ul style="list-style-type: none">产生了一个起始条件。竞争失败。发送了一个字节并收到一个 ACK/NACK收到一个字节。在起始条件或重复起始条件之后接收到一个从地址+R/W。收到一个停止条件。	<ul style="list-style-type: none">必须用软件清除。

21.4.3 硬件从地址识别

SMBus 具有硬件自动识别进入的从地址和发送 ACK 的能力，无需软件干预。通过将寄存器 SMB0ADM 中的 EHACK 位置 1 来使能自动从地址识别。这将同时使能自动从地址识别和自动硬件产生接收字节的 ACK（主或从方式）。有关自动硬件 ACK 产生的详细信息见 21.4.2.2 节。

用于定义要被硬件识别的地址的寄存器是 SMBus 从地址寄存器(SFR 定义 21.3)和 SMBus 从地址掩码寄存器（SFR 定义 21.4）。用这两个寄存器可以指定单个地址或一个地址范围（包括全局呼叫地址 0x00）。这两个寄存器的高 7 位用于定义要被确认的地址。从地址掩码 SLVM[6:0]中为 1 的位使能接收的从地址与硬件从地址 SLV[6:0]中对应位的比较。从地址掩码中某一位为 0 表示该位在比较时被忽略。在这种情况下，不管进入的从地址的对应位是 1 还是 0，该地址都是可接受的。另外，当寄存器 SMB0ADR 中的 GC 位被置 1 时，硬件会识别全局呼叫地址（0x00）。表 21.4 列出了一些示例参数设置和在这些条件下能被硬件识别从地址。

表 21.4 硬件地址识别示例（EHACK = 1）

硬件从地址 SLV[6:0]	从地址掩码 SLVM[6:0]	GC 位	被硬件识别的从地址
0x34	0x7F	0	0x34
0x34	0x7F	1	0x34, 0x00（全局呼叫）
0x34	0x7E	0	0x34, 0x35
0x34	0x7E	1	0x34, 0x35, 0x00（全局呼叫）
0x70	0x73	0	0x70, 0x74, 0x78, 0x7C

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 21.3 SMB0ADR: SMBus 从地址寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
SLV6	SLV5	SLV4	SLV3	SLV2	SLV1	SLV0	GC	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xD7
<p>位 7-1: SLV[6:0]: SMBus 硬件从地址 定义用于自动硬件确认的 SMBus 从地址。只有 SLVM[6:0]中的对应位为 1 的那些地址位与接收的地址比较。这样就允许多个地址被识别。</p>								
<p>位 0: GC: 全局呼叫地址使能 当硬件地址识别被使能时 (EHACK = 1), 该位决定硬件是否也识别全局呼叫地址 (0x00)。 0: 全局呼叫地址被忽略。 1: 全局呼叫地址被识别。</p>								

SFR 定义 21.4 SMB0ADM: SMBus 从地址掩码寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
SLVM6	SLVM5	SLVM4	SLVM3	SLVM2	SLVM1	SLVM0	EHACK	11111110
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xE7
<p>位 7-1: SLVM[6:0]: SMBus 从地址掩码 定义寄存器 SMB0ADR 中哪些位与接收的地址字节比较, 哪些位被忽略。SLVM[6:0]中任何被置 1 的位使能与 SLV[6:0]中对应位的比较。值为 0 的那些位被忽略 (这些位在接收的地址中可以是 0, 也可以是 1)。</p>								
<p>位 0: EHACK: 硬件确认使能 使能硬件对从地址和所接收的数据字节的确认。 0: 固件必须手动确认所有进入的地址和数据字节。 1: 自动从地址识别和硬件确认被使能。</p>								

21.4.4 数据寄存器

SMBus 数据寄存器 SMB0DAT 保存要发送或刚接收的串行数据字节。在 SI 标志被置 1 时，软件可以安全地读/写数据寄存器。当 SMBus 被使能但 SI 标志被清为逻辑 0 时软件不应访问 SMB0DAT 寄存器，因为接口可能正在对该寄存器中的数据字节进行移入或移出操作。

SMB0DAT 中的数据总是先移出 MSB。在收到一个字节后，接收数据的第一位位于 SMB0DAT 的 MSB。在数据被移出的同时，总线上的数据被移入，所以 SMB0DAT 中总是保存最后出现在总线上的数据字节。在竞争失败后，从主发送器变为从接收器时 SMB0DAT 中的数据或地址保持不变。

SFR 定义 21.5 SMB0DAT: SMBus 数据寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xC2
位 7-0: SMB0DAT: SMBus 数据								
SMB0DAT 寄存器保存要发送到 SMBus 串行接口上的一个数据字节，或刚从 SMBus 串行接口接收到的一个字节。一旦 SI 串行中断标志被置 1，CPU 即可读或写该寄存器。只要 SI 串行中断标志位（SMB0CN.0）为逻辑 1，该寄存器内的串行数据就是稳定的。当 SI 标志位不为 1 时，系统可能正在移入/移出数据，此时 CPU 不应访问该寄存器。								

21.5 SMBus 传输方式

SMBus 接口可以被配置为工作在主方式和/或从方式。在任一时刻，它将工作在下述 4 种方式之一：主发送器、主接收器、从发送器或从接收器。SMBus 在产生起始条件时进入主方式，并保持在该方式直到产生一个停止条件或在总线竞争中失败。SMBus 在每个字节帧结束后都产生一个中断。注意：在作为接收器工作时，ACK 中断的位置取决于硬件 ACK 产生是否被使能。作为接收器时，在硬件 ACK 产生被禁止的情况下，ACK 中断发生在 ACK 之前；在硬件 ACK 产生被使能的情况下，ACK 中断发生在 ACK 之后。作为发送器时中断发生在 ACK 之后，与硬件 ACK 产生是否被使能无关。

21.5.1 写序列（主方式）

在一次写序列期间，SMBus 主器件向从器件写数据。这次传送中的主器件在地址字节和所有数据字节期间都将作为发送器。SMBus 接口首先产生一个起始条件，然后发送含有目标从器件地址和数据方向位的第一个字节。在这种情况下，数据方向位（R/W）应为逻辑 0（WRITE），表示这是一个写操作。主器件接着发送一个或多个字节的串行数据。在每发送一个字节后，从器件产生一个确认位。当 STO 位被置 1 并产生一个停止条件后，本次传送结束。注意，如果在发生主发送器中断后没有向 SMB0DAT 写入数据，则接口将切换到主接收器方式。图 21.5 给出了典型的主器件写序列，只给出了发送两个字节的传输时序，尽管可以发送任意多个字节。注意，在该方式下，所有的“数据字节传输结束”中断都发生在 ACK 周期之后，与硬件 ACK 产生是否被使能无关。

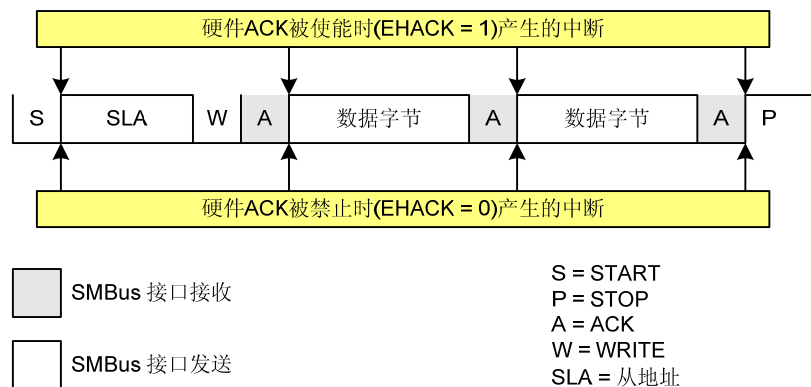


图 21.5 典型的主器件写序列

21.5.2 读序列（主方式）

在一次读序列期间，SMBus 主器件从一个从器件读数据。这次传送中的主器件在地址字节期间为发送器，在所有的数据字节期间都将作为接收器。SMBus 接口首先产生一个起始条件，然后发送含有目标从器件地址和数据方向位的第一个字节。在这种情况下，数据方向位（R/W）应为逻辑 1（READ），表示这是一个读操作。然后，SMBus 接口接收 SDA 线上来自从器件的串行数据，并输出 SMBus 串行时钟。从器件发送一个或多个字节的串行数据。

如果硬件 ACK 产生被禁止，则每接收到一个字节后 ACKRQ 位被置 1 并产生中断。此时软件必须写 ACK 位以确认（ACK）或否认（NACK）所接收的字节。

在硬件 ACK 产生被使能的情况下，SMBus 硬件会自动产生 ACK/NACK，并产生中断。

注意：当硬件 ACK 产生被使能时，软件应在读取该字节前写 ACK 或 NACK 值。

向 ACK 位写 1 产生一个 ACK，写 0 产生一个 NACK。软件应在最后一次数据传送时向 ACK 位写 0，以发送 NACK。接口电路将在对 STO 位置 1 并产生一个停止条件后退出主接收器方式。在主接收器方式，如果执行了 SMB0DAT 写操作，接口将切换到主发送器方式。图 21.6 给出了典型的主方式读序列，只给出了接收两个字节的传输时序，尽管可以接收任意多个字节。注意，在该方式下，“数据字节传输结束”中断可发生在序列中的不同位置，取决于硬件 ACK 产生是否被使能。当硬件 ACK 产生被禁止时，中断发生在 ACK 之前；当硬件 ACK 产生被使能时，中断发生在 ACK 之后。

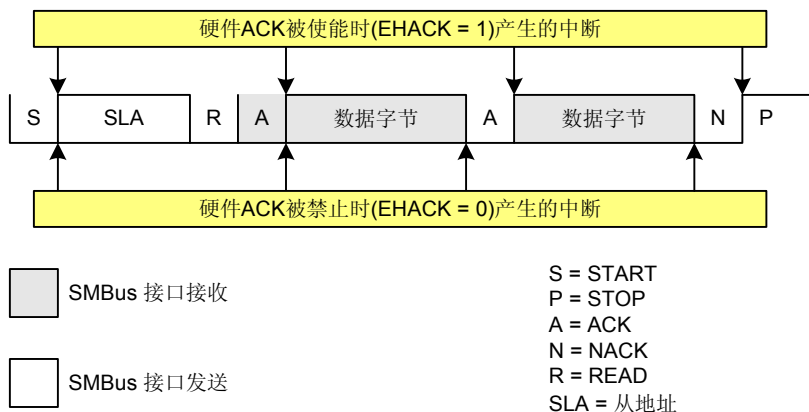


图 21.6 典型的主器件读序列

21.5.3 写序列（从方式）

在一次写序列期间，SMBus 主器件向从器件写数据。这次传送中的从器件在地址字节期间为接收器，在所有的数据字节期间作为接收器。在从事件被允许的情况下（INH = 0），当接收到一个起始条件（START）和一个含有从地址和数据方向位（此处应为写）的字节时，SMBus 接口进入从接收器方式。如果硬件 ACK 产生被禁止，在进入从接收器方式时将产生一个中断，并且 ACKRQ 被置 1。软件必须用一个 ACK 对接收到的从地址确认，或用一个 NACK 忽略接收到的从地址。如果硬件 ACK 产生被使能，硬件会对一个匹配的从地址产生 ACK，该地址要与由 SMB0ADR 和 SMB0ADM 设置的判别条件匹配。中断发生在 ACK 周期之后。

如果接收到的从地址被忽略（由硬件或软件），从事件中断将被禁止，直到检测到下一个起始条件。如果收到的从地址被确认，将接收 0 个或多个字节的数据。

如果硬件 ACK 产生被禁止，则每接收到一个字节后 ACKRQ 位被置 1 并产生中断。此时软件必须写 ACK 位以确认（ACK）或否认（NACK）所接收的字节。

在硬件 ACK 产生被使能的情况下，SMBus 硬件会自动产生 ACK/NACK，并产生中断。
注意：当硬件 ACK 产生被使能时，软件应在读取该字节前写 ACK 或 NACK 值。

在收到一个停止条件后，SMBus 接口退出从接收器方式。注意，如果在从接收器方式对 SMB0DAT 进行写操作，接口将切换到从发送器方式。图 21.7 给出了典型的从方式写序列，只给出了接收两个字节的传输时序，尽管可以接收任意多个字节。注意，在该方式下，“数据字节传输结束”中断可发生在序列中的不同位置，取决与硬件 ACK 产生是否被使能。当硬件 ACK 产生被禁止时，中断发生在 ACK 之前；当硬件 ACK 产生被使能时，中断发生在 ACK 之后。

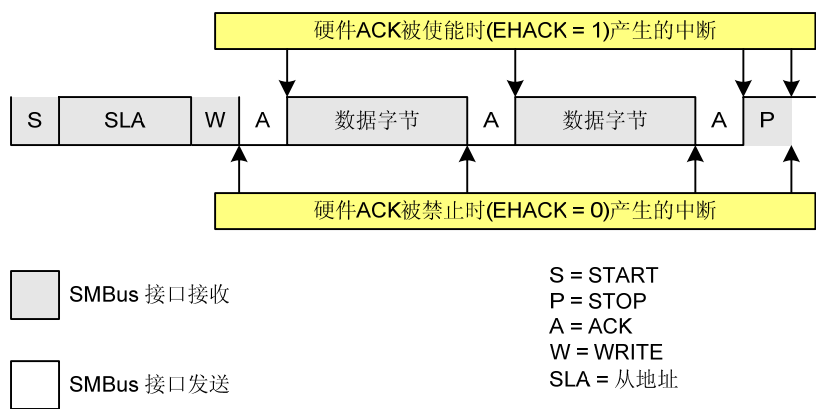


图 21.7 典型的从器件写序列

21.5.4 读序列（从方式）

在一次读序列期间，SMBus 主器件从一个从器件读数据。这次传送中的从器件在地址字节期间为接收器，在所有的数据字节期间都将作为发送器。在从事件被允许的情况下（INH = 0），当接收到一个起始条件（START）和一个含有从地址和数据方向位（此处应为读）的字节时，SMBus 接口进入从接收器方式（接收从地址）。如果硬件 ACK 产生被禁止，在进入从接收器方式时将产生一个中断，并且 ACKRQ 被置 1。软件必须用一个 ACK 对接收到的从地址确认，或用一个 NACK 忽略接收到的从地址。如果硬件 ACK 产生被使能，硬件会对一个匹配的从地址产生 ACK，该地址要与由 SMB0ADR 和 SMB0ADM 设置的判别条件匹配。中断发生在 ACK 周期之后。

如果接收到的从地址被忽略（由硬件或软件），从事件中断将被禁止，直到检测到下一个起始条件。如果收到的从地址被确认，将发送 0 个或多个字节的数据。如果收到的从地址被确认，软件应向 SMB0DAT 写入待发送的数据，SMBus 接口进入从发送器方式，并发送一个或多个字节的数据。在每发送一个字节后，主器件发送一个确认位。如果确认位为 ACK，应向 SMB0DAT 写入下一个数据字节；如果确认位为 NACK，在 SI 被清除前不应再写 SMB0DAT（注：在从发送器方式，如果在收到 NACK 后写 SMB0DAT，将会导致一个错误条件）。在收到停止条件后，SMBus 接口退出从发送器方式。注意，如果在一个从发送器中断发生之后没有对 SMB0DAT 进行写操作，接口将切换到从接收器方式。图 21.8 给出了典型的从方式读序列，只给出了发送两个字节的传输时序，尽管可以发送任意多个字节。注意，在该方式下“数据字节传输结束”中断发生在 ACK 周期之后，与硬件 ACK 产生是否被使能无关。

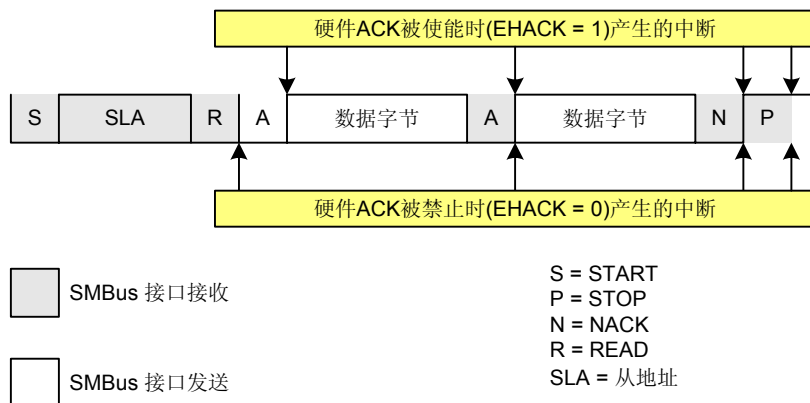


图 21.8 典型的从器件读序列

21.6 SMBus 状态译码

用 SMB0CN 寄存器可以很容易地对 SMBus 的当前状态译码。响应一个 SMBus 事件要采取的合适动作取决于硬件从地址识别和 ACK 产生是被使能还是被禁止。表 21.5 中描述了硬件从地址识别和 ACK 产生被禁止时的典型动作。表 21.6 中描述了硬件从地址识别和 ACK 产生被使能时的典型动作。在这两个表中，状态向量指的是 SMB0CN 中的高 4 位：MASTER、TXMODE、STA 和 STO。注意，表中只列出了典型的响应选项。只要符合 SMBus 规范，特定应用过程是允许的。表中被突出显示的响应选项是硬件允许的，但不符合 SMBus 规范。

表 21.5 硬件 ACK 产生被禁止时的 SMBus 状态译码 (EHACK = 0)

方式	读取值				SMBus 的当前状态	典型响应选项	写入值			下一个期望的状态向量
	状态向量	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
主发送器	1110	0	0	X	起始条件已发出。	将从地址+R/W 装入到 SMB0DAT。	0	0	X	1100
	1100	0	0	0	数据或地址字节已发出；收到 NACK。	置位 STA 以重新启动数据传输。	1	0	X	1110
						放弃发送。	0	1	X	-
		0	0	1	数据或地址字节已发出；收到 ACK。	将下一数据字节装入到 SMB0DAT。	0	0	X	1100
						用停止条件结束数据传输。	0	1	X	-
						用停止条件结束数据传输并开始另一次传输。	1	1	X	-
						发送重复起始条件。	1	0	X	1110
						切换到主接收器方式（清除 SI，不向 SMB0DAT 写新数据）。	0	0	X	1000
主接收器	1000	1	0	X	收到数据字节；请求 ACK。	确认接收字节；读 SMB0DAT。	0	0	1	1000
						发 NACK，表示这是最后一个字节，发停止条件。	0	1	0	-
						发 NACK，表示这是最后一个字节，接着发停止条件，再发起始条件。	1	1	0	1110
						发 ACK 后再发重复起始条件。	1	0	1	1110
						发 NACK，表示这是最后一个字节，接着发重复起始条件。	1	0	0	1110
						发 ACK 并切换到主发送器方式（在清除 SI 之前写 SMB0DAT）。	0	0	1	1100
						发 NACK 并切换到主发送器方式（在清除 SI 之前写 SMB0DAT）。	0	0	0	1100

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 21.5 硬件 ACK 产生被禁止时的 SMBus 状态译码 (EHACK = 0) (续)

方式	读取值				SMBus 的当前状态	典型响应选项	写入值			下一个期望的状态向量
	状态向量	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
从发送器	0100	0	0	0	字节已发送; 收到 NACK。	不需任何操作 (等待停止条件)。	0	0	X	0001
		0	0	1	字节已发送; 收到 ACK。	将下一个要发送的数据字节装入到 SMB0DAT。	0	0	X	0100
		0	1	X	字节已发送; 检测到错误。	不需任何操作 (等待主器件结束传输)。	0	0	X	0001
	0101	0	X	X	从发送期间检测到非法停止条件或总线错误。	清 STO。	0	0	X	-
从接收器	0010	1	0	X	接收到从地址+R/W; 请求 ACK。	如果写, 对接收到的地址进行确认。	0	0	1	0000
						如果读, 向 SMB0DAT 装入数据字节; 对接收到的地址进行确认。	0	0	1	0100
						否认接收到的地址。	0	0	0	-
	1	1	1	X	作为主器件竞争失败; 收到从地址+R/W; 请求 ACK。	如果写, 对接收到的地址进行确认。	0	0	1	0000
						如果读, 向 SMB0DAT 装入数据字节; 对接收到的地址进行确认。	0	0	1	0100
						否认接收到的地址。	0	0	0	-
						重新安排失败的传输; 否认接收到的地址。	1	0	0	1110
	0001	0	0	X	作为从发送器或从接收器被寻址时, 检测到停止条件。	清 STO。	0	0	X	-
		1	1	X	试图发送停止条件时竞争失败。	不需任何操作 (传输完成/放弃)。	0	0	0	-
	0000	1	0	X	接收到从字节; 请求 ACK。	确认接收的字节; 读 SMB0DAT。	0	0	1	0000
						否认接收到的字节。	0	0	0	-
总线错误条件	0010	0	1	X	试图发送重复起始条件时竞争失败。	放弃失败的传输。	0	0	X	-
						重新安排失败的传输。	1	0	X	1110
	0001	0	1	X	由于检测到停止条件而竞争失败。	放弃失败的传输。	0	0	X	-
						重新安排失败的传输。	1	0	X	1110
	0000	1	1	X	试图作为主器件发送数据字节时竞争失败。	放弃失败的传输。	0	0	0	-
						重新安排失败的传输。	1	0	0	1110

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 21.6 硬件 ACK 产生被使能时的 SMBus 状态译码 (EHACK = 1)

方式	读取值				SMBus 的当前状态	典型响应选项	写入值			下一个期望的状态向量
	状态向量	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
主发送器	1110	0	0	X	起始条件已发出。	将从地址+R/W 装入到 SMB0DAT。	0	0	X	1100
	1100	0	0	0	数据或地址字节已发出；收到 NACK。	置位 STA 以重新启动数据传输。	1	0	X	1110
						放弃发送。	0	1	X	-
		0	0	1	数据或地址字节已发出；收到 ACK。	将下一数据字节装入到 SMB0DAT。	0	0	X	1100
						用停止条件结束数据传输。	0	1	X	-
						用停止条件结束数据传输并开始另一次传输。	1	1	X	-
						发送重复起始条件。	1	0	X	1110
						切换到主接收器方式 (清除 SI, 不向 SMB0DAT 写新数据)。为初始数据字节置位 ACK。	0	0	1	1000
主接收器	1000	0	0	1	收到数据字节；已发送 ACK。	发 ACK 准备接收下一数据字节；读 SMB0DAT。	0	0	1	1000
						发 NACK, 表示下一数据字节是最后一个数据字节；读 SMB0DAT。	0	0	0	1000
						发送一个重复起始条件	1	0	0	1110
						切换到主发送器方式 (在清除 SI 之前写 SMB0DAT)。	0	0	X	1100
	0000	0	0	0	收到数据字节；已发送 NACK(最后一个字节)。	读 SMB0DAT, 发送停止条件。	0	1	0	-
						读 SMB0DAT, 发送停止条件后再发起始条件。	1	1	0	1110
						发送重复起始条件。	1	0	0	1110
						切换到主发送器方式 (在清除 SI 之前写 SMB0DAT)。	0	0	X	1100

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 21.5 硬件 ACK 产生被使能时的 SMBus 状态译码 (EHACK = 1) (续)

方式	读取值				SMBus 的当前状态	典型响应选项	写入值			下一个期望的状态向量
	状态向量	ACKRQ	ARBLOST	ACK			STA	STO	ACK	
从发送器	0100	0	0	0	字节已发送; 收到 NACK。	不需任何操作 (等待停止条件)。	0	0	X	0001
		0	0	1	字节已发送; 收到 ACK。	将下一个要发送的数据字节装入到 SMB0DAT。	0	0	X	0100
		0	1	X	字节已发送; 检测到错误。	不需任何操作 (等待主器件结束传输)。	0	0	X	0001
	0101	0	X	X	从发送期间检测到非法停止条件或总线错误。	清 STO。	0	0	X	-
从接收器	0010	0	0	X	接收到从地址+R/W; ACK 已发送。	如果写, 设置第一个数据字节的 ACK。 如果读, 向 SMB0DAT 装入数据字节。	0	0	1	0000
		0	1	X	作为主器件竞争失败; 收到从地址+R/W; ACK 已发送。	如果写, 设置第一个数据字节的 ACK。 如果读, 向 SMB0DAT 装入数据字节。 重新安排失败的传输。	0	0	1	0000
		0	0	X	作为从发送器或从接收器被寻址时, 检测到停止条件。	清 STO。	0	0	X	-
		0	1	X	试图发送停止条件时竞争失败。	不需任何操作 (传输完成/放弃)。	0	0	0	-
	0000	0	0	X	接收到从字节。	为下一数据字节设置 ACK; 读 SMB0DAT。	0	0	1	0000
		0	0	X		为下一数据字节设置 NACK; 读 SMB0DAT。	0	0	0	0000
	0010	0	1	X	试图发送重复起始条件时竞争失败。	放弃失败的传输。 重新安排失败的传输。	0	0	X	-
	0001	0	1	X	由于检测到停止条件而竞争失败。	放弃失败的传输。 重新安排失败的传输。	0	0	X	-
总线错误条件	0000	0	1	X	试图作为主器件发送数据字节时竞争失败。	放弃失败的传输。 重新安排失败的传输。	0	0	X	-
		0	1	X		放弃失败的传输。 重新安排失败的传输。	1	0	X	1110

22. UART0

UART0 是一个异步、全双工串口，它提供标准 8051 UART 的方式 1 和方式 3。UART0 具有增强的波特率发生器电路，有多个时钟源可用于产生标准波特率。接收数据缓冲机制允许 UART0 在软件尚未读取前一个数据字节的情况下开始接收第二个输入数据字节。

UART0 有两个相关的特殊功能寄存器：串行控制寄存器（SCON0）和串行数据缓冲器（SBUF0）。用同一个 SBUF0 地址可以访问发送寄存器和接收寄存器。**写 SBUF0 时总是访问发送寄存器；读 SBUF0 时总是访问缓冲的接收寄存器，不可能从发送数据寄存器中读数据。**

如果 UART0 中断被允许，则每次发送完成（SCON0 中的 TI0 位被置 1）或接收到一个数据字节（SCON0 中的 RI0 位被置 1）时将产生中断。当 CPU 转向中断服务程序时硬件不清除 UART0 中断标志。中断标志必须用软件清除，这就允许软件判断 UART0 中断的原因（发送完成或接收完成）。

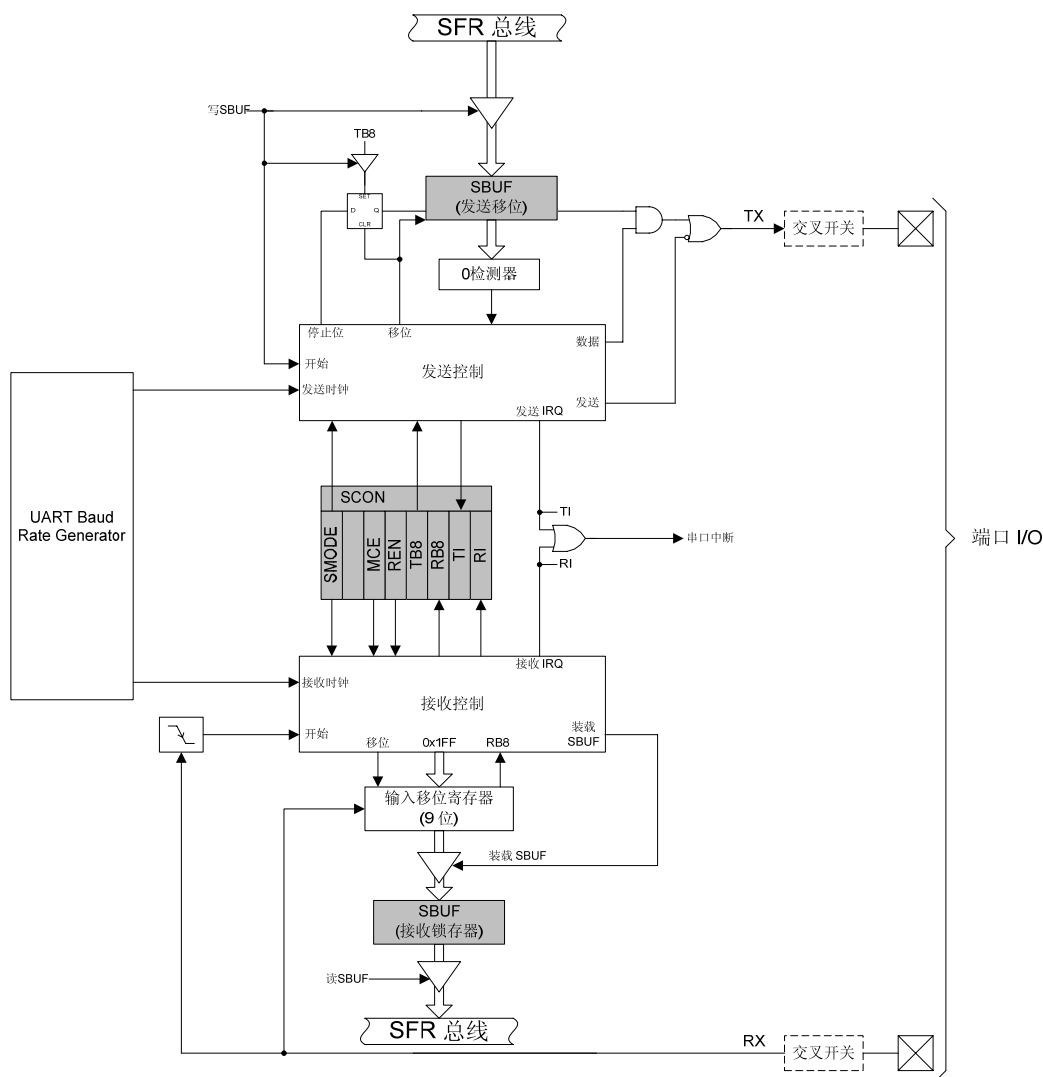


图 22.1 UART0 原理框图

22.1 增强的波特率发生器

UART0 波特率由定时器 1 工作在 8 位自动重载方式产生。发送 (TX) 时钟由 TL1 产生；接收 (RX) 时钟由 TL1 的副本寄存器 (图 22.2 中的 RX 定时器) 产生，该寄存器不能被用户访问。TX 和 RX 定时器的溢出信号经过二分频后用于产生 TX 和 RX 波特率。当定时器 1 被允许时，RX 定时器运行并使用与定时器 1 相同的重载值 (TH1)。在检测到 RX 引脚上的起始条件时 RX 定时器被强制重载，这允许在检测到起始位时立即开始接收过程，而与 TX 定时器的状态无关。

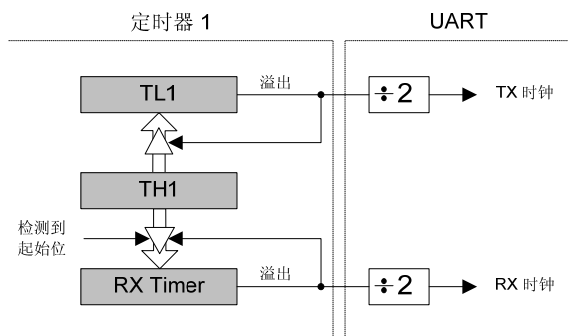


图 22.2 UART0 波特率逻辑

定时器 1 应被配置为方式 2，即 8 位自动重载方式（见“24.1.3 方式 2：8 位重载计数器/定时器”）。定时器 1 的重载值应设置为使其溢出频率为所期望的 UART 波特率频率的两倍。注意，定时器 1 的时钟可以在 6 个时钟源中选择：SYSCLK、SYSCLK/4、SYSCLK/12、SYSCLK/48、外部振荡器时钟/8 和外部输入 T1。对于任何给定的定时器 1 时钟源，UART0 的波特率由方程 22.1-A 和方程 22.1-B 决定：

$$\text{A) } \text{UART 波特率} = \frac{1}{2} \times T1 \text{ 溢出率}$$

$$\text{B) } T1 \text{ 溢出率} = \frac{T1_{CLK}}{(256 - TH1)}$$

方程 22.1 UART0 波特率

其中 $T1_{CLK}$ 是定时器 1 的时钟频率， $TH1$ 是定时器 1 的高字节（重载值）。

定时器 1 时钟频率的选择方法见“24. 定时器”。表 22.1 – 22.2 给出了典型波特率和系统时钟频率的对照表。注意，当外部振荡器驱动定时器 1 时，内部振荡器仍可产生系统时钟。

22.2 工作方式

UART0 提供标准的异步、全双工通信，其工作方式（8 位或 9 位）通过 S0MODE 位（SCON0.7）来选择。典型的 UART 连接方式如图 22.3 所示。

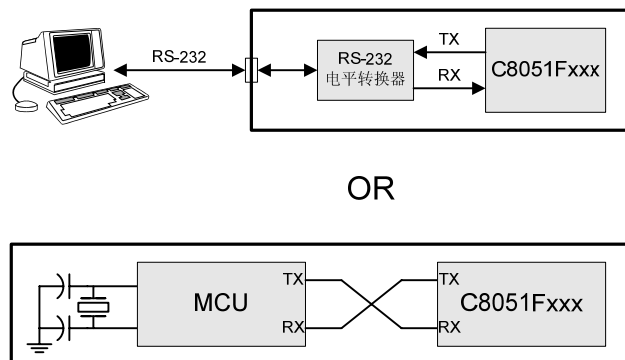


图 22.3 UART 连接图

22.2.1 8 位 UART

在 8 位 UART 方式，每个数据字节共使用 10 位：一个起始位、8 个数据位（LSB 在先）和一个停止位。数据从 TX0 引脚发送（LSB 在先），在 RX0 引脚接收。在接收时，8 个数据位存入 SBUF0，停止位进入 RB80（SCON0.2）。

当软件向 SBUF0 寄存器写入一个数据字节时开始数据发送。在发送结束时（停止位开始）发送中断标志 TI0（SCON0.1）被置 1。在接收允许位 REN0（SCON0.4）被置 1 后，数据接收可以在任何时刻开始。收到停止位后，如果满足下述条件则数据字节将被装入到接收寄存器 SBUF0：RI0 必须为逻辑 0；如果 MCE0 为逻辑 1，则停止位必须为 1。在发生接收数据溢出的情况下，先接收到的 8 位数据被锁存到 SBUF0，而后面的溢出数据被丢弃。

如果这些条件满足，则 8 位数据被存入 SBUF0，停止位被存入 RB80，RI0 标志被置 1。如果这些条件不满足，则不装入 SBUF0 和 RB80，RI0 标志也不会被置 1。如果中断被允许，在 TI0 或 RI0 置位时将产生一个中断。

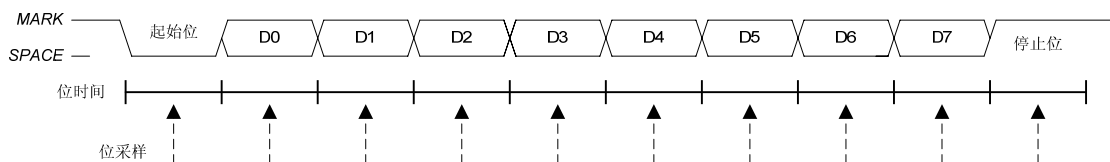


图 22.4 8 位 UART 时序图

22.2.2 9 位 UART

在 9 位 UART 方式，每个数据字节共使用 11 位：一个起始位、8 个数据位（LSB 在先）、一个可编程的第九位和一个停止位。第九发送数据位由 TB80（SCON0.3）中的值决定，由用户软件赋值。它可以被赋值为错误检测用的奇偶标志（PSW 中的位）的值，或用于多处理器通信。在接收时，第九数据位进入 RB80（SCON0.2），停止位被忽略。

当执行一条向 SBUF0 寄存器写一个数据字节的指令时开始数据发送。在发送结束时（停止位开始）发送中断标志 TI0 被置 1。在接收允许位 REN0（SCON0.4）被置 1 后，数据接收可以在任何时刻开始。收到停止位后如果满足下述条件则数据字节将被装入到接收寄存器 SBUF0：RI0 为逻辑 0；如果 MCE0 为逻辑 1，则第九位必须为逻辑 1（当 MCE0 为逻辑 0 时，第九位数据的状态并不重要）。如果这些条件满足，则 8 位数据被存入 SBUF0，第九位被存入 RB80，RI0 标志被置位。如果这些条件不满足，则不装入 SBUF0 和 RB80，RI0 标志也不会被置 1。如果中断被允许，在 TI0 或 RI0 置位时将产生 UART0 中断。

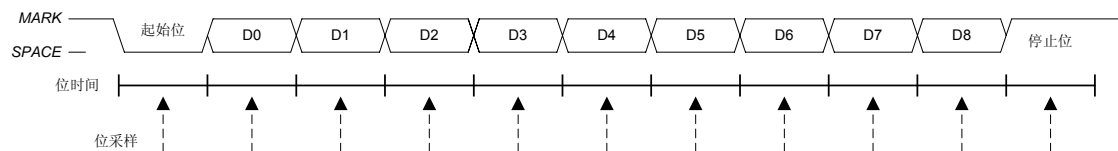


图 22.5 9 位 UART 时序图

22.3 多机通信

9 位 UART 方式通过使用第 9 数据位可以支持一个主处理器与一个或多个从处理器之间的多机通信。当主机要发送数据给一个或多个从机时，它先发送一个用于选择目标的地址字节。地址字节与数据字节的区别是：地址字节的第 9 位为逻辑 1；数据字节的第 9 位总是设置为逻辑 0。

如果从机的 MCE0 位（SCON.5）被置 1，则在收到有效的停止位后，只有当接收到的第九位为逻辑 1（RB80 = 1）时 UART 才会产生中断，意味着接收到一个地址字节。在 UART 的中断处理程序中，软件将接收到的地址与从机自身的 8 位地址进行比较。如果地址匹配，从机将清除它的 MCE0 位以允许后面接收数据字节时产生中断。未被寻址的从机仍保持其 MCE0 位为 1，在收到后续的数据字节时不产生中断，从而忽略收到的数据。一旦接收完整个消息，被寻址的从机将它的 MCE0 位重新置 1 以忽略所有的数据传输，直到它收到下一个地址字节。

可以将多个地址分配给一个从机，和/或将一个地址分配给多个从机从而允许同时向多个从机“广播”发送。主机可以被配置为接收所有的传输数据，或通过实现某种协议使主/从角色能临时变换以允许原来的主机和从机之间进行半双工通信。

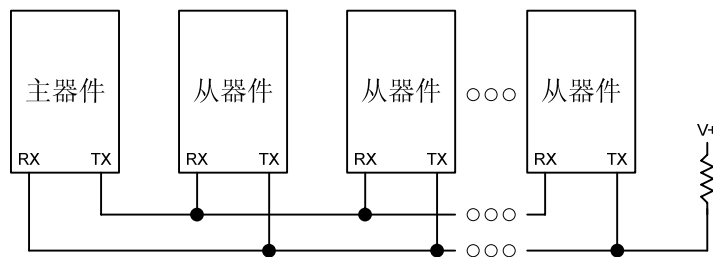


图 22.6 UART 多机方式连接图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 22.1 SCON0: UART0 控制寄存器

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	复位值
S0MODE	-	MCE0	REN0	TB80	RB80	TI0	RI0	01000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: (可位寻址) 0x98
位 7:	S0MODE: UART0 工作方式位 该位选择 UART0 的工作方式。 0: 方式 0: 波特率可编程的 8 位 UART。 1: 方式 1: 波特率可编程的 9 位 UART。							
位 6:	未使用。读 = 1b。写 = 忽略。							
位 5:	MCE0: 多处理器通信使能 该位的功能取决于 UART0 工作方式。 方式 0: 检查有效停止位。 0: 停止位的逻辑电平被忽略。 1: 只有当停止位为逻辑 1 时 RI0 激活。 方式 1: 多处理器通信使能。 0: 第 9 位的逻辑电平被忽略。 1: 只有当第 9 位为逻辑 1 时 RI0 才被置位并产生中断。							
位 4:	REN0: 接收允许 该位允许/禁止 UART 接收器。 0: UART0 接收禁止。 1: UART0 接收允许。							
位 3:	TB80: 第 9 发送位 该位的逻辑电平被作为 9 位 UART 方式（方式 1）的第 9 发送位。在 8 位 UART 方式（方式 0）中未用。							
位 2:	RB80: 第 9 接收位 在方式 0, 则 RB80 被赋值为停止位的值。在方式 1 该位被赋值为 9 位 UART 方式中第九数据位的值。							
位 1:	TI0: 发送中断标志 当 UART0 发送完一个字节数据后该位被硬件置 1（在 8 位 UART 方式时, 是在发送第 8 位后; 在 9 位 UART 方式时, 是在停止位的开始）。当 UART0 中断被允许时, 置 1 该位将导致 CPU 转到 UART0 中断服务程序。该位必须用软件清 0。							
位 0:	RI0: 接收中断标志 当 UART0 接收到一个字节数据时该位被硬件置 1（在停止位采样时）。当 UART0 中断被允许时, 置 1 该位将会使 CPU 转到 UART0 中断服务程序。该位必须用软件清 0。							

SFR 定义 22.2 SBUF0: UART0 串行数据缓冲寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x99
<p>位 7-0: SBUF0[7:0]: UART0 数据缓冲器位 7-0 (MSB-LSB)</p> <p>该 SFR 访问两个寄存器: 发送移位寄存器和接收锁存寄存器。当数据被写到 SBUF0 时, 它进入发送移位寄存器等待串行发送。向 SBUF0 写入一个字节即启动发送过程。读 SBUF0 时返回接收锁存器的内容。</p>								

表 22.1 对应标准波特率的定时器设置 (使用内部 24.5MHz 振荡器)

频率: 24.5MHz							
	目标波特率 (bps)	波特率 误差 (%)	振荡器 分频系数	定时器 时钟源	SCA1-SCA0 (分频选择) ¹	T1M ¹	定时器 1 重载值 (hex)
SYSCLK 源自 内部振荡器	230400	-0.32%	106	SYSCLK	XX ²	1	0xCB
	115200	-0.32%	212	SYSCLK	XX	1	0x96
	57600	0.15%	426	SYSCLK	XX	1	0x2B
	28800	-0.32%	848	SYSCLK/4	01	0	0x96
	14400	0.15%	1704	SYSCLK/12	00	0	0xB9
	9600	-0.32%	2544	SYSCLK/12	00	0	0x96
	2400	-0.32%	10176	SYSCLK/48	10	0	0x96
	1200	0.15%	20448	SYSCLK/48	10	0	0x2B
注: 1. SCA1-SCA0 和 T1M 位的定义见 24.1 节。 2. X=忽略。							

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 22.2 对应标准波特率的定时器设置（使用 22.1184MHz 外部振荡器）

频率：22.1184MHz							
	目标波特率 (bps)	波特率 误差 (%)	振荡器 分频系数	定时器 时钟源	SCA1-SCA0 (分频选择) ¹	T1M ¹	定时器 1 重载值 (hex)
SYSCLK 源自 外部振荡器	230400	0.00%	96	SYSCLK	XX ²	1	0xD0
	115200	0.00%	192	SYSCLK	XX	1	0xA0
	57600	0.00%	384	SYSCLK	XX	1	0x40
	28800	0.00%	768	SYSCLK/12	00	0	0xE0
	14400	0.00%	1536	SYSCLK/12	00	0	0xC0
	9600	0.00%	2304	SYSCLK/12	00	0	0xA0
	2400	0.00%	9216	SYSCLK/48	10	0	0xA0
	1200	0.00%	18432	SYSCLK/48	10	0	0x40
SYSCLK 源自 内部振荡器	230400	0.00%	96	EXTCLK/8	11	0	0xFA
	115200	0.00%	192	EXTCLK/8	11	0	0xF4
	57600	0.00%	384	EXTCLK/8	11	0	0xE8
	28800	0.00%	768	EXTCLK/8	11	0	0xD0
	14400	0.00%	1536	EXTCLK/8	11	0	0xA0
	9600	0.00%	2304	EXTCLK/8	11	0	0x70
注： 1. SCA1-SCA0 和 TIM 位的定义见 24.1 节。 2. X=忽略。							

23. 增强型串行外设接口（SPI0）

增强型串行外设接口（SPI0）提供访问一个全双工同步串行总线的能力。SPI0 可以作为主器件或从器件工作，可以使用 3 线或 4 线方式，并可在同一总线上支持多个主器件和从器件。从选择信号（NSS）可被配置为输入以选择工作在从方式的 SPI0，或在多主环境中禁止主方式操作，以避免两个以上主器件试图同时进行数据传输时发生 SPI 总线冲突。NSS 可以被配置为片选输出（在主方式），或在 3 线操作时被禁止。在主方式，可以用其他通用端口 I/O 引脚选择多个从器件。

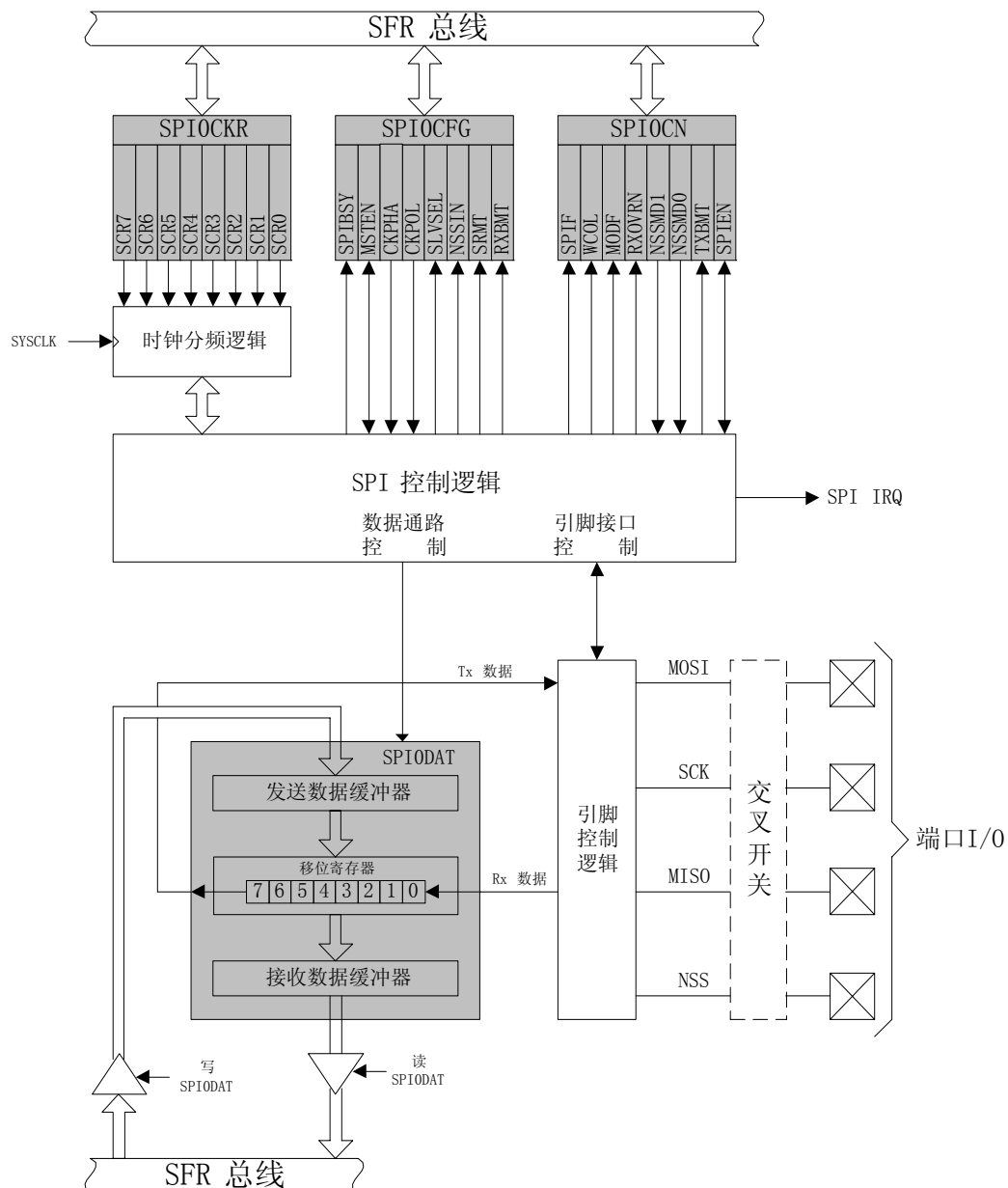


图 23.1 SPI 原理框图

23.1 信号说明

下面介绍 SPI0 所使用的 4 个信号（MOSI、MISO、SCK、NSS）。

23.1.1 主输出、从输入（MOSI）

主出从入（MOSI）信号是主器件的输出和从器件的输入，用于从主器件到从器件串行传送数据。当 SPI0 作为主器件时，该信号是输出；当 SPI0 作为从器件时，该信号是输入。数据传输时最高位在先。当被配置为主器件时，MOSI 由移位寄存器的 MSB 驱动。

23.1.2 主输入、从输出（MISO）

主入从出（MISO）信号是从器件的输出和主器件的输入，用于从从器件到主器件串行传送数据。当 SPI0 作为主器件时，该信号是输入；当 SPI0 作为从器件时，该信号是输出。数据传输时最高位在先。当 SPI 被禁止或工作在 4 线从方式而未被选中时，MISO 引脚被置于高阻态。当作为从器件工作在 3 线方式时，MISO 总是由移位寄存器的 MSB 驱动。

23.1.3 串行时钟（SCK）

串行时钟（SCK）信号是主器件的输出和从器件的输入，用于同步主器件和从器件之间在 MOSI 和 MISO 线上的串行数据传输。当 SPI0 作为主器件时产生该信号。在 4 线从方式，当从器件未被选中时（NSS=1），SCK 信号被忽略。

23.1.4 从选择（NSS）

从选择（NSS）信号的功能取决于 SPI0CN 寄存器中 NSSMD1 和 NSSMD0 位的设置。这两位可以选择 3 种可能的方式：

1. NSSMD[1:0] = 00: 3 线主方式或 3 线从方式：SPI0 工作在 3 线方式，NSS 被禁止。当作为从器件工作在 3 线方式时，SPI0 总是被选择。由于没有选择信号，SPI0 工作在 3 线方式时必须是总线唯一的从器件。这种情况用于一个主器件和一个从器件之间点对点通信。
2. NSSMD[1:0] = 01: 4 线从方式或多主方式：SPI0 工作在 4 线方式，NSS 被使能为输入。当作为从器件时，NSS 选择 SPI0 器件。当作为主器件时，NSS 信号的负跳变禁止 SPI0 的主器件功能，因此可以在同一个 SPI 总线上使用多个主器件。
3. NSSMD[1:0] = 1x: 4 线主方式：SPI0 工作在 4 线方式，NSS 被使能为输出。NSSMD0 的设置值决定 NSS 引脚的输出逻辑电平。这种配置只应在 SPI0 作为主器件时使用。

图 23.2 ~ 23.4 给出了不同工作方式下的典型连接图。注意：NSSMD 位的设置影响器件的引脚分配。当工作在 3 线主或从方式时，NSS 不被交叉开关分配引脚。在所有其他方式，NSS 将被映射到器件的一个引脚。有关通用端口 I/O 和交叉开关的信息见“20. 端口输入/输出”。

23.2 SPI0 主方式

SPI 总线上的所有数据传输都由 SPI 主器件启动。通过将主允许标志(MSTEN, SPI0CFG.6)置 1 将 SPI0 置于主方式。当处于主方式时, 向 SPI0 数据寄存器(SPI0DAT)写入一个数据字节时是写发送缓冲器。如果 SPI 移位寄存器为空, 发送缓冲器中的数据字节被传送到移位寄存器, 数据传输开始。SPI0 主器件立即在 MOSI 线上串行移出数据, 同时在 SCK 上提供串行时钟。在传输结束后 SPIF (SPI0CN.7) 标志被置为逻辑 1。如果中断被允许, 在 SPIF 标志置位时将产生一个中断请求。在全双工操作中, 当 SPI 主器件在 MOSI 线向从器件发送数据时, 被寻址的 SPI 从器件可以同时在 MISO 线上向主器件发送其移位寄存器中的内容。因此, SPIF 标志既作为发送完成标志又作为接收数据准备好标志。从从器件接收的数据字节以 MSB 在前的形式传送到主器件的移位寄存器。当一个数据字节被完全移入该寄存器时, 便被传送到接收缓冲器, 处理器通过读 SPI0DAT 来读该缓冲器。

当被配置为主器件时, SPI0 可以工作在下面的三种方式之一: 多主方式、3 线单主方式或 4 线单主方式。当 NSSMD1 (SPI0CN.3) =0 且 NSSMD0 (SPI0CN.2) =1 时, 是默认的多主方式。在该方式, NSS 是器件的输入, 用于禁止主 SPI0, 以允许另一主器件访问总线。在该方式, 当 NSS 被拉为低电平时, MSTEN (SPI0CN.6) 和 SPIEN (SPI0CN.0) 位被清 0, 以禁止 SPI 主器件, 且方式错误标志 (MODE, SPI0CN.5) 被置 1。如果中断被允许, 将产生方式错误中断。在这种情况下, 必须用软件重新使能 SPI0。在多主系统中, 当器件不作为系统主器件使用时, 一般被默认为从器件。在多主方式, 可以用通用 I/O 引脚对从器件单个寻址 (如果需要)。图 23.2 给出了两个主器件在多主方式下的连接图。

当 NSSMD1 (SPI0CN.3) =0 且 NSSMD0 (SPI0CN.2) =0 时, SPI0 工作在 3 线单主方式。在该方式, NSS 未被使用, 也不被交叉开关映射到外部端口引脚。在该方式, 应使用通用 I/O 引脚选择要寻址的从器件。图 23.3 给出了一个 3 线主方式主器件和一个从器件的连接图。

当 NSSMD1 (SPI0CN.3) =1 时, SPI0 工作在 4 线单主方式。在该方式, NSS 被配置为输出引脚, 可被用作从选择信号去选中一个 SPI 器件。在该方式, NSS 的输出值由 NSSMD0 (SPI0CN.2) 控制 (用软件)。可以用通用 I/O 引脚寻址另外的从器件。图 23.4 给出了一个 4 线主方式主器件和两个从器件的连接图。

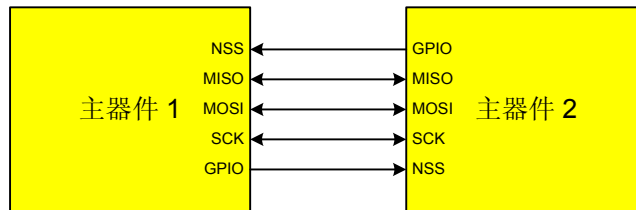


图 23.2 多主方式连接图

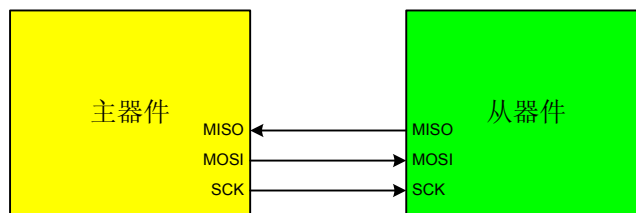


图 23.3 3 线单主方式和 3 线单从方式连接图

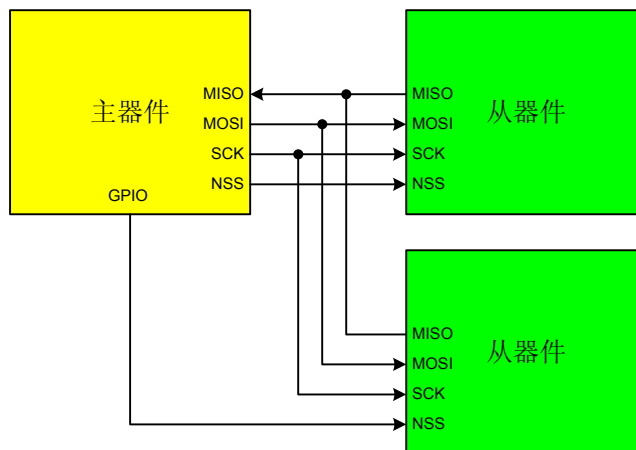


图 23.4 4 线单主方式和 4 线从方式连接图

23.3 SPI0 从方式

当 SPI0 被使能而未被配置为主器件时，它将作为 SPI 从器件工作。作为从器件，由主器件控制串行时钟（SCK），从 MOSI 移入数据，从 MISO 引脚移出数据。SPI0 逻辑中的位计数器对 SCK 边沿计数。当 8 位数据经过移位寄存器后，SPIF 标志被置为逻辑 1，接收到的字节被传送到接收缓冲器。通过读 SPI0DAT 来读取接收缓冲器中的数据。从器件不能启动数据传输。通过写 SPI0DAT 来预装要发送给主器件的数据。写往 SPI0DAT 的数据是双缓冲的，首先被放在发送缓冲器。如果移位寄存器为空，发送缓冲器中的数据会立即被传送到移位寄存器。当移位寄存器中已经有数据时，SPI 将在下一次（或当前）SPI 传输的最后一个 SCK 边沿过去后再将发送缓冲器的内容装入移位寄存器。

当被配置为从器件时，SPI0 可以工作 4 线或 3 线方式。当 NSSMD1（SPI0CN.3）=0 且 NSSMD0（SPI0CN.2）=1 时，是默认的 4 线从方式。在 4 线方式，NSS 被分配端口引脚并被配置为数字输入。当 NSS 为逻辑 0 时，SPI0 被使能；当 NSS 为逻辑 1 时，SPI0 被禁止。在 NSS 的下降沿，位计数器被复位。注意，对应每次字节传输，在第一个有效 SCK 边沿到来之前，NSS 信号必须被驱动到低电平至少两个系统时钟周期。图 23.4 给出了两个 4 线方式从器件和一个主器件的连接图。

当 NSSMD1（SPI0CN.3）=0 且 NSSMD0（SPI0CN.2）=0 时，SPI0 工作在 3 线从方式。在该方式，NSS 未被使用，也不被交叉开关映射到外部端口引脚。由于在 3 线从方式无法唯一地寻址从器件，所以 SPI0 必须是总线上唯一的从器件。需要注意的是，在 3 线从方式，没有外部手段对位计数器复位以判断是否收到一个完整的字节。只能通过用 SPIEN 位禁止并重新使能 SPI0 来复位位计数器。图 23.3 给出了一个 3 线从器件和一个主器件的连接图。

23.4 SPI0 中断源

如果 SPI0 中断被允许，在下述 4 个标志位被置 1 时将产生中断。

注意：这 4 个标志位都必须用软件清 0。

1. 在每次字节传输结束时，SPI 中断标志 SPIF（SPI0CN.7）被置 1。该标志适用于所有 SPI 方式。
2. 如果在发送缓冲器中的数据尚未被传送到 SPI 移位寄存器时写 SPI0DAT，写冲突标志 WCOL（SPI0CN.6）被置 1。发生这种情况时，写 SPI0DAT 的操作被忽略，不会对发送缓冲器写入。该标志适用于所有 SPI 方式。
3. 当 SPI0 被配置为工作于多主方式的主器件而 NSS 被拉为低电平时，方式错误标志 MODF（SPI0CN.5）被置 1。当发生方式错误时，SPI0CN 中的 MSTEN 和 SPIEN 位被清 0，以禁止 SPI0 并允许另一个主器件访问总线。
4. 当 SPI0 被配置为从器件并且一次传输结束，而接收缓冲器中还保持着上一次传输的数据未被读取时，接收溢出标志 RXOVRN（SPI0CN.4）被置 1。新接收的字节将不被传送到接收缓冲器，允许前面接收的字节被读取。引起溢出的数据字节丢失。

23.5 串行时钟相位和极性

使用 SPI0 配置寄存器 (SPI0CFG) 中的时钟控制选择位可以在串行时钟相位和极性的 4 种组合中选择其一。CKPHA 位 (SPI0CFG.5) 选择两种时钟相位 (锁存数据所用的边沿) 中的一种。CKPOL 位 (SPI0CFG.4) 在高电平有效和低电平有效的时钟之间选择。主器件和从器件必须被配置为使用相同的时钟相位和极性。在改变时钟相位和极性期间应禁止 SPI0 (通过清除 SPIEN 位, SPI0CN.0)。主方式下时钟和数据线的时序关系示于图 23.5; 从方式下时钟和数据的时序关系示于图 23.6 和图 23.7。注意: 当 C8051F04x, C8051F06x, C8051F12x, C8051F31x, C8051F32x, C8051F33x 中的两个器件通信时, 不论工作在主方式还是从方式, CKPHA 必须被置 0。

SFR 定义 23.3 所示的 SPI0 时钟速率寄存器 (SPI0CKR) 控制主方式的串行时钟频率。当工作于从方式时该寄存器被忽略。当 SPI 被配置为主器件时, 最大数据传输率 (位/秒) 是系统时钟频率的二分之一或 12.5MHz (取较低的频率)。当 SPI 被配置为从器件时, 全双工操作的最大数据传输率 (位/秒) 是系统时钟频率的十分之一, 前提是主器件与从器件系统时钟同步发出 SCK、NSS (在 4 线从方式) 和串行输入数据。如果主器件发出的 SCK、NSS 及串行输入数据不同步, 则最大数据传输率 (位/秒) 必须小于系统时钟频率的十分之一。在主器件只发送数据到从器件而不需要接收从器件发出的数据 (即半双工操作) 这一特殊情况下, SPI 从器件接收数据时的最大数据传输率 (位/秒) 是系统时钟频率的四分之一, 这是在假设由主器件发出 SCK、NSS 和串行输入数据与从器件系统时钟同步的情况下。

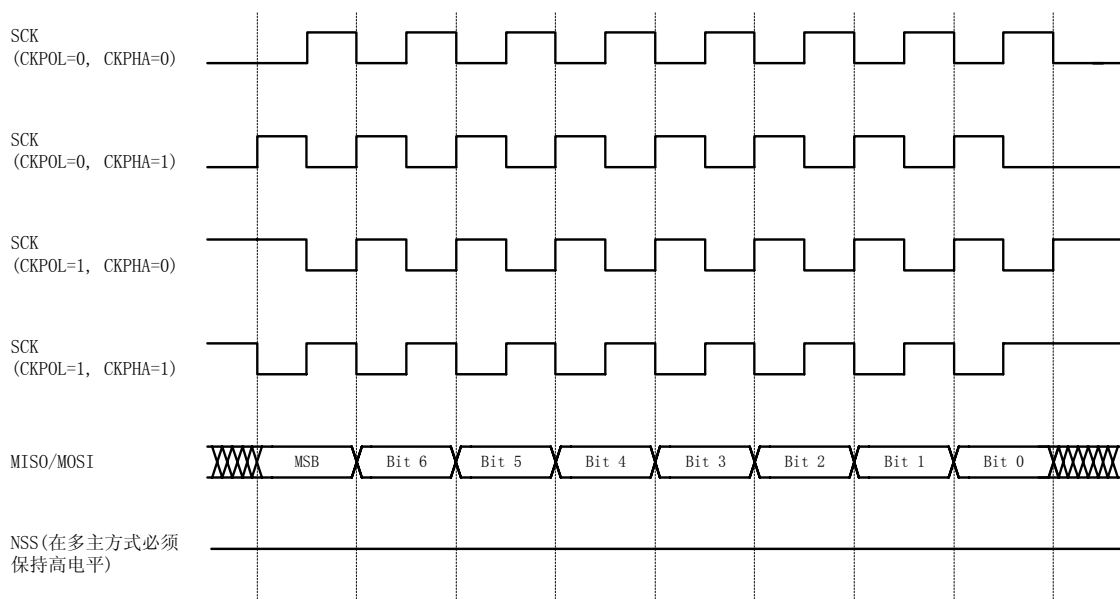


图 23.5 主方式数据/时钟时序图

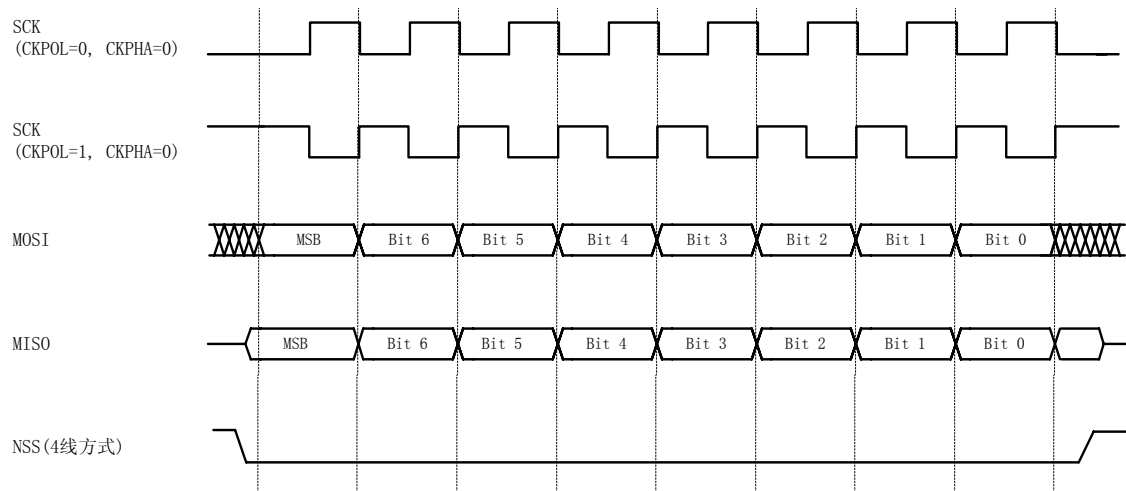


图 23.6 从方式数据/时钟时序图 (CKPHA = 0)

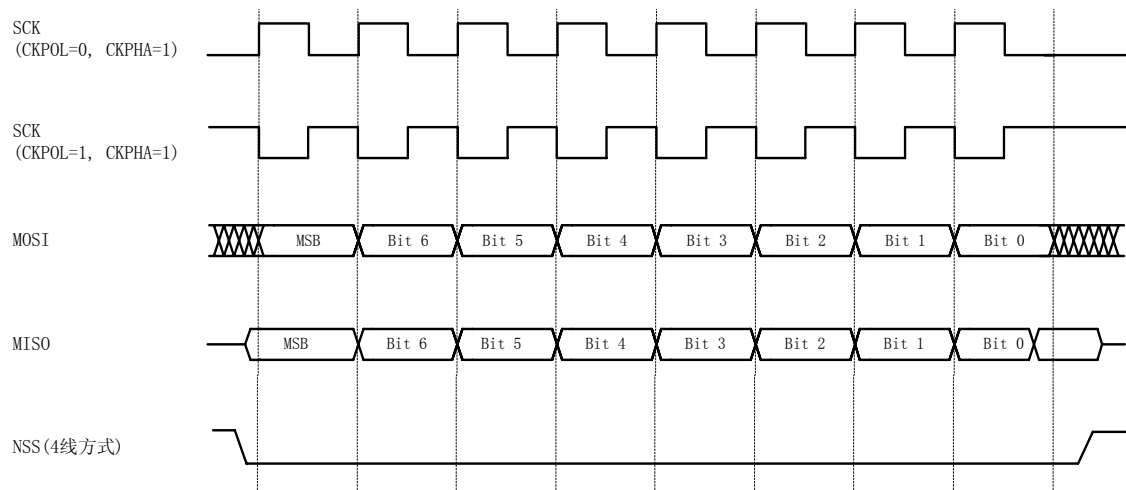


图 23.7 从方式数据/时钟时序图 (CKPHA = 1)

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

23.6 SPI 特殊功能寄存器

对 SPI0 的访问和控制是通过系统控制器中的 4 个特殊功能寄存器实现的：控制寄存器 SPI0CN、数据寄存器 SPI0DAT、配置寄存器 SPI0CFG 和时钟频率寄存器 SPI0CKR。下面将介绍这 4 个与 SPI0 总线操作有关的特殊功能寄存器。

SFR 定义 23.1 SPI0CFG: SPI0 配置寄存器

R	R/W	R/W	R/W	R	R	R	R	复位值
SPIBSY	MSTEN	CKPHA	CKPOL	SLVSEL	NSSIN	SRMT	RXBMT	00000111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xA1

位 7:

SPIBSY: SPI 忙标志（只读）
当一次 SPI 传输正在进行时（主或从方式），该位被置为逻辑 1。

位 6:

MSTEN: 主方式使能位
0: 禁止主方式，工作在从方式。
1: 使能主方式，工作在主器件方式。

位 5:

CKPHA: SPI0 时钟相位。
该位控制 SPI0 时钟的相位。
0: 数据以 SCK 周期的第一个边沿为中心*。
1: 数据以 SCK 周期的第二个边沿为中心*。

位 4:

CKPOL: SPI0 时钟极性
该位控制 SPI0 时钟的极性。
0: SCK 在空闲状态时处于低电平。
1: SCK 在空闲状态时处于高电平。

位 3:

SLVSEL: 从选择标志（只读）。
当 NSS 引脚为低电平时该位被置 1，指示 SPI0 是被选中的从器件。当 NSS 引脚为高电平时（未被选中为从器件）该位被清 0。该位不指示 NSS 引脚的即时值，而是该引脚输入的去噪信号。

位 2:

NSSIN: NSS 引脚的即时输入值（只读）
该位指示读该寄存器时 NSS 引脚的即时值。该信号未被去噪。

位 1:

SRMT: 移位寄存器空标志（只在从方式有效，（只读））。
当所有数据都被移入/移出移位寄存器并且没有新数据可以从发送缓冲器读出或向接收缓冲器写入时，该位被置 1。当数据字节被从发送缓冲器传送到移位寄存器或 SCK 发生变化时，该位被清 0。
注：在主方式时 SRMT = 1。

位 0:

RXBMT: 接收缓冲器空（在从方式有效，只读）
当接收缓冲器被读取且没有新数据时，该位被置 1。如果在接收缓冲器中有新数据未被读取，则该位被清 0。注意：在主方式时，RXBMT = 1。

*在从方式，MOSI 上的数据是在每个数据位的中间被采样。在主方式，MISO 上的数据是在每个数据结束前一个 SYSCLK 位被采样，为从器件提供最大的建立时间。时序参数见表 23.1。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 23.2 SPI0CN: SPI0 控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	复位值
SPIF	WCOL	MODF	RXOVRN	NSSMD1	NSSMD0	TXBMT	SPIEN	00000110
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: (可位寻址) 0xF8
<p>位 7: SPIF: SPI0 中断标志 该位在数据传输结束后被硬件置为逻辑 1。如果中断被允许, 该位置 1 将会使 CPU 转到 SPI0 中断处理服务程序。该位不能被硬件自动清 0, 必须用软件清 0。</p> <p>位 6: WCOL: 写冲突标志 该位由硬件置为逻辑 1 (并产生一个 SPI0 中断), 指示数据传送期间对 SPI0 数据寄存器进行了写操作。该位不能被硬件自动清 0, 必须用软件清 0。</p> <p>位 5: MODF: 方式错误标志 当检测到主方式冲突 (NSS 为低电平, MSTEN=1, NSSMD[1:0] = 01) 时, 该位由硬件置为逻辑 1 (并产生一个 SPI0 中断)。该位不能被硬件自动清 0, 必须用软件清 0。</p> <p>位 4: RXOVRN: 接收溢出标志 (只适用于从方式) 当前传输的最后一位已经移入 SPI0 移位寄存器, 而接收缓冲器中仍保存着前一次传输未被读取的数据时该位由硬件置为逻辑 1 (并产生一个 SPI0 中断)。该位不会被硬件自动清 0, 必须用软件清 0。</p> <p>位 3-2: NSSMD1-NSSMD0: 从选择方式位 选择 NSS 工作方式: 00: 3 线从方式或 3 线主方式。NSS 信号不连到端口引脚。 01: 4 线从方式或多主方式 (默认值)。NSS 是器件的输入。 1x: 4 线单主方式。NSS 信号被分配一个输出引脚并输出 NSSMD0 的值。</p> <p>位 1: TXBMT: 发送缓冲器空标志 当新数据被写入发送缓冲器时, 该位被清 0。当发送缓冲器中的数据被传送到 SPI 移位寄存器时, 该位被置 1, 表示可以安全地向发送缓冲器写新数据。</p> <p>位 0: SPIEN: SPI0 使能位 该位使能 / 禁止 SPI0。 0: 禁止 SPI0 1: 使能 SPI0</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 23.3 SPI0CKR: SPI0 时钟速率寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
SCR7	SCR6	SCR5	SCR4	SCR3	SCR2	SCR1	SCR0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xA2

位 7-0: SCR7-SCR0: SPI0 时钟频率

当 SPI0 模块被配置为工作于主方式时, 这些位决定 SCK 输出的频率。SCK 时钟频率是从系统时钟分频得到的, 由下面的方程给出, 其中: *SYSCLK* 是系统时钟频率, *SPI0CKR* 是 SPI0CKR 寄存器中的 8 位值。

$$f_{SCK} = \frac{SYSCLK}{2 \times (SPI0CKR + 1)}$$

(0 ≤ SPI0CKR ≤ 255)

例如: 如果 SYSCLK = 2MHz, SPI0CKR = 0x04, 则

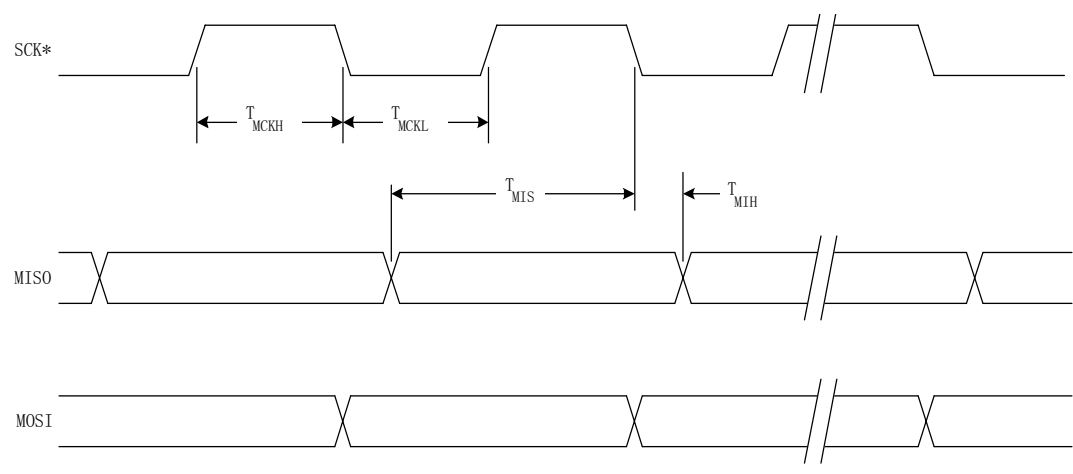
$$f_{SCK} = \frac{2000000}{2 \times (4 + 1)}$$
$$f_{SCK} = 200kHz$$

SFR 定义 23.4 SPI0DAT: SPI0 数据寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xA3

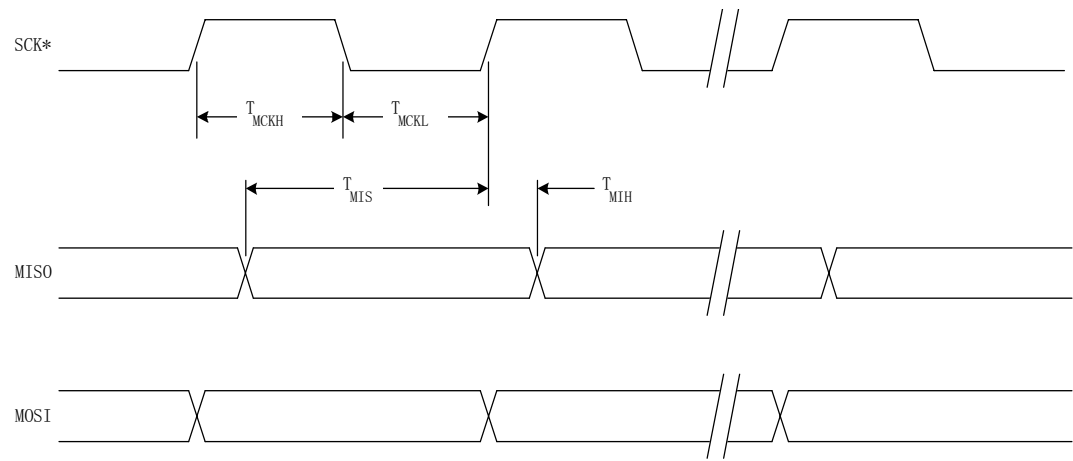
位 7-0: SPI0DAT: SPI0 发送和接收数据寄存器。

SPI0DAT 寄存器用于发送和接收 SPI0 数据。在主方式下, 向 SPI0DAT 写入数据时, 数据被放到发送缓冲器并启动发送。读 SPI0DAT 返回接收缓冲器的内容。



* 这是对应 CKPOL = 0时的 SCK 波形 对于 CKPOL = 1, SCK波形的极性反向。

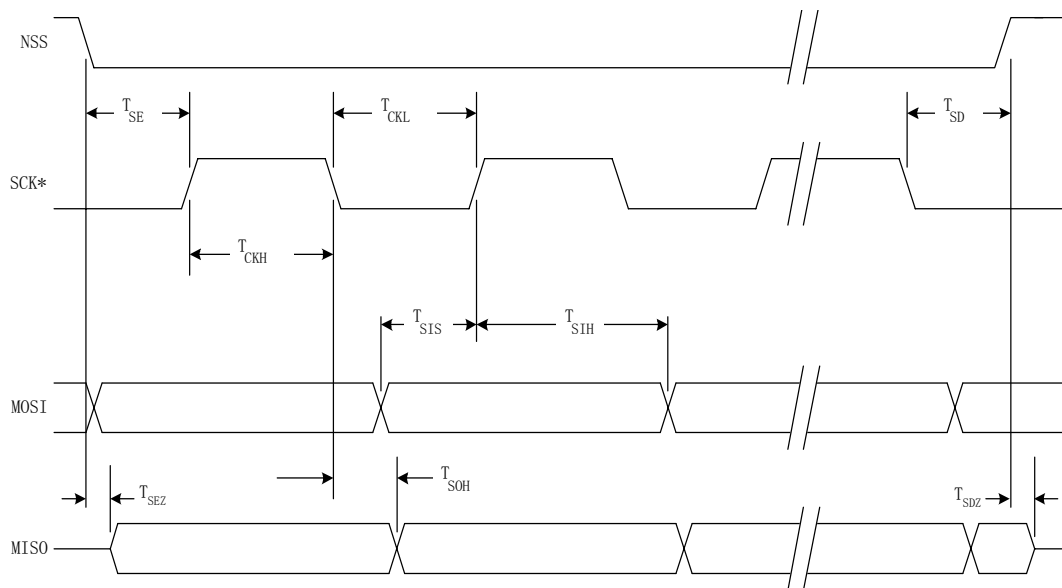
图 23.8 SPI 主方式时序 (CKPHA = 0)



* 这是对应 CKPOL = 0时的 SCK 波形。对于 CKPOL = 1, SCK波形的极性反向。

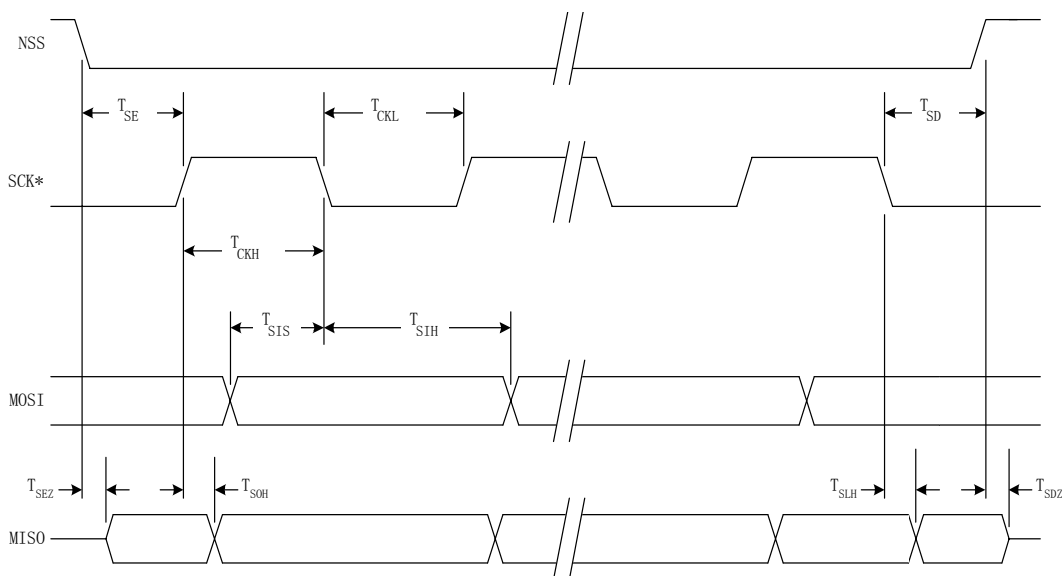
图 23.9 SPI 主方式时序 (CKPHA = 1)

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器



* 这是对应 CKPOL = 0时的 SCK 波形。对于 CKPOL = 1, SCK波形的极性反向。

图 23.10 SPI 从方式时序 (CKPHA = 0)



* 这是对应 CKPOL = 0时的 SCK 波形。对于 CKPOL = 1, SCK波形的极性反向。

图 23.11 SPI 从方式时序 (CKPHA = 1)

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

表 23.1 SPI 从方式时序参数

参 数	说 明	最小值	最大值	单位
主方式时序* (见图 23.8 和图 23.9)				
T_{MCKH}	SCK 高电平时间	$1 \times T_{SYSCLK}$	—	ns
T_{MCKL}	SCK 低电平时间	$1 \times T_{SYSCLK}$	—	ns
T_{MIS}	MISO 有效到 SCK 移位边沿	$1 \times T_{SYSCLK} + 20$	—	ns
T_{MIH}	SCK 移位边沿到 MISO 发生改变	0	—	ns
从方式时序* (见图 23.10 和图 23.11)				
T_{SE}	NSS 下降沿到第一个 SCK 边沿	$2 \times T_{SYSCLK}$	—	ns
T_{SD}	最后一个 SCK 边沿到 NSS 上升沿	$2 \times T_{SYSCLK}$	—	ns
T_{SEZ}	NSS 下降沿到 MISO 有效	—	$4 \times T_{SYSCLK}$	ns
T_{SDZ}	NSS 上升沿到 MISO 变为高阻态	—	$4 \times T_{SYSCLK}$	ns
T_{CKH}	SCK 高电平时间	$5 \times T_{SYSCLK}$	—	ns
T_{CKL}	SCK 低电平时间	$5 \times T_{SYSCLK}$	—	ns
T_{SIS}	MOSI 有效到 SCK 采样边沿	$2 \times T_{SYSCLK}$	—	ns
T_{SIH}	SCK 采样边沿到 MOSI 发生改变	$2 \times T_{SYSCLK}$	—	ns
T_{SOH}	SCK 移位边沿到 MISO 发生改变	—	$4 \times T_{SYSCLK}$	ns
T_{SLH}	最后一个 SCK 边沿到 MISO 发生改变 (只限于 CKPHA = 1)	$6 \times T_{SYSCLK}$	$8 \times T_{SYSCLK}$	ns
*注: T_{SYSCLK} 为器件系统时钟 (SYSCLK) 的一个周期。				

24. 定时器

C8051F336/7/8/9 内部有 4 个计数器/定时器: 其中两个 16 位计数器/定时器与标准 8051 中的计数器/定时器兼容, 另外两个是 16 位自动重装载定时器, 可用于 ADC、SMBus 或作为通用定时器使用。这些定时器可以用于测量时间间隔, 对外部事件计数或产生周期性的中断请求。定时器 0 和定时器 1 几乎完全相同, 有四种工作方式。定时器 2 和定时器 3 均可作为一个 16 位或两个 8 位自动重装载定时器。另外, 定时器 3 还可以在器件处于挂起模式时用外部振荡器作为时钟, 并可被用作唤醒源。该特性允许实现功耗很低的系统和 RTC 功能。

定时器 0 和定时器 1 工作方式	定时器 2 工作方式	定时器 3 工作方式
13 位计数器/定时器	16 位自动重装载定时器	16 位自动重装载定时器
16 位计数器/定时器		
8 位自动重装载的计数器/定时器	两个 8 位自动重装载定时器	两个 8 位自动重装载定时器
两个 8 位计数器/定时器 (仅限于定时器 0)		

定时器 0 和定时器 1 有 5 个可选择的时钟源, 由定时器方式选择位 (T1M-T0M) 和时钟分频位 (SCA1-SCA0) 决定。时钟分频位定义一个预分频时钟, 作为定时器 0 和/或定时器 1 的时钟源 (预分频时钟选择见 SFR 定义 24.1)。

定时器 0 和定时器 1 可以被配置为使用该预分频时钟或系统时钟。定时器 2 和定时器 3 可以使用系统时钟、系统时钟/12 或外部振荡器时钟/8 作为时钟源。

定时器 0 和定时器 1 也可以工作在计数器方式。当作为计数器使用时, 在为定时器所选择的输入引脚 (T0 或 T1) 上出现负跳变时计数器/定时器寄存器的值加 1。对事件计数的最大频率可达到系统时钟频率的四分之一。输入信号不需要是周期性的, 但在一个给定电平上的保持时间至少应为两个完整的系统时钟周期, 以保证该电平能够被正确采样。

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 24.1 CKCON: 时钟控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
T3MH	T3ML	T2MH	T2ML	T1M	T0M	SCA1	SCA0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x8E

位 7:

T3MH: 定时器 3 高字节时钟选择。

该位选择供给定时器 3 高字节的时钟(如果定时器 3 被配置为两个 8 位定时器, 定时器 3 工作在其他方式时该位被忽略)。

0: 定时器 3 高字节使用 TMR3CN 中的 T3XCLK 位定义的时钟。

1: 定时器 3 高字节使用系统时钟。

位 6:

T3ML: 定时器 3 低字节时钟选择。

该位选择供给定时器 3 的时钟。如果定时器 3 被配置为两个 8 位定时器, 该位选择供给低 8 位定时器的时钟。

0: 定时器 3 低字节使用 TMR3CN 中的 T3XCLK 位定义的时钟。

1: 定时器 3 低字节使用系统时钟。

位 5:

T2MH: 定时器 2 高字节时钟选择。

该位选择供给定时器 2 高字节的时钟(如果定时器 2 被配置为两个 8 位定时器, 定时器 2 工作在其他方式时该位被忽略)。

0: 定时器 2 高字节使用 TMR2CN 中的 T2XCLK 位定义的时钟。

1: 定时器 2 高字节使用系统时钟。

位 4:

T2ML: 定时器 2 低字节时钟选择。

该位选择供给定时器 2 的时钟。如果定时器 2 被配置为两个 8 位定时器, 该位选择供给低 8 位定时器的时钟。

0: 定时器 2 低字节使用 TMR2CN 中的 T2XCLK 位定义的时钟。

1: 定时器 2 低字节使用系统时钟。

位 3:

T1M: 定时器 1 时钟选择。

该位选择定时器 1 的时钟源。当 C/T1 被设置为逻辑 1 时, T1M 被忽略。

0: 定时器 1 使用由分频位 (SCA1[1:0]) 定义的时钟。

1: 定时器 1 使用系统时钟。

位 2:

T0M: 定时器 0 时钟选择。

该位选择定时器 0 的时钟源。当 C/T0 被设置为逻辑 1 时, T0M 被忽略。

0: 定时器 0 使用由分频位 (SCA1-SAC0) 定义的时钟。

1: 定时器 0 使用系统时钟。

位 1-0:

SCA1[1:0]: 定时器 0/1 预分频位

如果定时器 0/1 被配置为使用预分频时钟, 则这两位控制时钟分频数。

SCA1	SCA0	预分频时钟
0	0	系统时钟/12
0	1	系统时钟/4
1	0	系统时钟/48
1	1	外部时钟/8 (与系统时钟同步)

24.1 定时器 0 和定时器 1

每个计数器/定时器都是一个 16 位的寄存器，在被访问时分为两个字节：一个低字节（TL0 或 TL1）和一个高字节（TH0 或 TH1）。计数器/定时器控制寄存器（TCON）用于使能定时器 0 和定时器 1 以及指示它们的状态。通过将 IE 寄存器中的 ET0 位置 1 来允许定时器 0 中断，通过将 ET1 位置 1 来允许定时器 1 中断。这两个计数器/定时器都有四种工作方式，通过设置计数器/定时器方式寄存器（TMOD）中的方式选择位 T1M1-T0M0 来选择工作方式，每个定时器都可以被独立配置。下面对每种工作方式进行详细说明。

24.1.1 方式 0 — 13 位计数器/定时器

在方式 0，定时器 0 和定时器 1 被作为 13 位的计数器/定时器使用。图 24.1 给出了定时器 0 工作在方式 0 时的原理框图。下面介绍对定时器 0 的配置和操作。由于这两个定时器在工作上完全相同，所以定时器 1 的配置过程与定时器 0 一样。

TH0 寄存器保持 13 位计数器/定时器的 8 个 MSB，TL0 在 TL0.4-TL0.0 位置保持 5 个 LSB。TL0 的高 3 位（TL0.7-TL0.5）是不确定的，在读计数值时应屏蔽掉或忽略这 3 位。作为 13 位定时器寄存器，计到 0x1FFF（全 1）后再计一次将发生溢出，使计数值回到 0x0000，此时定时器溢出标志 TF0（TCON.5）被置位并产生一个中断（如果定时器 0 中断被允许）。

C/T0 位（TMOD.2）选择计数器/定时器的时钟源。当 C/T0 被设置为逻辑 1 时，出现在所选定时器 0 输入引脚（T0）上的负跳变使定时器寄存器加 1。清除 C/T0 位将选择由 T0M 位（CKCON.3）定义的时钟作为定时器的输入。当 T0M 被置 1 时，定时器 0 的时钟为系统时钟；当 T0M 位被清 0 时，定时器 0 的时钟源由 CKCON（见 SFR 定义 24.1）中的时钟分频位定义。

当 GATE0（TMOD.3）为逻辑 0 或输入信号/INT0 有效时（有效电平由 IT01CF 寄存器中的 IN0PL 位定义，见 SFR 定义 15.5），将 TR0 位（TCON.4）置 1 允许定时器 0 工作。设置 GATE0 为逻辑 1 允许定时器受外部输入信号/INT0 的控制，便于脉冲宽度测量。

TR0	GATE0	/INT0	计数器/定时器
0	X	X	禁止
1	0	X	允许
1	1	0	禁止
1	1	1	允许

X=任意

注意，置位 TR0 并不强制定时器 0 复位。应在定时器被使能前将定时器寄存器装入所需要的初值。

与上述的 TL0 和 TH0 一样，TL1 和 TH1 构成定时器 1 的 13 位寄存器。定时器 1 的配置和控制方法与定时器 0 一样，使用 TCON 和 TMOD 中的对应位。输入信号/INT1 为定时器 1 所用，其极性由 IT01CF 寄存器中的 IN1PL 位定义（见 SFR 定义 15.5）。

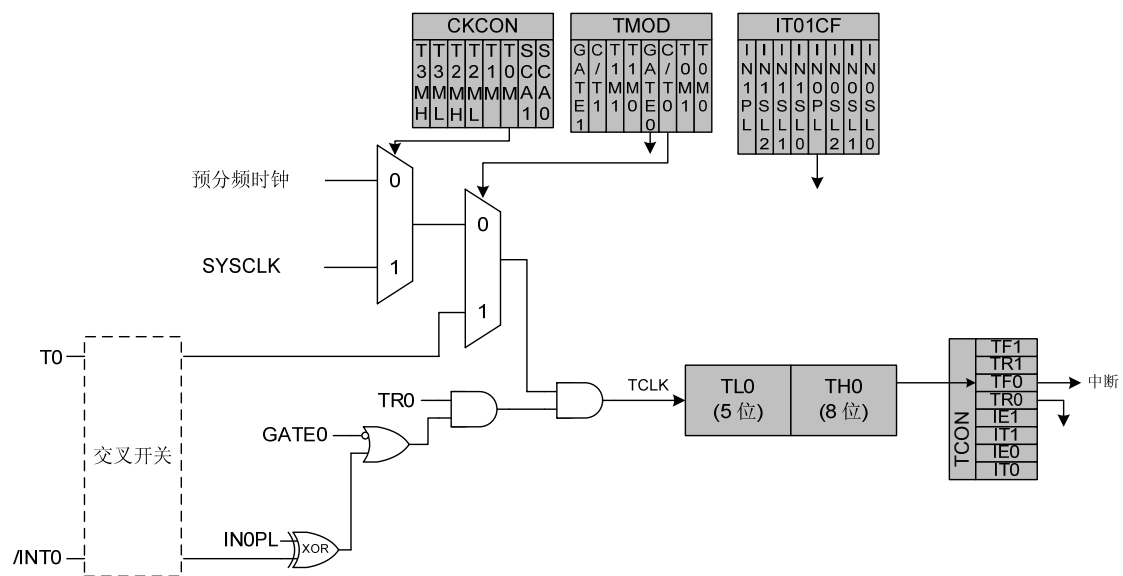


图 24.1 T0 方式 0 原理框图

24.1.2 方式 1 — 16 位计数器/定时器

方式 1 的操作与方式 0 完全一样，所不同的是计数器/定时器使用全部 16 位。用与方式 0 相同的方法使能和控制工作在方式 1 的计数器/定时器。

24.1.3 方式 2 — 自动重载的 8 位计数器/定时器

方式 2 将定时器 0 和定时器 1 配置为具有自动重新装入计数初值能力的 8 位计数器/定时器。TL0 保持计数值，而 TH0 保持重载值。当 TL0 中的计数值发生溢出（从全 1 到 0x00）时，定时器溢出标志 TF0（TCON.5）被置位，TH0 中的重载值被重新装入到 TL0。如果定时器 0 中断被允许，在 TF0 被置位时将产生一个中断。TH0 中的重载值保持不变。为了保证第一次计数正确，必须在允许定时器之前将 TL0 初始化为所希望的计数初值。当工作于方式 2 时，定时器 1 的操作与定时器 0 完全相同。

在方式 2，定时器 1 和定时器 0 的配置和控制方法与方式 0 一样。当 GATE0（TMOD.3）为逻辑 0 或输入信号/INT0 有效时（有效电平由 IT01CF 寄存器中的 IN0PL 为定义），置位 TR0（TCON.4）将允许定时器 0 工作。

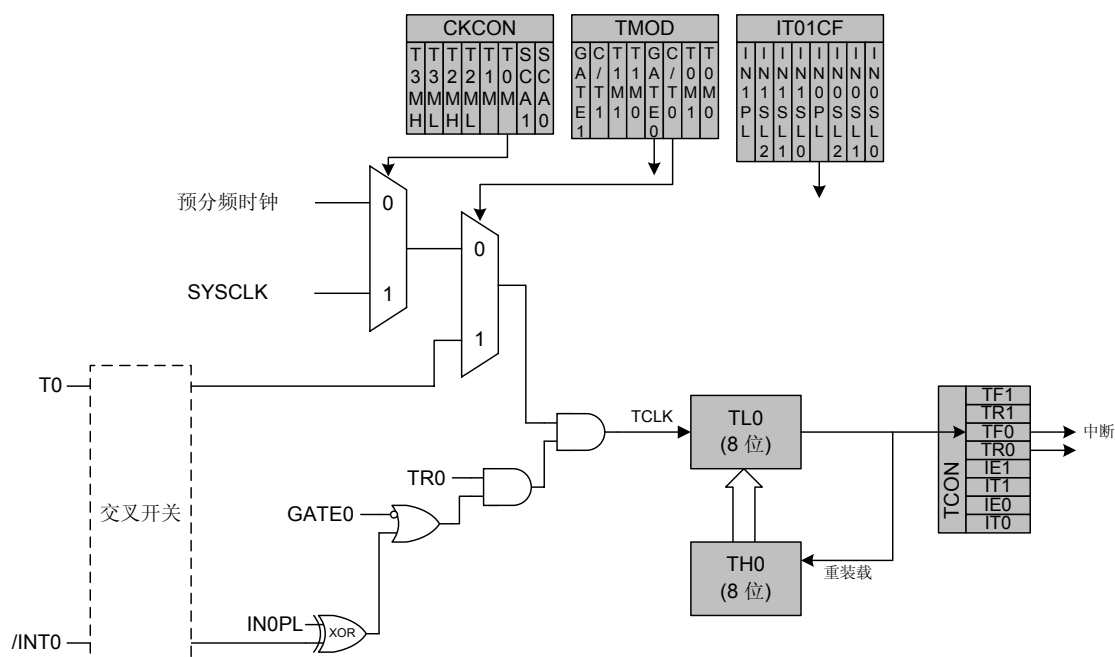


图 24.2 T0 方式 2 原理框图

24.1.4 方式 3 — 两个 8 位计数器/定时器（仅定时器 0）

在方式 3 时,定时器 0 被配置为两个独立的 8 位定时器/计数器,计数值分别在 TL0 和 TH0 中。在 TL0 中的计数器/定时器使用 TCON 和 TMOD 中定时器 0 的控制/状态位: TR0、C/T0、GATE0 和 TF0。TL0 既可以使用系统时钟也可以使用一个外部输入信号作为时基。TH0 寄存器只能作为定时器使用,由系统时钟或预分频时钟提供时基。TH0 使用定时器 1 的运行控制位 TR1,并在发生溢出时将定时器 1 的溢出标志位 TF1 置 1,所以它控制定时器 1 的中断。

定时器 1 在方式 3 时停止运行。在定时器 0 工作于方式 3 时,定时器 1 可以工作在方式 0、1 或 2,但不能用外部信号作为时钟,也不能设置 TF1 标志和产生中断。但是定时器 1 溢出可以用于为 SMBus 和/或 UART 产生波特率,也可以用于启动 ADC 转换。当定时器 0 工作在方式 3 时,定时器 1 的运行控制由其方式设置决定。为了在定时器 0 工作于方式 3 时使用定时器 1,应将定时器 1 的工作在方式设置为 0、1 或 2。可以通过将定时器 1 切换到方式 3 使其停止运行。

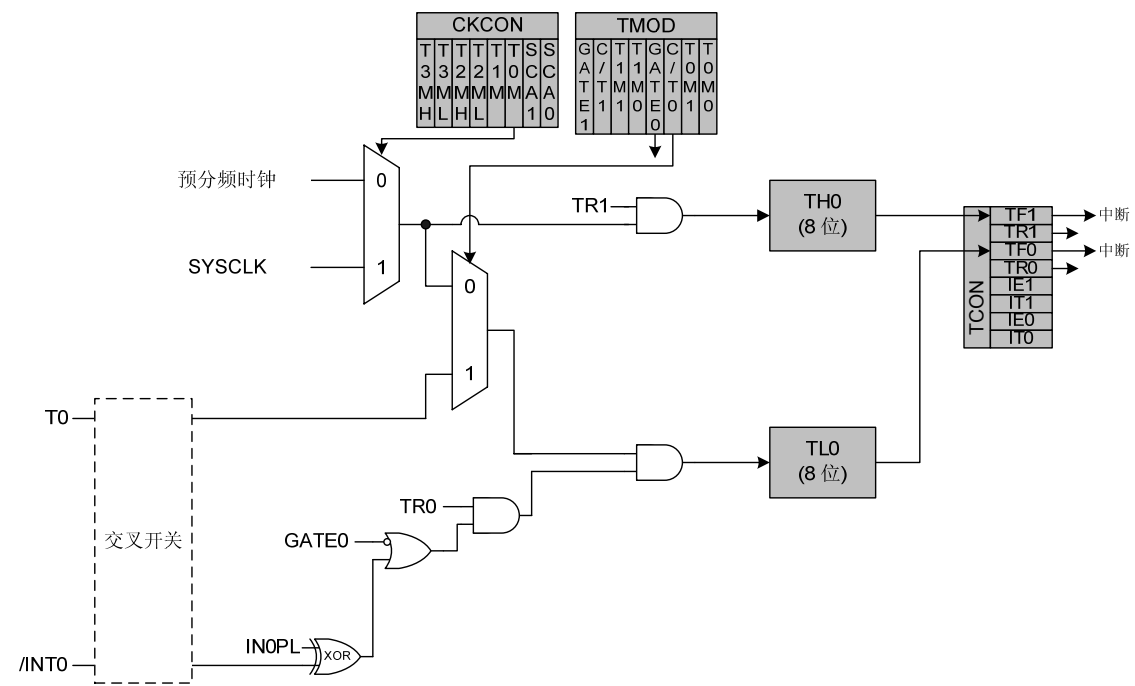


图 24.3 T0 方式 3 原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 24.2 TCON: 定时器控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: (可位寻址) 0x88
位 7:	TF1: 定时器 1 溢出标志 当定时器 1 溢出时由硬件置位。该位可以用软件清 0, 但当 CPU 转向定时器 1 中断服务程序时该位被自动清 0。							
位 6:	TR1: 定时器 1 运行控制 通过将该位置 1 来使能定时器 1。							
位 5:	TF0: 定时器 0 溢出标志 当定时器 0 溢出时由硬件置位。该位可以用软件清 0, 但当 CPU 转向定时器 0 中断服务程序时该位被自动清 0。							
位 4:	TR0: 定时器 0 运行控制 通过将该位置 1 来使能定时器 0。							
位 3:	IE1: 外部中断 1 当检测到一个由 IT1 定义的边沿/电平时, 该标志由硬件置位。该位可以用软件清 0, 但当 CPU 转向外部中断 1 的中断服务程序时该位被自动清 0 (如果 IT1=1)。当 IT1=0 时, 该标志在/INT1 有效时被置 1 (有效电平由 IT01CF 寄存器中的 IN1PL 位定义)。							
位 2:	IT1: 中断 1 类型选择 该位选择/INT1 中断是边沿触发还是电平触发。可以用 IT01CF 寄存器中的 IN1PL 位将/INT1 配置为低电平有效或高电平有效 (见 SFR 定义 15.5)。 0: /INT1 为电平触发。 1: /INT1 为边沿触发。							
位 1:	IE0: 外部中断 0 当检测到一个由 IT0 定义的边沿/电平时, 该标志由硬件置位。该位可以用软件清 0, 但当 CPU 转向外部中断 0 的中断服务程序时该位被自动清 0 (如果 IT0=1)。当 IT0=0 时, 该标志在/INT0 有效时被置 1 (有效电平由 IT01CF 寄存器中的 IN0PL 位定义)。							
位 0:	IT0: 中断 0 类型选择 该位选择/INT0 中断是边沿触发还是电平触发。可以用 IT01CF 寄存器中的 IN0PL 位将/INT0 配置为低电平有效或高电平有效 (见 SFR 定义 15.5)。 0: /INT0 为电平触发。 1: /INT0 为边沿触发。							

SFR 定义 24.3 TMOD：定时器方式寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
GATE1	C/T1	T1M1	T1M0	GATE0	C/T0	T0M1	T0M0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x89

位 7: GATE1: 定时器 1 门控位
0: 当 TR1=1 时定时器 1 工作，与/INT1 的逻辑电平无关。
1: 只有当 TR1=1 并且/INT1 有效时定时器 1 才工作（有效电平由 IT01CF 寄存器中的 IN1PL 位定义，见 SFR 定义 15.5）。

位 6: C/T1: 计数器/定时器 1 功能选择。
0: 定时器功能: 定时器 1 由 T1M 位（CKCON.4）定义的时钟加 1。
1: 计数器功能: 定时器 1 由外部输入引脚（T1）的负跳变加 1。

位 5-4: T1M1-T1M0: 定时器 1 方式选择
这些位选择定时器 1 的工作方式。

T1M1	T1M0	方式
0	0	方式 0: 13 位计数器/定时器
0	1	方式 1: 16 位计数器/定时器
1	0	方式 2: 自动重装载的 8 位计数器/定时器
1	1	方式 3: 定时器 1 停止运行

位 3: GATE0: 定时器 0 门控位
0: 当 TR0=1 时定时器 0 工作，与/INT0 的逻辑电平无关。
1: 只有当 TR0=1 并且/INT0 有效时定时器 0 才工作（有效电平由 IT01CF 寄存器中的 IN0PL 位定义，见 SFR 定义 15.5）。

位 2: C/T0: 计数器/定时器 0 功能选择。
0: 定时器功能: 定时器 0 由 T0M 位（CKCON.3）定义的时钟加 1。
1: 计数器功能: 定时器 0 由外部输入引脚（T0）的负跳变加 1。

位 1-0: T0M1-T0M0: 定时器 0 方式选择
这些位选择定时器 0 的工作方式。

T0M1	T0M0	方式
0	0	方式 0: 13 位计数器/定时器
0	1	方式 1: 16 位计数器/定时器
1	0	方式 2: 自动重装载的 8 位计数器/定时器
1	1	方式 3: 两个 8 位计数器/定时器

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 24.4 TL0: 定时器 0 低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
位7	位6	位5	位4	位3	位2	位1	位0	00000000
								SFR地址: 0x8A
位 7-0: TL0[7:0]: 定时器 0 低字节								
TL0 寄存器是 16 位定时器 0 的低字节。								

SFR 定义 24.5 TL1: 定时器 1 低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
位7	位6	位5	位4	位3	位2	位1	位0	00000000
								SFR地址: 0x8B
位 7-0: TL1[7:0]: 定时器 1 低字节								
TL1 寄存器是 16 位定时器 1 的低字节。								

SFR 定义 24.6 TH0: 定时器 0 高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
位7	位6	位5	位4	位3	位2	位1	位0	00000000
								SFR地址: 0x8C
位 7-0: TH0[7:0]: 定时器 0 高字节								
TH0 寄存器是 16 位定时器 0 的高字节。								

SFR 定义 24.7 TH1: 定时器 1 高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
位7	位6	位5	位4	位3	位2	位1	位0	00000000
								SFR地址: 0x8D
位 7-0: TH1[7:0]: 定时器 1 高字节								
TH1 寄存器是 16 位定时器 1 的高字节。								

24.2 定时器 2

定时器 2 是一个 16 位的定时器，由两个 8 位的 SFR 组成：TMR2L（低字节）和 TMR2H（高字节）。定时器 2 可以工作在 16 位自动重载方式或 8 位自动重载方式（两个 8 位定时器）。T2SPLIT 位（TMR2CN.3）定义定时器 2 的工作方式。

定时器 2 的时钟源可以是系统时钟、系统时钟/12 或外部振荡源时钟/8。在使用实时时钟（RTC）功能时，外部时钟方式是理想的选择，此时用内部振荡器驱动系统时钟，而定时器 2（和/或 PCA）的时钟由一个精确的外部振荡器提供。注意，外部振荡源时钟/8 与系统时钟同步。

24.2.1 16 位自动重载定时器

当 T2SPLIT 位（TMR2CN.3）为 0 时，定时器 2 工作在自动重载的 16 位定时器方式（见图 24.4）。定时器 2 可以使用 SYSCLK、SYSCLK/12 或外部振荡器时钟/8 作为其时钟源。当 16 位定时器寄存器加 1 后发生溢出（从 0xFFFF 到 0x0000）时，定时器 2 重载寄存器（TMR2RLH 和 TMR2RLL）中的 16 位计数初值被自动装入到定时器 2 寄存器，定时器 2 高字节溢出标志 TF2H（TMR2CN.7）被置 1。如果定时器 2 中断被允许（如果 IE.5 被置 1），每次溢出都将产生中断。如果定时器 2 中断被允许并且 TF2LEN 位（TMR2CN.5）被置 1，则每次低 8 位（TMR2L）溢出时（从 0xFF 到 0x00）将产生一个中断。

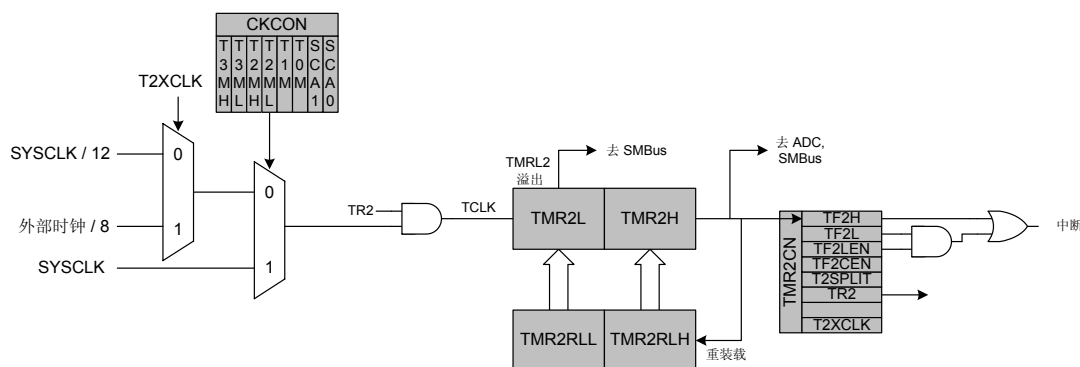


图 24.4 定时器 2 的 16 位方式原理框图

24.2.2 8 位自动重载定时器

当 T2SPLIT 位被置 1 时，定时器 2 工作在双 8 位定时器方式（TMR2H 和 TMR2L）。这两个 8 位定时器都工作在自动重载方式，如图 24.5 所示。TMR2RLL 保持 TMR2L 的重载值，而 TMR2RLH 保持 TMR2H 的重载值。TMR2CN 中的 TR2 是 TMR2H 的运行控制位。当定时器 2 被配置为 8 位方式时，TMR2L 总是处于运行状态。

每个 8 位定时器都可以被配置为使用 SYSCLK、SYSCLK/12 或外部振荡器时钟/8 作为其时钟源。定时器 2 时钟选择位 T2MH 和 T2ML（位于 CKCON 中）选择 SYSCLK 或由定时器 2 外部时钟选择位（TMR2CN 中的 T2XCLK）定义的时钟源。时钟源的选择情况如下所示。

T2MH	T2XCLK	TMR2H 时钟源	T2ML	T2XCLK	TMR2L 时钟源
0	0	SYSCLK/12	0	0	SYSCLK/12
0	1	外部时钟/8	0	1	外部时钟/8
1	X	SYSCLK	1	X	SYSCLK

当 TMR2H 发生溢出时（从 0xFF 到 0x00），TF2H 被置 1；当 TMR2L 发生溢出时（从 0xFF 到 0x00），TF2L 被置 1。如果定时器 2 中断被允许，则每次 TMR2H 溢出时都将产生一个中断。如果定时器 2 中断被允许并且 TF2LEN 位（TMR2CN.5）被置 1，则每当 TMR2L 或 TMR2H 发生溢出时将产生一个中断。在 TF2LEN 位被置 1 的情况下，软件应检查 TF2H 和 TF2L 标志，以确定中断的来源。TF2H 和 TF2L 标志不能被硬件自动清除，必须通过软件清除。

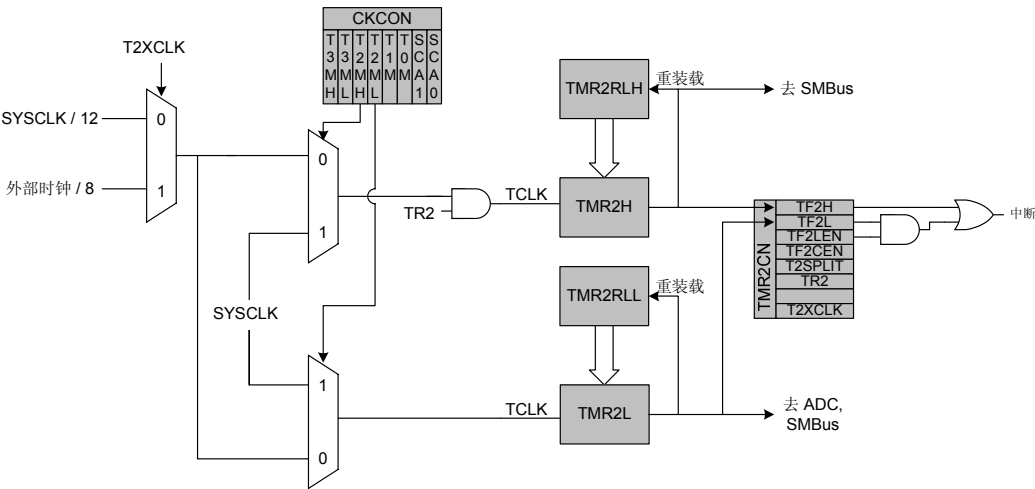


图 24.5 定时器 2 的 8 位方式原理框图

24.2.3 低频振荡器（LFO）捕捉方式

低频振荡器捕捉方式允许使用系统时钟或一个外部振荡源测量 LFO 时钟。定时器 2 可以使用系统时钟、系统时钟/12 或外部振荡器/8 作为其时钟源，由 T2ML (CKCON.4) 和 T2XCLK 的设置决定。

通过将 TF2CEN 置 1 使能定时器 2 的 LFO 捕捉方式。在该方式，T2SPLIT 应被清 0，使用全 16 位定时器。在低频振荡器的下降沿，定时器 2 (TMR2H:TMR2L) 的内容被装入定时器 2 重载寄存器 (TMR2RLH:TMR2RLL)，TF2H 标志被置位。通过计算两个连续的定时器捕捉值的差值，可以确定 LFO 的振荡频率（相对于定时器 2 时钟）。为获得精确的测量值，定时器 2 的时钟频率应远大于 LFO 频率。

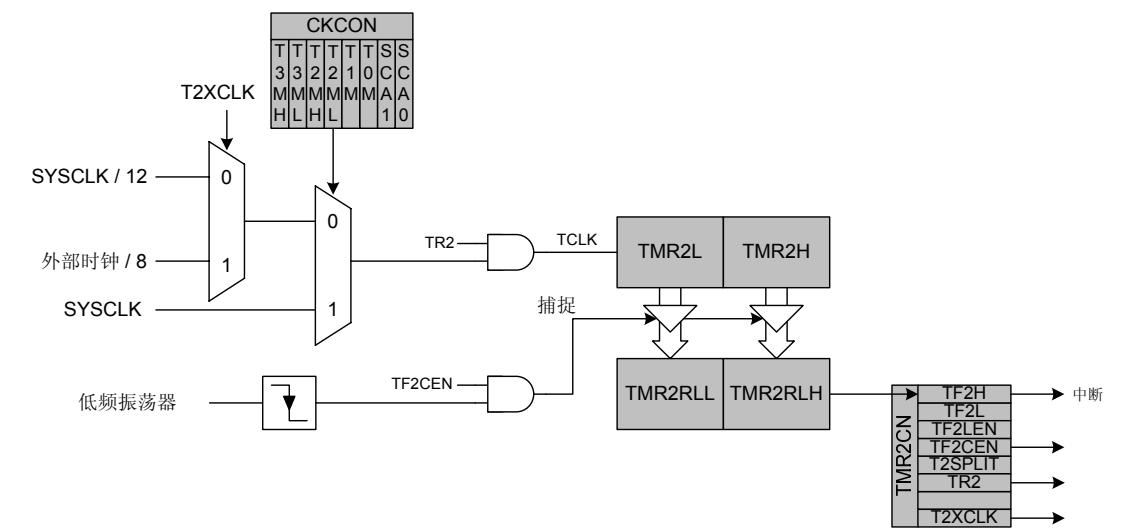


图 24.6 定时器 2 低频振荡器捕捉方式原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 24.8 TMR2CN: 定时器 2 控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	复位值
TF2H	TF2L	TF2LEN	TF2CEN	T2SPLIT	TR2	-	T2XCLK	00000000
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xC8
<p>位 7: TF2H: 定时器 2 高字节溢出标志</p> <p>当定时器 2 高字节发生溢出时 (从 0xFF 到 0x00) 由硬件置 1。在 16 位方式, 当定时器 2 发生溢出时 (从 0xFFFF 到 0x0000) 由硬件置 1。当定时器 2 中断被允许时, 该位置 1 将导致 CPU 转向定时器 2 的中断服务程序。该位不能由硬件自动清 0, 必须用软件清 0。</p> <p>位 6: TF2L: 定时器 2 低字节溢出标志</p> <p>当定时器 2 低字节发生溢出时 (从 0xFF 到 0x00) 由硬件置 1。当定时器 2 中断被允许并且 TF2LEN 位被设置为逻辑 1 时, 该位置 1 将产生中断。TF2L 在低字节溢出时置位, 与定时器 2 的工作方式无关。该位不能由硬件自动清 0, 必须用软件清 0。</p> <p>位 5: TF2LEN: 定时器 2 低字节中断允许位</p> <p>该位置 1 将允许定时器 2 低字节中断。如果定时器 2 中断被允许 (IE.5), 则当定时器 2 低字节发生溢出时将产生一个中断。当定时器 2 工作在 16 位方式时, 该位应被清 0。</p> <p>0: 禁止定时器 2 低字节中断。 1: 允许定时器 2 低字节中断。</p> <p>位 4: TF2CEN: 定时器 2 低频振荡器捕捉允许位</p> <p>该位允许/禁止定时器 2 低频振荡器捕捉方式。如果 TF2CEN 被置 1 且定时器 2 中断被允许, 在低频振荡器输出的下降沿将产生中断, TMR2H:TMR2L 中的 16 位定时器值被复制到 TMR2RLH:TMR2RL。</p> <p>0: 禁止定时器 2 低频振荡器捕捉方式。 1: 允许定时器 2 低频振荡器捕捉方式。</p> <p>位 3: T2SPLIT: 定时器 2 双 8 位方式允许位</p> <p>当该位被置 1 时, 定时器 2 工作在双 8 位自动重装载定时器方式。</p> <p>0: 定时器 2 工作在 16 位自动重装载方式。 1: 定时器 2 工作在双 8 位自动重装载定时器方式。</p> <p>位 2: TR2: 定时器 2 运行控制</p> <p>该位允许/禁止定时器 2。在 8 位方式, 该位只控制 TMR2H, TMR2L 总是处于运行状态。</p> <p>0: 定时器 2 禁止。 1: 定时器 2 允许。</p> <p>位 1: 未用。读 = 0b, 写 = 忽略。</p> <p>位 0: T2XCLK: 定时器 2 外部时钟选择</p> <p>该位选择定时器 2 的外部时钟源。如果定时器 2 工作在 8 位方式, 该位为两个 8 位定时器选择外部振荡器时钟源。但仍可用定时器 2 时钟选择位 (CKCON 中的 T2MH 和 T2ML) 在外部时钟和系统时钟之间作出选择。</p> <p>0: 定时器 2 时钟为系统时钟/12。 1: 定时器 2 时钟为外部时钟/8 (与系统时钟同步)。</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 24.9 TMR2RL: 定时器 2 重载寄存器低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xCA
位 7-0: TMR2RL[7:0]: 定时器 2 重载寄存器的低字节 TMR2RL 保持定时器 2 重载值的低字节。								

SFR 定义 24.10 TMR2RH: 定时器 2 重载寄存器高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xCB
位 7-0: TMR2RH[7:0]: 定时器 2 重载寄存器的高字节 TMR2RH 保持定时器 2 重载值的高字节。								

SFR 定义 24.11 TMR2L: 定时器 2 低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xCC
位 7-0: TMR2L[7:0]: 定时器 2 的低字节 在 16 位方式, TMR2L 寄存器保持 16 位定时器 2 的低字节。在 8 位方式, TMR2L 中保持 8 位低字节定时器的计数值。								

SFR 定义 24.12 TMR2H: 定时器 2 高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xCD
位 7-0: TMR2H[7:0]: 定时器 2 的高字节 在 16 位方式, TMR2H 寄存器保持 16 位定时器 2 的高字节。在 8 位方式, TMR2H 中保持 8 位高字节定时器的计数值。								

24.3 定时器 3

定时器 3 是一个 16 位的定时器，由两个 8 位的 SFR 组成：TMR3L（低字节）和 TMR3H（高字节）。定时器 3 可以工作在 16 位自动重载方式或 8 位自动重载方式（两个 8 位定时器）。T3SPLIT 位（TMR3CN.3）定义定时器 3 的工作方式。

定时器 3 的时钟源可以是系统时钟、系统时钟/12、外部振荡源时钟/8 或内部低频振荡器/8。在使用实时时钟（RTC）功能时，外部时钟方式是理想的选择，此时用内部振荡器驱动系统时钟，而定时器 3 的时钟由一个外部振荡源提供。注意，除了挂起方式，在所有工作方式中，外部振荡源时钟/8 和 LFO 源/8 与系统时钟同步。当内部振荡器被置于挂起方式时，外部时钟/8 信号或 LFO/8 输出可直接驱动定时器 3。这就允许使用外部时钟或 LFO 将器件从挂起方式唤醒。定时器在挂起方式继续运行并向上计数。当定时器发生溢出时，器件会从挂起方式唤醒，继续执行代码。可以在进入挂起方式之前设置定时器值，使定时器在期望的时间内（时钟数）溢出，将器件从挂起方式唤醒。如果将器件从挂起方式唤醒的唤醒源不是定时器，则可能要经过最多三个定时器周期才能读或写定时器的寄存器。在此期间，寄存器 OSCICN 中的 STSYNC 为被置 1，指示读或写定时器寄存器是不安全的。

注意：在内部 LFO/8 方式，内部 LFO 的分频系数必须被设置为 1 才能正常工作。如果内部 LFO 的分频系数设置值不为 1，则定时器将不工作。

24.3.1 16 位自动重载定时器

当 T3SPLIT 位（TMR3CN.3）为 0 时，定时器 3 工作在自动重载的 16 位定时器方式。定时器 3 可以使用 SYSCLK、SYSCLK/12 或外部振荡器时钟/8 作为其时钟源。当 16 位定时器寄存器发生溢出（从 0xFFFF 到 0x0000）时，定时器 3 重载寄存器（TMR3RLH 和 TMR3RLL）中的 16 位计数初值被自动装入到定时器 3 寄存器（如图 24.7 所示），并将定时器 3 高字节溢出标志 TF3H（TMR3CN.7）置 1。如果定时器 3 中断被允许（EIE1.7 被置 1），则每次溢出都将产生中断。如果定时器 3 中断被允许并且 TF3LEN 位（TMR3CN.5）被置 1，则每次低 8 位（TMR3L）溢出时（从 0xFF 到 0x00）将产生中断。

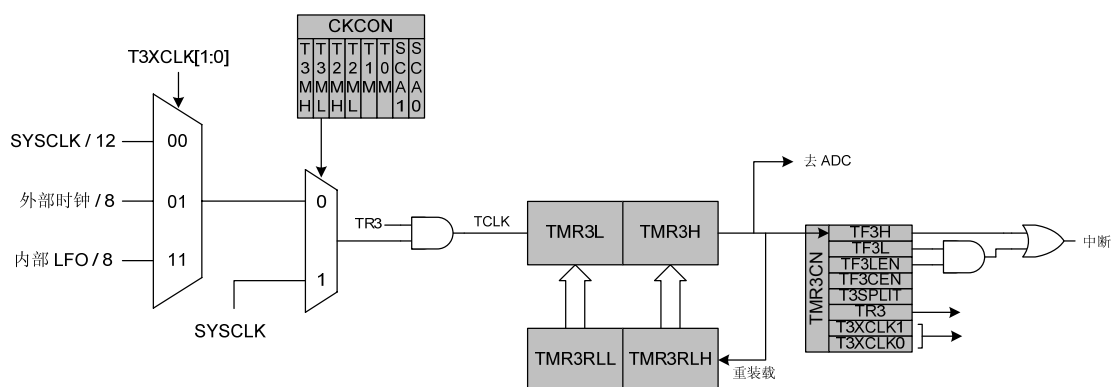


图 24.7 定时器 3 的 16 位方式原理框图

24.3.2 8 位自动重载定时器

当 T3SPLIT 被置 1 时，定时器 3 工作双 8 位定时器方式（TMR3H 和 TMR3L）。这两个 8 位定时器都工作在自动重载方式（如图 24.8 所示）。TMR3RLL 保持 TMR3L 的重载值，而 TMR3RLH 保持 TMR3H 的重载值。TMR3CN 中的 TR3 是 TMR3H 的运行控制位。当定时器 3 被配置为 8 位方式时，TMR3L 总是处于运行状态。

每个 8 位定时器都可以被配置为使用 SYSCLK、SYSCLK/12、外部振荡器时钟/8 或内部低频振荡器/8 作为其时钟源。定时器 3 时钟选择位(CKCON 的 T3MH 和 T3ML)选择 SYSCLK 或由定时器 3 外部时钟选择位（TMR3CN 中的 T3XCLK[1:0]）定义的时钟源。时钟源的选择情况如下所示。

T3MH	T3XCLK[1:0]	TMR3H 时钟源	T3ML	T3XCLK[1:0]	TMR3L 时钟源
0	00	SYSCLK/12	0	00	SYSCLK/12
0	01	外部时钟/8	0	01	外部时钟/8
0	10	保留	0	10	保留
0	11	内部 LFO	0	11	内部 LFO
1	X	SYSCLK	1	X	SYSCLK

当 TMR3H 发生溢出时（从 0xFF 到 0x00），TF3H 被置 1；当 TMR3L 发生溢出时（从 0xFF 到 0x00），TF3L 被置 1。如果定时器 3 中断被允许，则每次 TMR3H 溢出时都将产生一个中断。如果定时器 3 中断被允许并且 TF3LEN 位（TMR3CN.5）被置 1，则每当 TMR3L 或 TMR3H 发生溢出时将产生一个中断。在 TF3LEN 位被置 1 的情况下，软件必须检查 TF3H 和 TF3L 标志，以确定定时器 3 中断的来源。TF3H 和 TF3L 中断标志不能被硬件自动清除，必须通过软件手动清除。

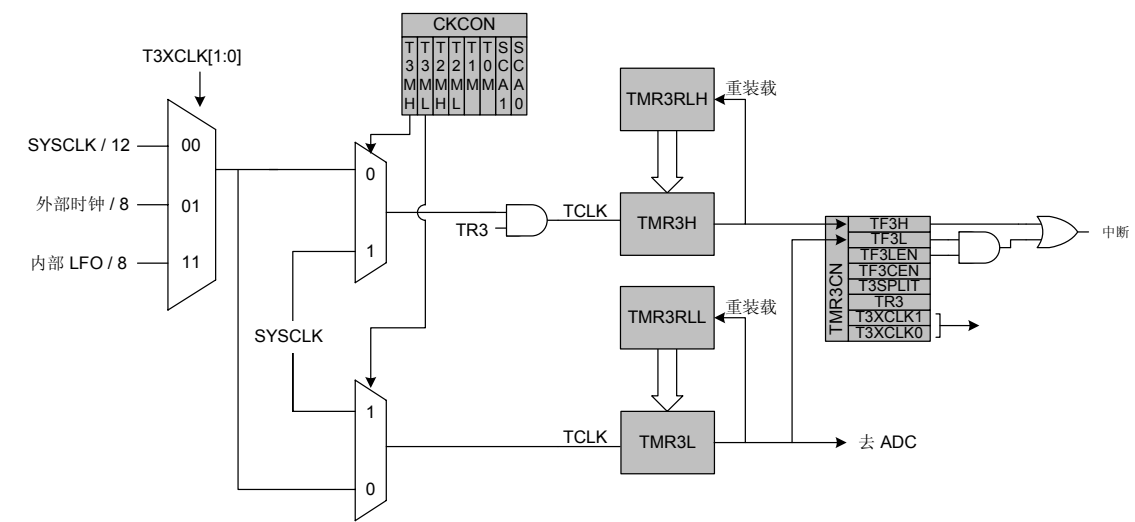


图 24.8 定时器 3 的 8 位方式原理框图

24.3.3 低频振荡器（LFO）捕捉方式

低频振荡器捕捉方式允许使用系统时钟或一个外部振荡源测量 LFO 时钟。定时器 3 可以使用系统时钟、系统时钟/12 或外部振荡器/8 作为其时钟源，由 T3ML (CKCON.6) 和 T3XCLK 的设置决定。

通过将 TF3CEN 置 1 使能定时器 3 的 LFO 捕捉方式。在该方式，T3SPLIT 应被清 0，使用全 16 位定时器。在低频振荡器的下降沿，定时器 3 (TMR3H:TMR3L) 的内容被装入定时器 3 重载寄存器 (TMR3RLH:TMR3RLL)，TF3H 标志被置位。通过计算两个连续的定时器捕捉值的差值，可以确定 LFO 的振荡频率（相对于定时器 3 时钟）。为获得精确的测量值，定时器 3 的时钟频率应远大于 LFO 频率。这意味着在该方式不应选择 LFO/8 作为定时器时钟源。

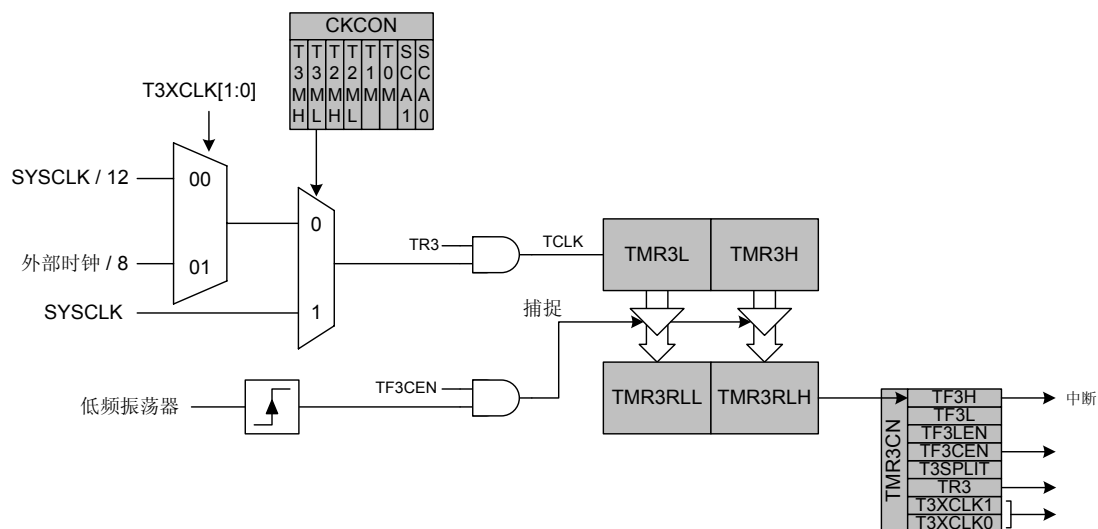


图 24.9 定时器 3 的低频振荡器捕捉方式原理框图

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 24.13 TMR3CN: 定时器 3 控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
TF3H	TF3L	TF3LEN	TF3CEN	T3SPLIT	TR3	T3XCLK1	T3XCLK0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x91
<p>位 7: TF3H: 定时器 3 高字节溢出标志 当定时器 3 高字节发生溢出时（从 0xFF 到 0x00）由硬件置 1。在 16 位方式，当定时器 3 发生溢出时（从 0xFFFF 到 0x0000）由硬件置 1。当定时器 3 中断被允许时，该位置 1 将导致 CPU 转向定时器 3 的中断服务程序。该位不能由硬件自动清 0，必须用软件清 0。</p> <p>位 6: TF3L: 定时器 3 低字节溢出标志 当定时器 3 低字节发生溢出时（从 0xFF 到 0x00）由硬件置 1。当定时器 3 中断被允许并且 TF3LEN 位被设置为逻辑 1 时，该位置 1 将产生中断。TF3L 在低字节溢出时置位，与定时器 3 的工作方式无关。该位不能由硬件自动清 0。</p> <p>位 5: TF3LEN: 定时器 3 低字节中断允许位 该位允许/禁止定时器 3 低字节中断。如果 TF3LEN 被置 1 并且定时器 3 中断被允许，则当定时器 3 低字节发生溢出时将产生一个中断。当定时器 3 工作在 16 位方式时，该位应被清 0。 0: 禁止定时器 3 低字节中断。 1: 允许定时器 3 低字节中断。</p> <p>位 4: TF3CEN: 定时器 3 低频振荡器捕捉允许位 该位允许/禁止定时器 3 低频振荡器捕捉方式。如果 TF3CEN 被置 1 且定时器 3 中断被允许，在低频振荡器输出的上升沿将产生中断，TMR3H:TMR3L 中的 16 位定时器值被复制到 TMR3RLH:TMR3RLL。 0: 禁止定时器 3 低频振荡器捕捉方式。 1: 允许定时器 3 低频振荡器捕捉方式。</p> <p>位 3: T3SPLIT: 定时器 3 双 8 位方式允许位 当该位被置 1 时，定时器 3 工作在双 8 位自动重装载定时器方式。 0: 定时器 3 工作在 16 位自动重装载方式。 1: 定时器 3 工作在双 8 位自动重装载定时器方式。</p> <p>位 2: TR3: 定时器 3 运行控制 该位允许/禁止定时器 3。在 8 位方式，该位只控制 TMR3H，TMR3L 总是处于运行状态。 0: 定时器 3 禁止。 1: 定时器 3 允许。</p> <p>位 1-0: T3XCLK[1:0]: 定时器 3 外部时钟选择 该位选择定时器 3 的“外部”时钟源。如果定时器 3 工作在 8 位方式，该位为两个 8 位定时器选择外部振荡器时钟源。但仍可用定时器 3 时钟选择位（CKCON 中的 T3MH 和 T3ML）在外部时钟和系统时钟之间作出选择。 00: 选择定时器 3 时钟为系统时钟/12。 01: 选择定时器 3 时钟为外部时钟/8（与系统时钟同步，挂起方式除外）。 10: 保留。 11: 内部 LFO/8（与系统时钟同步，挂起方式除外）。</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 24.14 TMR3RLL: 定时器 3 重载寄存器低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x92
位 7-0: TMR3RLL[7:0]: 定时器 3 重载寄存器的低字节 TMR3RLL 保存定时器 3 重载值的低字节。								

SFR 定义 24.15 TMR3RLH: 定时器 3 重载寄存器高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x93
位 7-0: TMR3RLH[7:0]: 定时器 3 重载寄存器的高字节 TMR3RLH 保存定时器 3 重载值的高字节。								

SFR 定义 24.16 TMR3L: 定时器 3 低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x94
位 7-0: TMR3L[7:0]: 定时器 3 的低字节 在 16 位方式, TMR3L 寄存器保持 16 位定时器 3 的低字节。在 8 位方式, TMR3L 中保持 8 位低字节定时器的计数值。								

SFR 定义 24.17 TMR3H: 定时器 3 高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x95
位 7-0: TMR3H[7:0]: 定时器 3 的高字节 在 16 位方式, TMR3H 寄存器保持 16 位定时器 3 的高字节。在 8 位方式, TMR3H 中保持 8 位高字节定时器的计数值。								

25. 可编程计数器阵列

可编程计数器阵列（PCA0）提供增强的定时器功能，与标准8051的计数器/定时器相比，它需要较少的CPU干预。PCA由一个专用的16位计数器/定时器和3个16位捕捉/比较模块组成。每个捕捉/比较模块有其自己的I/O线（CEX_n），这些I/O线在被允许时通过交叉开关连到端口I/O。计数器/定时器由一个可编程的时基信号驱动，时基信号可以在六个时钟源中选择：系统时钟、系统时钟/4、系统时钟/12、外部振荡器时钟/8、定时器0溢出或ECI输入引脚上的外部时钟信号。每个捕捉/比较模块都有六种工作方式：边沿触发捕捉、软件定时器、高速输出、频率输出、8~11位PWM和16位PWM，每个捕捉/比较模块的工作方式都可以被独立配置。在使用实时时钟（RTC）功能时，外部振荡器时钟选项是理想的选择，此时用内部振荡器驱动系统时钟，而PCA的时钟由一个精确的外部振荡器提供。对PCA的配置和控制是通过系统控制器的特殊功能寄存器来实现的。PCA的原理框图示于图25.1。

注意：PCA的模块2可被用作看门狗定时器（WDT），在系统复位后即被使能为该方式。在看门狗方式被使能时，对某些PCA寄存器的访问受到限制，详见25.4节。

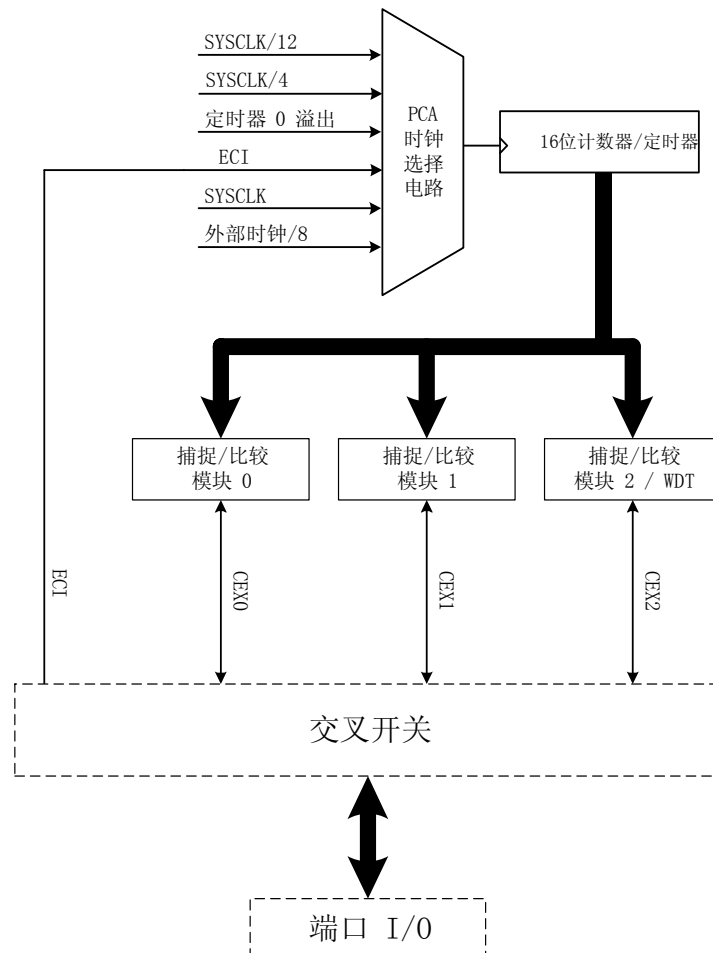


图 25.1 PCA 原理框图

25.1 PCA 计数器/定时器

16 位的 PCA 计数器/定时器由两个 8 位的 SFR 组成：PCA0L 和 PCA0H。PCA0H 是 16 位计数器/定时器的高字节（MSB），而 PCA0L 是低字节（LSB）。在读 PCA0L 时，“瞬象寄存器”自动锁存 PCA0H 的值，随后读 PCA0H 时将访问这个“瞬象寄存器”而不是 PCA0H 本身。**先读 PCA0L 寄存器可以保证正确读取整个 16 位 PCA0 计数器的值。**读 PCA0H 或 PCA0L 不影响计数器工作。PCA0MD 寄存器中的 CPS2-CPS0 位用于选择 PCA 计数器/定时器的时基，如表 25.1 所示。

当计数器/定时器溢出时（从 0xFFFF 到 0x0000），PCA0CN 中的计数器溢出标志（CF）被置为逻辑 1 并产生一个中断请求（如果 CF 中断被允许）。将 PCA0MD 中 ECF 位设置为逻辑 1 即可允许 CF 标志产生中断请求。当 CPU 转向中断服务程序时，CF 位不能被硬件自动清除，必须用软件清除。清除 PCA0MD 寄存器中的 CIDL 位将允许 PCA 在微控制器内核处于空闲方式时继续正常工作。

表 25.1 PCA 时基输入选择

CPS2	CPS1	CPS0	时间基准
0	0	0	系统时钟的 12 分频
0	0	1	系统时钟的 4 分频
0	1	0	定时器 0 溢出
0	1	1	ECI 下降沿（最大速率 = 系统时钟频率/4）
1	0	0	系统时钟
1	0	1	外部振荡器 8 分频*
1	1	X	保留
*注：外部振荡器 8 分频与系统时钟同步。			

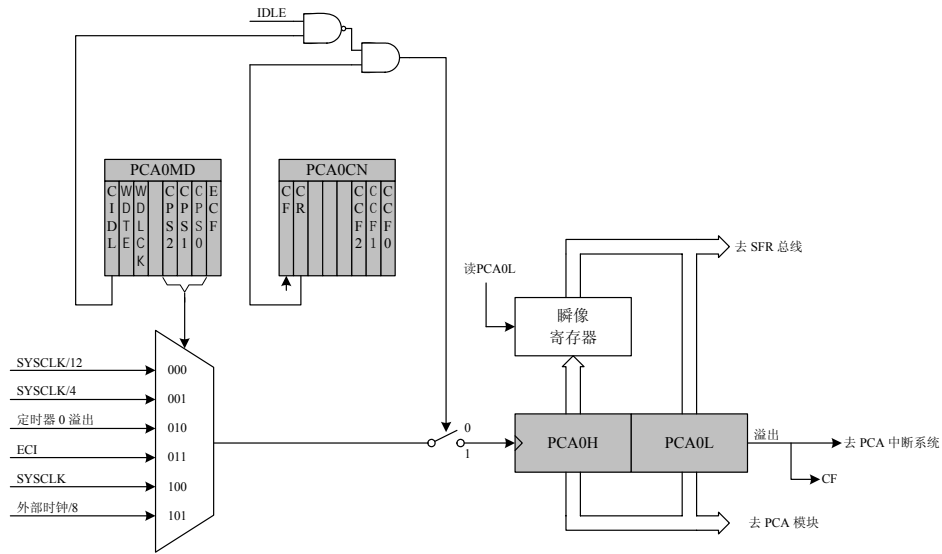


图 25.2 PCA 计数器/定时器原理框图

25.2 PCA0 中断源

图 25.3 给出了 PCA 中断树的原理框图。有 5 个独立的事件标志可用于产生 PCA0 中断。它们是：主 PCA 计数器溢出标志 (CF)，在 16 位 PCA0 计数器溢出时置 1；中间溢出标志 (COVF)，可以在 PCA0 计数器的第 8、9、10 或 11 位溢出时置 1；以及每个 PCA 通道的独立标志位 (CCF0、CCF1 和 CCF2)，**根据对应模块的工作方式被置 1**。当触发条件发生时，这些事件标志将被置 1。通过使用相应的中断允许位 (ECF 对应 CF，ECOV 对应 COVF，ECCFn 对应 CCFn) 来选择用于产生中断的标志位。要想使任何一个独立的中断源能被处理器识别，必须先总体允许 PCA0 中断。通过将 EA 位 (IE.7) 和 EPCA0 位 (EIE1.4) 设置为逻辑 1 来总体允许 PCA0 中断。

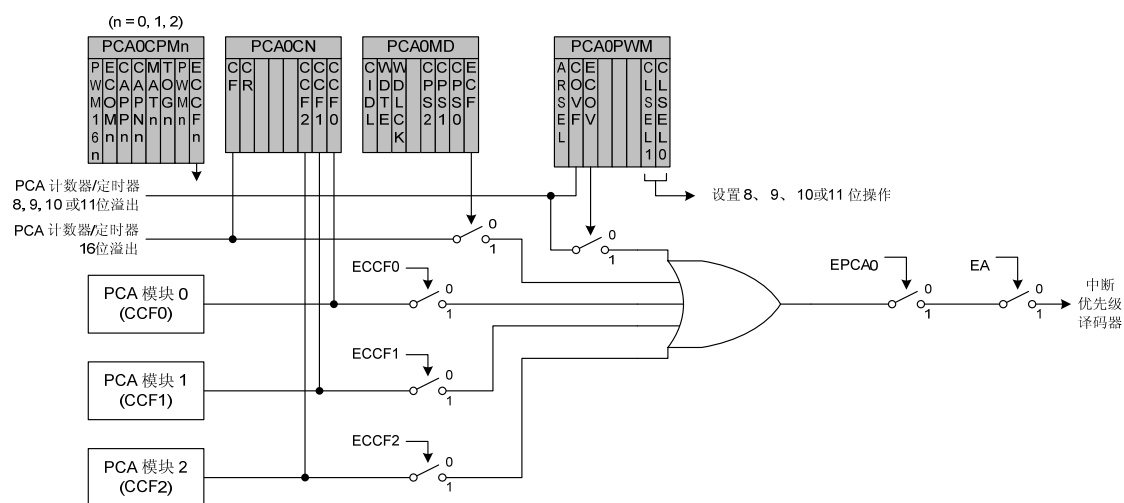


图 25.3 PCA 中断原理框图

25.3 捕捉/比较模块

每个模块都可被配置为独立工作，有六种工作方式：边沿触发捕捉、软件定时器、高速输出、频率输出、8~11位脉宽调制器和16位脉宽调制器。每个模块在CIP-51系统控制器中都有属于自己的特殊功能寄存器（SFR），这些寄存器用于配置模块的工作方式和与模块交换数据。表25.2概述了用于选择PCA捕捉/比较模块工作方式的PCA0CPMn和PCA0PWM寄存器中各位的设置情况。注意：被设置为8、9、10或11位PWM方式的所有模块使用相同的周期长度（8~11位）。置位PCA0CPMn寄存器中的ECCFn位将允许模块的CCFn中断。

表 25.2 PCA 捕捉/比较模块的 PCA0CPM 和 PCA0PWM 寄存器的位设置

工作方式	位号	PCA0CPMn								PCA0PWM				
		7	6	5	4	3	2	1	0	7	6	5	4-2	1-0
用 CEXn 的正沿触发捕捉		X	X	1	0	0	0	0	A	0	X	B	XXX	XX
用 CEXn 的负沿触发捕捉		X	X	0	1	0	0	0	A	0	X	B	XXX	XX
用 CEXn 的任一边沿触发捕捉		X	X	1	1	0	0	0	A	0	X	B	XXX	XX
软件定时器		X	C	0	0	1	0	0	A	0	X	B	XXX	XX
高速输出		X	C	0	0	1	1	0	A	0	X	B	XXX	XX
频率输出		X	C	0	0	0	1	1	A	0	X	B	XXX	XX
8 位脉冲宽度调制器（注 7）		0	C	0	0	E	0	1	A	0	X	B	XXX	00
9 位脉冲宽度调制器（注 7）		0	C	0	0	E	0	1	A	D	X	B	XXX	01
10 位脉冲宽度调制器（注 7）		0	C	0	0	E	0	1	A	D	X	B	XXX	10
11 位脉冲宽度调制器（注 7）		0	C	0	0	E	0	1	A	D	X	B	XXX	11
16 位脉冲宽度调制器		1	C	0	0	E	0	1	A	0	X	B	XXX	XX
注： 1. X = 任意值（对每个单独的模块而言，该位为 1 或为 0 没有功能差别）。 2. A = 使能该模块的中断（CCFn 被置 1 时触发 PCA 中断）。 3. B = 使能第 8、9、10、11 位溢出中断（取决于 CLSEL[1:0]的设置）。 4. C = 当被设置为 0 时，数字比较器不工作。对于高速和频率输出方式，相关引脚的电平不切换。在任何一种 PWM 方式，将产生 0%的占空比（输出 = 0）。 5. D = 选择通过地址 PCA0CPHn 和 PCA0CPLn 访问的是相关通道的捕捉/比较寄存器（0）还是自动重载寄存器（1）。 6. E = 当被置 1 时，一个匹配事件会导致相关通道的 CCFn 标志被置 1。 7. 被设置为 8、9、10 或 11 位 PWM 方式的所有模块都使用相同的周期长度设置。														

图 25.3 PCA 中断原理框图

25.3.1 边沿触发的捕捉方式

在该方式，CEXn引脚上出现的有效电平跳变导致PCA捕捉PCA计数器/定时器的值并将其装入到对应模块的16位捕捉/比较寄存器（PCA0CPLn和PCA0CPHn）。PCA0CPMn寄存器中的CAPPn和CAPNn位用于选择触发捕捉的电平变化类型：低电平到高电平（正沿）、高电平到低电平（负沿）或任何变化（正沿或负沿）。当捕捉发生时，PCA0CN中的捕捉/比较标志（CCFn）被置为逻辑1。如果该模块的CCFn中断被允许，将产生一个中断请求。当CPU转向中断服务程序时，CCFn位不能被硬件自动清除，必须用软件清0。如果CAPPn和CAPNn位都被设置为逻辑1，则可以通过直接读CEXn对应端口引脚的状态来确定本次捕捉是由上升沿触发还是由下降沿触发。

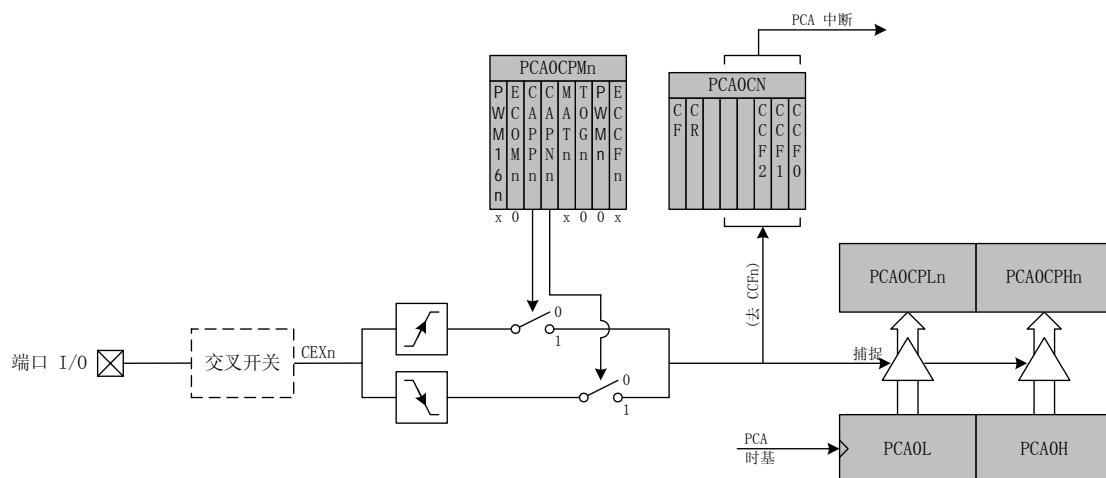


图25.4 PCA捕捉方式原理框图

注意：CEXn 输入信号必须在高电平或低电平期间至少保持两个系统时钟周期，以保证能够被硬件识别。

25.3.2 软件定时器（比较）方式

在软件定时器方式，PCA将计数器/定时器的值与模块的16位捕捉/比较寄存器(PCA0CPHn和PCA0CPLn) 进行比较。当发生匹配时，PCA0CN中的捕捉/比较标志（CCFn）被置为逻辑1。如果该模块的CCFn中断被允许，将产生一个中断请求。当CPU转向中断服务程序时，CCFn位不能被硬件自动清除，必须用软件清0。置位PCA0CPMn寄存器中的ECOMn和MATn位将使能软件定时器方式。

注意，当向PCA0的捕捉/比较寄存器写入一个16位数值时，应先写低字节。向PCA0CPLn的写入操作将ECOMn位清0；向PCA0CPHn写入时将ECOMn位置1。

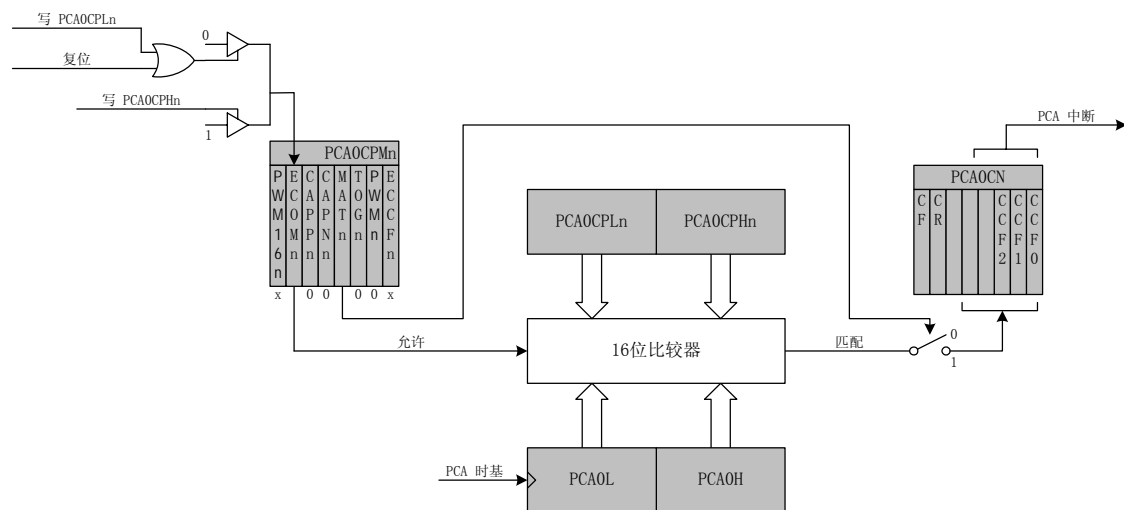


图 25.5 PCA 软件定时器方式原理框图

25.3.3 高速输出方式

在高速输出方式，每当PCA计数器与模块的16位捕捉/比较寄存器（PCA0CPHn和PCA0CPLn）发生匹配时，模块的CEXn引脚上的逻辑电平将发生变化。当发生匹配时，PCA0CN中的捕捉/比较标志（CCFn）被置为逻辑1。如果该模块的CCFn中断被允许，将产生一个中断请求。当CPU转向中断服务程序时，CCFn位不能被硬件自动清除，必须用软件清0。置位PCA0CPMn寄存器中的TOGn、MATn和ECOMn位将使能高速输出方式。如果ECOMn被清0，则模块的CEXn引脚将保持其状态，在下一个匹配事件到来时不会发生电平切换。

注意：当向PCA0的捕捉/比较寄存器写入一个16位数值时，应先写低字节。向PCA0CPLn的写入操作将ECOMn位清0；向PCA0CPHn写入时将ECOMn位置1。

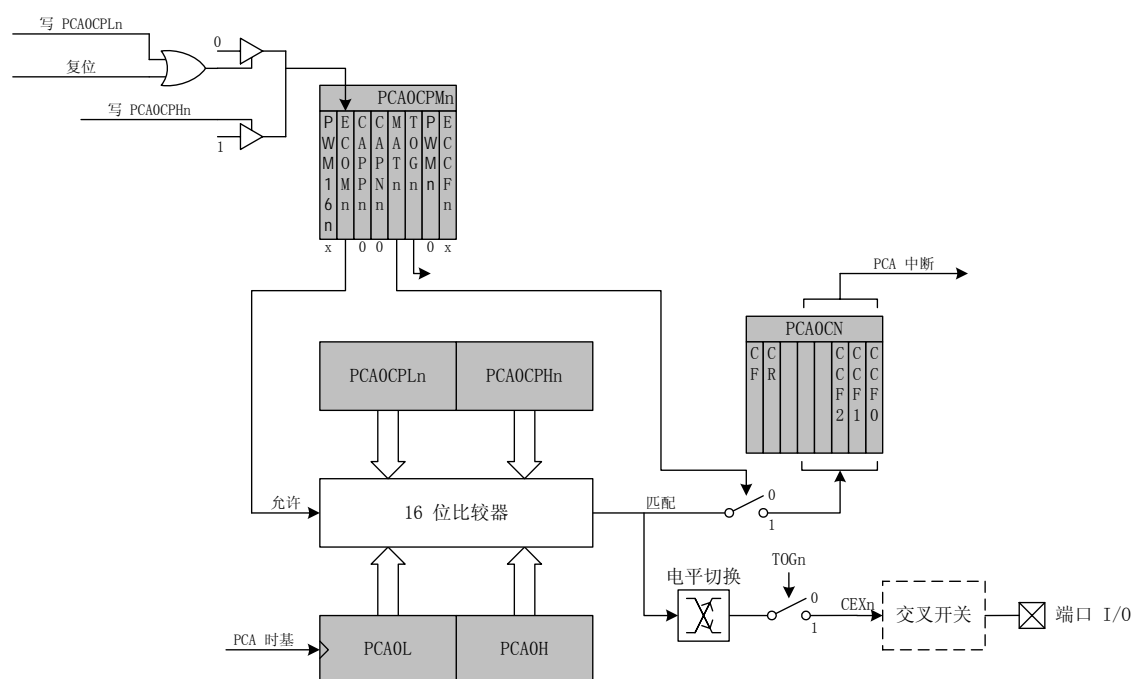


图 25.6 PCA 高速输出方式原理框图

25.3.4 频率输出方式

频率输出方式在模块的CEXn引脚产生可编程频率的方波。捕捉/比较模块的高字节保持输出电平改变前要计的PCA时钟数。所产生的方波的频率由方程25.1定义：

$$F_{CEXn} = \frac{F_{PCA}}{2 \times PCA0CPHn}$$

注：对于该方程，PCA0CPHn寄存器中的值为0x00时，相当于256。

方程25.1 方波输出频率

其中： F_{PCA} 是由PCA方式寄存器（PCA0MD）中的CPS2-0位选择的PCA时钟的频率。捕捉/比较模块的低字节与PCA0计数器的低字节比较；两者匹配时，CEXn的电平发生翻转，高字节中的偏移值被加到PCA0CPLn中的匹配值。通过将PCA0CPMn寄存器中ECOMn、TOGn和PWMn位置1来使能频率输出方式。注意：在该方式，MATn应被设置为0。如果MATn被设置为1，则当16位PCA0计数器与该模块的16位捕捉比较寄存器相等时，模块的CCFn标志被置1。

关于捕捉/比较寄存器的重要注意事项：当向PCA0的捕捉/比较寄存器写入一个16位值时，应先写低字节。向PCA0CPLn的写入操作将ECOMn位清0；向PCA0CPHn写入时将ECOMn位置1。

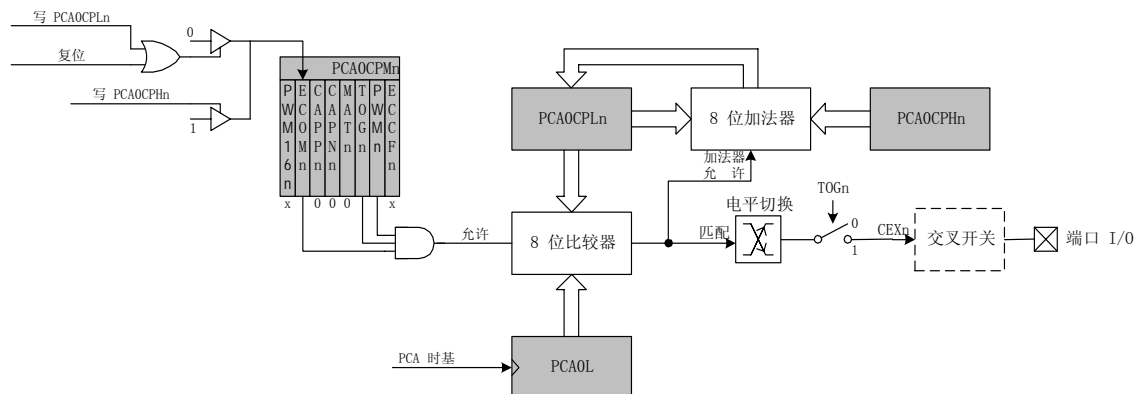


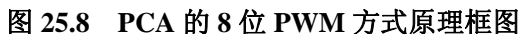
图 25.7 PCA 频率输出方式原理框图

每个模块都可以被独立地用于在对应的CEX_n引脚产生脉宽调制（PWM）输出。PWM输出的频率取决于PCA计数器/定时器的时基和PWM周期长度的设置（8、9、10或11位）。为了向后兼容其它器件的8位PWM方式，8位PWM方式与9、10和11位PWM方式稍有不同。注意：被设置为8、9、10或11位PWM方式的所有模块使用相同的周期长度。例如，不可能将一个通道配置为8位PWM方式，而将另一通道配置为11位方式。但其它PCA通道可以被独立地配置为引脚捕捉、高速输出、软件定时器、频率输出或16位PWM方式。

使用模块的捕捉/比较寄存器PCA0CPLn改变8位PWM方式PWM输出信号的占空比。当PCA计数器/定时器的低字节（PCA0L）与PCA0CPLn中的值相等时，CEXn引脚上的输出被置1；当PCA0L中的计数值溢出时，CEXn输出被复位（见图25.8）。当计数器/定时器的低字节PCA0L溢出时（从0xFF到0x00），保存在模块的捕捉/比较高字节（PCA0CPHn）中的值被自动装入到PCA0CPLn，不需软件干预。通过将PCA0CPMn寄存器中的ECOMn和PWMn位置1并将PCA0PWM寄存器中的CLSEL位设置为00b来使能8位脉冲宽度调制器方式。如果MATn被设置为1，则每当发生一次8位比较器匹配时（上升沿），模块的CCFn标志被置1。PCA0PWM中的COVF标志可用于检测溢出（下降沿），每256个PCA时钟周期发生一次溢出。8位PWM方式的占空比由方程25.2给出。

$$\text{占空比} = \frac{(256 - PCA0CPHn)}{256}$$

由方程25.2可知，最大占空比为100% (PCA0CPHn = 0)，最小占空比为0.39% (PCA0CPHn = 0xFF)。可以通过将ECOMn位清0来产生0%的占空比。



25.3.5.2 9/10/11 位脉宽调制器方式

通过写一个“自动重装载”寄存器（该寄存器被双映射到PCA0CPHn和PCA0CPLn地址）改变9/10/11位PWM方式PWM输出信号的占空比。定义占空比的写入数据在寄存器中应被右对齐。当PCA0PWM中的ARSEL位被设置为1时，访问自动重装载寄存器。当PCA0PWM中的ARSEL位被设置为0时，访问捕捉/比较寄存器。

当PCA0计数器的低N位与模块的捕捉/比较寄存器（PCA0CPn）中的值一致时，CEXn引脚上的输出被置为高电平。当计数从N位溢出时，CEXn输出被置为低电平（见图1）。当计数从N位溢出时，COVF标志被置位，保存在模块的自动重装载寄存器中的值被装入到捕捉/比较寄存器。值N由PCA0PWM寄存器中的CLSEL位确定。

通过将PCA0CPMn寄存器中的ECOMn和PWMn位置1并将PCA0PWM寄存器中的CLSEL位设置为所希望的周期长度（非8位）来使能9、10或11位脉冲宽度调制器方式。如果MATn被设置为1，则每当发生一次比较器匹配时（上升沿），模块的CCFn标志被置1。PCA0PWM中的COVF标志可用于检测溢出（下降沿），每512（9位）、1024（10位）或2048（11位）个PCA时钟周期发生一次溢出。9/10/11位PWM方式的占空比由方程25.3给出，其中N为PWM周期的位数。

注意：当向PCA0的捕捉/比较寄存器写入一个16位数值时，应先写低字节。向PCA0CPLn写入时将ECOMn位清0；向PCA0CPHn写入时将ECOMn位置1。

$$\text{占空比} = \frac{(2^N - \text{PCA0CPn})}{2^N}$$

方程25.3 9、10或11位PWM占空比

可以通过将ECOMn位清0来产生0%的占空比。

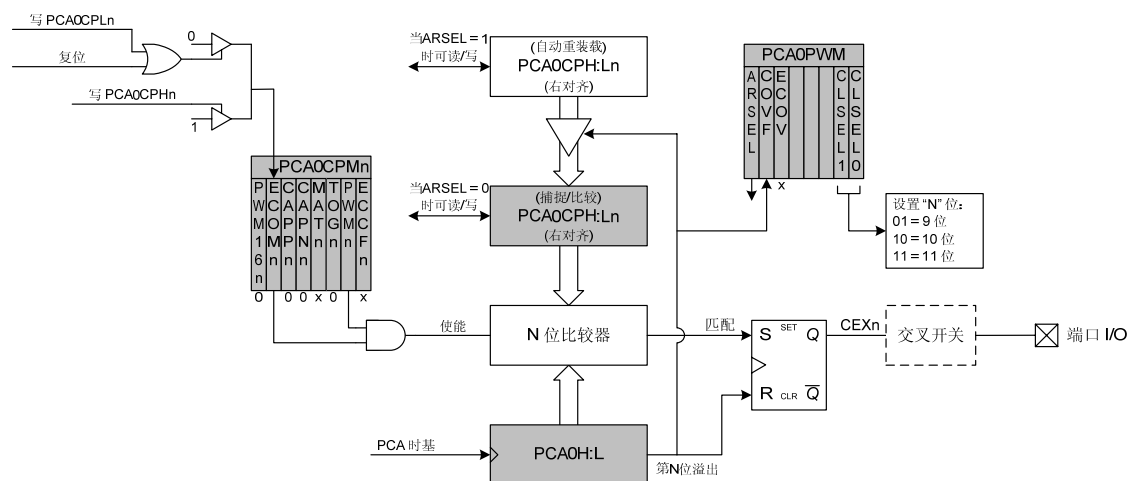


图1. PCA的9、10或11位PWM方式原理框图

PCA模块还可以工作在16位PWM方式。16位PWM方式与其它(8/9/10/11位)PWM方式无关。在该方式, 16位捕捉/比较模块定义PWM信号低电平时间的PCA时钟数。当PCA计数器与模块的内容一致时, CEXn的输出被置为高电平; 当16位计数器溢出时, CEXn输出被置为低电平。为了输出一个占空比可变的波形, 新值的写入应与PCA的CCFn匹配中断同步。通过将PCA0CPMn寄存器中的ECOMn、PWMn和PWM16n位置1来使能16位PWM方式。为了得到可变的占空比, 应允许匹配中断(ECCFn = 1并且MATn = 1), 以帮助对捕捉/比较寄存器的写操作同步。16位PWM方式的占空比由方程25.4给出。

方程25.4 16位PWM占空比

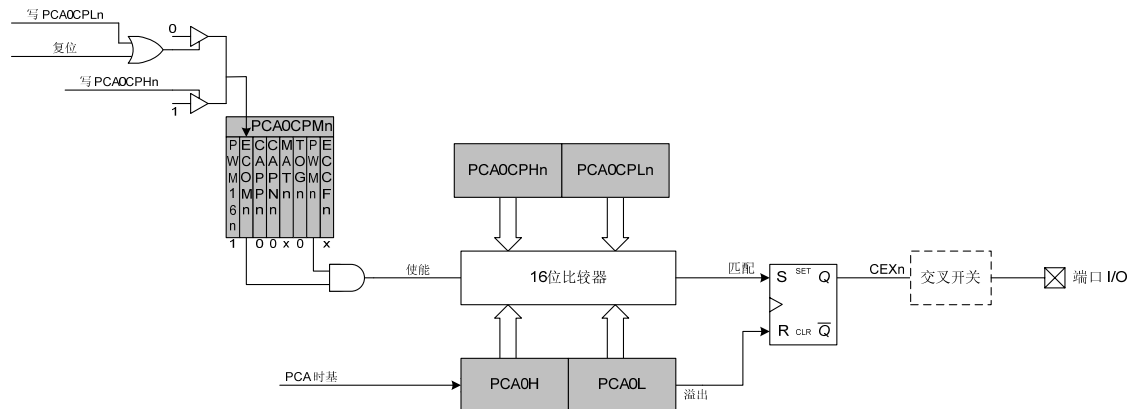


图 25.9 PCA 的 16 位 PWM 方式原理框图

25.4 看门狗定时器方式

通过 PCA 的模块 2 可以实现可编程看门狗定时器（WDT）功能。如果两次对 WDT 更新寄存器（PCA0CPH2）的写操作相隔的时间超过规定的极限，WDT 将产生一次复位。可以根据需要用软件配置和使能/禁止 WDT。

当 PCA0MD 寄存器中的 WDTE 位被置 1 时，模块 2 被作为看门狗定时器（WDT）使用。模块 2 高字节与 PCA 计数器的高字节比较；模块 2 低字节保持执行 WDT 更新时要使用的偏移值。在系统复位后看门狗定时器被使能。在看门狗被使能时，对某些 PCA 寄存器的写操作受到限制。

25.4.1 看门狗定时器操作

当 WDT 被使能时：

- PCA 计数器被强制运行。
- 不允许写 PCA0L 和 PCA0H。
- PCA 时钟源选择位（CPS2-CPS0）被冻结。
- PCA 空闲控制位（CIDL）被冻结。
- 模块 2 被强制进入软件定时器方式。
- 对模块 2 方式寄存器（PCA0CPM2）的写操作被禁止。

当 WDT 被使能时，写 CR 位并不改变 PCA 计数器的状态；计数器将一直保持运行状态，直到 WDT 被禁止。如果 WDT 被使能，但用户软件没有使能 PCA 计数器，则读 PCA 运行控制（CR）位时将返回 0。如果在 WDT 被使能时 PCA0CPH2 和 PCA0H 发生匹配，则会产生系统复位。为了防止 WDT 复位，需要通过向 PCA0CPH2 写入任意值来更新 WDT。在写 PCA0CPH2 时，PCA0H 的值加上 PCA0CPL2 中保存的偏移值后被装入到 PCA0CPH2（见图 25.10）。

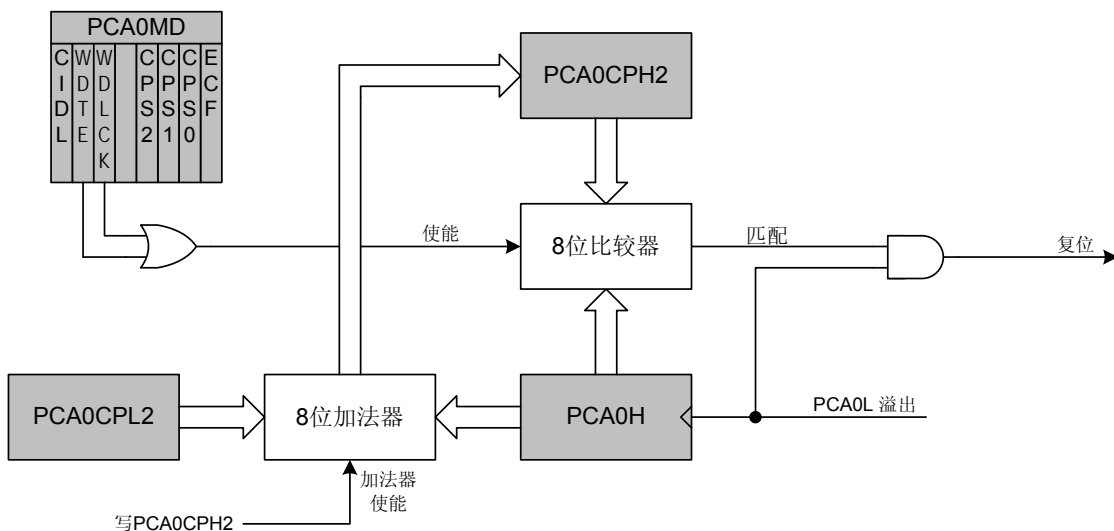


图 25.10 PCA 模块 2 的看门狗定时器方式

注意：保存在 PCA0CPH2 中的 8 位偏移值与 16 位 PCA 计数器的高字节进行比较，该偏

移值是复位前 PCA0L 的溢出次数。PCA0L 的第一次溢出周期取决于进行更新操作时 PCA0L 的值，最长可达 256 个 PCA 时钟。总偏移值（PCA 时钟数）由方程 25.5 给出，其中 PCA0L 是执行更新操作时 PCA0L 寄存器的值。

偏移值 = (256 × PCA0CPL2) + (256 − PCA0L)

方程25.5 看门狗定时器偏移值（PCA时钟数）

当 PCA0L 发生溢出并且 PCA0CPH2 和 PCA0H 匹配时，WDT 将产生一次复位。在 WDT 被使能的情况下，软件可以通过向 CCF2 标志（PCA0CN.2）写 1 来强制产生 WDT 复位。

25.4.2 看门狗定时器的使用

配置 WDT 的步骤如下：

- 通过向 WDTE 位写 0 来禁止 WDT。
- 选择 PCA 时钟源（用 CPS2-0 位）。
- 向 PCA0CPL2 装入所希望的 WDT 更新偏移值。
- 配置 PCA 的空闲方式位（如果希望在 CPU 处于空闲方式时 WDT 停止工作，则应将 CIDL 位置 1）。
- 通过向 WDTE 位写 1 来使能 WDT。
- 通过写 PCA0CPH2 来复位 WDT 定时器。

在 WDT 被使能时，不能改变 PCA 时钟源和空闲方式的设置值。通过向 PCA0MD 寄存器的 WDTE 或 WDLCK 位写 1 来使能 WDT。当 WDLCK 被置 1 时，在发生下一次系统复位之前将不能禁止 WDT。如果 WDCLK 未被置 1，清除 WDTE 位将禁止 WDT。

WDT 在任何一次系统复位之后都被使能。PCA0 计数器的缺省时钟为系统时钟的十二分频（SYSCLK/12）。PCA0L 和 PCA0CPL2 的缺省值均为 0x00。根据方程 25.5，这些值将产生 256 个 PCA 时钟周期或 3072 个系统时钟周期的 WDT 超时间隔。表 25.3 列出了对应某些典型系统时钟频率的超时间隔示例。

表 25.3 看门狗定时器超时间隔¹

系统时钟（Hz）	PCA0CPL2	超时间隔（ms）
24,500,000	255	32.1
24,500,000	128	16.2
24,500,000	32	4.1
3,062,500 ²	255	257
3,062,500 ²	128	129.5
3,062,500 ²	32	33.1
32,000	255	24576
32,000	128	12384
32,000	32	3168
注：		
1. 假设 PCA 使用 SYSCLK/12 作为时钟源，更新时 PCA0L 的值为 0x00。		
2. 内部 SYSCLK 复位频率 = 内部振荡器频率/8。		

25.5 PCA0 寄存器说明

下面对与 PCA 工作有关的特殊功能寄存器进行详细说明。

SFR 定义 25.1 PCA0CN: PCA 控制寄存器

R/W	R/W	R	R	R	R/W	R/W	R/W	复位值
CF	CR	-	-	-	CCF2	CCF1	CCF0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: (可位寻址) 0xD8
<p>位 7: CF: PCA 计数器/定时器溢出标志 当 PCA 计数器/定时器从 0xFFFF 到 0x0000 溢出时由硬件置位。在计数器/定时器溢出 (CF) 中断被允许时, 该位置 1 将导致 CPU 转向 PCA 中断服务程序。该位不能由硬件自动清 0, 必须用软件清 0。</p>								
<p>位 6: CR: PCA 计数器/定时器运行控制 该位允许/禁止 PCA 计数器/定时器。 0: 禁止 PCA 计数器/定时器。 1: 允许 PCA 计数器/定时器。</p>								
<p>位 5-3: 未用。读 = 000b, 写 = 忽略。</p>								
<p>位 2: CCF2: PCA 模块 2 捕捉/比较标志 在发生一次匹配或捕捉时该位由硬件置位。当 CCF2 中断被允许时, 该位置 1 将导致 CPU 转向 PCA 中断服务程序。该位不能由硬件自动清 0, 必须用软件清 0。</p>								
<p>位 1: CCF1: PCA 模块 1 捕捉/比较标志 在发生一次匹配或捕捉时该位由硬件置位。当 CCF1 中断被允许时, 该位置 1 将导致 CPU 转向 PCA 中断服务程序。该位不能由硬件自动清 0, 必须用软件清 0。</p>								
<p>位 0: CCF0: PCA 模块 0 捕捉/比较标志 在发生一次匹配或捕捉时该位由硬件置位。当 CCF0 中断被允许时, 该位置 1 将导致 CPU 转向 PCA 中断服务程序。该位不能由硬件自动清 0, 必须用软件清 0。</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 25.2 PCA0MD: PCA 方式寄存器

R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	复位值
CIDL	WDTE	WDLCK	-	CPS2	CPS1	CPS0	ECF	01000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xD9

位 7:

CIDL: PCA 计数器/定时器空闲控制
设置 CPU 空闲方式下的 PCA 工作方式。
0: 当系统控制器处于空闲方式时, PCA 继续正常工作。
1: 当系统控制器处于空闲方式时, PCA 停止工作。

位 6:

WDTE: 看门狗定时器使能位
如果该位被置 1, PCA 模块 2 被用作看门狗定时器。
0: 看门狗定时器被禁止。
1: PCA 模块 2 被用作看门狗定时器。

位 5:

WDLCK: 看门狗定时器锁定
该位对看门狗定时器使能位锁定/解锁。当 WDLCK 被置 1 时, 在发生下一次系统复位之前将不能禁止 WDT。
0: 看门狗定时器使能位未被锁定。
1: 锁定看门狗定时器使能位。

位 4:

未用。读 = 0b, 写 = 忽略。

位 3-1:

CPS2-CPS0: PCA 计数器/定时器时钟选择
这些位选择 PCA 计数器的时钟源。

CPS2	CPS1	CPS0	时钟源
0	0	0	系统时钟的 12 分频
0	0	1	系统时钟的 4 分频
0	1	0	定时器 0 溢出
0	1	1	ECI 负跳变 (最大速率 = 系统时钟频率/4)
1	0	0	系统时钟
1	0	1	外部时钟的 8 分频
1	1	0	保留
1	1	1	保留

位 0:

ECF: PCA 计数器/定时器溢出中断允许
该位是 PCA 计数器/定时器溢出 (CF) 中断的屏蔽位。
0: 禁止 CF 中断。
1: 当 CF (PCA0CN.7) 被置位时, 允许 PCA 计数器/定时器溢出的中断请求。

注:

当 WDTE 位被置 1 时, 不能改变 PCA0MD 寄存器的其它位。若要改变 PCA0MD 的内容, 必须先禁止看门狗定时器。

SFR 定义 25.3 PCA0PWM: PCA PWM 配置寄存器

R/W	R/W	R/W	R	R	R	R/W	R/W	复位值
ARSEL	ECOV	COVF	-	-	-	CLSEL1	CLSEL0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xF7
<p>位 7: ARSEL: 自动重装载寄存器选择</p> <p>该位选择是读和写正常的 PCA 捕捉/比较寄存器(PCA0CPn)还是位于相同 SFR 地址的自动重装载寄存器。该功能用于定义 9、10 或 11 位 PWM 方式的重载值。在所有其它方式, 自动重装载寄存器没有功能。</p> <p>0: 读/写位于 PCA0CPHn 和 PCA0CPLn 的捕捉/比较寄存器。</p> <p>1: 读/写位于 PCA0CPHn 和 PCA0CPLn 的自动重装载寄存器寄存器。</p>								
<p>位 6: ECOV: 周期溢出中断允许</p> <p>该位设置周期溢出标志 (COVF) 中断的掩码。</p> <p>0: COVF 将不产生 PCA 中断。</p> <p>1: 当 COVF 置位时将一个产生 PCA 中断。</p>								
<p>位 5: COVF: 周期溢出标志</p> <p>该位指示主 PCA 计数器的第 8、9、10 或 11 位溢出。该标志具体使用哪一位取决于周期长度选择位的设置。该位可由硬件或软件置位, 但必须用软件清 0。</p> <p>0: 自最后一次该位被清 0 以来没有发生溢出。</p> <p>1: 自最后一次该位被清 0 以来发生了溢出。</p>								
<p>位 4-2: 未用。读 = 000b, 写 = 忽略。</p>								
<p>位 1-0: CLSEL[1:0]: 周期长度选择位</p> <p>当未选择 16 位 PWM 方式时, 这些位选择 8、9、10 或 11 位的 PWM 周期长度。这种选择影响不使用 16 位 PWM 方式的所有被配置为 PWM 方式的通道。这些位对配置为 16 位 PWM 方式的通道而言被忽略。</p> <p>00: 8 位。</p> <p>01: 9 位。</p> <p>10: 10 位。</p> <p>11: 11 位。</p>								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 25.4 PCA0CPMn: PCA 捕捉/比较寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
PWM16n	ECOMn	CAPPn	CAPNn	MATn	TOGn	PWMn	ECCFn	00000000
位7	位6	位5	位4	位3	位2	位1	位0	

PCA0CPMn 地址： PCA0CPM0=0xDA PCA0CPM1=0xDB
PCA0CPM2=0xDC

位 7： PWM16n: 16 位脉冲宽度调制使能
当脉冲宽度调制方式被使能时（PWMn = 1），该位选择 16 位方式。
0：选择 8~11 位 PWM。
1：选择 16 位 PWM。

位 6： ECOMn: 比较器功能使能
该位置 1 时使能 PCA 模块 n 的比较器功能。

位 5： CAPPn: 正沿捕捉功能使能
该位置 1 时使能 PCA 模块 n 的正边沿捕捉。

位 4： CAPNn: 负沿捕捉功能使能
该位置 1 时使能 PCA 模块 n 的负边沿捕捉。

位 3： MATn: 匹配功能使能
该位置 1 时使能 PCA 模块 n 的匹配功能。如果被使能，当 PCA 计数器与一个模块的捕捉/比较寄存器匹配时，PCA0MD 寄存器中的 CCFn 位被置 1。

位 2： TOGn: 电平切换功能使能
该位置 1 时使能 PCA 模块 n 的电平切换功能。如果被使能，当 PCA 计数器与一个模块的捕捉/比较寄存器匹配时，CEXn 引脚的逻辑电平发生切换。如果 PWMn 位也被置 1，模块将工作在频率输出方式。

位 1： PWMn: 脉宽调制方式使能
该位置 1 时使能 PCA 模块 n 的 PWM 功能。当被使能时，CEXn 引脚输出脉冲宽度调制信号。PWM16n 为 0 时使用 8~11 位 PWM 方式；PWM16n 为 1 时使用 16 位方式。如果 TOGn 位也被置为逻辑 1，则模块工作在频率输出方式。

位 0： ECCFn: 捕捉/比较标志中断允许
该位设置捕捉/比较标志（CCFn）的中断屏蔽。
0：禁止 CCFn 中断。
1：允许 CCFn 位被置 1 时捕捉/比较标志的中断请求。

注: 当 WDTE 位被置 1 时, 不能修改 PCA0CPM2 寄存器, 模块 2 作为看门狗定时器。若要改变 PCA0CPM2 寄存器的内容或模块 2 的功能, 必须先禁止看门狗定时器。

SFR 定义 25.5 PCA0L: PCA 计数器 / 定时器低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xF9
位 7-0: PCA0[7:0]: PCA 计数器/定时器的低字节 PCA0L 寄存器保存 16 位 PCA 计数器/定时器的低字节 (LSB)。								
注: 当 WDTE 位被置 1 时, 不能用软件修改 PCA0L 寄存器。若要改变 PCA0L 寄存器的内容, 必须先禁止看门狗定时器。								

SFR 定义 25.6 PCA0H: PCA 计数器 / 定时器高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xFA
位 7-0: PCA0[15:8]: PCA 计数器/定时器高字节 PCA0H 寄存器保存 16 位 PCA 计数器/定时器的高字节 (MSB)。读该寄存器时读的是“瞬像”寄存器的内容, 是在读 PCA0L 的内容时被锁存的 (见 25.1 节)。								
注: 当 WDTE 位被置 1 时, 不能用软件修改 PCA0H 寄存器。若要改变 PCA0H 寄存器的内容, 必须先禁止看门狗定时器。								

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

SFR 定义 25.7 PCA0CPLn: PCA 捕捉模块低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	
PCA0CPLn 地址: PCA0CPL0 = 0xFB (n=0) PCA0CPL1 = 0xE9 (n=1) PCA0CPL2 = 0xEB (n=2)								
位 7-0: PCA0CPn[7:0]: PCA 捕捉模块低字节 PCA0CPLn 寄存器保存 16 位捕捉模块 n 的低字节 (LSB)。该寄存器还允许访问对应的 PCA 通道之 9、10 或 11 位 PWM 方式自动重载值的低字节。寄存器 PCA0PWM 中的 ARSEL 位控制访问哪个寄存器。								
注: 写该寄存器时将模块的 ECOMn 位清 0。								

SFR 定义 25.8 PCA0CPHn: PCA 捕捉模块高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	
PCA0CPHn 地址: PCA0CPH0=0xFC PCA0CPH1=0xEA PCA0CPH2=0xEC								
位 7-0: PCA0CPn[15:8]: PCA 捕捉模块高字节 PCA0CPHn 寄存器保存 16 位捕捉模块 n 的高字节 (MSB)。该寄存器还允许访问对应的 PCA 通道之 9、10 或 11 位 PWM 方式自动重载值的低字节。寄存器 PCA0PWM 中的 ARSEL 位控制访问哪个寄存器。								
注: 写该寄存器时将模块的 ECOMn 位置 1。								

26. C2 接口

C8051F336/7/8/9 器件内部有一个 Silicon Labs 2 线（C2）调试接口，支持 FLASH 编程和使用安装在最终应用系统中的器件进行在系统调试。C2 接口使用一个时钟信号（C2CK）和一个双向的 C2 数据信号（C2D）在器件和宿主机之间传送信息。有关 C2 协议的详细信息见 C2 接口规范。

26.1 C2 接口寄存器

下面对与 FLASH 编程（通过 C2 接口）有关的 C2 寄存器进行说明。对所有 C2 寄存器的访问都要通过 C2 接口实现（详见 C2 接口规范）。

C2 寄存器定义 26.1 C2ADD：C2 地址寄存器

复位值
00000000

位7	位6	位5	位4	位3	位2	位1	位0

位 7-0： C2ADD[7:0]： C2 地址寄存器
C2ADD 寄存器选择 C2 数据读和数据写命令的目标数据寄存器。

地址	说明
0x00	选择器件 ID 寄存器（数据读指令）。
0x01	选择版本 ID 寄存器（数据读指令）。
0x02	选择 C2 FLASH 编程控制寄存器（数据读/写指令）
0xB4	选择 C2 FLASH 编程数据寄存器（数据读/写指令）

C2 SFR 定义 26.2 DEVICEID：C2 器件 ID 寄存器

复位值
00010100

位7	位6	位5	位4	位3	位2	位1	位0

位 7-0： DEVICEID[7:0]： 器件 ID 寄存器
该只读寄存器返回 8 位的器件 ID 号： 0x14（C8051F336/7/8/9）。

C2地址：
0x00

C2 SFR 定义 26.3 REVID：C2 版本 ID 寄存器

复位值
00000000

位7	位6	位5	位4	位3	位2	位1	位0

位 7-0： REVID[7:0]： 版本 ID 寄存器
该只读寄存器返回 8 位的版本 ID 号： 如 0x00（版本 A）。

C2地址：
0x01

C8051F336/7/8/9 混合信号 ISP FLASH 微控制器

C2 SFR 定义 26.4 FPCTL: C2 FLASH 编程控制寄存器

位7	位6	位5	位4	位3	位2	位1	位0

复位值
00000000
C2地址:
0x02

位 7-0: FPCTL[7:0]: C2 FLASH 编程控制寄存器。

该寄存器用于使能通过 C2 接口对 FLASH 编程。为了使能 C2 FLASH 编程，必须按顺序写代码：0x02、0x01。注意：一旦 C2 FLASH 编程被使能，必须进行一次复位才能使系统恢复正常工作。

C2 SFR 定义 26.5 FPDAT: C2 FLASH 编程数据寄存器

位7	位6	位5	位4	位3	位2	位1	位0

复位值
00000000
C2地址:
0xB4

位 7-0: FPDAT[7:0]: FLASH 编程数据寄存器。

该寄存器用于在 C2 FLASH 访问期间传递 FLASH 编程命令、地址和数据。下面列出了有效的编程命令。

代码	命令
0x06	读 FLASH 块
0x07	写 FLASH 块
0x08	擦除 FLASH 页
0x03	擦除器件

C2 协议允许 C2 引脚与用户功能共享，可以进行在系统调试和 FLASH 编程。这种共享之所以可能，是因为 C2 通信通常发生在器件的停止运行状态。在这种状态下所有片内外设和用户软件都被冻结，C2 接口可以安全地“借用”C2CK (/RST) 和 C2D 引脚。在大多数应用中，需要使用外部电阻对 C2 接口和用户应用进行隔离。典型的隔离电路如图 26.1 所示。



1. 在目标器件的停止运行状态，用户输入（b）不能改变状态。
2. 目标器件的/RST 引脚只能被作为输入使用。

新华龙电子有限公司 电话: 0755-83645240 83645242 传真: 0755-83645243 209

联系信息

Silicon Laboratories Inc.

4635 Boston Lane
Austin, TX 78735
Tel: 1+(512) 416-8500
Fax: 1+(512) 416-9669
Toll Free: 1+(877) 444-3032
Email: mcuinfo@silabs.com
Internet: www.silabs.com

新华龙电子有限公司

电话: 0755-83645240 83645242 83645244 83645251
技术支持: 0755-83645259
传真: 0755-83645243
地址: 深圳市福田区华强北路现代之窗大厦 A 座 13F C 室(518013)
Email: sales@xhl.com.cn
Email: shenzhen@xhl.com.cn
网站: www.xhl.com.cn
技术支持: support-sz@xhl.com.cn