

C8051F52x/52xA/53x/53xA
混合信号 ISP FLASH 微控制器
数 据 手 册

潘 琢 金 译

Rev 0.5 2008.04

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

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

模拟外设

- 12 位 ADC
 - $\pm 1\text{LSB}$ INL; 无失码
 - 可编程转换速率, 最高 200ksps
 - 最多 6/16 个外部输入
 - 数据窗口中断发生器
 - 内建温度传感器
- 比较器
 - 可编程回差电压和响应时间
 - 可配置为唤醒或复位源
 - 低电流
- 上电复位/欠压检测器
- 电压基准 — 1.5V 和 2.2V (可编程)

在片调试

- 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
- 完备的开发套件

供电电压 1.8V ~ 5.25V

- 内建 LDO 稳压器: 2.1 或 2.5 V

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

存储器

- 8/4/2KB FLASH; 可在系统编程, 扇区大小为 512 字节
- 256 字节内部数据 RAM

数字外设

- 16/6 个端口 I/O; 推挽或漏极开路, 耐 5 V 电压
- 硬件 SPI 和 UART 串口
- 硬件 LIN (支持主和从方式, 兼容 V1.3 和 V2.0)
- 3 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 WDT

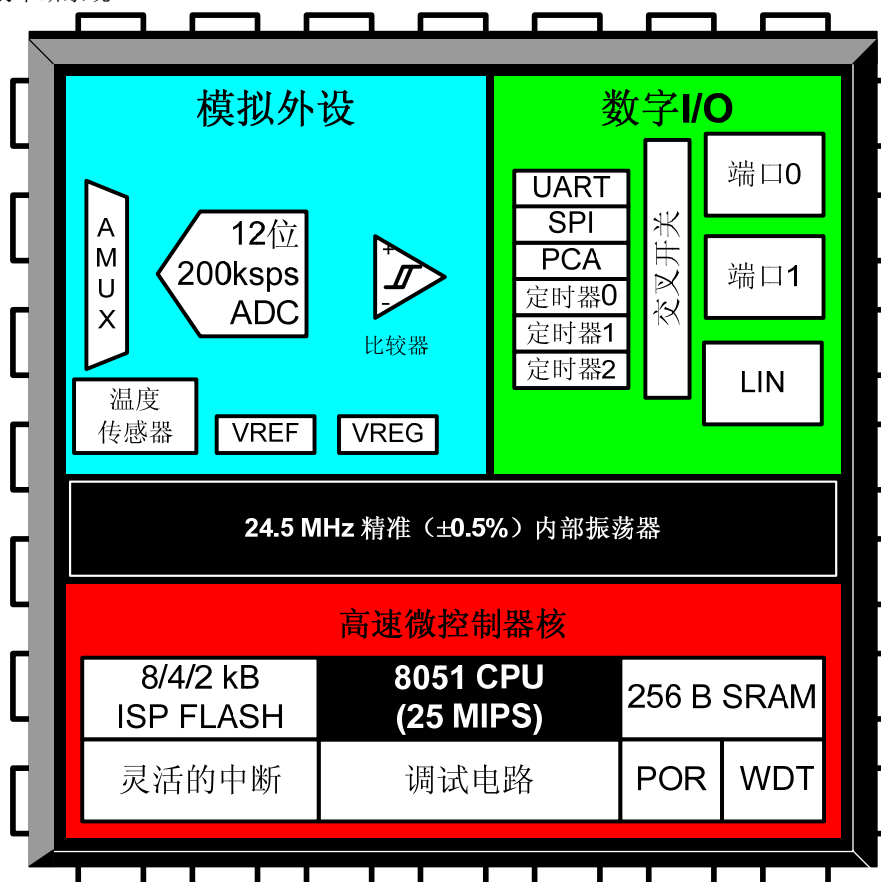
时钟源

- 内部振荡器: 24.5MHz $\pm 0.5\%$ 精度, 可支持 UART 操作和 LIN 主操作
- 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

封装:

- 10 脚 QFN (3x3mm)
- 20 脚 QFN (4x4mm)
- 20 脚 TSSOP

温度范围: -40°C - +125°C



C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

目 录

1. 系统概述	7
1.1 CIP-51™ 微控制器核	11
1.1.1 与 8051 指令集完全兼容	11
1.1.2 速度提升	11
1.1.3 增加的功能	11
1.1.4 片内调试电路	12
1.2 片内存储器	13
1.3 工作方式	14
1.4 12 位模/数转换器	15
1.5 可编程比较器	16
1.6 稳压器	16
1.7 串行端口	16
1.8 端口输入/输出	16
2. 极限参数	18
3. 总体直流电气特性	20
4. 引脚和封装定义	21
5. 12 位ADC (ADC0)	34
5.1 模拟多路选择器	35
5.2 温度传感器	35
5.3 工作方式	36
5.3.1 启动转换	36
5.3.2 跟踪方式	36
5.3.3 时序	37
5.3.4 突发方式	39
5.3.5 输出转换码	40
5.3.6 建立时间要求	41
5.4 可选择的增益	41
5.4.1 计算增益值	42
5.4.2 设置增益值	43
5.5 可编程窗口检测器	50
5.5.1 单端方式下的窗口检测器	51
6. 电压基准	53
7. 稳压器 (REG0)	56

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

8. 比较器	58
9. CIP-51 微控制器	64
9.1 指令集	65
9.1.1 指令和CPU时序	65
9.1.2 MOVX指令和程序存储器	65
9.2 寄存器说明	69
9.3 电源管理方式	72
9.3.1 空闲方式	72
9.3.2 停机方式	72
9.3.3 挂起方式	72
10. 存储器组织和SFR	74
10.1 程序存储器	74
10.2 数据存储器	75
10.3 通用寄存器	75
10.4 位寻址空间	75
10.5 堆栈	75
10.6 特殊功能寄存器	76
11. 中断系统	80
11.1 MCU中断源和中断向量	80
11.2 中断优先级	80
11.3 中断响应时间	80
11.4 中断寄存器说明	82
11.5 外部中断	86
12. 复位源	88
12.1 上电复位	89
12.2 掉电复位和VDD监视器	90
12.3 外部复位	91
12.4 时钟丢失检测器复位	91
12.5 比较器复位	91
12.6 PCA看门狗定时器复位	92
12.7 FLASH错误复位	92
12.8 软件复位	92
13. FLASH存储器	95
13.1 FLASH存储器编程	95
13.1.1 FLASH锁定和关键码功能	95
13.1.2 FLASH擦除过程	96
13.1.3 FLASH写过程	96

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

13.2	FLASH写和擦除指南	97
13.2.1	VDD维护和VDD监视器	97
13.2.2	PSWE维护	98
13.2.3	系统时钟	98
13.3	非易失性数据存储	98
13.4	安全选项	99
14.	端口输入/输出	104
14.1	优先权交叉开关译码器	105
14.2	端口I/O初始化	110
14.3	通用端口I/O	113
15.	振荡器	119
15.1	可编程内部振荡器	120
15.1.1	内部振荡器挂起方式	120
15.2	外部振荡器驱动电路	122
15.2.1	外部振荡器直接用作定时器时钟	122
15.2.2	外部晶体示例	123
15.2.3	外部RC示例	124
15.2.4	外部电容示例	124
15.3	系统时钟选择	126
16.	UART0	128
16.1	增强的波特率发生器	129
16.2	工作方式	130
16.2.1	8位UART	130
16.2.2	9位UART	131
16.3	多机通信	132
17.	LIN (C8051F520/0A/3/3A/526/6A和C8051F530/3A/6/6A)	135
17.1	与LIN外设的软件接口	135
17.2	LIN接口设置和操作	136
17.2.1	方式定义	136
17.2.2	波特率选项: 手动或自动	136
17.2.3	波特率计算: 手动方式	136
17.2.4	波特率计算: 自动方式	138
17.3	LIN主方式操作	139
17.4	LIN从方式操作	140
17.5	休眠方式和唤醒	141
17.6	错误检测和处理	141
17.7	LIN寄存器	142
17.7.1	LIN直接访问SFR寄存器定义	142

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

17.3.2 LIN间接访问SFR寄存器定义.....	143
18. 增强型串行外设接口 (SPI0)	151
18.1 信号说明.....	152
18.1.1 主输出、从输入 (MOSI)	152
18.1.2 主输入、从输出 (MISO)	152
18.1.3 串行时钟 (SCK)	152
18.1.4 从选择 (NSS)	152
18.2 SPI0 主方式	153
18.3 SPI0 从方式	155
18.4 SPI0 中断源	155
18.5 串行时钟时序.....	156
18.6 SPI特殊功能寄存器.....	157
19. 定时器	163
19.1 定时器 0 和定时器 1.....	163
19.1.1 方式 0—13 位计数器/定时器.....	163
19.1.2 方式 1—16 位计数器/定时器.....	165
19.1.3 方式 2—自动重载的 8 位计数器/定时器.....	165
19.1.4 方式 3—两个 8 位计数器/定时器 (仅定时器 0)	166
19.2 定时器 2.....	171
19.2.1 16 位自动重载定时器.....	171
19.2.2 8 位自动重载定时器.....	172
19.2.3 外部捕捉方式.....	173
20. 可编程计数器阵列 (PCA0)	176
20.1 PCA计数器/定时器	177
20.2 捕捉/比较模块.....	178
20.2.1 边沿触发的捕捉方式.....	179
20.2.2 软件定时器 (比较) 方式.....	180
20.2.3 高速输出方式.....	181
20.2.4 频率输出方式.....	182
20.2.5 8 位脉宽调制器方式.....	183
20.2.6 16 位脉宽调制器方式.....	184
20.3 看门狗定时器方式	185
20.3.1 看门狗定时器操作.....	185
20.3.2 看门狗定时器的使用.....	186
20.4 PCA寄存器说明.....	188
21. 器件行为差别	192
21.1 器件标识.....	192
21.2 复位引脚行为.....	193

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

21.3 复位时延.....	193
21.4 UART引脚.....	193
21.5 LIN.....	193
21.5.1 停止位检查.....	193
21.5.2 SYNCH BREAK和SYNCH FIELD长度检查.....	194
22. C2 接口.....	195
22.1 C2 接口寄存器.....	195
22.2 C2 引脚共享.....	197

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

1. 系统概述

C8051F52x/52xA/53x/53xA 器件是完全集成的极低功耗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 1.1。

- 高速、流水线结构的与 8051 兼容的微控制器核（可达 25MIPS）
- 全速、非侵入式的在系统调试接口（片内）
- 真 12 位、200 ksps ADC，最多 16 路模拟输入，带模拟多路器
- 高精度可编程的 24.5MHz 内部振荡器，在整个电压和温度范围内精度为 $\pm 0.5\%$ 。
- 达 7680 字节的片内 FLASH 存储器
- 256 字节片内 RAM
- 硬件实现的增强型 UART 和 SPI 串行接口
- LIN2.0 外设（兼容 V2.0 和 V1.3，主方式和从方式）
- 3 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
- 多达 16 个端口 I/O

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 C8051F52x/52xA/53x/53xA 器件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，使在系统编程和调试功能不占用封装引脚。

每种器件都可在汽车级温度范围（-40℃到+125℃）内用 1.8V ~ 5.25V 的电压工作（使用片内稳压器时电源电压可达 5.25V）。C8051F52x/52xA 封装为 DFN10（3x3 mm），C8051F53x/53xA 的封装有 QFN20（4x4 mm）和 TSSOP20。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 1.1 产品选择指南

器件型号	MIPS (峰值)	FLASH 存储器 (KB)	RAM (字节)	校准的内部 24.5MHz 振荡器	SPI	UART	定时器(16位)	可编程计数器阵列	端口 I/O	12 位 ADC	LIN	内部电压基准	温度传感器	模拟比较器	建议新设计采用	封装
C8051F520A-IM	25	8	256	0.5%	✓	✓	3	✓	6	✓	✓	✓	✓	✓	✓	DFN-10
C8051F521A-IM	25	8	256	0.5%	✓	✓	3	✓	6	✓	—	✓	✓	✓	✓	DFN-10
C8051F523A-IM	25	4	256	0.5%	✓	✓	3	✓	6	✓	✓	✓	✓	✓	✓	DFN-10
C8051F524A-IM	25	4	256	0.5%	✓	✓	3	✓	6	✓	—	✓	✓	✓	✓	DFN-10
C8051F526A-IM	25	2	256	0.5%	✓	✓	3	✓	6	✓	✓	✓	✓	✓	✓	DFN-10
C8051F527A-IM	25	2	256	0.5%	✓	✓	3	✓	6	✓	—	✓	✓	✓	✓	DFN-10
C8051F530A-IM	25	8	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	✓	QFN-20
C8051F531A-IM	25	8	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	✓	QFN-20
C8051F533A-IM	25	4	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	✓	QFN-20
C8051F534A-IM	25	4	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	✓	QFN-20
C8051F536A-IM	25	2	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	✓	QFN-20
C8051F537A-IM	25	2	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	✓	QFN-20
C8051F530A-IT	25	8	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	✓	TSSOP-20
C8051F531A-IT	25	8	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	✓	TSSOP-20
C8051F533A-IT	25	4	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	✓	TSSOP-20
C8051F534A-IT	25	4	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	✓	TSSOP-20
C8051F536A-IT	25	2	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	✓	TSSOP-20
C8051F537A-IT	25	2	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	✓	TSSOP-20
C8051F520-IM	25	8	256	0.5%	✓	✓	3	✓	6	✓	✓	✓	✓	✓	—	DFN-10
C8051F521-IM	25	8	256	0.5%	✓	✓	3	✓	6	✓	—	✓	✓	✓	—	DFN-10
C8051F523-IM	25	4	256	0.5%	✓	✓	3	✓	6	✓	✓	✓	✓	✓	—	DFN-10
C8051F524-IM	25	4	256	0.5%	✓	✓	3	✓	6	✓	—	✓	✓	✓	—	DFN-10
C8051F526-IM	25	2	256	0.5%	✓	✓	3	✓	6	✓	✓	✓	✓	✓	—	DFN-10
C8051F527-IM	25	2	256	0.5%	✓	✓	3	✓	6	✓	—	✓	✓	✓	—	DFN-10
C8051F530-IM	25	8	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	—	QFN-20
C8051F531-IM	25	8	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	—	QFN-20
C8051F533-IM	25	4	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	—	QFN-20
C8051F534-IM	25	4	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	—	QFN-20
C8051F536-IM	25	2	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	—	QFN-20
C8051F537-IM	25	2	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	—	QFN-20
C8051F530-IT	25	8	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	—	TSSOP-20
C8051F531-IT	25	8	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	—	TSSOP-20
C8051F533-IT	25	4	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	—	TSSOP-20
C8051F534-IT	25	4	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	—	TSSOP-20
C8051F536-IT	25	2	256	0.5%	✓	✓	3	✓	16	✓	✓	✓	✓	✓	—	TSSOP-20
C8051F537-IT	25	2	256	0.5%	✓	✓	3	✓	16	✓	—	✓	✓	✓	—	TSSOP-20

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

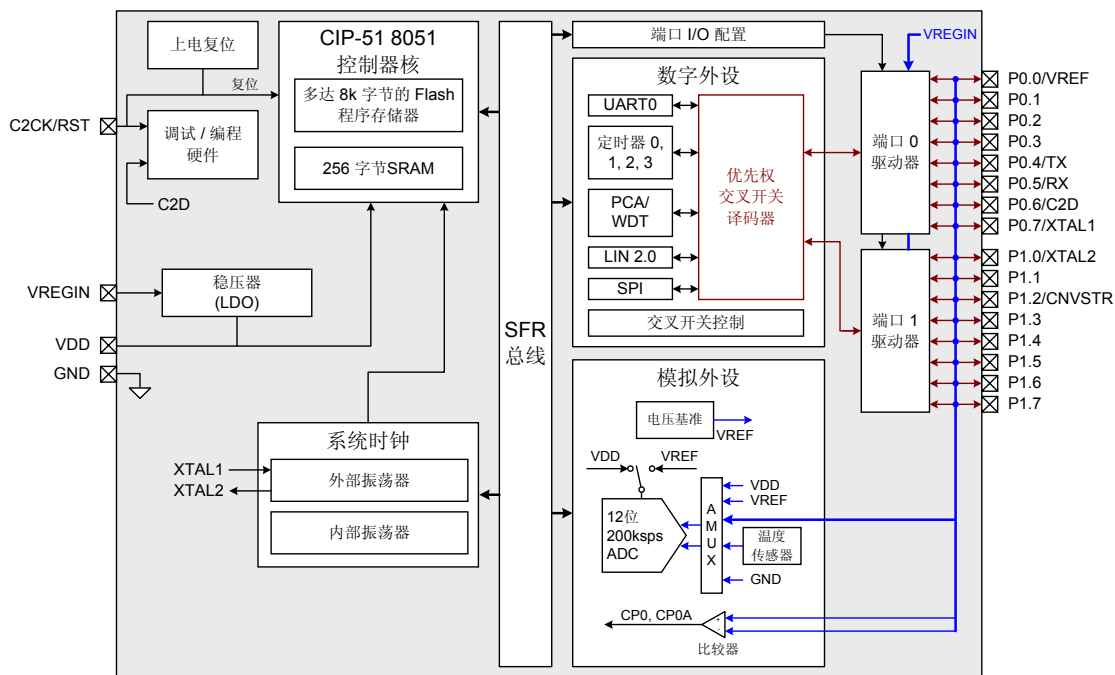


图 1.1 C8051F53xA 原理框图

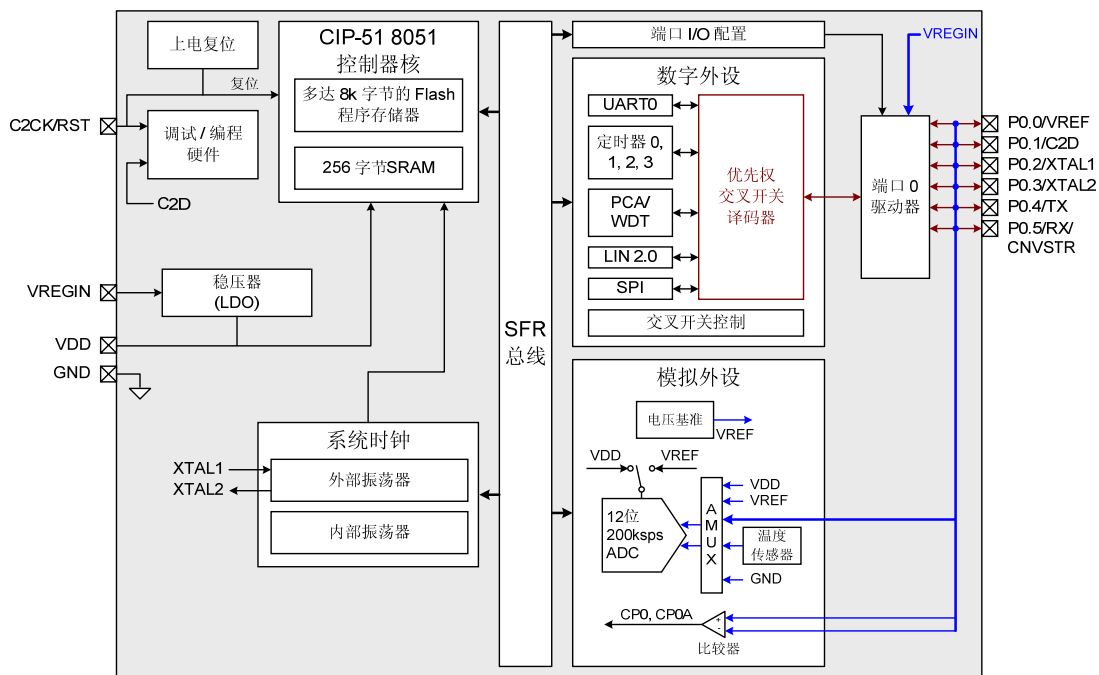


图 1.2 C8051F52xA 原理框图

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

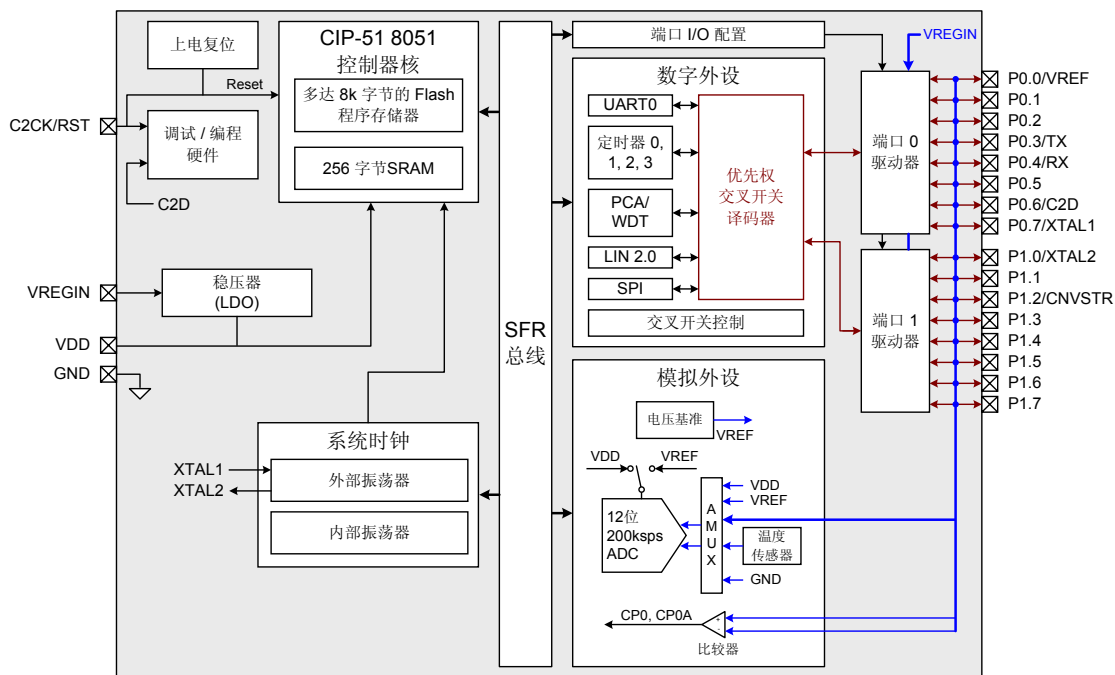


图 1.3 C8051F53x 原理框图

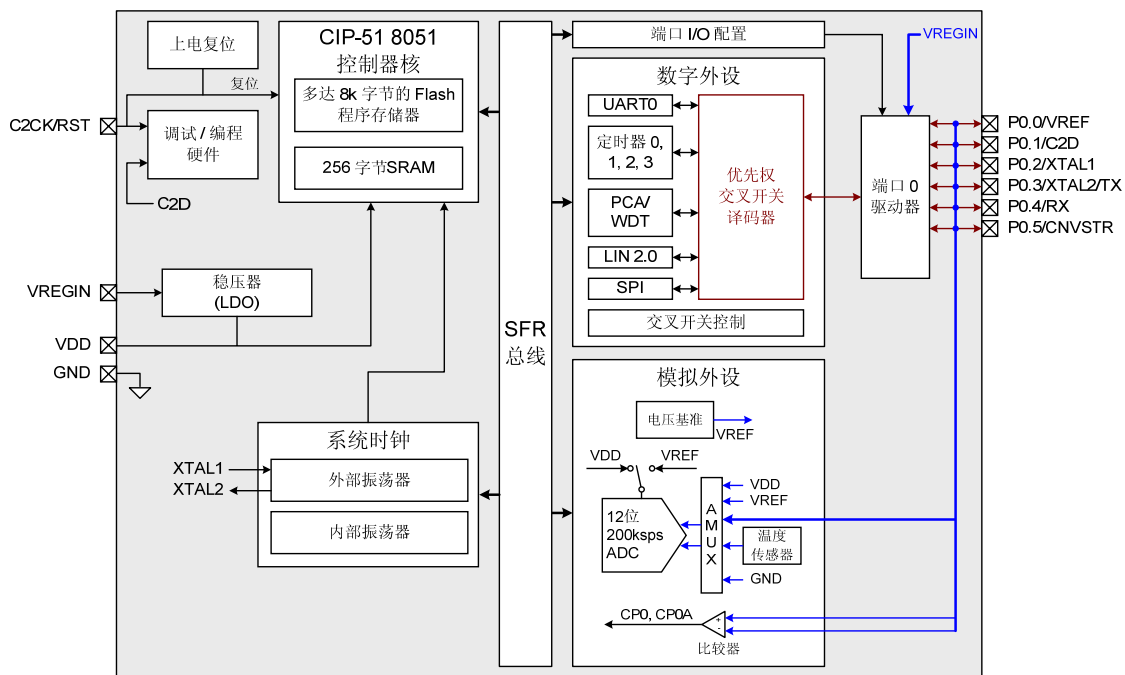


图 1.4 C8051F52x 原理框图

1.1 CIP-51™ 微控制器核

1.1.1 与 8051 指令集完全兼容

C8051F52x/52xA/53x/53xA系列器件使用Silicon Labs的专利CIP-51 微控制器核。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。C8051F52x/52xA/53x/53xA系列器件的外设是标准 8052 的所有外设的超集。

1.1.2 速度提升

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 核，70%的指令的执行时间为 1 或 2 个系统时钟周期，没有执行时间大于 8 个系统时钟周期的指令。

CIP-51 工作在 25MHz 的时钟频率时，它的峰值速度达到 25MIPS。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

1.1.3 增加的功能

C8051F52x/52xA/53x/53xA 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。

扩展的中断系统允许大量的模拟和数字外设独立于微控制器工作，只在必要时中断微控制器。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，并使多任务实时系统的实现更加容易。

MCU 有 8 个复位源：上电复位电路（POR）、片内 VDD 监视器、看门狗定时器、时钟丢失检测器、由比较器 0 提供的电压检测器、软件强制复位、外部复位引脚复位和 FLASH 非法访问保护电路复位。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 可以通过软件被永久性使能。

内部振荡器在出厂时已经被校准为 24.5MHz，在整个工作电压范围和温度范围精度为±0.5%。器件内还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。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

1.1.4 片内调试电路

C8051F52x/52xA/53x/53xA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除外）都停止运行，以保持与指令执行同步。

开发套件C8051F530ADK具有开发应用代码和对C8051F52x/52xA/53x/53xAMCU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USB调试适配器、一块安装了相应MCU的目标应用板和所必需的电缆及墙装电源。开发套件需要一个安装了Windows的计算机。如图1.5所示，PC机连接到USB调试适配器，六英吋的扁平电缆将USB调试适配器连接到用户的应用板（使用两个C2引脚和GND）。

对于开发和调试来说，Silicon Laboratories 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 Laboratories的调试环境既便于使用又能保持高精度模拟外设的性能。

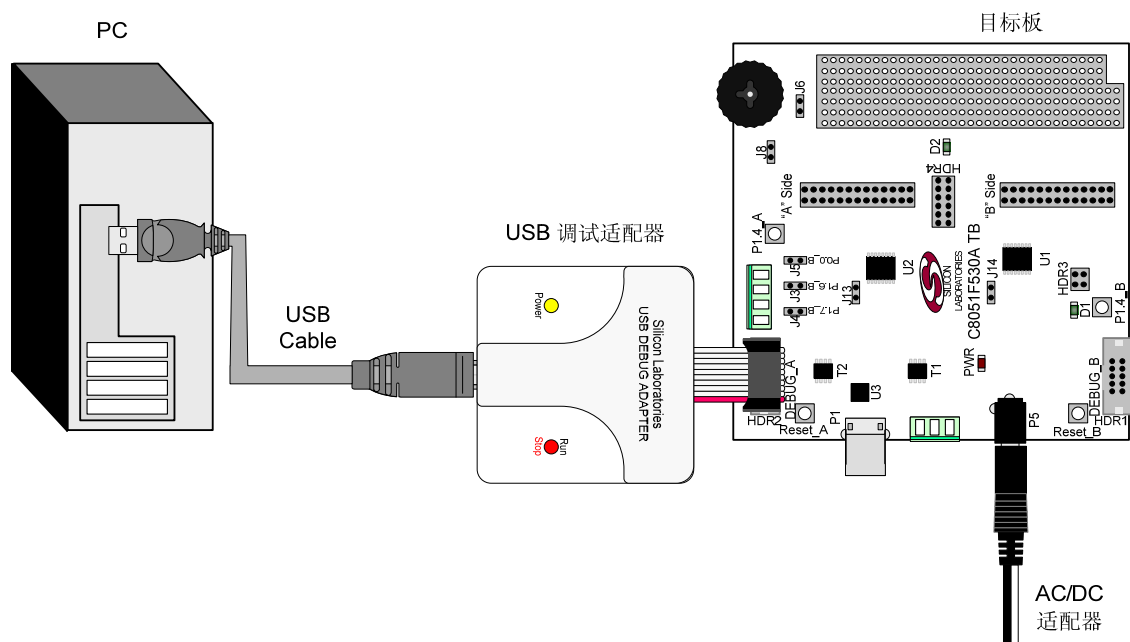


图1.5 开发/在系统调试示意图

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 7680 字节（F520/0A/1/1A 和 F530/0A/1/1A）、4KB（F523/3A/4/4A 和 F533/3A/4/4A）或 2KB（F526/6A/7/7A 或 F536/6A/7/7A）的 FLASH。该存储器可按字节写入，以 512 字节为扇区擦除，且不需特别的片外编程电压。图 1.6 给出了 MCU 系统的存储器组织。

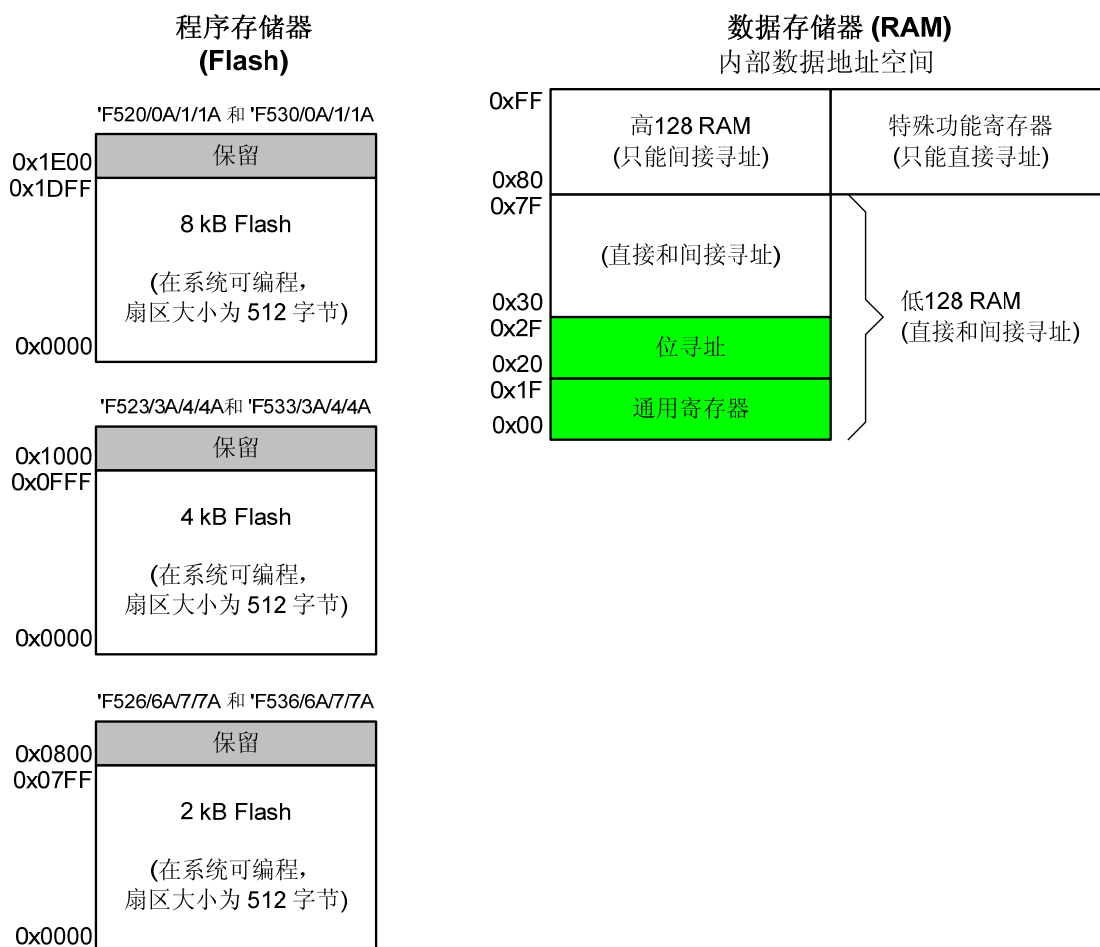


图 1.6 存储器组织

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

1.3 工作方式

C8051F52x/52xA/53x/53xA 器件有四种工作方式：活动（正常）、空闲、挂起和停机。当振荡器和外设处于活动状态时，MCU 工作在正常方式。在空闲方式，CPU 停止运行，而外设和内部时钟处于活动状态。在挂起和停机方式，CPU 停止运行，所有的中断和定时器都处于非活动状态，内部振荡器停止。表 1.2 给出了这些工作方式的说明。

表1.2 工作方式一览表

	特性	功耗	进入条件	退出条件
活动	<ul style="list-style-type: none">• SYSCLK活动• CPU活动（访问FLASH）• 外设是否活动取决于用户设置	全功耗	—	—
空闲	<ul style="list-style-type: none">• SYSCLK活动• CPU不活动（不访问FLASH）• 外设是否活动取决于用户设置	低于全功耗	IDLE (PCON.0)	任何被使能的中断或器件复位
挂起	<ul style="list-style-type: none">• 内部振荡器不活动• 如果SYSCLK源自内部振荡器，则外设和CIP-51停止运行	低	SUSPEND (OSCICN.5)	端口0事件匹配 端口1事件匹配 比较器0使能且输出为逻辑0
停机	<ul style="list-style-type: none">• SYSCLK不活动• CPU不活动（不访问FLASH）• 数字外设不活动；模拟外设活动与否取决于用户设置	很低	STOP (PCON.1)	器件复位

对空闲和停机方式的详细说明见“9.3 电源管理方式”；对挂起方式的详细说明见“15.1.1 内部振荡器挂起方式”。

1.4 12 位模/数转换器

C8051F52x/52xA/53x/53xA器件内部有一个12位SAR ADC，其最大转换速率为200ksps。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（相对于GND）。端口0和1可以作为ADC的输入；另外，包含一个创新的半倍增益选择功能，允许采样大于VREF电压的输入。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置于关断状态或使用突发模式以节省功耗。

A/D转换可以有3种转换启动方式：软件命令、定时器2溢出和外部转换启动信号。这种灵活性允许用软件事件、周期性信号（定时器溢出）或外部硬件信号触发转换。在完成1、4、8或16次采样并由硬件累加器完成累加后，一个状态位指示转换完成并产生中断（如果被允许）。转换结束后，12~16位结果数据字被锁存到ADC数据寄存器中。当系统时钟频率很低时，突发模式允许ADC0自动从低功耗停机状态被唤醒，采集和累加样本值，然后重新进入低功耗停机状态，不需要CPU干预。

ADC数据的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可以用后台方式连续监视一个关键电压，当转换数据位于规定的范围之内/外时才中断控制器。

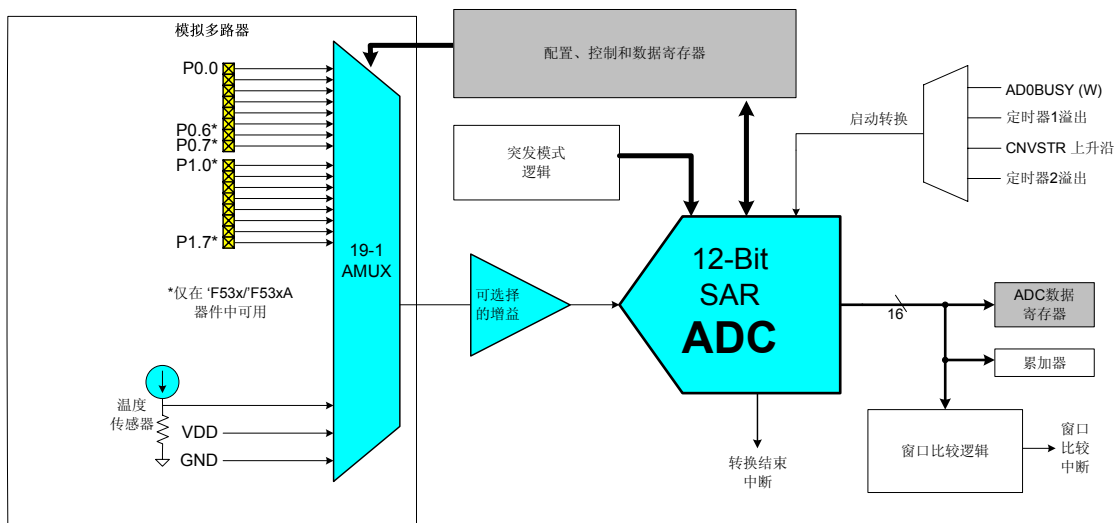


图1.5 12位ADC原理框图

1.5 可编程比较器

C8051F52x/52xA/53x/53xA 器件内部有一个带有输入多路选择器的可软件编程的电压比较器。比较器提供可编程的响应时间和两个可选择连到端口引脚的输出：一个同步锁存输出（CP0），一个异步直接输出（CP0A）。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。当处理器工作在空闲或挂起方式时，这些中断可用作处理器的唤醒源。比较器还可以被配置为复位源。比较器的原理框图示于图1.6。

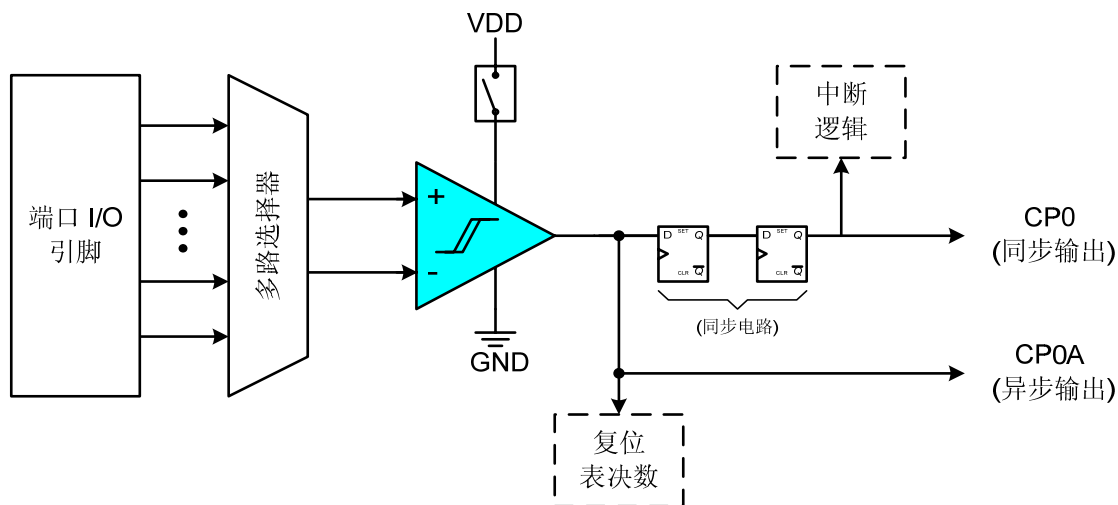


图1.6 比较器原理框图

1.6 稳压器

C8051F52x/52xA/53x/53xA 器件内部包含一个低压差稳压器（REG0）。输入到REG0的 V_{REGIN} 引脚的电压可以高达5.25V。稳压器的输出可以用软件选择为2.1V或2.6V。当被使能时，REG0输出为器件供电并驱动VDD引脚。稳压器可用于为连接到VDD的外部器件提供电源。

1.7 串行端口

C8051F52x/52xA/53x/53xA系列MCU内部有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。

1.8 端口输入/输出

C8051F52x/52xA/53x/53xA 器件最多有16个I/O引脚，端口引脚被组织为两个8位端口。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数字或模拟I/O引脚。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，以降低功耗。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

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1.7）。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、硬件中断以及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应用需要选择通用端口I/O、数字资源和模拟资源的组合。

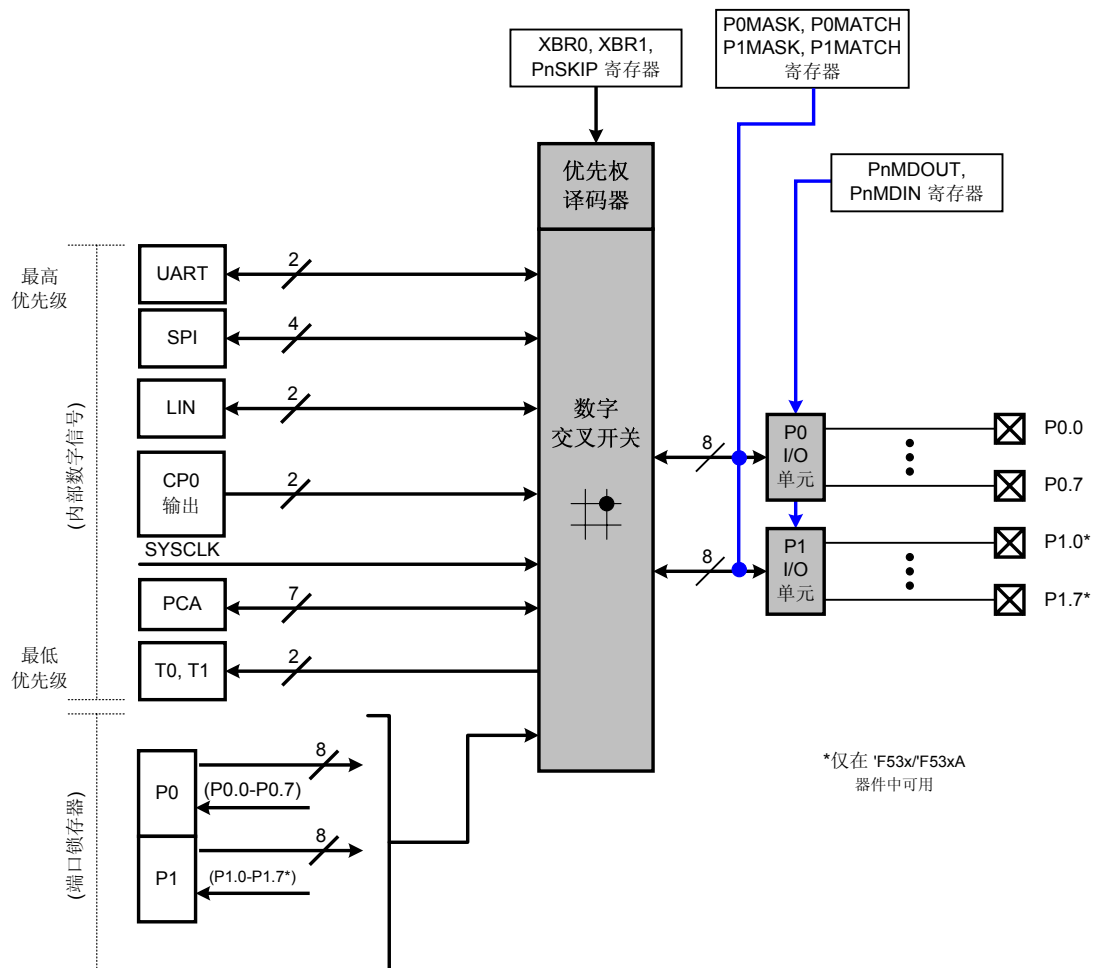


图1.7 端口I/O原理框图

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

2. 极限参数

表 2.1 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	—	135	℃
储存温度		-65	—	150	℃
V _{REGIN} 引脚相对GND的电压		-0.3	—	5.5	V
V _{DD} 引脚相对GND的电压		-0.3	—	2.8	V
V _{XTAL1} 引脚相对GND的电压		-0.3	—	V _{DD} +0.3	V
V _{XTAL2} 引脚相对GND的电压		-0.3	—	V _{DD} +0.3	V
任何端口I/O引脚或/RST相对GND的电压		-0.3	—	V _{REGIN} +0.3	V
任何端口引脚的最大输出灌电流		—	—	100	mA
任何端口引脚的最大输出拉电流		—	—	100	mA
通过V _{REGIN} 和GND的最大总电流		—	—	500	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。					

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 3.1 总体直流电气特性

-40℃到+125℃，25MHz 系统时钟（除非特别说明）。典型值在 25℃给出。

参 数	条 件	最小值	典型值	最大值	单 位
电源输入电压 (V_{REGIN}) ¹	输出电流 = 1mA				
	C8051F52x/53x C8051F52xA/53xA	2.7 1.8	— —	5.25 5.25	V V
数字电源电压 (V_{DD})	C8051F52x/53x	2.0	—	2.7	V
	C8051F52xA/53xA	1.8	—	2.7	V
内核RAM数据保持电源电压		—	1.5	—	V
SYSCLK（系统时钟） ²	（注3和注4）	0	—	25	MHz
额定工作温度范围		-40	—	+125	℃
数字电源电流——CPU 活动（正常方式，从 FLASH 中取指令）					
I_{DD} ³	$V_{\text{DD}} = 2.1\text{V}$				
	Clock = 32 KHz	—	13	—	μA
	Clock = 200 KHz	—	40	—	μA
	Clock = 1 MHz	—	0.25	—	mA
	Clock = 25 MHz	—	9	—	mA
	$V_{\text{DD}} = 2.6\text{V}$				
	Clock = 32 KHz	—	21	—	μA
	Clock = 200 KHz	—	100	—	μA
I_{DD} 电源敏感度 ³	F = 25 MHz	—	TBD	—	%/V
	F = 1 MHz	—	TBD	—	%/V
I_{DD} 频率敏感度 ^{3,4}	T = 25℃				
	VDD = 2.1V, F ≤ 15 MHz	—	TBD	—	mA/MHz
	VDD = 2.1V, F > 15 MHz	—	TBD	—	mA/MHz
	VDD = 2.7V, F ≤ 15 MHz	—	TBD	—	mA/MHz
	VDD = 2.7V, F > 15 MHz	—	TBD	—	mA/MHz
注： <ol style="list-style-type: none"> 有关V_{REGIN}特性的详细信息，见表 7.1。 为能使用调试功能，SYSCLK 必须至少为 32kHz。 基于器件特性数据，非产品测试结果。 在频率≤15MHz时，可以用频率乘以该范围的频率敏感度来估算电源电流I_{DD}。当使用频率敏感度估算SYSCLK>15MHz的I_{DD}时，估算值应为 25MHz时的电流减去由频率敏感度决定的电流差值。例如，VDD = 2.6V，F =20MHz，则$I_{\text{DD}}=11\text{mA}-(25\text{MHz}-20\text{MHz}) \times \text{TBD mA/MHz} = \text{TBDmA}$。 在频率≤1MHz时，空闲$I_{\text{DD}}$可以简单地用频率乘以该范围的频率敏感度来估算。当使用频率敏感度估算F>1MHz的空闲I_{DD}时，估算值应为 25MHz时的电流减去由频率敏感度决定的电流差值。例如，VDD = 2.6V，F =5MHz，则空闲$I_{\text{DD}}= 4 \text{ mA}-(25\text{MHz}-5\text{MHz}) \times \text{TBD mA/MHz} = \text{TBD mA}$。 					

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

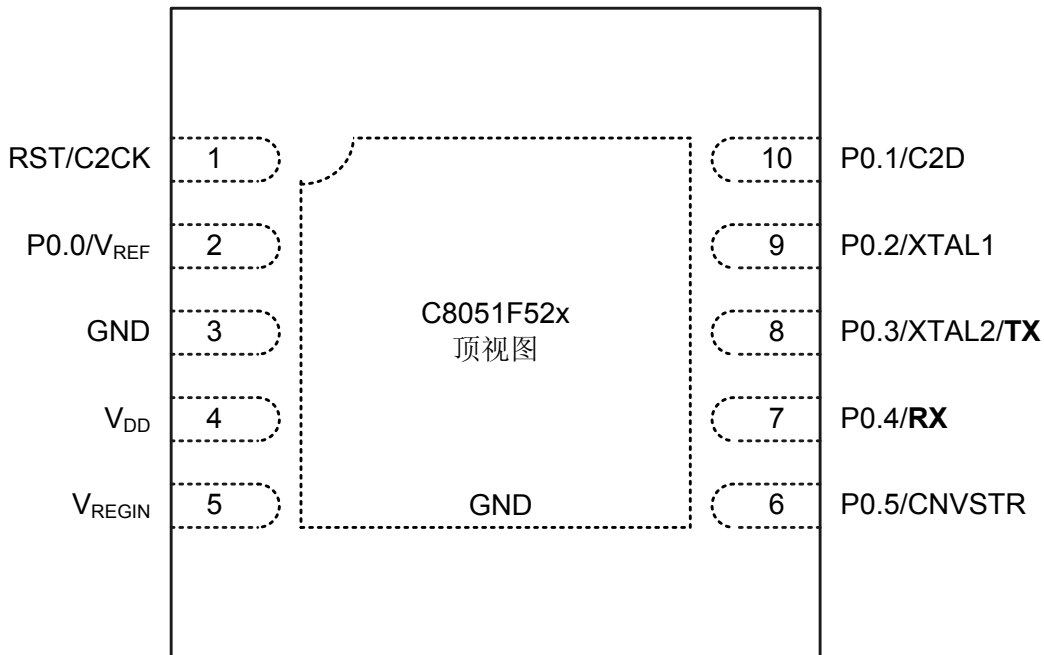
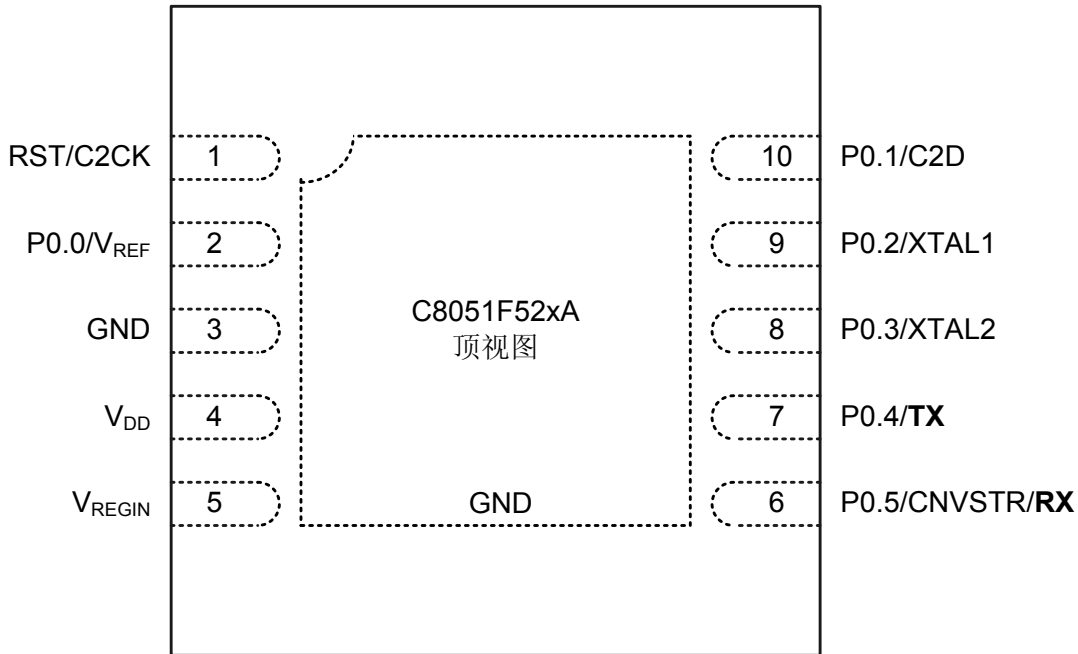
3. 总体直流电气特性

表 3.1 总体直流电气特性

-40℃到+125℃，25MHz 系统时钟（除非特别说明）。典型值在 25℃给出。

参 数	条 件	最小值	典型值	最大值	单 位
数字电源电流——CPU 不活动（空闲方式，不从 FLASH 中取指令）					
空闲 I_{DD}^3	$V_{DD} = 2.1V$	—	10	—	μA
	Clock = 32 KHz	—	22	—	μA
	Clock = 200 KHz	—	0.15	—	mA
	Clock = 1 MHz	—	3	—	mA
	Clock = 25 MHz	—	—	—	—
	$V_{DD} = 2.6V$	—	15	—	μA
	Clock = 32 KHz	—	34	—	μA
	Clock = 200 KHz	—	0.23	—	mA
I_{DD} 电源敏感度 ³	F = 25 MHz	—	TBD	—	%/V
	F = 1 MHz	—	TBD	—	%/V
I_{DD} 频率敏感度 ^{3,5}	T = 25℃	—	TBD	—	mA/MHz
	VDD = 2.1V, F <= 1 MHz	—	TBD	—	mA/MHz
	VDD = 2.1V, F > 1 MHz	—	TBD	—	mA/MHz
	VDD = 2.7V, F <= 1 MHz	—	TBD	—	mA/MHz
数字电源电流（挂起方式） ²	振荡器不运行，VDD 监视器禁止	—	2	—	μA
	振荡器不运行，VDD 监视器禁止	—	2	—	μA
注： <ol style="list-style-type: none"> 有关 V_{REGIN} 特性的详细信息，见表 7.1。 为能使用调试功能，SYSCLK 必须至少为 32kHz。 基于器件特性数据，非产品测试结果。 在频率≤15MHz时，可以用频率乘以该范围的频率敏感度来估算电源电流I_{DD}。当使用频率敏感度估算SYSCLK>15MHz的I_{DD}时，估算值应为 25MHz时的电流减去由频率敏感度决定的电流差值。例如，VDD = 2.6V，F =20MHz，则I_{DD}=11mA- (25MHz-20MHz) × TBD mA/MHz = TBDmA。 在频率≤1MHz时，空闲I_{DD}可以简单地用频率乘以该范围的频率敏感度来估算。当使用频率敏感度估算F>1MHz的空闲I_{DD}时，估算值应为 25MHz时的电流减去由频率敏感度决定的电流差值。例如，VDD = 2.6V，F =5MHz，则空闲I_{DD}= 4 mA- (25MHz-5MHz) × TBD mA/MHz = TBD mA。 					

4. 引脚和封装定义



C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 4.1 C8051F52x 和 C8051F52xA 引脚定义 (DFN-10)

引脚名称	引脚号		引脚类型	说 明
	F52xA	F52x		
/RST/ C2CK	1	1	数字 I/O 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 15μs）来启动一次系统复位。建议在该引脚与V _{DD} 之间接 1KΩ的上拉电阻。 C2 调试接口的时钟信号
P0.0/ VREF	2	2	数字 I/O 或模拟输入 模拟输入	端口 P0.0 外部 VREF 输入
GND	3	3		地。
V _{DD}	4	4		内核电源电压
V _{REGIN}	5	5		内部稳压器输入
P0.5/RX*/ CNVSTR	6	—	数字 I/O 或模拟输入 数字输入	端口 P0.5 /UART RX 引脚 ADC0 的外部转换启动输入
P0.5/ CNVSTR	—	6	数字 I/O 或模拟输入 数字输入	端口 P0.5 ADC0 的外部转换启动输入
P0.4/TX*	7	—	数字 I/O 或模拟输入	端口 P0.4/UART TX 引脚
P0.4/RX*	—	7	数字 I/O 或模拟输入	端口 P0.4/UART RX 引脚
P0.3 XTAL2	8	—	数字 I/O 或模拟输入 数字 I/O	端口 P0.3 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。
P0.3/TX*/ XTAL2	—	8	数字 I/O 或模拟输入 数字 I/O	端口 P0.3/ UART TX 引脚 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。
P0.2/ XTAL1	9	9	数字 I/O 或模拟输入 模拟输入	端口 P0.2 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。
P0.1/ C2D	10	10	数字 I/O 或模拟输入 数字 I/O	端口 P0.1 C2 调试接口的双向数据信号。
*注：参见“21. 器件特定行为特性”。				

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

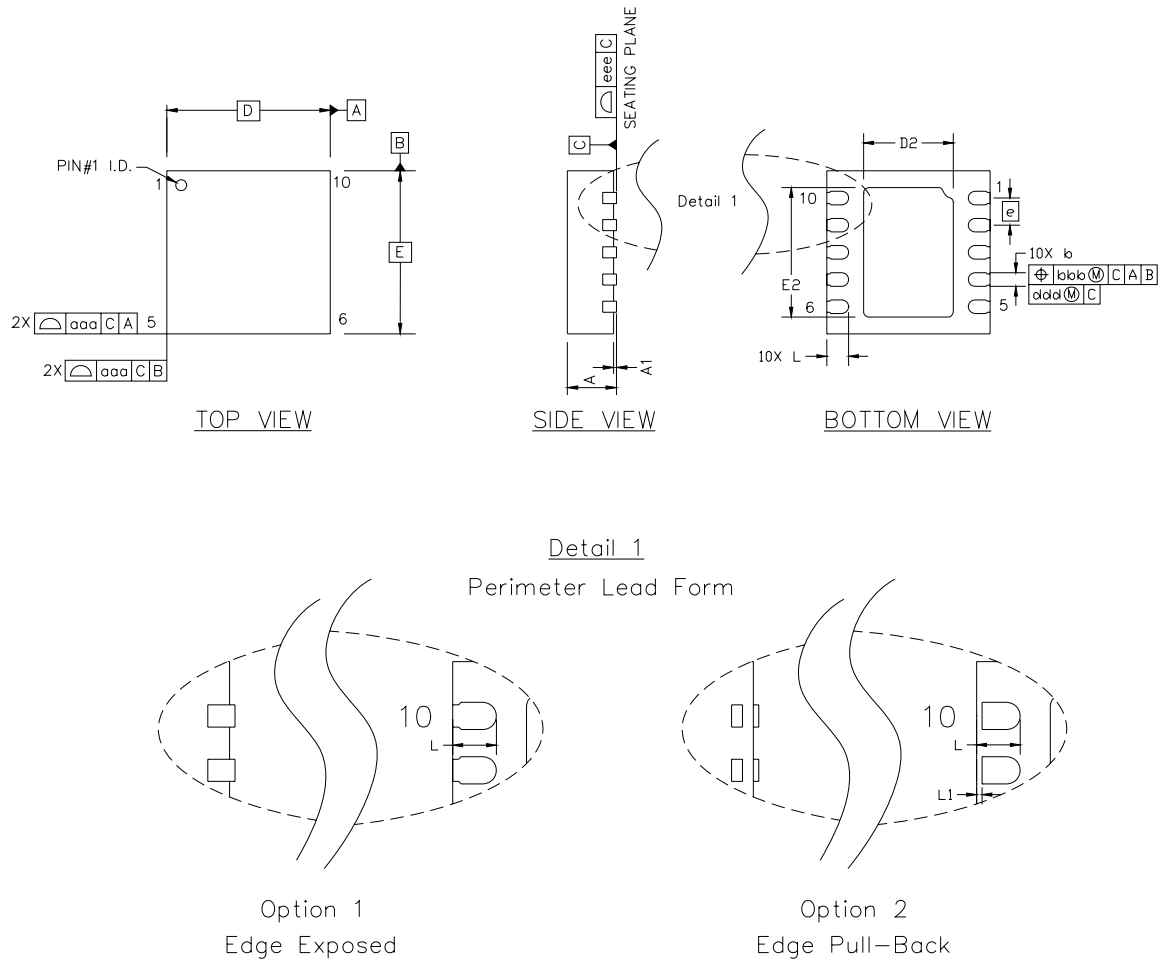


图 4.1 DFN-10 封装图*

*注：封装尺寸见表 4.2 “DFN-10 封装尺寸”。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 4.2 DFN-10 封装尺寸

尺寸	最小值	典型值	最大值
A	0.80	0.90	1.00
A1	0.03	0.07	0.11
b	0.18	0.25	0.30
D	3.0 BSC		
D2	1.50	1.65	1.80
e	0.50 BSC		
E	3.0 BSC		
E2	2.23	2.38	2.53
L	0.30	0.40	0.50
L1	0.00	—	0.15
aaa	—	—	0.15
bbb	—	—	0.15
ddd	—	—	0.05
eee	—	—	0.08
注：所有尺寸都以 mm 为单位，除非特别说明			

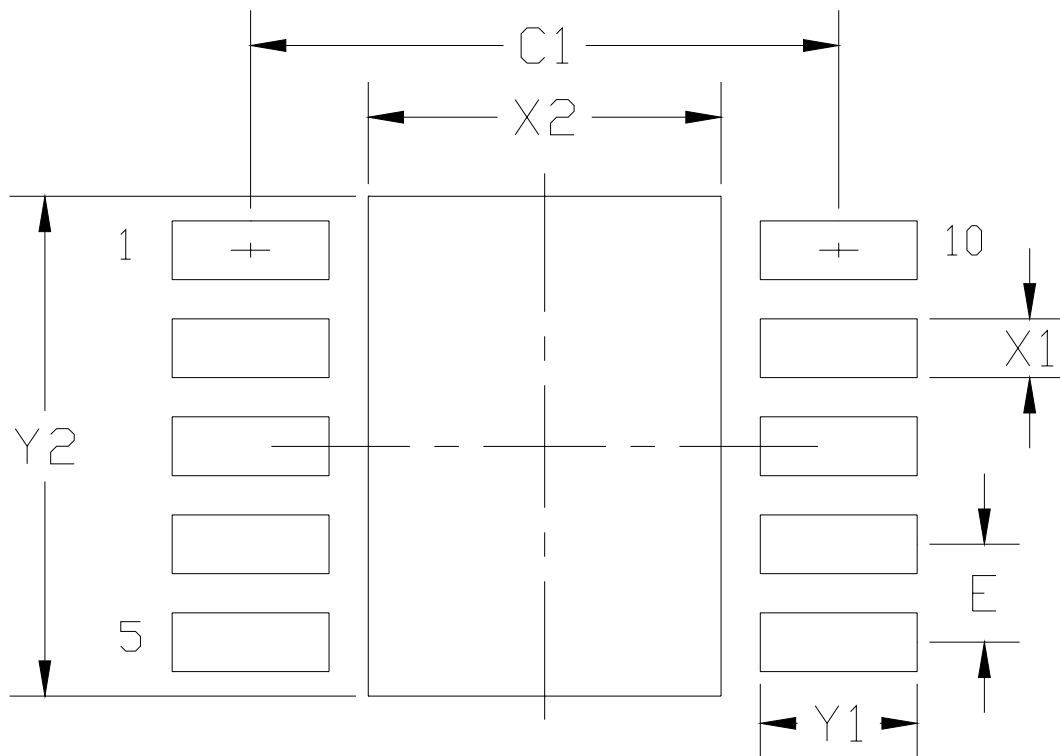
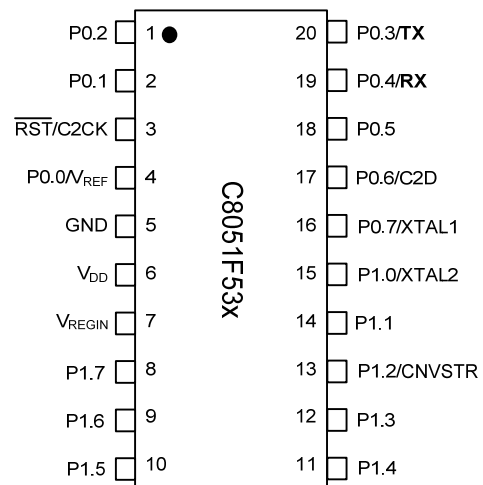
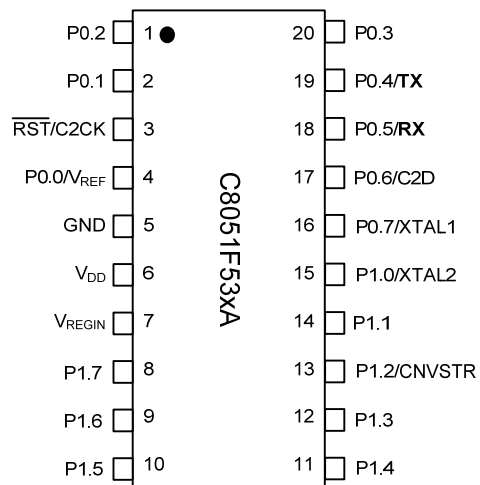


图 4.2 DFN-10 焊盘图

表 4.3 DFN-10 焊盘图尺寸

尺寸	最小值	最大值
C1	2.90	3.00
E	0.50 BSC	
X1	0.20	0.30
X2	1.70	1.80
Y1	0.70	0.80
Y2	2.45	2.55
注：所有尺寸都以 mm 为单位，除非特别说明		

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器



C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 4.4 C8051F53x 和 C8051F53xA 引脚定义 (TSSOP-20)

引脚名	引脚号		引脚类型	说 明
	F53xA	F53x		
P0.2	1	1	数字 I/O 或模拟输入	端口 P0.2
P0.1	2	2	数字 I/O 或模拟输入	端口 P0.1
/RST/ C2CK	3	3	数字 I/O 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 15μs）来启动一次系统复位。建议在该引脚与V _{DD} 之间接 1KΩ的上拉电阻。 C2 调试接口的时钟信号。
P0.0/ VREF	4	4	数字 I/O 或模拟输入 模拟输入	端口 P0.0 外部 VREF 输入
GND	5	5		地。
V _{DD}	6	6		内核电源电压
V _{REGIN}	7	7		内部稳压器输入
P1.7	8	8	数字 I/O 或模拟输入	端口 P1.7
P1.6	9	9	数字 I/O 或模拟输入	端口 P1.6
P1.5	10	10	数字 I/O 或模拟输入	端口 P1.5
P1.4	11	11	数字 I/O 或模拟输入	端口 P1.4
P1.3	12	12	数字 I/O 或模拟输入	端口 P1.3
P1.2 CNVSTR	13	13	数字 I/O 或模拟输入 数字输入	端口 P1.2 ADC0 的外部转换启动输入。
P1.1	14	14	数字 I/O 或模拟输入	端口 P1.1
P1.0 XTAL2	15	15	数字 I/O 或模拟输入 数字 I/O	端口 P1.0 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。
P0.7/ XTAL1	16	16	数字 I/O 或模拟输入 模拟输入	端口 P0.7 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。
P0.6/ C2D	17	17	数字 I/O 或模拟输入 数字 I/O	端口 P0.6 C2 调试接口的双向数据信号。
P0.5/RX*	18	—	数字 I/O 或模拟输入	端口 P0.5/UART RX 引脚。
P0.5	—	18	数字 I/O 或模拟输入	端口 P0.5。
P0.4/TX*	19	—	数字 I/O 或模拟输入	端口 P0.4/ UART TX 引脚。
P0.4/RX*	—	19	数字 I/O 或模拟输入	端口 P0.4/ UART TX 引脚。
P0.3	20	—	数字 I/O 或模拟输入	端口 P0.3
P0.3/TX*	—	20	数字 I/O 或模拟输入	端口 P0.3/ UART TX 引脚。
*注：参见“21. 器件特定行为特性”。				

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

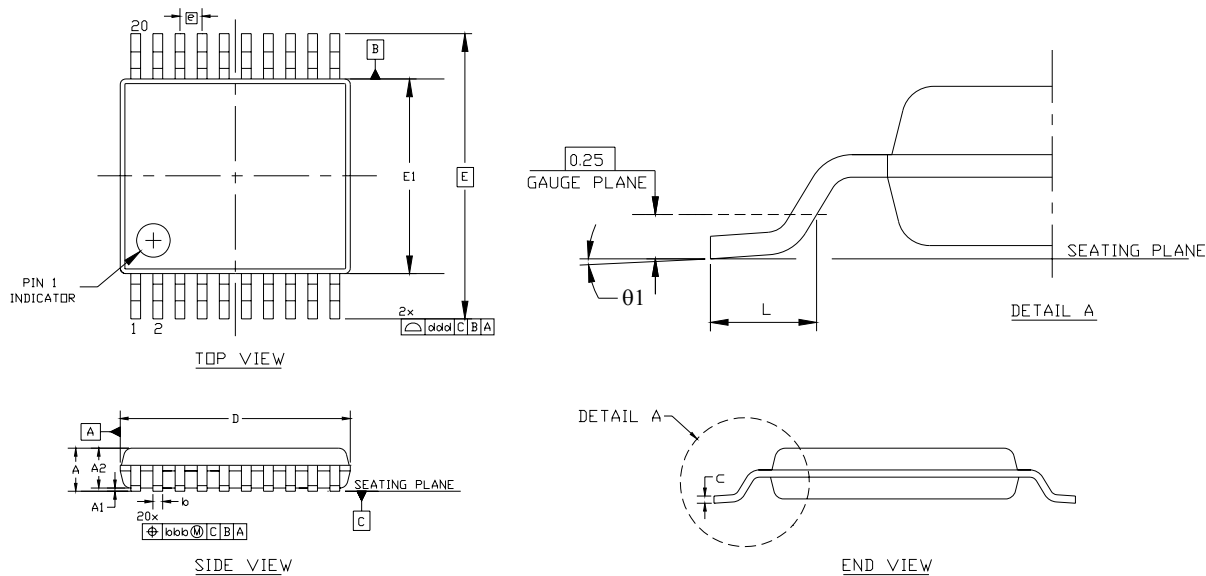


图 4.3 TSSOP-20 封装图*

表 4.5 DFN-10 封装尺寸

尺寸	最小值	典型值	最大值
A	—	—	1.20
A1	0.05	—	0.15
A2	0.80	1.00	1.05
b	0.19	—	0.30
c	0.09	—	0.20
D	6.40	6.50	6.60
e	0.65 BSC		
E	6.40 BSC		
E1	4.30	4.40	4.50
L	0.45	0.60	0.75
θ1	0°	—	8°
bbb	0.10		
ddd	0.20		
注：所有尺寸都以 mm 为单位，除非特别说明			

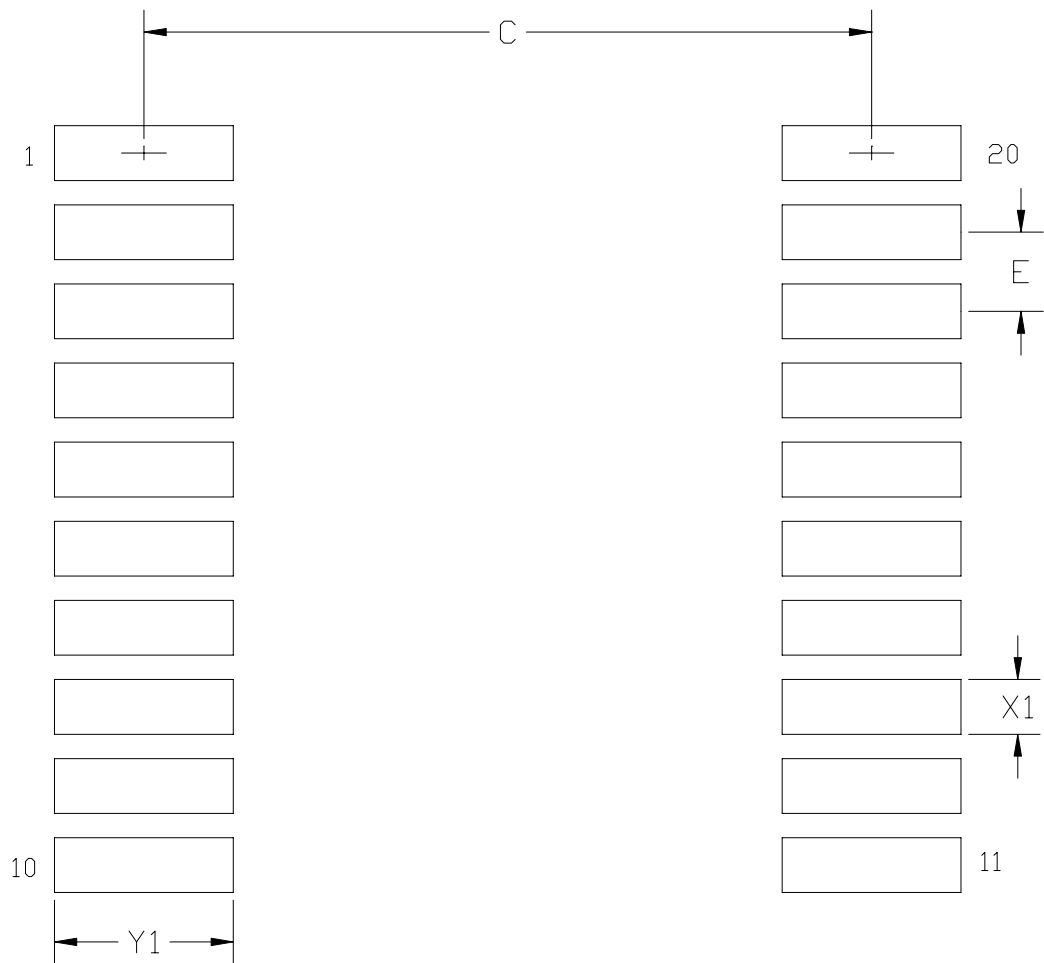
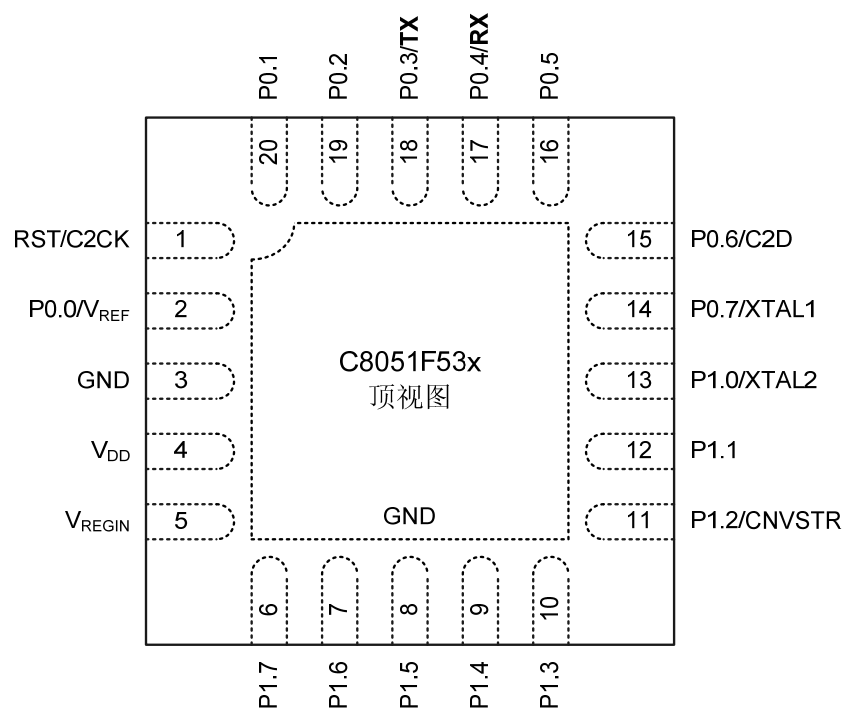
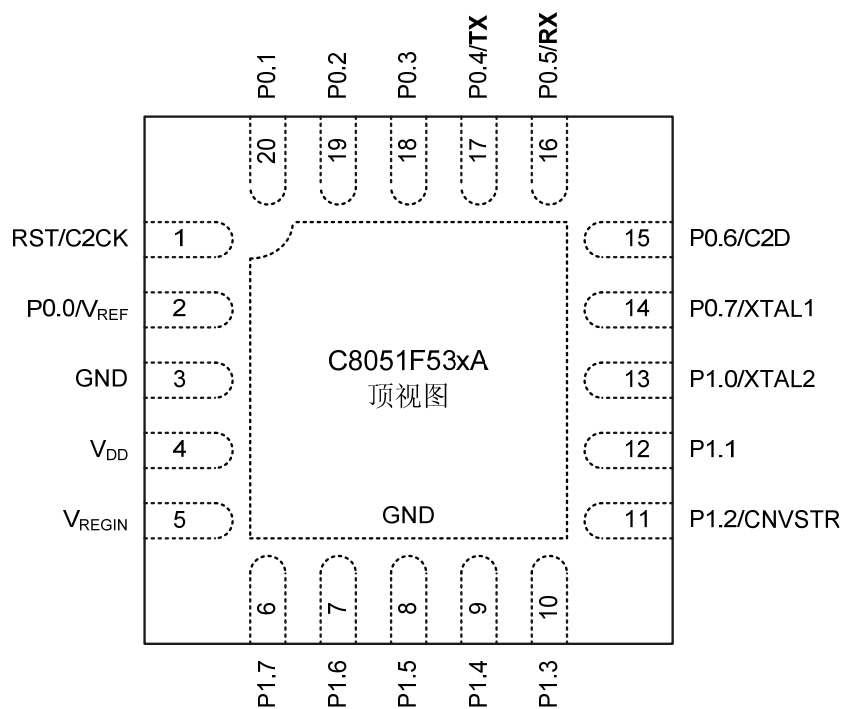


图 4.4 TSSOP-20 焊盘图

表 4.6 TSSOP-20 焊盘图尺寸

尺寸	最小值	最大值
C	5.80	5.90
E	0.65 BSC	
X1	0.35	0.45
Y1	1.35	1.45
注：所有尺寸都以 mm 为单位，除非特别说明		

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器



C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 4.7 C8051F53x 和 C8051F53xA 引脚定义 (QFN-20)

引脚名称	引脚号		引脚类型	说 明
	F53xA	F53x		
/RST/ C2CK	1	1	数字 I/O 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 15μs）来启动一次系统复位。建议在该引脚与V _{DD} 之间接 1KΩ的上拉电阻。 C2 调试接口的时钟信号。
P0.0/ VREF	2	2	数字 I/O 或模拟输入 模拟输入	端口 P0.0 外部 VREF 输入
GND	3	3		地。
V _{DD}	4	4		内核电源电压
V _{REGIN}	5	5		内部稳压器输入
P1.7	6	6	数字 I/O 或模拟输入	端口 P1.7
P1.6	7	7	数字 I/O 或模拟输入	端口 P1.6
P1.5	8	8	数字 I/O 或模拟输入	端口 P1.5
P1.4	9	9	数字 I/O 或模拟输入	端口 P1.4
P1.3	10	10	数字 I/O 或模拟输入	端口 P1.3
P1.2 CNVSTR	11	11	数字 I/O 或模拟输入 数字输入	端口 P1.2 ADC0 的外部转换启动输入。
P1.1	12	12	数字 I/O 或模拟输入	端口 P1.1
P1.0 XTAL2	13	13	数字 I/O 或模拟输入 数字 I/O	端口 P1.0 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。
P0.7/ XTAL1	14	14	数字 I/O 或模拟输入 模拟输入	端口 P0.7 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。
P0.6/ C2D	15	15	数字 I/O 或模拟输入 数字 I/O	端口 P0.6 C2 调试接口的双向数据信号。
P0.5/RX*	16	—	数字 I/O 或模拟输入	端口 P0.5/ UART RX 引脚。
P0.5	—	16	数字 I/O 或模拟输入	端口 P0.5。
P0.4/TX*	17	—	数字 I/O 或模拟输入	端口 P0.4 /UART TX 引脚。
P0.4/RX*	—	17	数字 I/O 或模拟输入	端口 P0.4 /UART RX 引脚。
P0.3	18	—	数字 I/O 或模拟输入	端口 P0.3。
P0.3/TX*	—	18	数字 I/O 或模拟输入	端口 P0.3 /UART TX 引脚。
P0.2	19	19	数字 I/O 或模拟输入	端口 P0.2
P0.1	20	20	数字 I/O 或模拟输入	端口 P0.1
*注：参见“21. 器件特定行为特性”。				

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

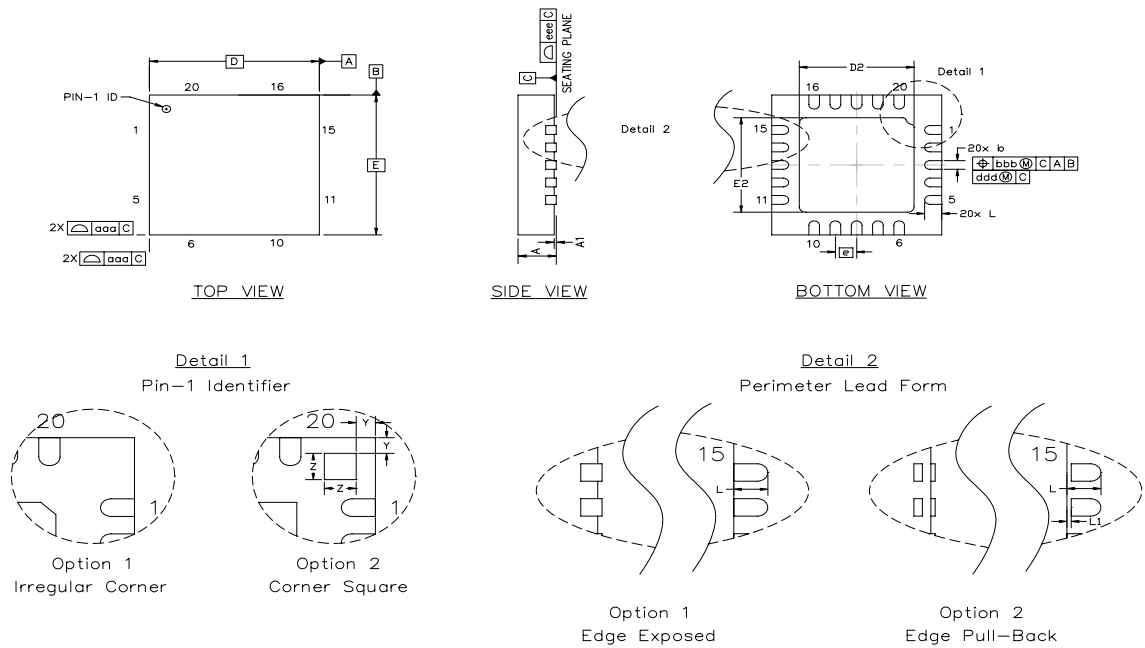


图 4.5 QFN-20 封装图

*注：封装尺寸见表 4.8 “QFN-20 封装尺寸”。

表 4.8 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	2.55	2.70	2.85
e	0.50 BSC		
E	4.0 BSC		
E2	2.55	2.70	2.85
L	0.30	0.40	0.50
L1	0.00	—	0.15
aaa	—	—	0.15
bbb	—	—	0.10
ddd	—	—	0.05
eee	—	—	0.08
Z	—	0.43	—
Y	—	0.18	—

注：所有尺寸都以 mm 为单位，除非特别说明

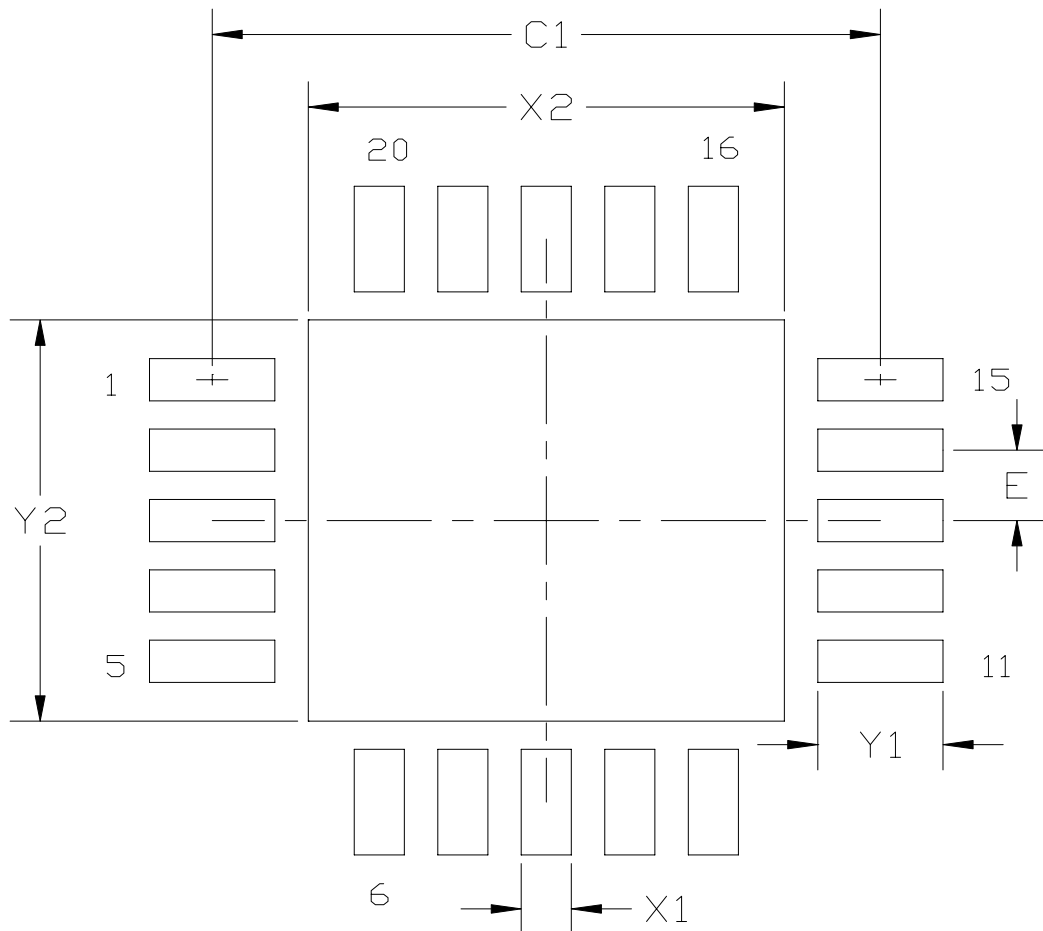


图 4.6 QFN-20 焊盘图

表 4.9 QFN-20 焊盘图尺寸

尺寸	最小值	最大值
C1	3.90	4.00
E	0.50 BSC	
X1	0.20	0.30
X2	2.75	2.85
Y1	0.65	0.75
Y2	2.75	2.85
注：所有尺寸都以 mm 为单位，除非特别说明		

5.12 位 ADC (ADC0)

C8051F52x/52xA/53x/53xA 的 ADC0 子系统集成了一个 16/6 通道的模拟多路选择器 (AMUX0) 和一个 200ksps 的 12 位逐次逼近寄存器型 ADC，ADC 中集成了跟踪保持电路、可编程窗口检测器、可编程衰减器和硬件累加器。ADC0 子系统有一种特殊的突发方式 (Burst mode)，该方式可自动使能 ADC0，采集和累加样本值，然后将 ADC0 置于低功耗停机方式，而不需 CPU 干预。AMUX0、数据转换方式及窗口检测器都可用软件通过特殊功能寄存器来配置，如图 5.1 所示)。ADC0 输入为单端方式，可以被配置为用于测量 P0.0 ~ P1.7、温度传感器输出、VDD 或 GND (相对于 GND)。只有当 ADC 控制寄存器 (ADC0CN) 中的 AD0EN 位被置 1 或在突发方式执行转换时，ADC0 子系统才被使能。当 AD0EN 位为 0 时和在突发方式下不进行转换时，ADC0 子系统处于低功耗关断方式。

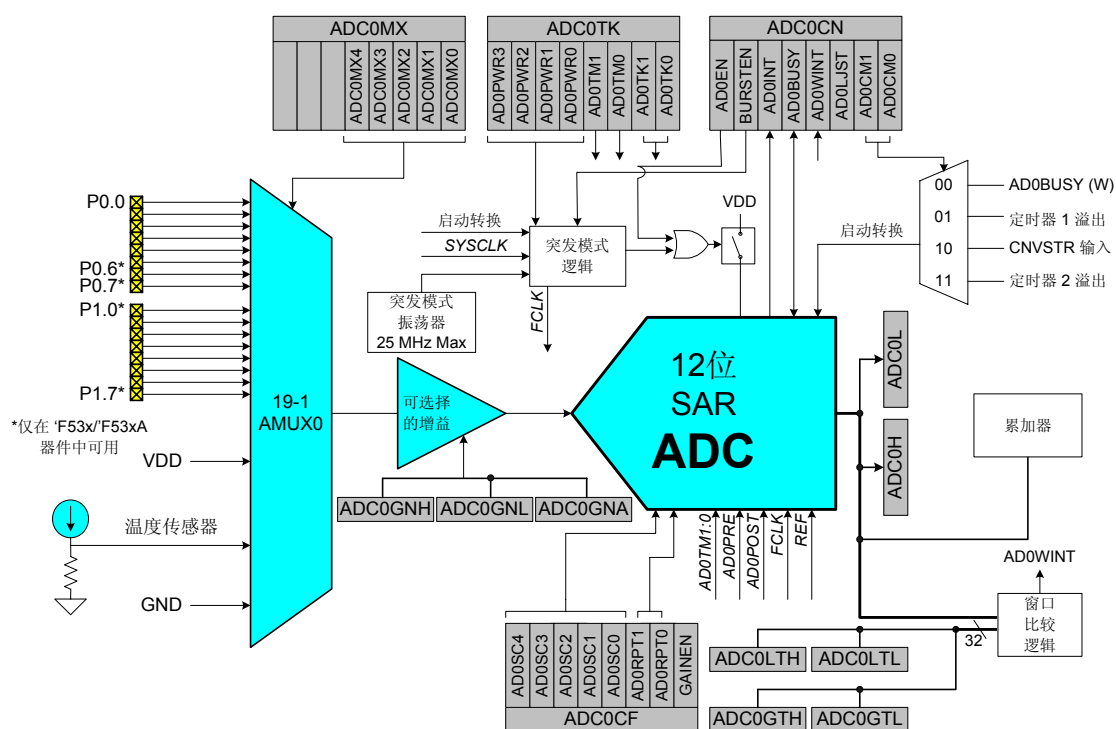


图 5.1 ADC0 功能框图

5.1 模拟多路选择器

AMUX0 选择 ADC 的输入通道。P0.0 ~ P1.7、片内温度传感器输出、内核电源 (VDD) 或 GND 中的任何一个都可以被选择为 ADC 输入。ADC0 工作在单端方式，所有信号测量都是相对于 GND 的。ADC0 的输入通道由寄存器 ADC0MX 选择 (见 SFR 定义 5.1)。

需要特别注意的是，被选择为 ADC0 输入的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 (n=0,1) 寄存器中的对应位置 0，并向对应的端口锁存器 Pn(n=0,1) 写 1。为了使交叉开关跳过一个端口引脚，应将 PnSKIP (n=0,1) 寄存器中的对应位置 1。有关端口 I/O 配置的详细信息见“14. 端口输入/输出”。

5.2 温度传感器

C8051F52x/52xA/53x/53xA 内部包含一个温度传感器，可以通过 ADC0 多路器直接测量其输出。为了使用 ADC0 测量温度传感器的输出，ADC 的多路器通道应被配置为连接到温度传感器。温度传感器的典型传输函数示于图 5.2，输出电压 (V_{TEMP}) 由寄存器 ADC0MX 中的 AD0MX[4:0] 选择为 ADC 的正输入。寄存器 REF0CN 中的 TEMPE 位使能/禁止温度传感器，见 SFR 定义 6.1 的说明。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 ADC 测量都是无意义的。温度传感器的斜率和偏移参数见表 5.1。

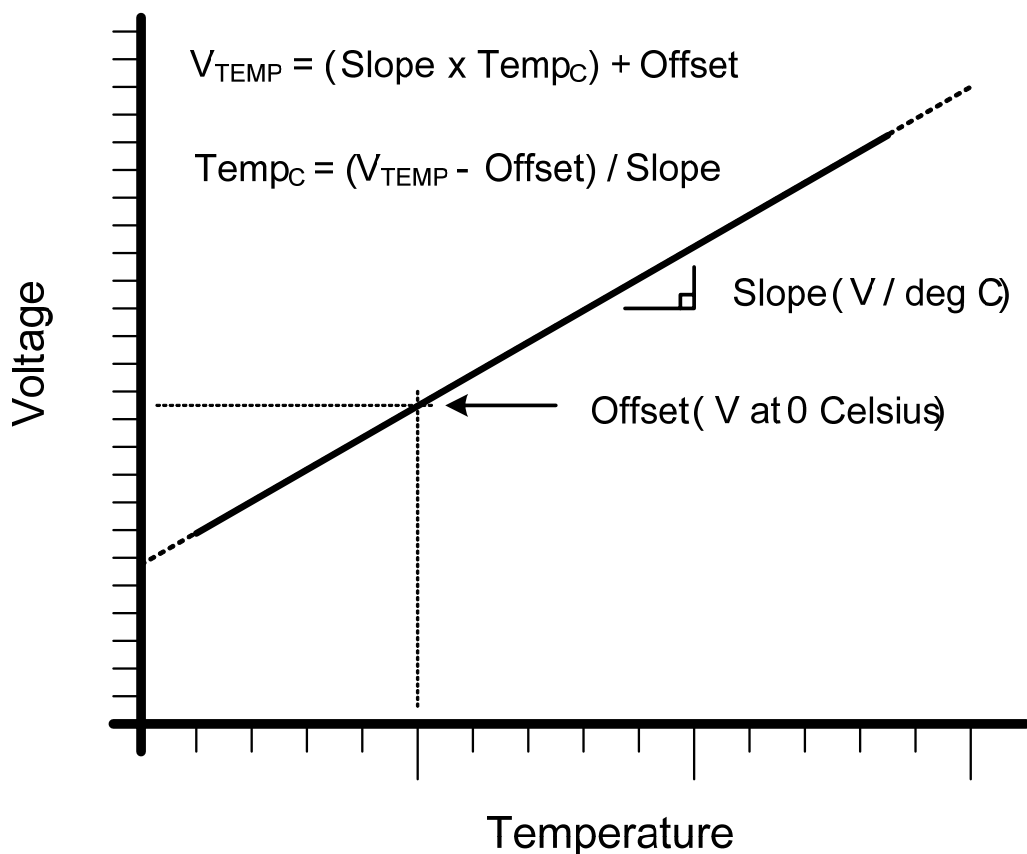


图 5.2 温度传感器典型传输函数

5.3 工作方式

在一个典型系统中，用下面的步骤来配置 ADC0：

1. 如果需要选择衰减 (1:2) 功能，请参见“5.5 可选择的衰减功能”。
2. 选择转换启动源。
3. 选择正常方式或突发方式。
4. 如果使用突发方式，选择 ADC0 空闲电源状态并设置上电时间。
5. 选择跟踪方式。注意：预跟踪方式只能用于正常转换方式。
6. 计算需要的建立时间，并用 AD0TK 位设置转换启动后的跟踪时间。
7. 选择重复次数。
8. 选择输出字对齐方式（右对齐或左对齐）。
9. 使能或禁止转换结束及窗口比较器中断。

5.3.1 启动转换

有 4 种 A/D 转换启动方式，由 ADC0CN 中的 ADC0 转换启动方式位 (AD0CM1-0) 的状态决定采用哪一种方式。转换触发源有：

- 写 1 到 ADC0CN 的 AD0BUSY 位；
- CNVSTR 输入信号 (P0.6) 的上升沿；
- 定时器 1 溢出 (即定时连续转换)；
- 定时器 2 溢出 (即定时连续转换)。

向 AD0BUSY 写 1 方式提供了用软件根据需要控制 ADC0 转换的能力。AD0BUSY 位在转换期间被置 1，转换结束后复 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：当工作在查询方式时，应使用 ADC0 中断标志 (AD0INT) 来查询 ADC 转换是否完成。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 溢出时，如果定时器 2 工作在 8 位方式，使用定时器 2 的低字节溢出；如果定时器 2 工作在 16 位方式，则使用定时器 2 的高字节溢出。有关定时器配置方面的信息见“19. 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，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 POSKIP 中的位 6 置 1。有关端口 I/O 配置的详细信息，见“14. 端口输入/输出”。

5.3.2 跟踪方式

根据表 5.1 和表 5.2，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。ADC0 有三种跟踪方式：预跟踪、后跟踪和双跟踪。预跟踪方式在转换启动信号有效前连续跟踪，提供最小的转换延时 (转换启动信号有效到转换结束)。该方式需要软件管理，以保证满足最短跟踪时间要求。在后跟踪方式，在转换启动信号有效之后进行跟踪的时间长度是可编程的，并由硬件管理。双跟踪方式在转换启动信号有效之前和之后都跟踪，使跟踪时间最大化。图 5.3 给出了这三种跟踪方式的例子。

当 AD0TM 被设置为 10b 时选择预跟踪方式。该方式在转换启动信号开始后立即启动转换。ADC0 在不转换时会一直跟踪。软件必须在每次转换结束和下一次转换启动信号之间保证最小

的跟踪时间。在 ADC0 被使能后的第一个转换启动信号之前也必须满足最小跟踪时间。

当 AD0TM 被设置为 01b 时选择后跟踪方式。该方式在转换启动信号开始后立即启动跟踪，跟踪时间用 AD0TK 编程。在编程的跟踪时间结束后开始转换。转换结束后，ADC0 不再跟踪输入信号。但采样电容仍保持与输入断开的状态，使输入引脚呈现高阻抗，直到下一个转换启动信号有效。

当 AD0TM 被设置为 11b 时选择双跟踪方式。该方式在转换启动信号开始后立即启动跟踪，跟踪时间用 AD0TK 编程。在编程的跟踪时间结束后开始转换。转换结束后，ADC0 继续跟踪输入信号，直到下一次转换开始。

随着连接到 ADC 输入的信号不同，在改变 MUX 设置之后，实际需要的跟踪时间可能比表 5.1 和表 5.1 给出的最小跟踪时间要长。对建立时间的要求见“5.3.6 建立时间要求”。



图 5.3 ADC0 跟踪方式

5.3.3 时序

表 5.1 和表 5.2 给出了 ADC0 的最大转换速度指标。ADC0 由 ADC0 子系统时钟（FCLK）定时。FCLK 的时钟源由 BURSTEN 位选择。当 BURSTEN 为逻辑 0 时，FCLK 源自当前的系统时钟；当 BURSTEN 为逻辑 1 时，FCLK 源自突发方式振荡器，这是一个独立的时钟源，其最高频率为 25 MHz。

当 ADC0 执行一次转换时，它需要一个一般来说比 FCLK 慢的时钟。ADC0 SAR 转换时钟（SAR 时钟）由 FCLK 分频得到。分频系数用 ADC0CF 寄存器中的 AD0SC 位控制。最大 SAR 时钟频率列于表 5.1 和表 5.2 中。

在任一给定时刻，ADC0 处于这三种状态之一：跟踪、转换或空闲。跟踪时间取决于所选择的跟踪方式。对于前跟踪方式，跟踪时间由软件管理，ADC0 在转换启动信号开始后立即启动转换。对于后跟踪和双跟踪方式，转换启动信号有效后的跟踪时间等于由 AD0TK 决定的时间加上两个 FLCK 周期。跟踪结束后立即开始转换。ADC0 转换时间（从转换开始到转换结束）总是为 13 个 SAR 时钟加上两个 FCLK 周期。图 5.4 给出了前跟踪方式的一次转换和后跟踪或双跟踪方式跟踪加转换的时序图。在该例中，重复次数被设置为 1。

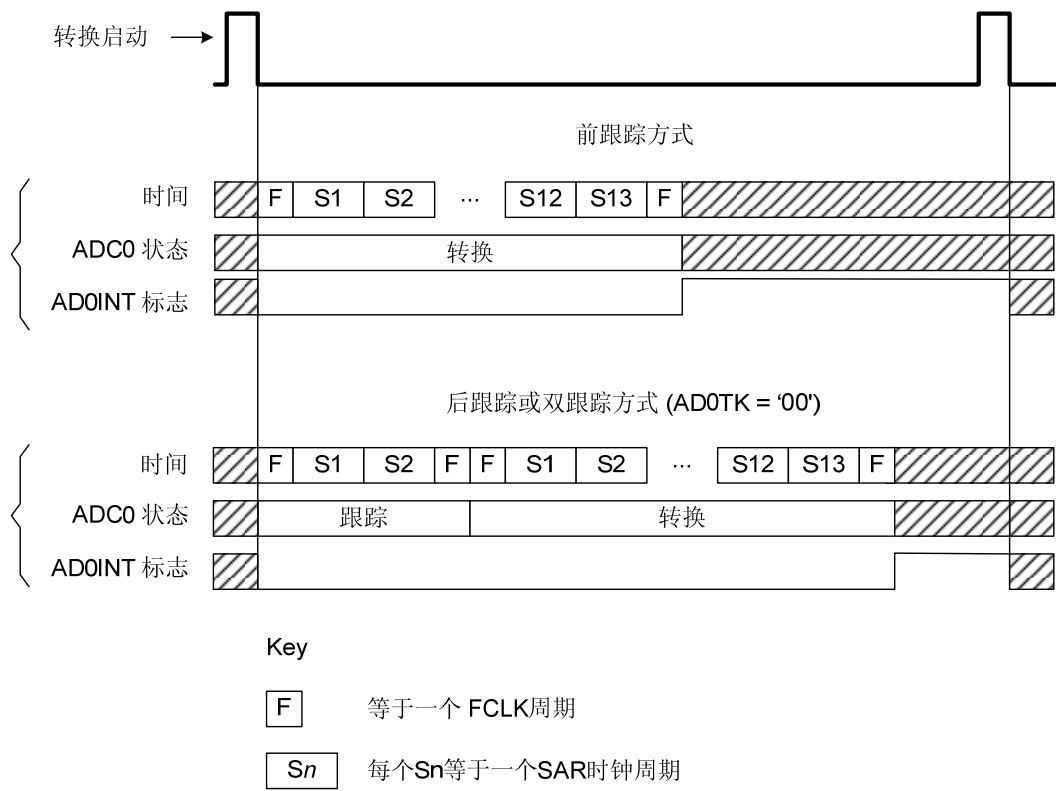


图 5.4 12 位 ADC 跟踪方式示例

5.3.4 突发方式

突发模式是一种节省功耗的功能特性，允许 ADC0 在两次转换期间保持低功耗状态。当突发模式被使能时，ADC0 从低功耗状态被唤醒，使用内部突发模式时钟（约 25 MHz），累加 1、4、8 或 16 个采样值，然后又重新进入低功耗状态。由于突发模式时钟独立于系统时钟，ADC0 可以在一个系统时钟周期内完成多次转换并重新进入低功耗状态，即使系统时钟频率很低（如 32.768 KHz）或被挂起。

将 BURSTEN 设置为逻辑 1 即使能突发模式。当工作在突发模式时，AD0EN 控制 ADC0 的空闲电源状态（即 ADC0 不跟踪也不执行转换时进入的状态）。如果 AD0EN 被设置为逻辑 0，ADC0 在每次突发转换后进入断电状态；如果 AD0EN 被设置为逻辑 1，ADC0 在每次突发转换后保持使能状态。每来一次转换启动信号，ADC0 被从其低功耗状态唤醒。如果 ADC0 被断电，它会自动上电并等待一个可编程的上电时间，该上电时间由 AD0PWR 位控制。否则，ADC0 会立即启动跟踪和转换。图 5.5 给出了使用慢速系统时钟且重复次数为 4 时的突发模式示例。

注意：当突发模式被使能时，只能使用后跟踪或双跟踪方式。

当突发模式被使能时，一次转换启动将进行多次转换，转换次数等于重复次数。当突发模式被禁止时，每次转换都需要有转换启动信号。在这两种情况下，在完成“重复次数”次转换和累加后，ADC0 转换结束中断会被置 1。类似地，在完成“重复次数”次转换和累加之前，窗口比较器不会将结果与“大于”或“小于”寄存器进行比较。

注：使用突发模式时必须谨慎，不能以高于 SYSCLK 频率的 1/4 发出转换启动信号，包括外部转换启动信号。

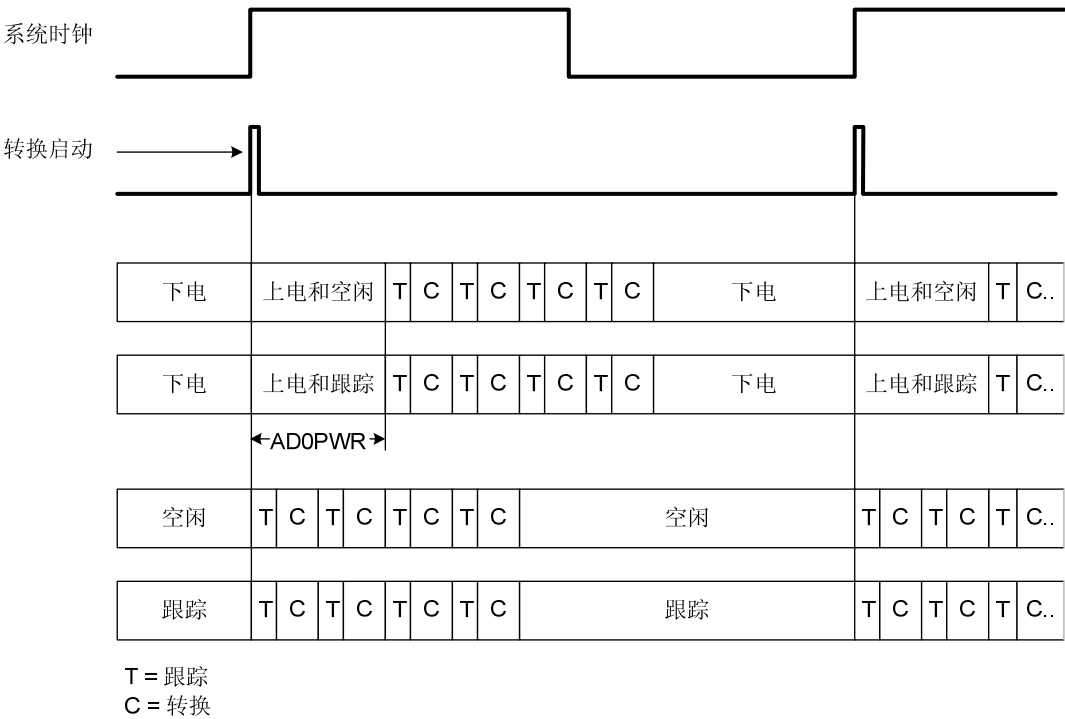


图 5.5 12 位 ADC 突发模式示例（重复次数为 4）

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

5.3.5 输出转换码

寄存器 ADC0H 和 ADC0L 保存输出转换码的高字节和低字节。当重复次数被设置为 1 时，转换码以 12 位无符号整数形式表示，并且输出转换码在每次转换后被更新。输入测量范围为 $0 \sim VREF \times 4095/4096$ 。数据可以是右对齐或左对齐，由 AD0LJST 位（ADC0CN.2）的设置决定。ADC0H 和 ADC0L 寄存器中未使用的位被清 0。表 5.0 给出了右对齐和左对齐的转换码示例。

表 5.0a. ADC0 右对齐和左对齐数据示例

输入电压	右对齐 ADC0H:ADC0L (AD0LJST=0)	左对齐 ADC0H:ADC0L (AD0LJST=1)
$VREF \times 4095/4096$	0x0FFF	0xFFFF0
$VREF \times 2048/4096$	0x0800	0x8000
$VREF \times 2047/4096$	0x07FF	0x7FF0
0	0x0000	0x0000

当ADC0 重复次数大于 1 时，输出转换码代表所有转换值累加的结果，并在最后一次转换结束后被更新。根据所选择的重复次数不同，输出值可以是 14 位（4 次采样）、15 位（8 次采样）或 16 位（16 次采样）的无符号整数。重复次数用ADC0CF寄存器中的AD0RPT位选择。结果值必须是右对齐的（AD0LJST = 0），ADC0H和ADC0L寄存器中未使用的位被清 0。表 5.2 给出了对应不同输入电压和重复次数的右对齐结果示例。注意：当从ADC返回的所有采样结果都相同时，累加 2^n 个采样值等价于左移n位。

表 5.0b. 不同输入电压下的 ADC0 重复次数示例

输入电压	重复次数 = 4	重复次数 = 8	重复次数 = 16
$VREF \times 4095/4096$	0x3FFC	0x7FF8	0xFFFF0
$VREF \times 2048/4096$	0x2000	0x4000	0x8000
$VREF \times 2047/4096$	0x1FFC	0x3FF8	0x7FF0
0	0x0000	0x0000	0x0000

5.3.6 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。该跟踪时间由 AMUX0 的电阻、ADC0 采样电容、外部信号源阻抗及所要求的转换精度决定。

图 5.6 给出了等效的ADC0 输入电路。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 5.1 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 5.3 和表 5.4 给出了ADC0 的最小建立时间要求。

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

方程 5.1 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 电阻与外部信号源电阻之和；

n 为 ADC 的分辨率，用比特表示（12）。

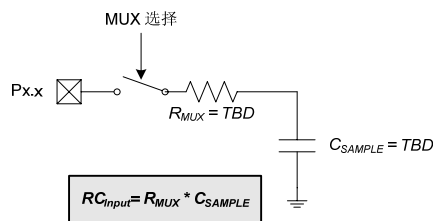


图 5.6 ADC0 等效输入电路

5.4 可选择的增益

C8051F52x/52xA/53x/53xA 系列器件的 ADC0 实现了一个可选择的增益调节选项。通过向增益调节地址范围写入一个数值，用户可以在 0 和 1.016 之间选择增益值。

例如，有三个要测量的模拟源，其满度输出分别为 5.0V、4.0V 和 3.0V。每个 ADC 测量最好都使用 ADC 的完全动态范围，使用 1.5V 或 2.2V 的内部电压基准（本例中设置为 2.2V）。当选择信号 1 时（满度值为 5.0V）且输入信号为 5.0V 时，增益值 0.44（5.0V 满度值*0.44 = 2.2V 满度值）提供一个 2.2V 的满度信号。同样，增益值 0.55（4.0V 满度值*0.55 = 2.2V 满度值）和增益值 0.73（3.0V 满度值*0.73 = 2.2V 满度值）分别为信号 2 和信号 3 的满度输入提供满度 ADC0 测量。

另外，有些传感器或其它输入源的特性存在器件之间的微小差别，为了获得精确的结果，必须考虑这些差别。在这种情况下，可编程增益值可以被用作校准值，以消除器件之间的这些差别。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

5.4.1 计算增益值

ADC0 的可选择增益功能由 3 个寄存器中的 13 个位控制。ADC0GNH 包含增益值的高 8 位，ADC0GNL 包含增益值的低 4 位。最后一位 GAINADD (ADC0GNA.0) 控制一个可选的额外增益值 1/64 (0.016)，该值可以被加到 ADC0GNH 和 ADC0GNL 增益中。ADC0GNA.0 位在上电复位后被置 1。

由 ADC0GNH、ADC0GNL 和 ADC0GNA 寄存器定义的等效增益为：

$$\text{增益} = \left(\frac{GAIN}{4096} \right) + GAINADD \times \left(\frac{1}{64} \right)$$

方程 5.2 由 ADC0GNH、ADC0GNL 和 ADC0GNA 寄存器定义的等效增益

其中：

GAIN 为 ADC0GNH[7:0]和 ADC0GNL[7:4]组成的 12 位字；

GAINADD 为 GAINADD 位 (ADC0GNA.0) 的值；

增益是取值范围为 0 ~ 1.016 的增益值。

例如，如果 ADC0GNH = 0xFC，ADC0GNL = 0x00，GAINADD = 1，GAIN = 0xFC0 = 4032，则由该方程计算得到的增益值为：

$$\text{增益} = \left(\frac{4032}{4096} \right) + 1 \times \left(\frac{1}{64} \right) = 0.984 + 0.016 = 1.0$$

下表列出了一些 ADC0GNH、ADC0GNL 和 ADC0GNA 寄存器值及使用上述方程计算出等效增益。

ADC0GNH 值	ADC0GNL 值	GAINADD 值	增益值	等效增益
0xFC (默认)	0x00 (默认)	1 (默认)	4032+64	1.0 (默认)
0x7C	0x00	1	1984+64	0.5
0xBC	0x00	1	3008+64	0.75
0x3C	0x00	1	960+64	0.25
0xFF	0xF0	0	4095+0	~1.0
0xFC	0xF0	1	4096+64	1.016

对于任何一个期望的增益值，可以用下面的方程计算 GAIN 寄存器：

$$GAIN = \left(\text{增益} - GAINADD \times \left(\frac{1}{64} \right) \right) \times 4096$$

方程 5.3 由期望增益计算 ADC0GNH 和 ADC0GNL 的值

其中：

GAIN 为 ADC0GNH[7:0]和 ADC0GNL[7:4]组成的 12 位字；

GAINADD 为 GAINADD 位 (ADC0GNA.0) 的值；

增益是取值范围为 0 ~ 1.016 的增益值。

当计算要装载到 ADC0GNH 和 ADC0GNL 寄存器的 GAIN 值时，可以将 GAINADD 位置 1 或清 0，以获得一个接近于期望增益值的数值。

例如，本节开始时的例子需要一个值为 0.44 的增益将 5V 满度值转换为 2.2V 满度值。使用方程 5.3：

$$GAIN = \left(0.44 - GAINADD \times \left(\frac{1}{64} \right) \right) \times 4096$$

如果将 GAINADD 置 1，则：

$$GAIN = \left(0.44 - 1 \times \left(\frac{1}{64} \right) \right) \times 4096 = 0.424 \times 4096 = 1738 = 0x06CA$$

将 GAINADD 位置 1 和将 ADC0GNH 和 ADC0GNL 设置为 0x06CA 所对应的实际增益为 0.4399。将 GAINADD 位清 0 和将 ADC0GNH 和 ADC0GNL 设置为一个不同的值也可以得到一个相近的增益值。

5.4.2 设置增益值

当 GAINEN 位（ADC0CF.0）被置 1 时，可以使用 ADC0H 和 ADC0L 寄存器间接访问三个可编程增益寄存器。此时 ADC0H 作为地址寄存器，ADC0L 作为数据寄存器。可编程增益寄存器只能写入，不能读出。详见增益寄存器定义 5.1、增益寄存器定义 5.2 和增益寄存器定义 5.3。

使用下面的步骤对增益进行编程：

第一步：将 GAINEN 位（ADC0CF.0）置 1；

第二步：向 ADC0H 装载 ADC0GNH、ADC0GNL 或 ADC0GNA 的地址；

第三步：向 ADC0L 装载所选增益寄存器的期望值；

第四步：将 GAINEN 位（ADC0CF.0）清 0。

注：

1. 在 GAINEN 位被置 1 期间，不应进行 ADC 转换。
2. 即使在增益被使能的情况下，最大输入电压也必须小于 V_{REGIN} ，并且在增益级之后的信号电压必须小于或等于 V_{REF} 。

将增益值改为 0.44 的代码如下：

```
// C语言代码
ADC0CF |= 0x01;    // GAINEN = '1'
ADC0H = 0x04;      // 装载ADC0GNH的地址
ADC0L = 0x6C;      // 将0x6CA的高字节装入ADC0GNH
ADC0H = 0x07;      // 装载ADC0GNL的地址
ADC0L = 0xA0;      // 将0x6CA的低4位装入ADC0GNL
ADC0H = 0x08;      // 装载ADC0GNA的地址
ADC0L = 0x01;      // 将GAINADD位置1
ADC0CF &= ~0x01;   // GAINEN = '0'
```

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

; 汇编语言代码

```
ORL ADC0CF, #01H      ; GAINEN = '1'
MOV ADC0H, #04H        ; 装载ADC0GNH的地址
MOV ADC0L, #06CH       ; 将0x6CA的高字节装入ADC0GNH
MOV ADC0H, #07H        ; 装载ADC0GNL的地址
MOV ADC0L, #0A0H       ; 将0x6CA的低4位装入ADC0GNL
MOV ADC0H, #08H        ; 装载ADC0GNA的地址
MOV ADC0L, #01H        ; 将GAINADD位置1
ANL ADC0CF, #0FEH      ; GAINEN = '0'
```

增益寄存器定义 5.1 ADC0GNH: ADC0 可选择增益高字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
GAINH[7:0]								11111100
位7	位6	位5	位4	位3	位2	位1	位0	地址: 0x04
位 7-0: 可选择增益字的高字节。								

增益寄存器定义 5.2 ADC0GNL: ADC0 可选择增益低字节

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
GAINL[3:0]				保留	保留	保留	保留	00000000
位7	位6	位5	位4	位3	位2	位1	位0	地址: 0x07
位 7-4: 可选择增益字的低 4 位。								
位 3-0: 保留。必须写 000b。								

增益寄存器定义 5.3 ADC0GNA: ADC0 附加可选择增益

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
保留	保留	保留	保留	保留	保留	保留	GAINADD	00000001
位7	位6	位5	位4	位3	位2	位1	位0	地址: 0x08
位 7-1: 保留。必须写 0000000b。								
位 0: GAINADD: 附加增益位。								
该位置 1 时, 向 ADC0GNH 和 ADC0GNL 寄存器中的增益值加 1/64 (0.016)。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 5.4 ADC0MX: ADC0 通道选择寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-	-	-	AD0MX					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: AD0MX4-0: AMUX0 输入选择								
AD0MX4-0			ADC0 输入通道					
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*					
11000			温度传感器					
11001			VDD					
11010~11111			GND					

*注：仅适用于 C8051F53x/53xA。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 5.5 ADC0CF: ADC0 配置寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
		AD0SC			AD0RPT		GAINEN	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 FCLK, 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 5.1。 BURSTEN = 0: FCLK 为当前系统时钟。 BURSTEN = 1: FCLK 独立于系统时钟, 最大值为 25 MHz。</p> $AD0SC = \frac{FCLK}{CLK_{SAR}} - 1^* \quad \text{或} \quad CLK_{SAR} = \frac{FCLK}{AD0SC + 1}$ <p>*注: 结果向上取整。</p> <p>位 2-1: AD0RPT1-0: ADC0 重复次数 控制 ADC0 转换结束 (AD0INT) 和 ADC0 窗口比较中断 (AD0WINT) 之间的转换和累加次数。在突发模式未被使能时, 每次转换都需要一次转换启动。在突发模式, 一次转换启动能触发多个自定时的转换。在这两种模式下, 转换结果都被累加到 ADC0H:ADC0L 寄存器。当 AD0RPT1-0 的设置值不为 '00' 时, ADC0CN 寄存器中的 AD0LJST 位必须被清 0 (右对齐)。 00: 执行 1 次转换。 01: 执行 4 次转换转换并累加。 10: 执行 8 次转换转换并累加。 11: 执行 16 次转换转换并累加。</p> <p>位 0: GAINEN: 增益使能位。 控制增益编程。详细信息请参见 “5.4 可选择的增益”。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 5.6 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 数据字高位。</p> <p>对于 AD0LJST = 0 和下面的 AD0RPT 取值:</p> <p>00: 位 3-0 为 12 位结果的高 4 位。位 7-4 为 0000b。</p> <p>01: 位 5-0 为 14 位结果的高 6 位。位 7-6 为 00b。</p> <p>10: 位 6-0 为 15 位结果的高 7 位。位 7 为 0b。</p> <p>11: 位 7-0 为 16 位结果的高 8 位。</p> <p>对于 AD0LJST = 1 (AD0RPT 必须为 '00'): 位 7~0 是 12 位 ADC0 结果的高 8 位。</p>								

SFR 定义 5.7 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 数据字低位。</p> <p>AD0LJST = 0: 位 7~0 是 ADC0 累加结果的低 8 位。</p> <p>AD0LJST = 1 (AD0RPT 必须为 '00'): 位 7~4 是 12 位 ADC0 结果的低 4 位, 位 3~0 为 0000b。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 5.8 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	BURSTEN	AD0INT	AD0BUSY	AD0WINT	AD0LJST	AD0CM1	AD0CM0	00000000
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xE8
<p>位 7: AD0EN: ADC0 使能位 0: ADC0 禁止。ADC0 处于低功耗断电状态。 1: ADC0 使能。ADC0 处于活动状态, 可以进行转换数据。</p>								
<p>位 6: BURSTEN: ADC0 突发模式使能位 0: 突发模式禁止。 1: 突发模式使能。</p>								
<p>位 5: AD0INT: ADC0 转换结束中断标志 0: 自最后一次 AD0INT 清 0 后, ADC0 还没有完成一次数据转换。 1: ADC0 完成了一次转换。</p>								
<p>位 4: AD0BUSY: ADC0 忙标志位 读: 0: ADC0 转换结束或当前不在进行数据转换。AD0INT 在 AD0BUSY 的下降沿被置 1。 1: ADC0 正在进行转换。 写: 0: 无作用。 1: 若 AD0CM1-0 = 00b 则启动 ADC0 转换。</p>								
<p>位 3: AD0WINT: ADC0 窗口比较中断标志 该位必须用软件清 0。 0: 自该标志最后一次被清 0 后, 未发生 ADC0 窗口比较数据匹配。 1: 发生了 ADC0 窗口比较数据匹配。</p>								
<p>位 2: AD0LJST: ADC0 左对齐选择位。 0: ADC0H:ADC0L 中的数据为右对齐。 1: ADC0H:ADC0L 中的数据为左对齐。在重复次数大于 1 时 (AD0RPT 为 01b、10b 或 11b) 不应使用该选项。</p>								
<p>位 1-0: AD0CM1-0: ADC0 转换启动方式选择 00: 每向 AD0BUSY 写 1 时启动 ADC0 转换。 01: 定时器 1 溢出启动 ADC0 转换。 10: 外部 CNVSTR 输入信号的上升沿启动 ADC0 转换。 11: 定时器 2 溢出启动 ADC0 转换。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 5.9 ADC0TK: ADC0 跟踪方式选择寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
AD0PWR				AD0TM		AD0TK		11111111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xBA
<p>位 7-4: AD0PWR3-0: ADC0 突发模式上电时间控制位 BURSTEN = 0: ADC0 电源状态受 AD0EN 控制。 BURSTEN = 1 且 AD0EN = 1: ADC0 保持使能状态, 不会进入低功耗状态。 BURSTEN = 1 且 AD0EN = 0: ADC0 进入低功耗状态 (见表 5.1 和表 5.2), 并在每次转换启动信号有效时被使能。上电时间根据下面的方程编程:</p> $AD0PWR = \frac{\text{启动时间}}{200ns} - 1 \quad \text{或} \quad \text{启动时间} = (AD0PWR + 1) \times 200ns$								
<p>位 3-2: AD0TM1-0: ADC0 跟踪方式选择位 00: 保留。 01: ADC0 配置为后跟踪方式。 10: ADC0 配置为前跟踪方式。 11: ADC0 配置为双跟踪方式 (默认)。</p>								
<p>位 1-0: AD0TK1-0: ADC0 后跟踪时间 AD0TK 对后跟踪时间的控制如下: 00: 后跟踪时间等于 2 个 SAR 时钟周期+2 个 FCLK 周期。 01: 后跟踪时间等于 4 个 SAR 时钟周期+2 个 FCLK 周期。 10: 后跟踪时间等于 8 个 SAR 时钟周期+2 个 FCLK 周期。 11: 后跟踪时间等于 16 个 SAR 时钟周期+2 个 FCLK 周期。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

5.5 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，并在检测到所要求的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。窗口检测器标志既可以在测量数据位于用户编程的极限值以内时有效，也可以在测量数据位于用户编程的极限值以外时有效，这取决于 ADC0GT 和 ADC0LT 寄存器的编程值。

SFR 定义 5.10 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 5.11 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 5.12 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 5.13 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。								

5.5.1 单端方式下的窗口检测器

图 5.7 给出了使用右对齐数据窗口比较的两个例子。左边的例子所使用的极限值为：ADC0LTH:ADC0LTL = 0x0200 (512d) 和 ADC0GTH:ADC0GTL = 0x0100 (256d)；右边的例子所使用的极限值为：ADC0LTH:ADC0LTL = 0x0100 和 ADC0GTH:ADC0GTL = 0x0200。输入电压范围（相对于 GND）是 $0 \sim VREF \times (4095/4096)$ ，转换码为 12 位无符号整数。重复次数设置为 1。对于左边的例子，如果 ADC0 转换字（ADC0H:ADC0L）位于由 ADC0GTH:ADC0GTL 和 ADC0LTH:ADC0LTL 定义的范围之内（即 $0x0100 < \text{ADC0H:ADC0L} < 0x0200$ ），则会产生一个 AD0WINT 中断。对于右边的例子，如果 ADC0 转换结果数据字位于由 ADC0GT 和 ADC0LT 定义的范围之外（即 $\text{ADC0H:ADC0L} < 0x0100$ 或 $\text{ADC0H:ADC0L} > 0x0200$ ），则会产生一个 AD0WINT 中断。图 5.8 给出了使用相同比较值的左对齐数据窗口比较的例子。

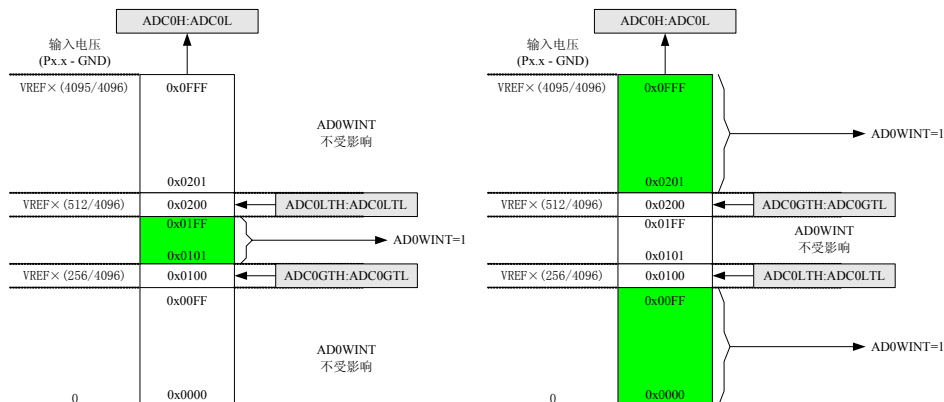


图 5.7 ADC 窗口比较示例（右对齐数据）

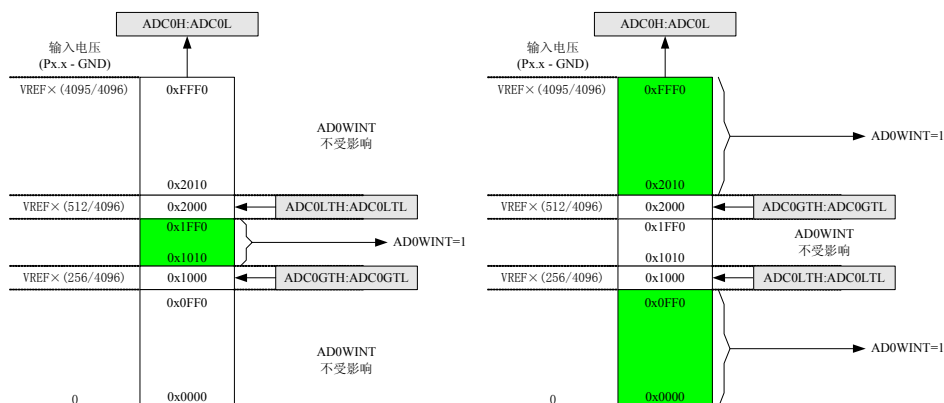


图 5.8 ADC 窗口比较示例（左对齐数据）

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 5.1 ADC0 电气特性

VDD=2.1V, VREF=1.5V(REFSL=0), -40℃到+125℃ (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		12			位
积分非线性		—	—	±1	LSB
微分非线性	保证单调	—	—	±1	LSB
偏移误差		-7	±1	+15	LSB
满度误差		-10	±1	+4	LSB
动态性能 (10kHz 正弦波差分输入, 满度值之下 0~1dB, 200ksps)					
信号与噪声失真比		60	66	—	dB
总谐波失真	到 5 次谐波	—	-74	—	dB
无失真动态范围		—	88	—	dB
转换速率					
SAR 转换时钟		—	—	3	MHz
转换时间 (SAR时钟数) ¹		—	13	—	时钟
跟踪/保持捕获时间 ²		1	—	—	μs
转换速率		—	—	200	ksps
模拟输入					
ADC 输入电压范围	增益 = 1.0 (默认)	0	—	V _{REF}	V
	增益 = n	0	—	V _{REF} /n	
绝对引脚电压 (相对于 GND)		0	—	V _{REGIN}	V
输入电容		—	12	—	pF
温度传感器					
线性度 ⁴		—	0.1	—	℃
增益 ⁴		—	2.84	—	mV/℃
增益误差 ³		—	±100	—	μV/℃
偏移 ⁴	(温度=25℃)	—	890	—	mV
偏移误差 ³	(温度=25℃)	—	±15	—	mV
电源特性					
电源电流 (VDD 给 ADC0 供电)	工作方式, 200ksps	—	880	TBD	μA
突发模式 (空闲)		—	930	—	μA
上电时间		—	5	—	μs
电源抑制比		—	1	—	mV/V
注:					
1. 需要两个额外的 FCLK 周期用于启动和结束转换。					
2. 根据连接到 ADC 输入的信号源输出阻抗不同, 可能需要增加跟踪时间。见“5.3.6 建立时间要求”。					
3. 代表偏离平均值一个标准差。					
4. 包括 ADC 偏移、增益和线性度变化。					

6. 电压基准

C8051F52x/52xA/53x/53xA 的电压基准 MUX 可以被配置为使用连接到外部的电压基准、内部基准电压发生器或电源电压 VDD（见图 6.1）。基准控制寄存器 REF0CN 中的 REFSL 位用于选择基准源。选择驱动 VREF 引脚的外部或内部基准时，REFSL 位应被清 0；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。ADC、温度传感器和内部振荡器都要使用偏置电压发生器提供的偏置电压。当这些外设中的任何一个被使能时，BIASE 位被自动置 1，也可以通过向 REF0CN 中的 BIASE 位写 1 来使能偏置电压发生器，见 SFR 定义 6.1 对 REF0CN 寄存器的详细说明。表 6.1 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温度特性稳定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。可以选择 1.5V 或 2.25V 的输出电压。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（见 SFR 定义 6.1）。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 6.1 给出了电压基准电路的电气特性。

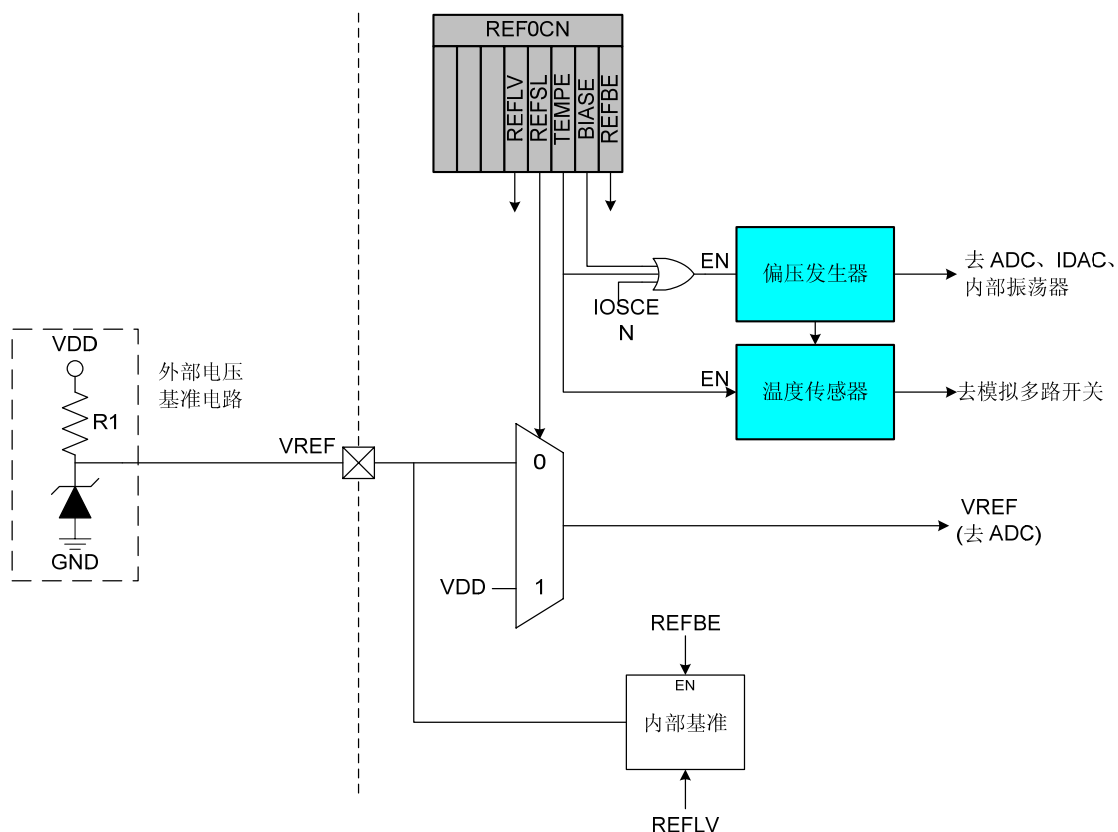


图 6.1 电压基准功能框图

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

注意：端口引脚 P0.0 被用作外部 VREF 输入和内部 VREF 的输出。当使用外部电压基准和内部基准中的任何一个时，P0.0 应被配置为模拟输入并被数字交叉开关跳过。为了将 P0.0 配置为模拟输入，应将 P0MDIN 寄存器的位 0 清 0。为使交叉开关跳过 P0.0，应将 P0SKIP 寄存器的位 0 置 1。有关端口 I/O 配置的详细信息，见“14. 端口输入/输出”。REF0CN 中的 TEMPE 位用于使能/禁止温度传感器。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。

SFR 定义 6.1 REF0CN: 电压基准控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
保留	保留	ZTCEN	REFLV	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 7-6: 保留。读 = 00b。必须写 00b。								
位 5: ZTCEN: 零温度系数偏置使能位								
0: 零温度系数偏置发生器在需要时被自动使能。								
1: 零温度系数偏置发生器被强制使能。								
位 4: REFLV: 电压基准输出电平选择								
该位选择内部电压基准的输出电压。								
0: 内部电压基准设置为 1.5V。								
1: 内部电压基准设置为 2.2V。								
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: 当需要时内部偏压发生器被自动使能。								
1: 内部偏压发生器总是被使能。								
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。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 6.1 电压基准的电气特性

VDD=2.1V, -40℃到+125℃ (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℃ (REFLV=0) 环境温度 25℃ (REFLV=1), VDD =2.6V	1.45 2.15	1.5 2.2	1.55 2.25	V
VREF 短路电流		—	2.5	—	mA
VREF 温度系数		—	33	—	ppm/℃
负载调整	负载 = 0 ~ 200μA 到 GND	—	10	—	ppm/μA
VREF 开启时间 1	4.7μF 钽电容, 0.1μF 陶瓷旁路电容	—	2	—	ms
VREF 开启时间 2	0.1μF 陶瓷旁路电容	—	TBD	—	μs
电源抑制比		—	TBD	—	ppm/V
外部基准 (REFBE=0)					
输入电压范围		0	—	VDD	V
输入电流	采样频率 = 200ksps, VREF=1.5 V	—	35	—	μA
偏压发生器					
ADC 偏压发生器	BIASE = 1	—	30	—	μA
功耗 (内部)		—	35	—	μA

7. 稳压器（REG0）

C8051F52x/52xA/53x/53xA器件内部包含一个低压降稳压器（REG0）。从 V_{REGIN} 引脚输入到REG0的电压可高达5.25V。REG0的输出可以用软件选择为2.1V或2.6V。当被使能时，REG0的输出连接到VDD引脚，为微控制器核供电，并可为外部器件提供电源。复位后REG0被使能，可以用软件禁止。

稳压器的输入（ V_{REGIN} ）和输出（VDD）与地之间都应接一个起保护作用的大电容（ $4.7\mu\text{F}+0.1\mu\text{F}$ ）。该电容能消除电源尖峰并提供微控制器所需要的即时电源。稳压器的建立时间见表 7.1。

稳压器也可以产生中断（如果被EREGR0（EIE1.6）使能），当 V_{REGIN} 输入电压降低到低于压差阈值时该中断被触发。

该电压降落中断没有中断标志，建议按下述步骤使用该中断：

步骤 1：等待足够的时间以保证 V_{REGIN} 输入电压；

步骤 2：使能电压降落中断（EREGR0，EIE1.6），选择合适的优先级（PREGR0，PIE.6）；

步骤 3：如果被触发，在中断服务程序内禁止该中断（清除 EREG0，EIE1.6），执行所有必要的过程以保护您的应用（将其置于一个安全模式，然后退出这个已被禁止的中断）。

步骤 4：在主应用中（现在运行在安全模式），周期性地检查 DROPOUT 位（REG0CN.0）。一旦该位被稳压器硬件清 0，应用可以重新使能中断（EREGR0，EIE1.6）并返回到正常模式。

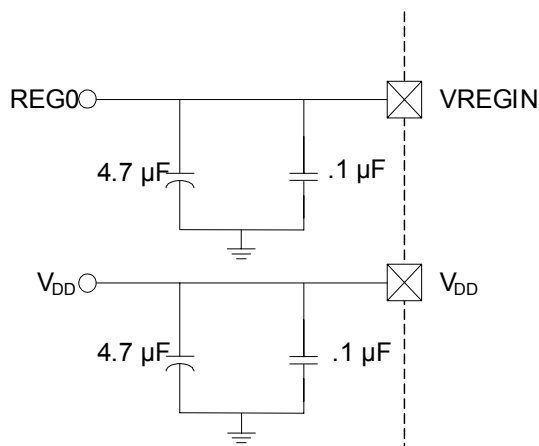


图 7.1 稳压器输入/输出的外部电容

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 7.1 REG0CN: 稳压器控制寄存器

R/W	R/W	R	R/W	R	R	R	R	复位值
REGDIS	保留	—	REG0MD	—	—	—	DROPOUT	00010000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xC9
<p>位 7: REGDIS: 稳压器禁止位 该位禁止/使能稳压器。 0: 稳压器使能。 1: 稳压器禁止。</p> <p>位 6: 保留。读 = 0b, 必须写 0b。</p> <p>位 5: 未用。读 = 0b, 写 = 忽略。</p> <p>位 4: REG0MD: 稳压器方式选择位 该位选择稳压器的输出电压。 0: 稳压器输出为 2.1V。 1: 稳压器输出为 2.6V (默认)。</p> <p>位 3-1: 未用。读 = 000b, 写 = 忽略。</p> <p>位 0: DROPOUT: 稳压器降压指示位 0: 稳压器输出没有电压降落。 1: 稳压器输出处于或接近电压降落状态。</p>								

表 7.1 稳压器电气特性

VDD = 2.1 或 2.6V, -40℃到+125℃ (除非特别说明)。

参 数	条 件	最小值	典型值	最大值	单 位
输入电压范围 (V _{REGIN})	C8051F52x/53x	2.7 ¹	—	5.25	V
	C8051F52xA/53xA	—	—	—	—
	V _{DD} 连接到V _{REGIN} V _{DD} 不连接到V _{REGIN}	1.8 2.2 ²	— —	2.7 5.25	—
电压降落 (V _{DO})	输出电流 = 1~ 50 mA	—	10	TBD	mV/mA
输出电压 (V _{DD})	输出电流 = 1 ~ 50 mA	—	—	—	—
	REG0MD = 0 REG0MD = 1	2.0 2.5	2.1 2.6	2.2 2.7	V
偏置电流	2.1V 工作 (REG0MD = 0)	—	1	TBD	μA
	2.6V 工作 (REG0MD = 1)	—	1	TBD	μA
降落指示器检测阈值		—	65	—	mV
输出电压温度系数		—	2	—	mV/℃
VREG 建立时间	V _{REGIN} = 2.4V, 50mA负载 VDD 负载电容为 4.8μF	—	250	—	μs
注:					
1. 最小输入电压为 2.7V或V _{DD} + V _{DO} (最大负载) 中的较大者。					
2. 最小输入电压为 2.2V或V _{DD} + V _{DO} (最大负载) 中的较大者。					

8. 比较器

C8051F52x/52xA/53x/53xA 器件内部有一个可编程电压比较器，如图 8.1 所示。

比较器的响应时间和回差电压都是可编程的。比较器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，异步 CP0A 信号仍然可用，这就允许比较器在器件处于停机方式或挂起（SUSPEND）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“14.2 端口 I/O 初始化”）。比较器 0 还可以被用作复位源（见“12.5 比较器复位”）。

比较器的输入用 CPT0MX 寄存器（SFR 定义 8.2）来选择。CMX0P3 ~ CMX0P0 位选择比较器 0 的正输入；CMX0N3 ~ 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”。

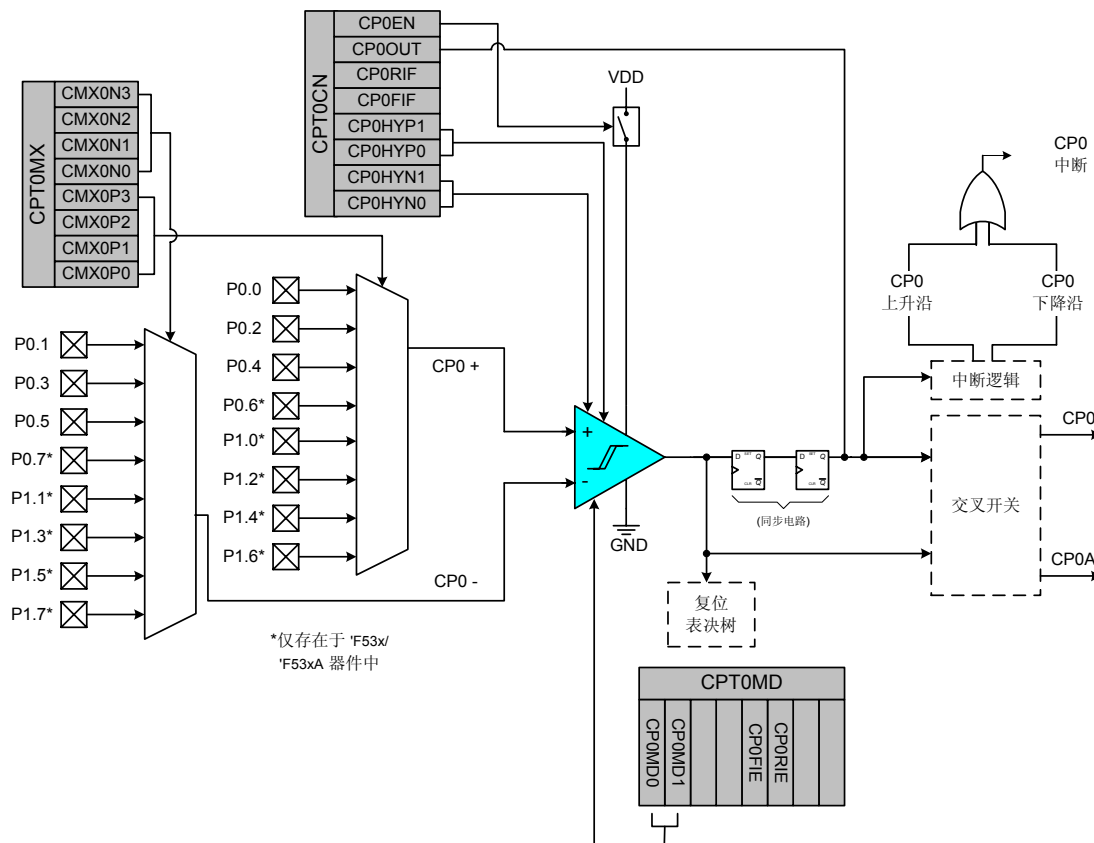


图 8.1 比较器功能框图

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

比较器的输出可以被软件查询，可以作为中断源和内部振荡器挂起唤醒源，还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。即使在停机或挂起方式（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，其电源电流降到小于 100 nA。有关通过交叉开关配置比较器输出的详细信息见“14.1 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 8.1 给出了比较器的电气特性。

可以通过对寄存器 CPT0MD（见 SFR 定义 8.3）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电源电流。8.1 给出了时序和电流消耗特性。

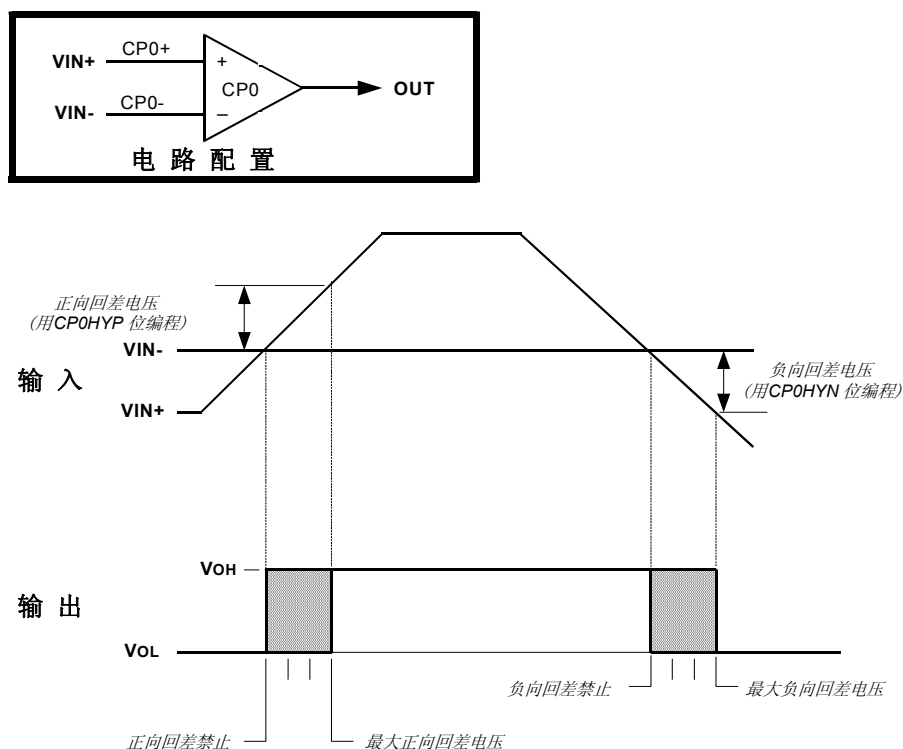


图 8.3 比较器回差电压曲线

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 8.1）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。如表 8.1 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。

比较器输出的上升沿和下降沿都可以产生中断。比较器的下降沿置位 CP0FIF 中断标志，比较器的上升沿置位 CP0RIF 中断标志。这些位一旦被置 1，将一直保持 1 状态直到被软件清 0。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

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

允许比较器下降沿中断。

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。当比较器被使能时，如果比较器输出为逻辑 0，则内部振荡器被从 SUSPEND 模式唤醒。

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，可能会检测到假上升沿或下降沿。建议在比较器被使能或方式位改变后经过一段延时将上升沿和下降沿标志显式清 0。表 8.1 中给出了上电时间。

SFR 定义 8.1 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: 自该标志位最后一次被清除后，未检测到比较器 0 上升沿。							
	1: 检测到比较器 0 上升沿。							
位 4:	CP0FIF: 比较器 0 下降沿中断标志，必须用软件清 0							
	0: 自该标志位最后一次被清除后，未检测到比较器 0 下降沿。							
	1: 检测到比较器 0 下降沿。							
位 3-2:	CP0HYP1-0: 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:	CP0HYN1-0: 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。							

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 8.2 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	01110111
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x9F

位 7-4: CMX0N3-0: 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。

CMX0N3	CMX0N2	CMX0N1	CMX0N0	负输入
0	0	0	0	P0.1
0	0	0	1	P0.3
0	0	1	0	P0.5
0	0	1	1	P0.7*
0	1	0	0	P1.1*
0	1	0	1	P1.3*
0	1	1	0	P1.5*
0	1	1	1	P1.7*

*注: 仅存在于 C8051F53x/53xA 器件中。

位 3-0: CMX0P3-0: 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。

CMX0P3	CMX0P2	CMX0P1	CMX0P0	正输入
0	0	0	0	P0.0
0	0	0	1	P0.2
0	0	1	0	P0.4
0	0	1	1	P0.6*
0	1	0	0	P1.0*
0	1	0	1	P1.2*
0	1	1	0	P1.4*
0	1	1	1	P1.6*

*注: 仅存在于 C8051F53x/53xA 器件中。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 8.3 CPT0MD: 比较器 0 方式选择寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
保留	-	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: 保留。读 = 0b, 必须写 0。

位 6: 未用。读 = 0b, 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

注: 需要同时使能 CP0xIE 和位于 EIE1 SFR 中的 ECPx 位。

位 3-2: 未用。读 = 00b, 写 = 忽略。

位 1-0: CP0MD1-0: 比较器 0 方式选择
这两位选择比较器 0 的响应时间。

方式	CP0MD1	CP0MD0	CP0 下降沿响应时间 (典型值)
0	0	0	最快响应时间
1	0	1	—
2	1	0	—
3	1	1	最低功耗

注: 上升沿响应时间约为下降沿响应时间的两倍。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 8.1 比较器电气特性

VDD = 2.1V, -40°C ~ +125°C (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, $V_{cm}^1=1.5V$	(CP0+) - (CP0-) = 100mV	—	780	—	ns
	(CP0+) - (CP0-) = -100mV	—	980	—	ns
响应时间: 方式 1, $V_{cm}^1=1.5V$	(CP0+) - (CP0-) = 100mV	—	850	—	ns
	(CP0+) - (CP0-) = -100mV	—	1120	—	ns
响应时间: 方式 2, $V_{cm}^1=1.5V$	(CP0+) - (CP0-) = 100mV	—	870	—	ns
	(CP0+) - (CP0-) = -100mV	—	1310	—	ns
响应时间: 方式 3, $V_{cm}^1=1.5V$	(CP0+) - (CP0-) = 100mV	—	1980	—	ns
	(CP0+) - (CP0-) = -100mV	—	4770	—	ns
共模抑制比		—	1.5	TBD	mV/V
正向回差电压 1	CP0HYP1-0 = 00	—	0.7	2	mV
正向回差电压 2	CP0HYP1-0 = 01	2	5	10	mV
正向回差电压 3	CP0HYP1-0 = 10	5	10	20	mV
正向回差电压 4	CP0HYP1-0 = 11	13	20	40	mV
负向回差电压 1	CP0HYN1-0 = 00	—	0.7	2	mV
负向回差电压 2	CP0HYN1-0 = 01	2	5	10	mV
负向回差电压 3	CP0HYN1-0 = 10	5	10	20	mV
负向回差电压 4	CP0HYN1-0 = 11	13	20	40	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		—	0.5	—	nA
输入偏移电压		-10	—	+10	mV
输入阻抗		—	TBD	—	K Ω
电源					
电源抑制比 ²		—	0.2	4	mV/V
上电时间		—	2.3	—	μ s
电源电流 (DC)	方式 0	—	13	TBD	μ A
	方式 1	—	6	TBD	μ A
	方式 2	—	3	TBD	μ A
	方式 3	—	1	TBD	μ A
注: <ol style="list-style-type: none"> 1. V_{cm} 是 CP0+和 CP0-上的共模电压。 2. 由设计和/或特性测试保证。 					

9. 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。C8051F52x/52xA/53x/53xA的外设是标准 8051 的所有外设的超集。C8051F52x/52xA/53x/53xA具有的外设部件见“1. 系统概述”。CIP-51 片内还包含与模拟和数字子系统直接接口的调试硬件，在一片集成电路内提供了完全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 9.1 的原理框图）。CIP-51 核具有下列特点：

- 与 MCS-51 指令集完全兼容
- 峰值速度为 25MIPS
- 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
- 集成的调试逻辑
- 程序和数据存储器安全

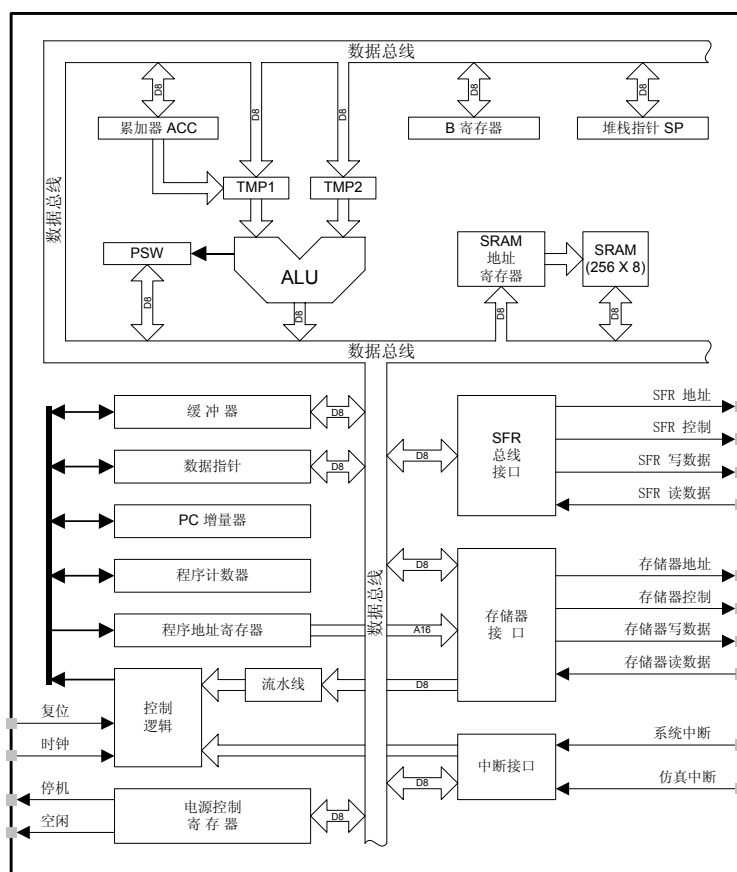


图 9.1 CIP-51 原理框图

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

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。注意，可以在用户软件中使用 MOVC 和 MOVX 指令对可再编程的 FLASH 读和写，每次读或写一个字节。这一特性允许将程序存储器用于非易失性数据存储以及在软件控制下更新代码。

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，不需要额外的目标 RAM、堆栈、定时器或其它片内资源。

CIP-51 有 Silicon Labs 公司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、评估版编译器、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，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。

9.1 指令集

CIP-51 系统控制器的指令集与标准MCS-51™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-51™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。

9.1.1 指令和 CPU 时序

在很多的 8051 产品中，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 2 个。表 9.1 给出了 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。

9.1.2 MOVX 指令和程序存储器

MOVX 指令一般用于访问外部数据存储器空间的数据。在 CIP-51 中，MOVX 指令还可用于写或擦除可重编程的片内 FLASH 程序存储器。这一特性为 CIP-51 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“13. FLASH 存储器”。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 9.1 CIP-51 指令集一览表¹

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

C8051F52x/52xA/53x/53xA 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

C8051F52x/52xA/53x/53xA 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-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 位立即数

#data16 – 16 位立即数

bit – 数据 RAM 或 SFR 中的直接寻址位

addr11 – ACALL 或 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 或 LJMP 使用的 16 位目的地址。目的地址可以是 7680 字节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。

9.2 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 9.1 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>								

SFR 定义 9.2 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 用于访问间接寻址的 XRAM 和 FLASH 存储器。</p>								

SFR 定义 9.3 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 XRAM 和 FLASH 存储器。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 9.4 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。
该两位选择访问寄存器时所使用的寄存器区。

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。
累加器中 8 个位的和为奇数时该位被置 1，为偶数时被清 0。

SFR 定义 9.5 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 9.6 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>								

9.3 电源管理方式

CIP-51 有两种可软件编程的电源管理方式：空闲和停机。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）。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。停机方式消耗最少的功率。SFR 定义 9.7 对用于控制 CIP-51 电源管理方式的电源控制寄存器（PCON）作出了说明。

虽然 CIP-51 具有空闲和停机方式（与任何标准 8051 结构一样），但通过使能和禁止外设，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功耗，但需要复位来重新启动 MCU。

C8051F52x/52xA/53x/53xA 器件有一个低功耗的挂起（SUSPEND）方式，该方式停止内部振荡器，直到有唤醒时间发生。详见“15.1.1 内部振荡器挂起方式”。

9.3.1 空闲方式

将空闲方式选择位（PCON.0）置 1 使 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。

如果被使能，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。如果不需要这种功能，可以在进入空闲方式之前用软件禁止 WDT（如果 WDT 被初始化为允许该操作）。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。

9.3.2 停机方式

将停机方式选择位（PCON.1）置 1 使 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。

9.3.3 挂起方式

C8051F52x/52xA/53x/53xA 器件有一个低功耗的挂起（SUSPEND）方式，该方式停止内部振荡器，直到有唤醒时间发生。详见“15.1.1 内部振荡器挂起方式”。

SFR 定义 9.7 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
保留	保留	保留	保留	保留	保留	STOP	IDLE	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0x87
<p>位 7-2： 保留。</p>								
<p>位 1： STOP： 停机方式选择。</p>								
<p>将该位置 1 使 CIP-51 进入停机方式。该位的读出值总是为 0。</p>								
<p>1： 进入停机方式（内部振荡器停止运行）</p>								
<p>位 0： IDLE： 空闲方式选择。</p>								
<p>将该位置 1 使 CIP-51 进入空闲方式。该位读出值总是为 0。</p>								
<p>1： CPU 进入空闲方式。（断开供给 CPU 的时钟信号，但供给定时器、中断和所有外设的时钟保持活动状态。）</p>								

10. 存储器组织和 SFR

C8051F52x/52xA/53x/53xA 的存储器组织与标准 8051 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。C8051F52x/52xA/53x/53xA 的存储器组织如图 10.1 所示。

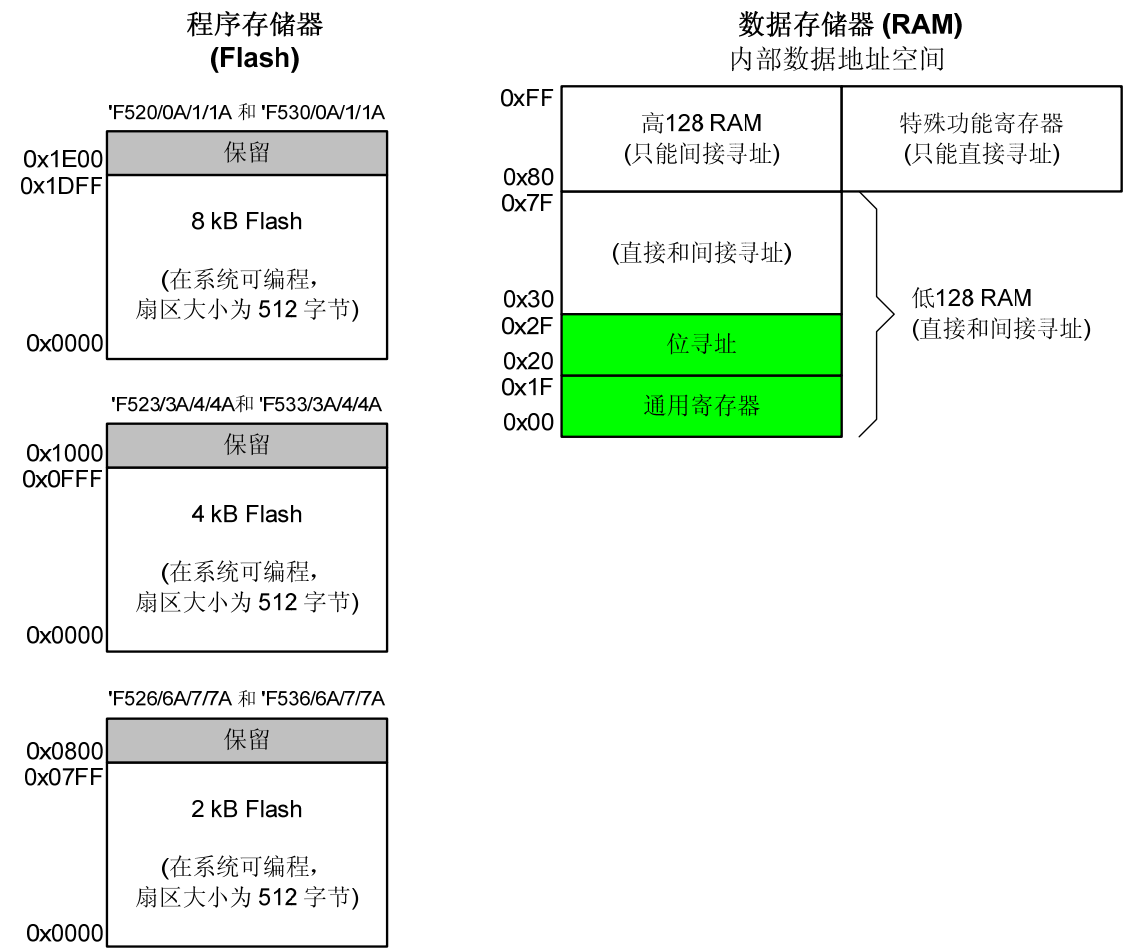


图 10.1 存储器组织图

10.1 程序存储器

CIP-51 有 64KB 的程序存储器空间。C8051F520/0A/1/1A 和 C8051F530/0A/1/1A 在这个程序存储器空间中实现了 8KB 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 – 0x1FFF）。在 8KB 器件中，0x1DFF 以上的地址保留。C8051F523/3A/4/4A 和 C8051F533/3A/4/4A 实现了 4KB 的 FLASH 存储器，地址范围为 0x0000 ~ 0x0FFF。C8051F526/6A/7/7A 和 C8051F536/6A/7/7A 实现了 2KB 的 FLASH 存储器，地址范围为 0x0000 ~ 0x07FF。

程序存储器通常被认为是只读的。但 C8051F52x/52xA/53x/53xA 可以通过设置程序存储

写允许位 (PSCTL.0) 并用 MOVX 写指令对程序存储器写入。这一特性为 CIP-51 提供了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。更详细的信息见“13. FLASH 存储器”。

10.2 数据存储器

C8051F52x/52xA/53x/53xA 的数据存储器空间中有 256 字节的内部 RAM，位于 0x00 ~ 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的低 128 字节。0x00 ~ 0x1F 可作为 4 个通用寄存器区来访问，每个区有 8 个 8 位寄存器。接下来的 16 字节，地址 0x20 ~ 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 10.1 示出了 C8051F52x/52xA/53x/53xA 的数据存储器组织。

10.3 通用寄存器

数据存储器的低 32 字节 (地址 0x00 ~ 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 9.4 对 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。

10.4 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-51™ 汇编语言允许用 XX.B 的形式替代位地址，XX 为字节地址，B 为寻址位在该字节中的位置。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。

10.5 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。

10.6 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 CIP-51 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 10.1 列出了 CIP-51 系统控制器中的全部 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、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 10.2 中已标明)。

表 10.1 特殊功能寄存器 (SFR) 存储器映象

F8	SPI0CN	PCA0L	PCA0H	PCA0CPL0	PCA0CPH0			VDDMON
F0	B	P0MDIN	P1MDIN				EIP1	
E8	ADC0CN	PCA0CPL1	PCA0CPH1	PCA0CPL2	PCA0CPH2			RSTSRC
E0	ACC	XBR0	XBR1		IT01CF		EIE1	
D8	PCA0CN	PCA0MD	PCA0CPM0	PCA0CPM1	PCA0CPM2			
D0	PSW	REF0CN			P0SKIP	P1SKIP		P0MAT
C8	TMR2CN	REG0CN	TMR2RLL	TMR2RLH	TMR2L	TMR2H		P1MAT
C0				ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH	P0MASK
B8	IP		ADC0TK	ADC0MX	ADC0CF	ADC0L	ADC0H	P1MASK
B0	OSCIFIN	OSCXCN	OSCICN	OSCICL				FLKEY
A8	IE	CLKSEL						
A0		SPI0CFG	SPI0CKR	SPI0DAT	P0MDOUT	P1MDOUT		
98	SCON0	SBUF0		CPT0CN		CPT0MD		CPT0MX
90	P1		LINADR	LINDATA		LINCF		
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)

表 10.2 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	
ADC0H	0xBE	ADC0 数据字高字节	
ADC0L	0xBD	ADC0 数据字低字节	
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	
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	
ADC0MX	0xBB	ADC0 通道选择寄存器	
ADC0TK	0xBA	ADC0 跟踪方式选择寄存器	
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	
FLKEY	0xB7	FLASH 锁定和关键码寄存器	
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	
LINADDR	0x92	LIN 间接地址指针	
LINCF	0x95	LIN 主-从和自动波特率选择	
LINDATA	0x93	LIN 间接数据缓冲器	
OSCICL	0xB3	内部振荡器校准寄存器	
OSCICN	0xB2	内部振荡器控制寄存器	
OSXCXCN	0xB1	外部振荡器控制寄存器	
P0	0x80	端口 0 锁存器	
P0MASK	0xC7	端口 0 屏蔽寄存器	
P0MAT	0xD7	端口 0 匹配寄存器	
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	

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 10.2 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	0xBF	端口 1 屏蔽寄存器	
P1MAT	0xCF	端口 1 匹配寄存器	
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	
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	
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	
REG0CN	0xC9	稳压器控制寄存器	
RSTSRC	0xEF	复位源配置/状态寄存器	
SBUF0	0x99	UART0 数据缓冲器	
SCON0	0x98	UART0 控制寄存器	
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	
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	

表 10.2 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
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	
VDDMON	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	

11. 中断系统

C8051F52x/52xA/53x/53xA 包含一个扩展的中断系统，有两个可选择的优先级。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中断请求。一旦当前指令执行结束，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中断允许或扩展中断允许寄存器中的使能位来独立地允许或禁止，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，清除 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。

某些中断标志在 CPU 进入 ISR 时被自动清除，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。

11.1 MCU 中断源和中断向量

C8051F52x/52xA/53x/53xA MCU 支持 15 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 11.1 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。

11.2 中断优先级

每个中断源都可以被独立地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 11.1)。

11.3 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请求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 个周期用于完成对 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

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

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。

表 11.1 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)
ADC0 窗口比较	0x003B	7	AD0WINT (ADC0CN.3)	Y	N	EWADC0 (EIE1.0)	PWADC0 (EIP1.0)
ADC0 转换结束	0x0043	8	AD0INT (ADC0CN.5)	Y	N	EADC0 (EIE1.1)	PADC0 (EIP1.1)
可编程计数器阵列	0x004B	9	CF (PCA0CN.7) CCFn (PCA0CN.n)	Y	N	EPCA0 (EIE1.2)	PPCA0 (EIP1.2)
比较器 0 下降沿	0x0053	10	CP0FIF(CPT0CN.4)	N	N	ECPF (EIE1.3)	PCPF (EIP1.3)
比较器 0 上升沿	0x005B	11	CP0RIF(CPT0CN.5)	N	N	ECPR (EIE1.4)	PCPR (EIP1.4)
LIN 中断	0x0063	12	LININT(LINST.3)	N	N*	ELIN (EIE1.5)	PLIN (EIP1.5)
稳压器电压降落	0x006B	13	N/A	N/A	N/A	EREG0 (EIE1.6)	PREG0 (EIP1.6)
端口匹配	0x0073	14	N/A	N/A	N/A	EMAT (EIE1.7)	PMAT (EIP1.7)
*注：为了清除 LININT 标志，软件需要将 RSTINT 位（LINCTRL.3）置 1。							

11.4 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 11.1 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>								

SFR 定义 11.2 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
-	PSP10	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:	PSP10: 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。							

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 11.3 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
EMAT	EREG0	ELIN	ECPR	ECPF	EPCA0	EADC0	EWADC0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xE6
位 7:	EMAT: 端口匹配中断允许位 该位设置端口匹配中断屏蔽。 0: 禁止端口匹配中断。 1: 允许端口匹配中断。							
位 6:	EREG0: 稳压器中断允许位 该位设置稳压器电压降落中断屏蔽。 0: 禁止稳压器电压降落中断。 1: 允许稳压器电压降落中断。							
位 5:	ELIN: LIN 中断允许位 该位设置 LIN 的中断屏蔽。 0: 禁止 LIN 中断。 1: 允许 LIN 中断。							
位 4:	ECPR: 比较器 0 上升沿中断允许位 该位设置 CP0 上升沿的中断屏蔽。 0: 禁止 CP0 上升沿中断。 1: 允许 CP0 上升沿中断。							
位 3:	ECPF: 比较器 0 下降沿中断允许位 该位设置 CP0 下降沿的中断屏蔽。 0: 禁止 CP0 下降沿中断。 1: 允许 CP0 下降沿中断。							
位 2:	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 1:	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 0:	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。							

SFR 定义 11.4 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
PMAT	PREG0	PLIN	PCPR	PCPF	PPCA0	PADC0	PWADC0	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xF6
位 7:	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 6:	PREG0: 稳压器中断优先级控制 该位设置稳压器电压降落中断的优先级。 0: 稳压器中断为低优先级。 1: 稳压器中断为高优先级。							
位 5:	PLIN: LIN 中断优先级控制 该位设置 LIN 中断的优先级。 0: LIN 中断为低优先级。 1: LIN 中断为高优先级。							
位 4:	PCPR: 比较器上升沿中断优先级控制 该位设置比较器上升沿中断的优先级。 0: 比较器上升沿中断为低优先级。 1: 比较器上升沿中断为高优先级。							
位 3:	PCPF: 比较器下降沿中断优先级控制 该位设置比较器下降沿中断的优先级。 0: 比较器下降沿中断为低优先级。 1: 比较器下降沿中断为高优先级。							
位 2:	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 1:	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 0:	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 窗口比较中断为低优先级。 1: ADC0 窗口比较中断为高优先级。							

11.5 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。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 11.5)。注意,/INT0 和/INT1 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“14.1 优先权交叉开关译码器”）。

在典型配置中，外部中断引脚应被交叉开关跳过，并被配置为漏极开路，且引脚锁存器被置 1。详见“14. 端口输入/输出”。

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。

SFR 定义 11.5 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*

*注：仅存在于 C8051F53x/53xA 器件中。

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*

*注：仅存在于 C8051F53x/53xA 器件中。

12. 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“15. 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（有关使用看门狗定时器的详细信息见“20.3 看门狗定时器方式”）。程序从地址 0x0000 开始执行。

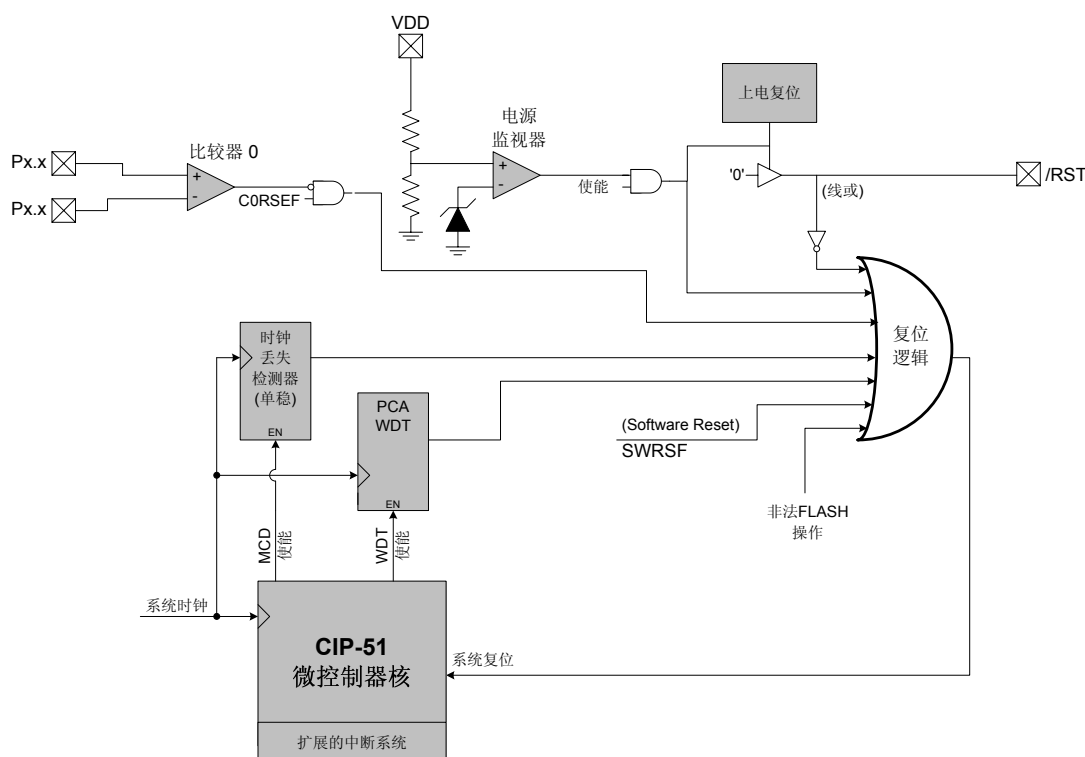


图 12.1 复位源框图

12.1 上电复位

在上电期间，器件保持在复位状态， $\overline{\text{RST}}$ 引脚被驱动到低电平，直到VDD上升到超过 V_{RST} 电平。从复位开始到退出复位状态要经过一个延时；该延时随着VDD上升时间的增大而减小（VDD上升时间被定义为VDD从0V上升到 V_{RST} 的时间）。图12.2给出了上电和VDD监视器复位的时序。对于有效的上升时间（小于1ms），上电复位延时（ T_{PORDelay} ）通常小于0.3ms。

注：最大的VDD上升时间为1ms；上升时间超过该最大值时可能导致器件在VDD达到 V_{RST} 电平之前退出复位状态。

在退出复位状态时，PORSF标志（RSTSRC.1）被硬件置为逻辑1。当PORSF标志被置位时，RSTSRC寄存器中的所有其它复位标志都是不确定的。PORSF被任何其它复位源清0。由于所有的复位都导致程序从同一个地址（0x0000）开始执行，软件可以通过读PORSF标志来确定是否为上电产生的复位。在一次上电复位后，内部数据存储器中的内容应被认为是不确定的。在上电复位后，VDD监视器被禁止。

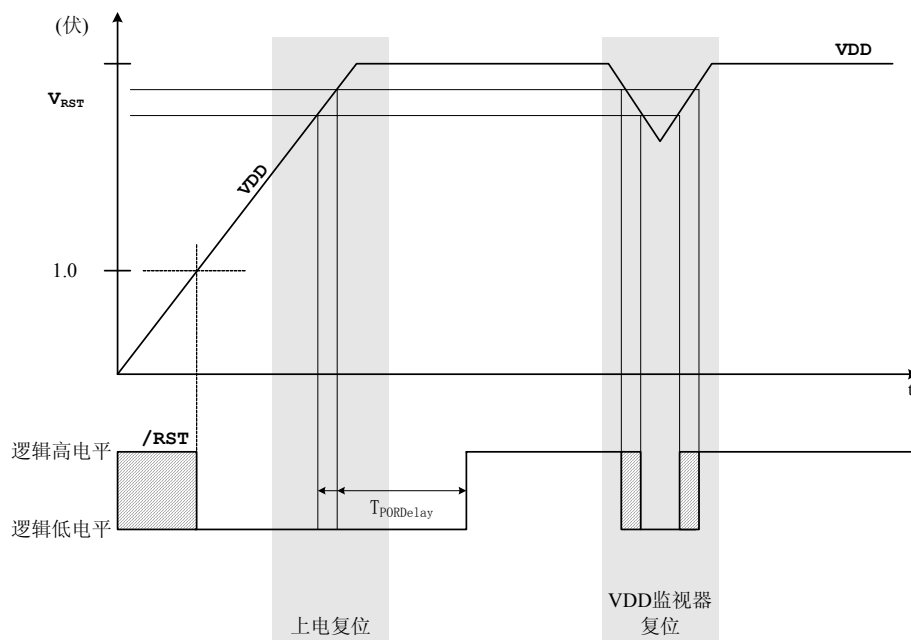


图 12.2 上电和 VDD 监视器复位时序

12.2 掉电复位和 VDD 监视器

当VDD监视器被选择为复位源并且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/RST引脚驱动为低电平并使CIP-51保持复位状态（见图12.2）。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，未被选择为复位源，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监视器被软件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。**如果软件包含擦除或写FLASH存储器的例程，为了保护FLASH内容的完整性，必须将VDD监视器使能为较高的电平设置（VDMLVL = 1）并将其选择为复位源。如果VDD监视器未被使能，对FLASH存储器执行任何擦除或写操作都将导致FLASH错误器件复位。**

在选择VDD监视器作为复位源之前，必须先使能VDD监视器。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。重新使能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（VDDMON中的VDMEN位 = 1）；
- 2) 等待VDD监视器稳定（见表12.1的VDD监视器开启时间）；
注：如果软件中包含擦除或写FLASH存储器的程序，则该延时应被省略。
- 3) 选择VDD监视器作为复位源（RSTSRC中的PORSF位 = 1）。

图12.2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监视器的电气特性见表12.1。

注意：当写RSTSRC以使能其他复位源或触发一次软件复位时，软件操作应谨慎，以防止意外禁止VDD监视器作为复位源。所有写RSTSRC的操作都应显式地将PORSF置1，以保持VDD监视器被使能为复位源。

SFR 定义 12.1 VDDMON: VDD 监视器控制寄存器

R/W	R	R/W	R	R	R	R	R	复位值
VDMEN	VDDSTAT	VDMLVL	保留	保留	保留	保留	保留	1v000000
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。 VDD 监视器的最小启动时间见表 12.1。 0: 禁止 VDD 监视器。 1: 使能 VDD 监视器 (缺省)。</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: VDMLVL: VDD 电平选择。 0: VDD监视器阈值被设置为V_{RST-LOW} (默认)。 1: VDD监视器阈值被设置为V_{RST-HIGH}。在任何包含写和/或擦除FLASH代码的系统中, 需要使用该设置。</p> <p>位 4-0: 保留。读 = 可变, 写 = 忽略。</p>								

12.3 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产生复位, 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。

12.4 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 MCD 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 使能时钟丢失检测器; 写 0 禁止时钟丢失检测器。/RST 引脚的状态不受该复位的影响。

12.5 比较器复位

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-), 则器件被置于复位状态。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。

12.6 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（见“20.3 看门狗定时器方式”）。在每次复位后，WDT 被使能并使用 SYSCLK/12 作为时钟。如果因系统出错使用户软件不能更新 WDT，则 WDT 将产生复位，WDTRSF 位（RSTSRC.5）被置 1。/RST 引脚的状态不受该复位的影响。

12.7 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 操作错误复位：

- 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 写操作的地址大于锁定字节地址时。
- FLASH 读地址超出了用户代码空间，即 MOVC 操作的地址大于锁定字节地址。
- 程序读超出了用户代码地址空间。这种情况发生在用户代码试图转移到大于锁定字节地址地址时。
- 当 FLASH 读、写或擦除被安全设置禁止时（见“13.4 安全选项”）。
- 当 VDD 监视器被禁止时，试图进行 FLASH 写或擦除操作。

在发生 FLASH 错误复位后，FERROR 位（RSTSRC.6）被置 1。/RST 引脚的状态不受该复位的影响。

12.8 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 12.2 RSTSRC: 复位源寄存器

R/W	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

注：当向 RSTSRC 写值时，软件应避免使用读-修改-写指令。

位 7: 未用。读 = 1b, 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
0: 读: 最后一次复位不是来自比较器 0。
写: 比较器 0 不是复位源。
1: 读: 最后一次复位来自比较器 0。
写: 比较器 0 是复位源 (低电平有效)。

位 4: SWRSF: 软件强制复位和标志
0: 读: 最后一次复位不是来自写 SWRSF 位。
写: 无作用。
1: 读: 最后一次复位来自写 SWRSF 位。
写: 强制产生一次系统复位。

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
0: 读: 最后一次复位不是来自时钟丢失检测器超时。
写: 禁止时钟丢失检测器。
1: 读: 最后一次复位来自时钟丢失检测器超时。
写: 使能时钟丢失检测器; 检测到时钟丢失条件时触发复位。

位 1: PORSF: 上电复位强制和标志
该位在上电复位后被置 1。对该位写入可以使能/禁止 VDD 监视器作为复位源。
注意: 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 VDDMON (SFR 定义 12.1)。
0: 读: 最后一次复位不是上电复位或 VDD 监视器复位。
写: 禁止 VDD 监视器为复位源。
1: 读: 最后一次复位是上电或 VDD 监视器复位, 所有其它复位标志不确定。
写: 使能 VDD 监视器为复位源。

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。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 12.1 复位源电气特性

-40℃到+125℃（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	$I_{OL} = 8.5 \text{ mA}$ $V_{DD} = 2.1 \text{ V}$	—	—	0.8	V
/RST 输入高电平		$0.7 \times V_{REGIN}$	—	—	V
/RST 输入低电平		—	—	$0.3 \times V_{REGIN}$	V
/RST 输入上拉阻抗	$V_{REGIN} = 3.3 \text{ V}$	—	126	—	K Ω
/RST输入上拉 V_{REGIN} 敏感度 ¹		—	-34.8	—	K Ω /V
V_{DD} 监视器门限($V_{RST-LOW}$)	C8051F52x/53x C8051F52xA/53xA	1.8 1.7	1.9 1.75	2.0 1.8	V
V_{DD} 监视器门限($V_{RST-HIGH}$)	C8051F52x/53x C8051F52xA/53xA	2.1 2.25	2.2 2.3	2.3 2.4	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， $V_{DD} = 2.5 \text{ V}$	TBD	350	650	μs
复位时间延迟 ²	从退出复位到开始执行位于 0x0000 地址的代码之间的延时	—	—	350	μs
产生系统复位的最小 /RST 低电平时间		TBD	—	—	μs
V_{DD} 监视器开启时间		—	TBD	—	μs
V_{DD} 监视器电源电流	$V_{DD} = 2.1 \text{ V}$	—	23	TBD	μA
注： 1. /RST输入上拉阻抗可以用 V_{REGIN} 为 3.3V时的阻抗减去由敏感度指示的阻抗差值。例如： $V_{REGIN} = 5 \text{ V}$ ，输入上拉阻抗 = $126 \text{ K}\Omega - (3.3 \text{ V} - 5 \text{ V}) * -34.8 \text{ K}\Omega/\text{V} = 60 \text{ K}\Omega$ 。 2. 参见第 21 章。					

13. FLASH 存储器

C8051F52x/52xA/53x/53xA 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 写指令对 FLASH 存储器进行在系统编程。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，程序暂停执行。表 13.2 给出了 FLASH 存储器的电气特性。

13.1 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“22. C2 接口”。

任何系统中如果软件包含擦除或写 FLASH 存储器的代码，为了保证 FLASH 内容的完整性，必须将 VDD 监视器使能为较高的电平设置（VDMVLV = 1），并随后立即将其选择为复位源。在 VDD 监视器被禁止期间，对 FLASH 存储器执行任何擦除或写操作都将导致 FLASH 错误器件复位。

在选择 VDD 监视器作为复位源之前，必须先使能 VDD 监视器。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。重新使能 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) 等待 VDD 监视器稳定（见表 12.1 的 VDD 监视器开启时间）；
注：如果软件中包含擦除或写 FLASH 存储器的程序，则该延时应被省略。
- 3) 选择 VDD 监视器作为复位源（RSTSRC 中的 PORSF 位 = 1）。

13.1.1 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 13.2。

13.1.2 FLASH 擦除过程

可以用软件使用 MOVX 写指令对 FLASH 存储器编程，像一般的操作数一样为 MOVX 写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字节地址。FLASH 存储器是以 512 字节的页为单位组织的，一次擦除操作将擦除整个页 (将页内的所有字节置为 0xFF)。擦除一个页的步骤如下：

1. 禁止中断 (建议这样做)。
2. 向 FLKEY 写第一个关键码：0xA5。
3. 向 FLKEY 写第二个关键码：0xF1。
4. 将 PSEE 位置 1 (寄存器 PSCTL)。
5. 将 PSWE 位置 1 (寄存器 PSCTL)。
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。
8. 重新使能中断。

13.1.3 FLASH 写过程

软件写 FLASH 字节的建议步骤如下：

1. 禁止中断。
2. 向 FLKEY 写第一个关键码：0xA5。
3. 向 FLKEY 写第二个关键码：0xF1。
4. 将 PSWE 位 (寄存器 PSCTL) 置 1。
5. 清除 PSEE 位 (寄存器 PSCTL)。
6. 用 MOVX 指令向扇区内的目标地址写入一个数据字节。
7. 清除 PSWE 位。
8. 重新使能中断。

重复步骤 2-7，直到写完每个字节。在 FLASH 写操作结束后，PSWE 应被清 0，以使 MOVX 指令的目标不再是程序存储器。

13.2 FLASH 写和擦除指南

如果 CPU 工作在 VDD、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 的代码。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。

13.2.1 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，确保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 1.8V 之前保持复位状态和在 VDD 下降到低于 1.8V 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 监视器和将其使能为复位源。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。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”可以快速完成检查。

13.2.2 PSWE 维护

1. 在代码中尽量减少将 PSWE 位 (PSCCTL 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)。
2. 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。
3. 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。
4. 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。
5. 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。

13.2.3 系统时钟

1. 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。
2. 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”。

13.3 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和存储类似标定系数这样的数据。数据写入时用 MOVX 写指令, 读出时用 MOVC 指令。注意: MOVX 读指令总是指向 XRAM。

13.4 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，以及防止产权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 位）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 1；在用软件擦除 FLASH 存储器之前，PSWE 位和 PSEE 位都必须被置 1。此外，CIP-51 还提供了可以防止通过 C2 接口读取或修改产权程序代码和数据常数这一安全功能。

位于 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 安全锁定字节的页不被锁定。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。**见下面的例子。

安全锁定字节： 11111101 b
反码： 00000010 b

被锁定的 FLASH 页： 3（前两个 FLASH 页 + 锁定字节页）
被锁定的地址： 0x0000 ~ 0x03FF（前两个 FLASH 页）和
0x1C00 ~ 0x1DFF（C8051F520/0A/1/1A 和 C8051F530/0A/1/1A）
0x0E00 ~ 0x0FFF（C8051F523/3A/4/4A 和 C8051F533/3A/4/4A）
0x0600 ~ 0x07FF（C8051F526/6A/7/7A 和 C8051F536/6A/7/7A）

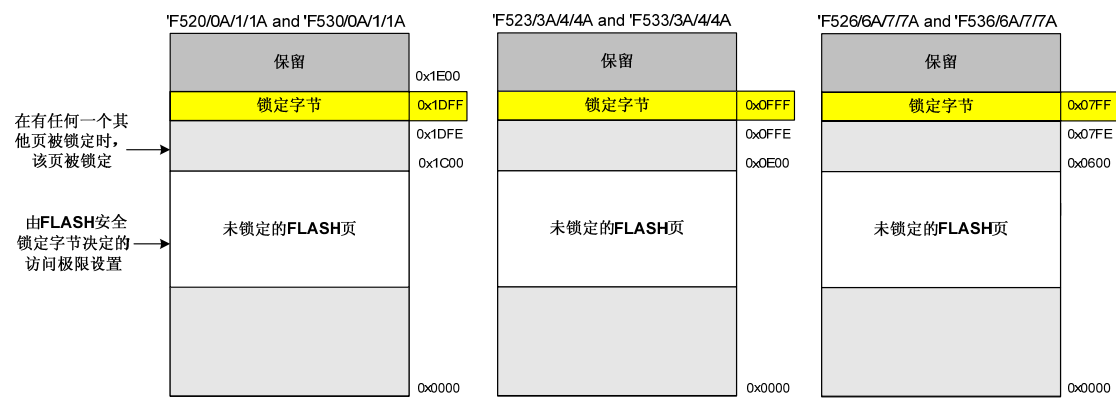


图 13.1 FLASH 程序存储器组织

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、在非锁定页执行的用户固件、在锁定页执行的用户固件对 FLASH 的读、写和擦除。表 13.1 概述了 C8051F52x/52xA/53x/53xA 器件的 FLASH 安全特性。

经 C2 调试接口访问 FLASH:

1. 任何未锁定的页均可被读、写或擦除。
2. 被锁定的页不能被读、写或擦除。
3. 包含锁定字节的页在未被锁定时可以被读、写或擦除。
4. 在没有页被锁定时，读锁定字节的内容总是被允许。
5. 不允许追加锁定页（将锁定字节中的 1 改写为 0）。
6. 对 FLASH 页解除锁定（将锁定字节中的 0 改写为 '1'）需要使用 C2 器件擦除命令，这将擦除所有页，包括含有锁定字节的页和锁定字节本身。
7. 保留区不能被读、写或擦除。

在未锁定页执行的用户固件访问 FLASH:

1. 除包含锁定字节的页之外，任何未被锁定的页均可被读、写或擦除。
2. 被锁定的页不能被读、写或擦除。试图擦除包含锁定字节的页将导致 FLASH 错误型器件复位。
3. 包含锁定字节的页不能被擦除，在未被锁定时可以被读或写。
4. 读锁定字节的内容总是被允许。
5. 不允许追加锁定页（将锁定字节中的 1 改写为 0）；
6. 不允许对 FLASH 页解除锁定（将锁定字节中的 0 改写为 1）。
7. 保留区不能被读、写或擦除。访问保留区或任何被锁定页的操作将导致 FLASH 错误型器件复位。

在被锁定页执行的用户固件访问 FLASH:

1. 除包含锁定字节的页之外，任何未被锁定的页均可被读、写或擦除。
2. 除包含锁定字节的页之外，任何被锁定的页都可以被读、写或擦除。试图擦除包含锁定字节的页将导致 FLASH 错误型器件复位。
3. 包含锁定字节的页不能被擦除，只能被读或写。试图擦除包含锁定字节的页将导致 FLASH 错误型器件复位。
4. 读锁定字节的内容总是被允许。
5. 不允许追加锁定页（将锁定字节中的 1 改写为 0）；
6. 不允许对 FLASH 页解除锁定（将锁定字节中的 0 改写为 1）。
7. 保留区不能被读、写或擦除。访问保留区或任何被锁定页的操作将导致 FLASH 错误型器件复位。

表 13.1 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"> — 经 C2 接口的所有被禁止的操作都被忽略 (不会导致器件复位)。 — 锁定任何一个 FLASH 页时，包含锁定字节的页也被锁定。 — 锁定字节一旦被写入便不能被修改，除非执行一次 C2 器件擦除。 — 如果用户代码写锁定字节，则在下一次复位之前锁定功能不会生效。 			

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 13.1 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
<p>位 7-2: 未使用。读 = 000000b, 写 = 忽略。</p> <p>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。</p> <p>位 0: PSWE: 程序存储写允许 将该位置 1 后允许用 MOVX 写指令向 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。</p>								

SFR 定义 13.2 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 写: 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 操作不正确或在写或擦除操作被禁止时试图写或擦除 FLASH, 则 FLASH 将被永久性锁定(不能写或擦除), 直到下一次器件复位。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: 位 1-0 指示当前的 FLASH 锁定状态 00: FLASH 写/擦除被锁定。 01: 第一个关键码已被写入 (0xA5)。 10: FLASH 处于解锁状态 (允许写/擦除) 11: FLASH 写/擦除操作被禁止, 直到下一次复位。</p>								

表 13.2 FLASH 存储器电气特性

VDD = 1.8V ~ 2.75V, -40℃到+125℃（除非特别说明）。

参 数	条 件	最小值	典型值	最大值	单 位
FLASH 尺寸	C8051F520/0A/1/1A 和 F530/0A/1/1A	7680	—	—	字节
	C8051F523/3A/4/4A 和 F533/3A/4/4A	4096			
	C8051F526/6A/7/7A 和 F536/6A/7/7A	2048			
擦写寿命	VDD ≥ 2.25V	40k	150k	—	擦/写
擦除周期		32	40	48	ms
写周期		76	92	114	μs
VDD	写/擦除操作	2.25	—	—	V

14. 端口输入/输出

数字和模拟资源可以通过 16 个 I/O 引脚使用。端口引脚被组织为两个 8 位端口。每个端口引脚都可以被定义为通用 I/O（GPIO）或模拟输入/输出。P0.0 ~ P1.7 可以被分配给内部数字资源，如图 14.3 所示。设计者完全控制数字功能的引脚分配，只受物理 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 14.3 和图 14.4）的外设优先顺序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 14.1 和 SFR 定义 14.2）用于选择内部数字功能。

端口 I/O 在 V_{REGIN} 的工作范围内耐 5.25V 电压。端口 I/O 单元电路示于图 14.2。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， $n = 0, 1$ ）。表 14.1 给出了端口 I/O 的电气特性。

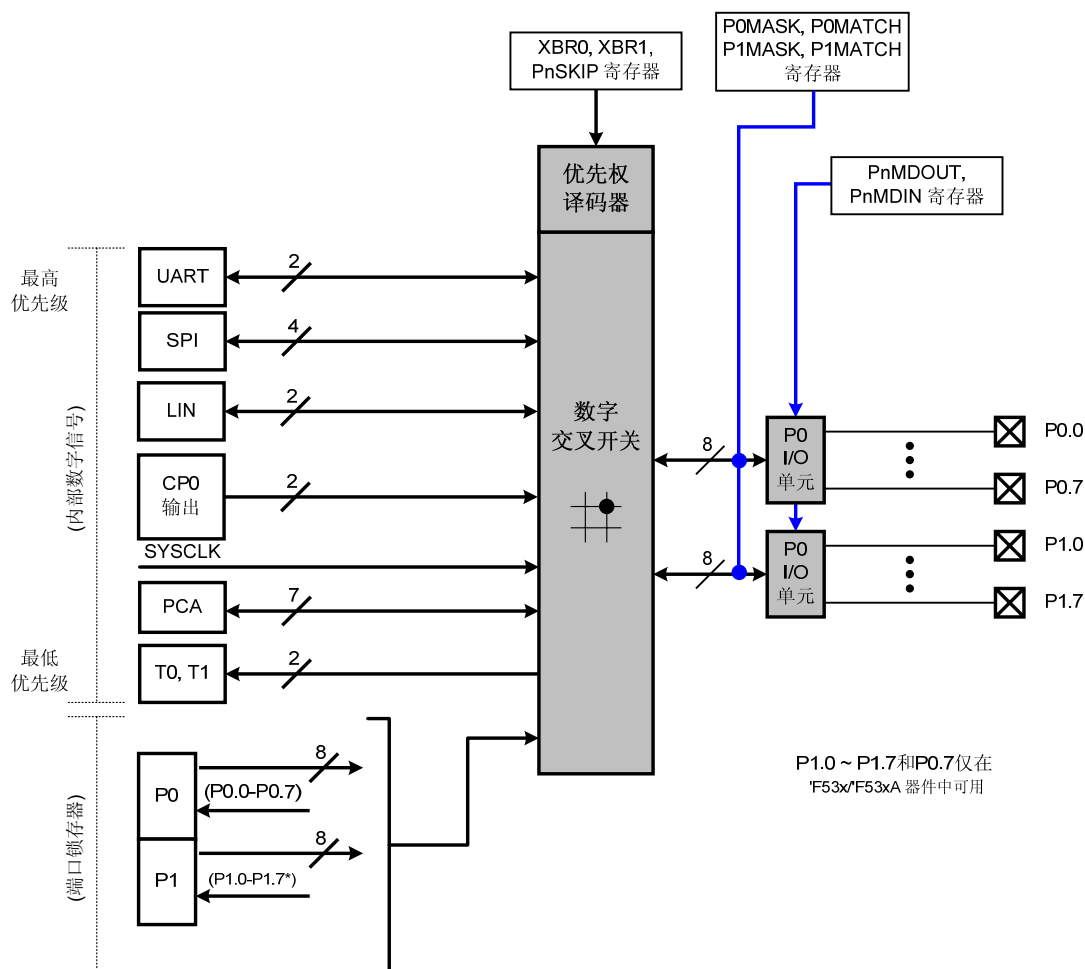


图 14.1 端口 I/O 功能框图

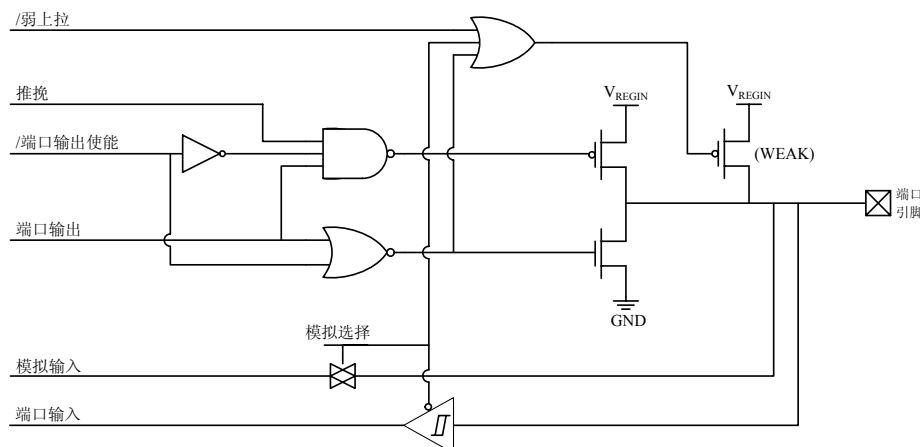


图 14.2 端口 I/O 单元框图

14.1 优先权交叉开关译码器

优先权交叉开关译码器（图 14.3）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 除外，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。这种情况适用于外部振荡器引脚 P1.0 和/或 P0.7（F53x/53xA）或 P0.2 和/或 P0.3（F52x/52xA），VREF 引脚 P0.0，外部 CNVSTR 信号引脚 P1.2（F53x/53xA）或 P0.5（F52x/52xA）以及任何被选择为 ADC 或比较器输入的引脚。交叉开关跳过那些被选择的引脚（如同将它们已分配），移向下一个未被分配的引脚。图 14.3 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 14.4 给出了 XTAL1(P1.0)脚和 XTAL2(P1.1)脚被跳过情况下（P1SKIP = 0x03）的交叉开关优先权译码表。

有关 UART 引脚的重要注意事项：在 C8051F52xA/53xA 器件中，如果 UART 被使能，则 UART 引脚必须被跳过，以使交叉开关为外设分配端口引脚时跳过 UART 引脚。例如，在 SPI 和 UART 被使能，并且为 SPI 分配引脚 P1.0 ~ P1.3 的情况下，为了正确分配 SPI 引脚，必须用 P0SKIP 跳过 UART 引脚。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

		P0								P1							
特殊功能信号		VREF								XTAL2							
TSSOP 20 和 QFN 20		XTAL1								CNVSTR							
PIN I/O		0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
TX0										C8051F53xA 器件							
RX0																	
TX0										C8051F53x 器件							
RX0																	
SCK																	
MISO																	
MOSI																	
NSS*																	
LIN-TX																	
LIN_RX																	
CP0																	
CP0A																	
/SYSCLK																	
CEX0																	
CEX1																	
CEX2																	
ECI																	
T0																	
T1																	
		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		P0SKIP[0:7]								P1SKIP[0:7]							
		可分配给外设的端口引脚															
特殊功能信号		不由交叉开关分配的特殊功能信号。															
		当这些信号被使能时，交叉开关必须被手动配置，															
		以跳过它们对应的端口引脚。															

图 14.3 没有引脚被跳过的交叉开关优先级译码表（TSSOP 20 和 QFN 20）

	P0								P1							
特殊功能信号 TSSOP 20 和 QFN 20	VREF								XTAL1	XTAL2	CNVSTR					
PIN I/O	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
TX0																
RX0																
TX0																
RX0																
SCK																
MISO																
MOSI																
NSS*																
LIN-TX																
LIN-RX																
CP0																
CP0A																
/SYSCLK																
CEX0																
CEX1																
CEX2																
ECI																
T0																
T1																
	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
	P0SKIP[0:7] = 0x80								P1SKIP[0:7] = 0x01							
特殊功能信号	不由交叉开关分配的特殊功能信号。															
	当这些信号被使能时，交叉开关必须被手动配置，															
	以跳过它们对应的端口引脚。															

图 14.4 晶体引脚被跳过的交叉开关优先级译码表（TSSOP 20 和 QFN 20）

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

特殊功能信号 DFN10		VREF	XTAL1	XTAL2	CNVSTR								
PIN I/O		0	1	2	3	4	5						
TX0								C8051F52xA 器件					
RX0													
TX0								C8051F52x 器件					
RX0													
SCK													
MISO													
MOSI													
NSS*													
LIN-TX													
LIN_RX													
CP0													
CP0A													
/SYSCLK													
CEX0													
CEX1													
CEX2													
ECI													
T0													
T1													
		0	0	0	0	0	0						
		P0SKIP[0:5]											
		可分配给外设的端口引脚。											
特殊功能信号		不由交叉开关分配的特殊功能信号。											
		当这些信号被使能时，交叉开关必须被手动配置，											
		以跳过它们对应的端口引脚。											

图 14.5 没有引脚被跳过的交叉开关优先权译码表（DFN 10）

		P0																
特殊功能信号 DFN 10		VREF	XTAL1		XTAL2	CNVSTR												
PIN I/O		0	1	2	3	4	5											
TX0								C8051F52xA 器件										
RX0																		
TX0									C8051F52x 器件									
RX0																		
SCK																		
MISO																		
MOSI																		
NSS*																		
LIN-TX																		
LIN-RX																		
CP0																		
CP0A																		
/SYSCLK																		
CEX0																		
CEX1																		
CEX2																		
ECI																		
T0																		
T1																		
		0	1	1	0	0	0											
		P0SKIP[0:5] = 0x06																
		可分配给外设的端口引脚。																
特殊功能信号		不由交叉开关分配的特殊功能信号。																
		当这些信号被使能时，交叉开关必须被手动配置，																
		以跳过它们对应的端口引脚。																

图 14.6 晶体引脚被跳过的交叉开关优先级译码表（DFN 10）

寄存器 XBR0 和 XBR1 用于为数字 I/O 资源分配物理 I/O 引脚。注意，当 SMBus 被选择时，交叉开关为其分配两个引脚（SDA 和 SCL）；当 UART 被选择时，交叉开关为其分配两个引脚（TX 和 RX）。UART0 的引脚分配是固定的（这是出于引导装载的目的）：UART TX0

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

总是被分配到 P0.3 (C8051F52x/53x) 或 P0.4 (C8051F52xA/53xA); UART RX0 总是被分配到 P0.4 (C8051F52x/53x) 或 P0.5 (C8051F52xA/53xA)。在优先功能和要跳过的引脚被分配之后, 标准端口 I/O 是连续的。

注意: SPI 可以工作在三线或四线方式, 由 SPI0CN 寄存器中的 NSSMD1-NSSMD0 位指定。根据 SPI 方式, NSS 信号可以连到端口引脚, 也可以不连到端口引脚。

14.2 端口 I/O 初始化

端口 I/O 初始化包括以下步骤:

第一步: 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)。

第二步: 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)。

第三步: 用端口跳过寄存器 (PnSKIP) 选择应被交叉开关跳过的那些引脚。

第四步: 用 XBRn 寄存器将引脚分配给要使用的外设。

第五步: 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, 其弱上拉、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仍可被模拟外设使用, 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 14.4。

注意: 端口 0 和端口 1 在 V_{REGIN} 的工作范围内耐 5.25V 电压。

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。即使对于在 XBRn 中被选择的数字资源, 也需要进行这种选择。当 XBR1 寄存器中的 WEAKPUD 位为 0 时, 输出方式为漏极开路的所有引脚的弱上拉都被使能。WEAKPUD 不影响被配置为推挽方式的端口 I/O。当漏极开路输出被驱动为逻辑 0 或引脚被配置为模拟输入方式时, 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。将 XBR1 中的 XBARE 位置 1 即使能交叉开关。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 寄存器设置, 可以使用优先权译码表确定 I/O 引脚分配。

注意: 为使端口引脚工作在标准端口 I/O 的输出方式, 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。

SFR 定义 14.1 XBR0: 端口 I/O 交叉开关寄存器 0

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
—	—	CP0AE	CP0E	SYSCKE	LINE	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; 必须写 00b。								
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: LINE: LIN 输出使能位								
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 (P0.3 和 P0.4) 或 (P0.4 和 P0.5)。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 14.2 XBRI: 端口 I/O 交叉开关寄存器 1

R/W	R/W	R/W	R/W	R/W	R/W	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
<p>位 7: WEAKPUD: 端口 I/O 弱上拉禁止位。 0: 弱上拉使能 (被配置为模拟输入的端口 I/O 除外)。 1: 弱上拉禁止。</p> <p>位 6: XBARE: 交叉开关使能位 0: 交叉开关禁止。 1: 交叉开关使能。</p> <p>位 5: T1E: T1 使能位 0: T1 不连到端口引脚。 1: T1 连到端口引脚。</p> <p>位 4: T0E: T0 使能位 0: T0 不连到端口引脚。 1: T0 连到端口引脚。</p> <p>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。</p> <p>位 2: 保留。</p> <p>位 2-0: PCA0ME: PCA 模块 I/O 使能位 000: 所有的 PCA I/O 都不连到端口引脚。 001: CEX0 连到端口引脚。 010: CEX0、CEX1 连到端口引脚。 011: CEX0、CEX1、CEX2 连到端口引脚。</p>								

14.3 通用端口 I/O

未被交叉开关分配的端口引脚和未被模拟外设使用的端口引脚都可以作为通用 I/O。通过对应的端口数据寄存器访问端口 P0 ~ P1，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，数据被锁存到端口数据寄存器中，以保持引脚上的输出数据值不变。读端口数据寄存器（或端口位）将总是返回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其它信号时，端口寄存器总是读其对应的端口 I/O 引脚。但在对端口锁存器执行下面的读-修改-写指令（ANL、ORL、XRL、JBC、CPL、INC、DEC、DJNZ）和对端口 SFR 中的某一位执行 MOV、CLR、SETB 期间例外。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。

除了执行通用 I/O 功能之外，P0 和 P1 还可以产生端口匹配事件（如果端口输入引脚的逻辑电平与一个软件控制值匹配）。如果（P0 & P0MASK）不等于（P0MATCH & P0MASK）或如果（P1 & P1MASK）不等于（P1MATCH & P1MASK），则会产生端口匹配事件。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。如果 EMAT（EIE2.1）被置 1，端口匹配事件可以产生中断。端口匹配事件可以将内部振荡器从 SUSPEND 方式唤醒，详见“15.1.1 内部振荡器挂起方式”。

SFR 定义 14.3 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 — 输出出现在 I/O 引脚（根据交叉开关寄存器的设置）。								
0: 逻辑低电平输出。								
1: 逻辑高电平输出。（若相应的 P0MDOUT.n 位 = 0，则为高阻态）。								
读 — 读那些在 P0MDIN 中被选择为模拟输入的引脚时总是返回 0。被配置为数字输入时直接读端口引脚。								
0: P0.n 为逻辑低电平。								
1: P0.n 为逻辑高电平。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 14.4 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 模拟输入配置位（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。

SFR 定义 14.5 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 输出方式配置位（分别对应）。如果 P0MDIN 寄存器中的对应位为逻辑 0，则输出方式配置位被忽略。
0：对应的 P0.n 引脚输出为漏极开路。
1：对应的 P0.n 引脚输出为推挽方式。

注：当 SDA 和 SCL 出现在任何端口 I/O 引脚时，不管 P0MDOUT 的值为何，它们都是漏极开路的。

SFR 定义 14.6 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引脚。作为模拟输入（ADC 或比较器）或特殊功能（VREF 输入、外部振荡器电路、CNVSTR 输入）的引脚应被交叉开关跳过。
0：对应的 P0.n 引脚不被交叉开关跳过。
1：对应的 P0.n 引脚被交叉开关跳过。

SFR 定义 14.7 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: 0xD7
位 7-0: P0MAT[7:0]: 端口 0 匹配值 这些位控制未被屏蔽的 P0 端口引脚的比较值。如果 (P0 & P0MASK) 不等于 (P0MAT & P0MASK), 则会产生端口匹配事件。								

SFR 定义 14.8 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: 0xC7
位 7-0: P0MASK[7:0]: 端口 0 屏蔽值 这些位选择哪些端口引脚与 P0MAT 中存储器的值比较。 0: 对应的 P0.n 引脚被忽略, 不能产生端口匹配事件。 1: 对应的 P0.n 引脚被与 P0MAT 中的对应位比较。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 14.9 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 — 输出出现在 I/O 引脚（根据交叉开关寄存器的设置）。

0: 逻辑低电平输出。

1: 逻辑高电平输出。（若相应的 P1MDOUT.n 位 = 0，则为高阻态）。

读 — 读那些在 P1MDIN 中被选择为模拟输入的引脚时总是返回 0。被配置为数字输入时直接读端口引脚。

0: P1.n 引脚为逻辑低电平。

1: P1.n 引脚为逻辑高电平。

SFR 定义 14.10 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: P1.7 – P1.0 模拟输入配置位（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 14.11 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: P1.7 – P1.0 输出方式配置位（分别对应）。如果 P1MDIN 寄存器中的对应位为逻辑 0，则输出方式配置位被忽略。

0: 对应的 P1.n 引脚输出为漏极开路。

1: 对应的 P1.n 引脚输出为推挽方式。

注：当 SDA 和 SCL 出现在任何端口 I/O 引脚时，不管 P0MDOUT 的值为何，它们都是漏极开路的。

SFR 定义 14.12 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引脚。用作模拟输入 (ADC 或比较器) 或特殊功能 (VREF 输入、外部振荡器电路、CNVSTR 输入) 的引脚应被交叉开关跳过。
 0: 对应的 P1.n 引脚不被交叉开关跳过。
 1: 对应的 P1.n 引脚被交叉开关跳过。

SFR 定义 14.13 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: 0xCF

位 7-0: P1MAT[7:0]: 端口 1 匹配值
 这些位控制未被屏蔽的 P1 端口引脚的比较值。如果 (P1 & P1MASK) 不等于 (P1MAT & P1MASK), 则会产生端口匹配事件。

SFR 定义 14.14 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: 0xBF

位 7-0: P1MASK[7:0]: 端口 1 屏蔽值
 这些位选择哪些端口引脚与 P1MAT 中存储器的值比较。
 0: 对应的 P1.n 引脚被忽略, 不能产生端口匹配事件。
 1: 对应的 P1.n 引脚被与 P1MAT 中的对应位比较。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 14.1 端口 I/O 直流电气特性

$V_{\text{REGIN}} = 2.7\text{V} \sim 5.25\text{V}$, -40°C 到 $+125^{\circ}\text{C}$ (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	$I_{\text{OH}} = -3\text{ mA}$, 端口I/O为推挽方式 $I_{\text{OH}} = -10\text{ }\mu\text{A}$, 端口I/O为推挽方式 $I_{\text{OH}} = -10\text{ mA}$, 端口I/O为推挽方式	$V_{\text{REGIN}}-0.4$ $V_{\text{REGIN}}-0.2$ —	— — $V_{\text{REGIN}}-0.7$	— — —	V
输出低电压	V = 1.8 V: $I_{\text{OL}} = 70\text{ }\mu\text{A}$ $I_{\text{OL}} = 8.5\text{ mA}$ V = 2.7 V: $I_{\text{OL}} = 70\text{ }\mu\text{A}$ $I_{\text{OL}} = 8.5\text{ mA}$ V = 5.25 V: $I_{\text{OL}} = 70\text{ }\mu\text{A}$ $I_{\text{OL}} = 8.5\text{ mA}$	— — — — — —	— — — — — —	50 750 45 550 40 400	mV
输入高电压		$V_{\text{REGIN}} \times 0.65$	—	—	V
输入低电压		—	—	$V_{\text{REGIN}} \times 0.3$	V
输入漏电流	弱上拉禁止 C8051F52xA/53xA: 弱上拉使能, $V_{\text{IN}} = 0\text{V}$; $V_{\text{REGIN}} = 1.8\text{V}$ C8051F52x/52xA/53x/53xA: 弱上拉使能, $V_{\text{IN}} = 0\text{V}$; $V_{\text{REGIN}} = 2.7\text{V}$ 弱上拉使能, $V_{\text{IN}} = 0\text{V}$; $V_{\text{REGIN}} = 5.25\text{V}$	— — — —	— < 5 < 20 < 65	± 2 TBD 50 115	μA

15. 振荡器

C8051F52x/52xA/53x/53xA 器件有一个可编程内部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/禁止内部振荡器和调节其输出频率(如图 15.1 所示)。系统时钟 (SYSCLK) 可以源自内部振荡器和外部振荡器电路。表 15.1 给出了内部振荡器的电气特性。

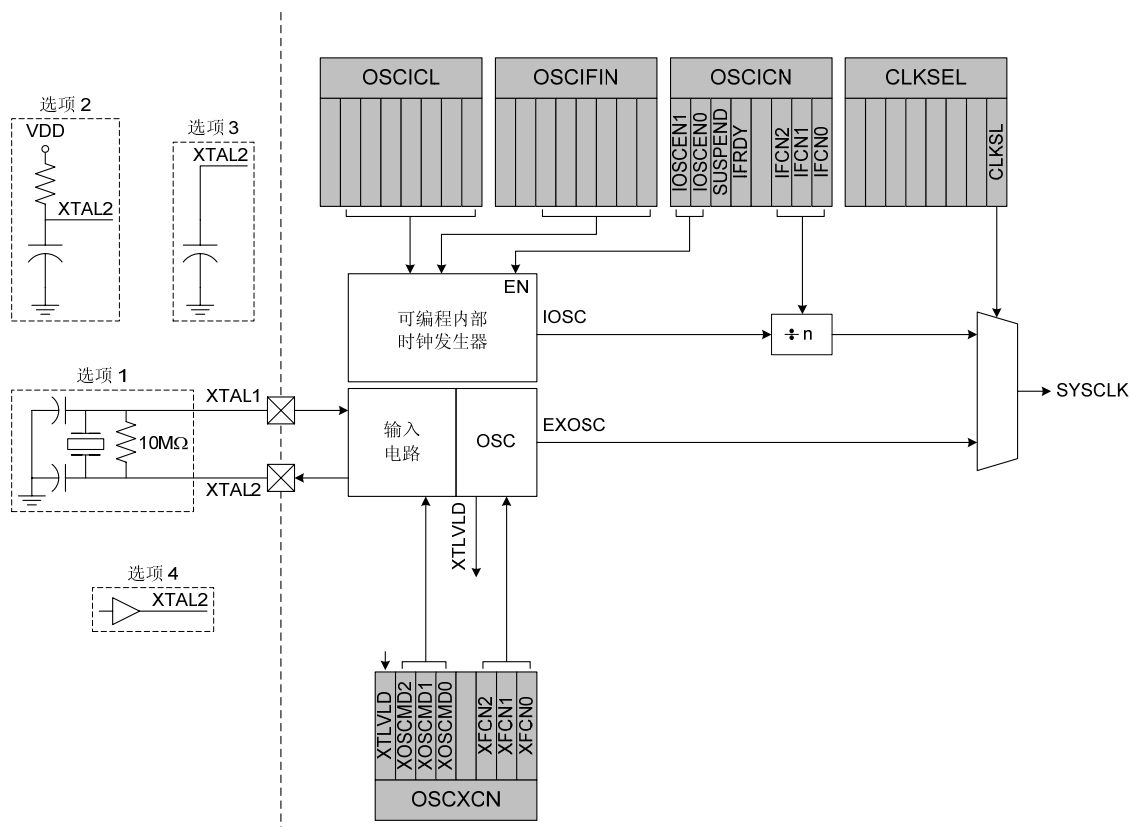


图 15.1 振荡器框图

15.1 可编程内部振荡器

C8051F52x/52xA/53x/53xA 器件包含一个可编程内部振荡器，该振荡器在系统复位后被默认为系统时钟。该振荡器的周期可以通过 OSCICL 和 OSCIFIN 寄存器（见 SFR 定义 15.2 和 SFR 定义 15.3）编程。对于 C8051F52x/53x 器件，OSCICL 和 OSCIFIN 已经过工厂校准，对应的频率为 24.5 MHz。

表 15.1 给出了精密内部振荡器的电气特性。注意：系统时钟可以从内部振荡器分频得到，分频系数由寄存器 OSCICN 中的 IFCN 位设定，可为 1、2、4、8、16、32、64 或 128。复位后的缺省分频系数为 128。

15.1.1 内部振荡器挂起方式

通过向 SUSPEND 位（OSCICN.5）写 1 可以将内部振荡器置于挂起方式。如果系统时钟来自内部振荡器，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。

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 15.1 OSCICN: 内部振荡器控制寄存器

R/W	R	R/W	R	R	R/W	R/W	R/W	复位值
IOSCEN1	IOSCEN0	SUSPEND	IFRDY	-	IFCN2	IFCN1	IFCN0	11000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0xB2
<p>位 7-6: IOSCEN[1:0]: 内部振荡器使能位 00: 内部振荡器禁止。 01: 保留。 10: 保留。 11: 内部振荡器在正常方式下被使能, 在 SUSPEND 方式下被禁止。</p> <p>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。</p> <p>位 4: IFRDY: 内部振荡器频率准备好标志 0: 内部振荡器未运行在编程频率。 1: 内部振荡器按编程频率运行。</p> <p>位 3: 未用。读 = 0b, 写 = 忽略。</p> <p>位 2-0: IFCN2-0: 内部振荡器频率控制位 000: SYSCLK 为内部振荡器 128 分频 (缺省)。 001: SYSCLK 为内部振荡器 64 分频。 010: SYSCLK 为内部振荡器 32 分频。 011: SYSCLK 为内部振荡器 16 分频。 100: SYSCLK 为内部振荡器 8 分频。 101: SYSCLK 为内部振荡器 4 分频。 110: SYSCLK 为内部振荡器 2 分频。 111: SYSCLK 为内部振荡器 1 分频。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 15.2 OSCICL: 内部振荡器校准寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-	OSCICL							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。
这些位决定内部振荡器的周期。对于 C8051F52x/53x 器件, 复位值已经过工厂校准, 对应 24.5 MHz 的内部振荡器频率。

SFR 定义 15.3 OSCIFIN: 内部振荡器微调寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
-	-	OSCIFIN						不确定
位7	位6	位5	位4	位3	位2	位1	位0 (可位寻址)	SFR地址: 0xB0

位 7-6: 未用。读 = 00b, 写 = 忽略。

位 5-0: OSCIFIN: 内部振荡器微调位。
有效范围在 0x00 和 0x27 之间。

该寄存器微调内部振荡器的周期。对于 C8051F52x/52xA/53x/53xA 器件, 复位值已经过工厂校准, 对应 24.5 MHz 的内部振荡器频率。

15.2 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 15.1, 选项 1), 还必须在 XTAL1 和 XTAL2 引脚之间并接一个 10MΩ 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 15.1, 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 XFNCN (见 SFR 定义 15.4)。

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 P0.7 和 P1.0 (F53x/53xA) 或 P0.2 和 P0.3 (F52x/52xA) 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 P1.0 (F53x/53xA) 或 P0.3 (F52x/52xA) 被用作 XTAL2。端口 I/O 交叉开关应配置为跳过被振荡器占用的引脚。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“14.2 端口 I/O 初始化”。

15.2.1 外部振荡器直接用作定时器时钟

外部振荡器 8 分频是定时器和可编程计数器阵列 (PCA) 的一个时钟选项 (见“19. 定时器”和“20. 可编程计数器阵列 (PCA0)”)。当外部振荡器被用作这些外设的时钟而不用作系统时钟时, 外部振荡器频率必须小于或等于系统时钟频率。在这种配置下, 提供给外设的时

钟（外部振荡器/8）与系统时钟同步，这种同步的抖动被限制为±0.5 个系统时钟周期。

15.2.2 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，则电路为图 15.1 中的选项 1。应从 SFR 定义 15.4（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 12MHz 的晶体要求 XFCN 设置值为 111b。

在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见的后果。建议的步骤如下：

- 第一步：通过向端口锁存器写 1 来配置 XTAL1 和 XTAL2 引脚。
- 第二步：配置 XTAL1 和 XTAL2 为模拟输入。
- 第三步：使能外部振荡器。
- 第四步：等待至少 1ms。
- 第五步：查询 XTLVLD =>1。
- 第六步：将系统时钟切换到外部振荡器。

注：对于音叉晶体，可能需要等待较长的时间 XTLVLD 才能有效。

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。从晶体来看，这两个电容是“串联”的，但与 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。

确定两个负载电容的方程如下：

$$C_L = \frac{C_A \times C_B}{C_A + C_B} + C_S$$

其中：

C_A 和 C_B 是连接到晶体引脚的电容。

C_S 是PCB的总寄生电容。

在晶体与 XTAL1 和 XTAL2 引脚尽可能接近的情况下，一般布局的寄生电容为每个引脚 2~5pF。

如果 C_A 和 C_B 相同（C），则上面的方程变为：

$$C_L = \frac{C}{2} + C_S$$

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 15.1 中的配置选项 1。如果每个 XTAL 引脚的寄生电容值为 3 pF（合计 6 pF），则采用 13 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 15.2 所示。

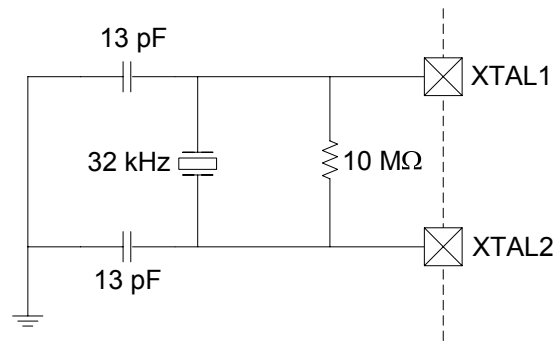


图 15.2 32.768 KHz 外部晶体示例

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，连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。

15.2.3 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 15.1 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。为了确定 OSCXCN 寄存器所需要的外部振荡器频率控制值（XFCN），首先选择能产生所要求的振荡频率的 RC 网络值。如果所希望的频率是 100kHz，选 $R = 246k\Omega$ 和 $C = 50pF$ ：

$$f = \frac{1.23(10^3)}{RC} = \frac{1.23(10^3)}{246 \times 50} = 0.1MHz = 100kHz$$

查 SFR 定义 15.3 中的表，得到所需要的 XFCN 值为 010b。在 RC 方式，将 XFCN 编程为较高的设置值会改善频率精度，但外部振荡器消耗的电源电流会有少许增加。

15.2.4 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 15.1 中的选项 3。电容应不大于 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），先选择振荡频率并利用下面的公式计算电容值。假设 $VDD = 2.1V$ 和 $f = 75 KHz$ ：

$$f = \frac{KF}{C \times V_{DD}}$$

$$0.075MHz = \frac{KF}{C \times 2.1}$$

由于所需要的频率大约为 75 kHz，从 SFR 定义 15.3 的表中选择 K 因子，得到 $KF = 7.7$ ：

$$0.075 MHz = 7.7/(C \times 2.1)$$

$$(C \times 2.1) = 7.7/0.075 MHz$$

$$C = 102.6/2.0 pF = 51.3 pF$$

因此，本例中要用的 XFCN 值为 010b。

SFR 定义 15.4 OSCXCN: 外部振荡器控制寄存器

R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
XTLVLD	XOSCMD2	XOSCMD1	XOSCMD0	保留	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
 (在 XOSCMD=11x 时有效, 只读)
 0: 晶体振荡器未用或未稳定。
 1: 晶体振荡器稳定运行。

位 6-4: XOSCMD2-0: 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 = 0b, 必须写 0b。

位 2-0: XFCN2-0: 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	$f \leq 20\text{kHz}$	$f \leq 25\text{kHz}$	K 因子 = 0.87
001	$20\text{kHz} < f \leq 58\text{kHz}$	$25\text{kHz} < f \leq 50\text{kHz}$	K 因子 = 2.6
010	$58\text{kHz} < f \leq 155\text{kHz}$	$50\text{kHz} < f \leq 100\text{kHz}$	K 因子 = 7.7
011	$155\text{kHz} < f \leq 415\text{kHz}$	$100\text{kHz} < f \leq 200\text{kHz}$	K 因子 = 22
100	$415\text{kHz} < f \leq 1.1\text{MHz}$	$200\text{kHz} < f \leq 400\text{kHz}$	K 因子 = 65
101	$1.1\text{MHz} < f \leq 3.1\text{MHz}$	$400\text{kHz} < f \leq 800\text{kHz}$	K 因子 = 180
110	$3.1\text{MHz} < f \leq 8.2\text{MHz}$	$800\text{kHz} < f \leq 1.6\text{MHz}$	K 因子 = 664
111	$8.2\text{MHz} < f \leq 25\text{MHz}$	$1.6\text{MHz} < f \leq 3.2\text{MHz}$	K 因子 = 1590

晶体方式 (电路见图 15.1, 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 15.1, 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 15.1, 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)

15.3 系统时钟选择

内部振荡器的启动时间很短，因此可以在使能内部振荡器的 OSCICN 写操作后立即选择内部振荡器为系统时钟。外部晶体和陶瓷谐振器通常需要一定的启动时间才能达到稳定。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。RC 和 C 方式通常不需要启动时间。

寄存器 CLKSEL 中的 CLKSL 位选择用作系统时钟的振荡源。当选择外部振荡器作为系统时钟时，CLKSL 必须被置 1。当选择另一振荡源作为系统时钟时，外部振荡器仍然可以给外设（定时器、PCA）提供时钟。系统时钟可以在内部振荡器和外部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。

SFR 定义 15.5 CLKSEL：时钟选择寄存器

R	R	R/W	R/W	R	R/W	R/W	R/W	复位值
-	-	保留	保留	-	保留	保留	CLKSL	00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址： 0xA9
<p>位 7-6： 未用。读 = 00b，写 = 忽略。</p> <p>位 5-4： 保留。读 = 00b，必须写 00b。</p> <p>位 3： 未用。读 = 0b，写 = 忽略。</p> <p>位 2-1： 保留。读 = 00b，必须写 00b。</p> <p>位 0： CLKSL1：系统时钟选择位 这些位选择系统时钟源。</p> <p>0：内部振荡器（由寄存器 OSCICN 中的 IFCN 位选择分频系数）。</p> <p>1：外部振荡器。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 15.1 内部振荡器电气特性

$V_{DD} = 1.8 \sim 2.75V$, $-40^{\circ}C$ 到 $+125^{\circ}C$ (除非特别说明)

参 数	条 件	最小值	典型值	最大值	单位
振荡器频率	IFCN = 11b	24.5-0.5%	24.5*	25 + 0.5%	MHz
振荡器电源电流 (来自 V_{DD})	内部振荡器使能 OSCICN[7:6] = 11b	—	800	TBD	μA
	内部振荡器挂起 OSCICN[7:6] = 00b ZTCEN = 1	—	50	—	
唤醒时间 (从挂起方式)	OSCICN[7:6] = 00b ZTCEN = 0	—	1	—	μs
	OSCICN[7:6] = 00b ZTCEN = 1	—	5	—	指令周期
*注: 这是在整个工作温度范围内的平均频率。					

16. UART0

UART0 是一个异步、全双工串口，它提供标准 8051 串行口的方式 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）或接收到数据字节（SCON0 中的 RI0 位被置 1）时将产生一个中断。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）。

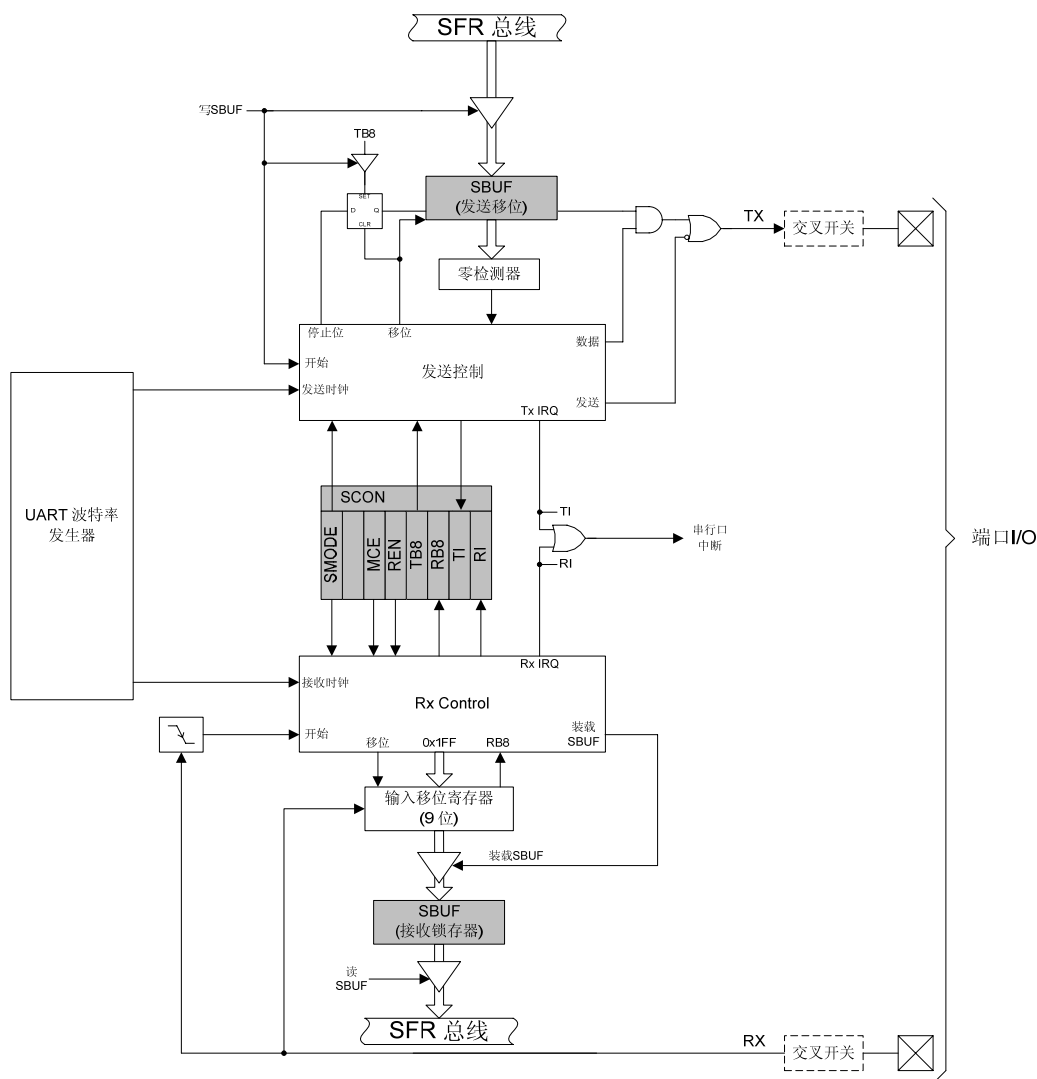


图 16.1 UART0 原理框图

16.1 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 16.2 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。

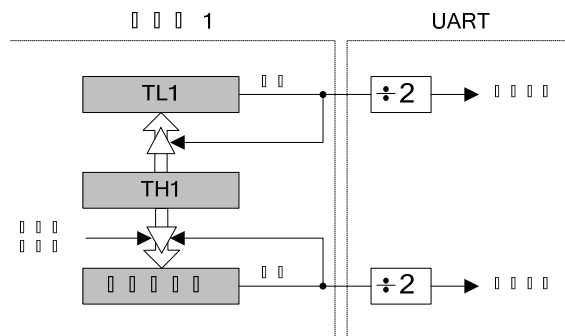


图 16.2 UART0 波特率逻辑

定时器 1 应被配置为方式 2，即 8 位自动重载方式。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 16.1-A 和方程 16.1-B 决定：

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

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

方程 16.1 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 (8 位自动重载方式的重载值)。

定时器 1 时钟频率的选择方法见“19. 定时器”。表 16.1 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。

16.2 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 16.3 所示。

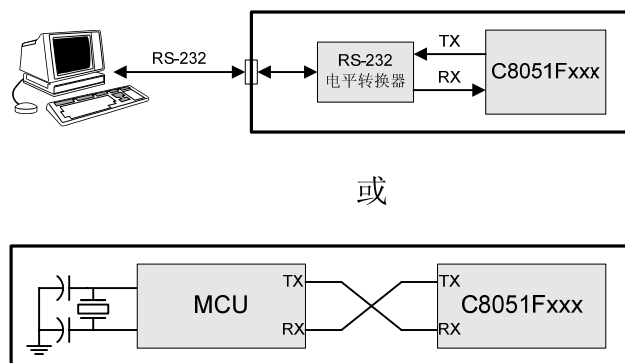


图 16.3 UART 连接图

16.2.1 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 置位时将产生中断。

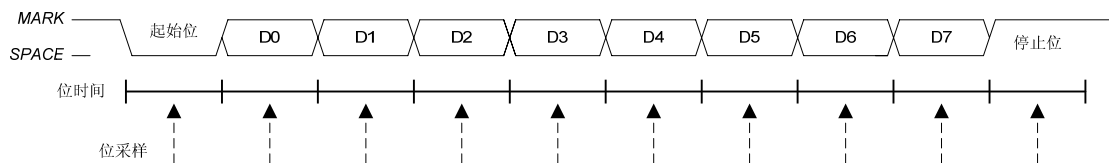


图 16.4 8 位 UART 时序图

16.2.2 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 PSW 中的奇偶位 P（用于错误检测），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。

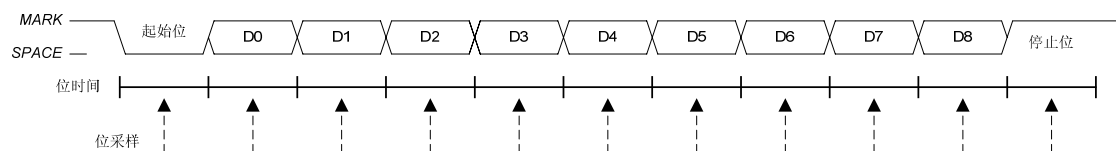


图 16.5 9 位 UART 时序图

16.3 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，则只有当 UART 接收到的第九位为逻辑 1 (RB80 = 1) 并收到有效的停止位后 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，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。

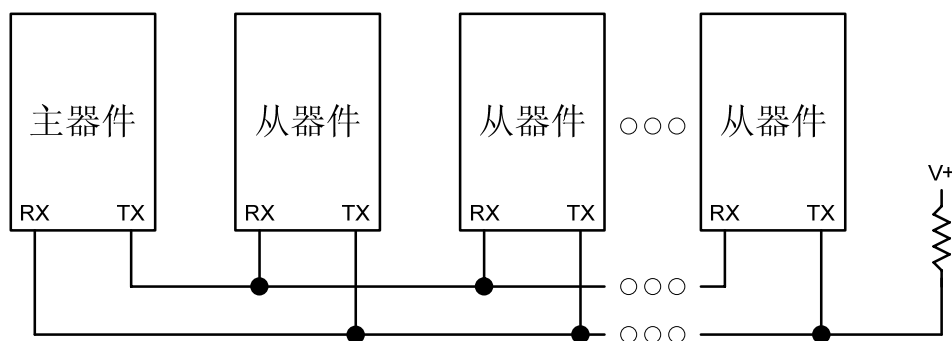


图 16.6 UART 多机方式连接图

SFR 定义 16.1 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 工作方式选择位 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: 多处理器通信允许 该位的功能取决于 UART0 工作方式。 S0MODE = 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。 S0MODE = 1: 多处理器通信允许。 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 该位的逻辑电平被赋值给 9 位 UART 方式的第 9 发送位。在 8 位 UART 方式中未用。根据需要用软件置 1 或清 0。							
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。							

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 16.2 SBUF0: UART0 数据缓冲器

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 16.1 对应标准波特率的定时器设置 (使用内部振荡器)

频率: 24.5MHz							
SYSCLK 源自 内部振荡器	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) *	T1M*	定时器 1 重载值 (hex)
	230400	-0.32%	106	SYSCLK	XX	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

X = 忽略。

*注: SCA1-SCA0 和 T1M 位的定义见 19.1 节。

17. LIN（C8051F520/0A/3/3A/526/6A 和 C8051F530/3A/6/6A）

注意：本章假设读者理解局部互连网络（LIN）协议。有关 LIN 协议的详细信息，包括技术规范，请参见 LIN 联盟网站（<http://www.lin-subbus.org/>）。

LIN 是一个异步串行通信接口，主要用于汽车网络中。Silicon Laboratories 的 LIN 控制器符合 LIN 2.0 规范，实现了一个完整的硬件 LIN 接口，该接口具有以下功能：

- 可选择主方式或从方式。
- 在从方式下有自动波特率选项。
- 24.5 MHz 内部振荡器在整个电压和温度范围内达到 0.5% 的精度，所以在主方式不需要使用外部振荡器。

注意：当使用 LIN 外设时，所需要的最小系统时钟（SYSCLK）应为 8 MHz。

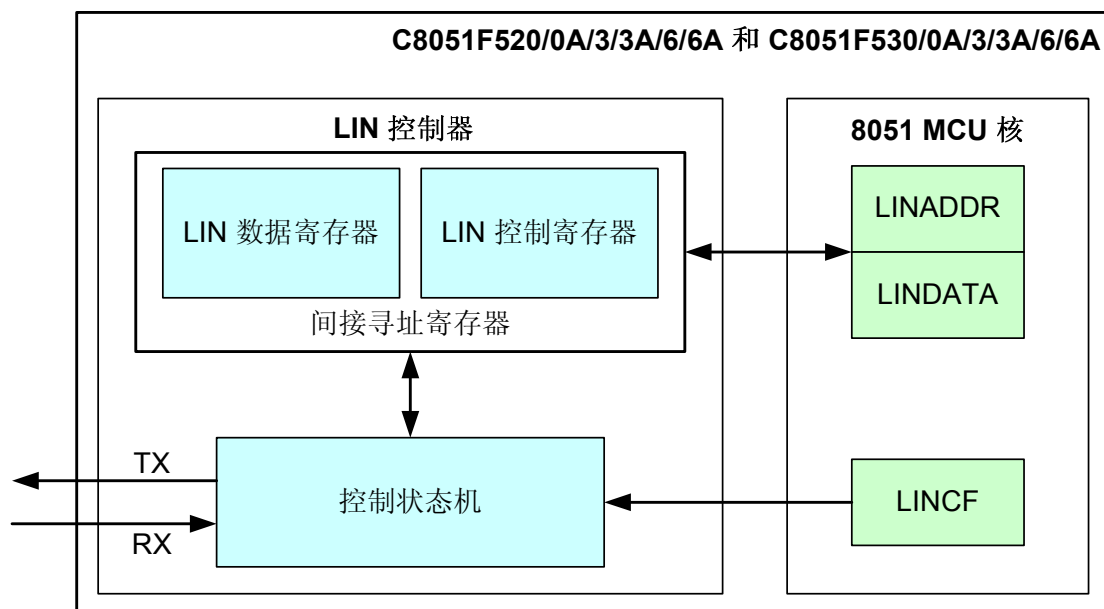


图 17.1 LIN 原理框图

LIN 外设由四个主要部分组成：

- LIN 访问寄存器——提供 MCU 核与 LIN 外设之间的接口。
- LIN 数据寄存器——用于存放发送和接收的报文数据字节的寄存器。
- LIN 控制寄存器——控制 LIN 接口的功能。
- 控制状态机和位流逻辑——包含用于报文串行化和控制总线时序的硬件。

17.1 与 LIN 外设的软件接口

方式（主或从）选择和自动波特率功能是通过 LIN0 的控制方式寄存器（LIN0CF）实现的。其他 LIN 寄存器是通过 LIN0 地址（LINADDR）和 LIN0 数据（LINDATA）这两个特殊功能寄存器间接访问的。LINADDR 寄存器选择读/写 LINDATA 寄存器时的目标地址。表 17.4

列出了所有可间接访问的 LIN 寄存器。

17.2 LIN 接口设置和操作

基于硬件的 LIN 外设允许以最小的固件开销实现从和主节点方式，并允许用中断和查询方式对接口状态实现完全控制。

使用 LIN 外设的第一步是定义节点的基本特性。

- 方式——主方式或从方式。
- 波特率——手动定义或使用自动波特率功能（仅限于从方式）。
- 校验和类型——在基本和增强校验和之间选择，这两种校验和算法均以硬件实现。

17.2.1 方式定义

该外设用硬件实现了符合 LIN 规范的从和主工作方式。工作方式使用 MODE 位（LIN0CF.6）配置。

17.2.2 波特率选项：手动或自动

LIN 外设可以选择手动或自动计算其波特率。主节点必须总是使用手动方式设置其波特率，而从节点可以选择手动或自动设置。使用 ABAUD 位（LIN0CF.5）选择手动或自动设置。

手动或自动波特率配置都需要进行一些其他设置。下面两节对这两种不同的选项及其与波特率的关系，以及为获得所需要的波特率所必须进行的步骤进行了详细解释。

17.2.3 波特率计算：手动方式

LIN 外设使用的波特率是系统时钟（SYSCLK）和位定时寄存器的函数，由下面的方程给出：

$$\text{波特率} = \frac{\text{SYSCLK}}{2^{\text{预分频系数}+1} \times \text{分频系数} \times (\text{倍频系数} + 1)}$$

预分频系数、分频系数和倍频系数是 LIN0DIV 和 LIN0MUL 寄存器中的组成部分，可以根据下面的范围设置它们的值：

表 17.1 波特率计算变量范围

系数	范围
预分频系数	0 ~ 3
倍频系数	0 ~ 31
分频系数	200 ~ 511

注意：LIN 外设工作的最小系统时钟（SYSCLK）为 8 MHz。

用下面的方程计算波特率方程中的这些系数：

$$\text{倍频系数} = \frac{20000}{\text{波特率}} - 1$$

$$\text{预分频系数} = \ln \left[\frac{\text{SYSCLK}}{(\text{倍频系数} + 1) \times \text{波特率} \times 200} \right] \times \frac{1}{\ln 2} - 1$$

$$\text{分频系数} = \frac{\text{SYSCLK}}{2^{\text{预分频系数}+1} \times \text{倍频系数} \times \text{波特率}}$$

注意：在所有这些方程中，结果都必须向下取整。

下面的例子计算一个主节点的这些系数，该主节点运行在 24.5 MHz，通信速率为 19.2 Kbits/s。首先计算倍频系数：

$$\text{倍频系数} = \frac{20000}{19200} - 1 = 0.0417 \cong 0$$

然后计算预分频系数：

$$\text{预分频系数} = \ln \left[\frac{24500000}{(0+1) \times 19200 \times 200} \right] \times \frac{1}{\ln 2} - 1 = 1.674 \cong 1$$

再计算分频系数：

$$\text{分频系数} = \frac{24500000}{2^{(1+1)} \times (0+1) \times 19200} = 319.010 \cong 319$$

将这些值代入波特率方程：

$$\text{波特率} = \frac{24500000}{2^{(1+1)} \times (0+1) \times 319} = 19200.63$$

下面的代码对主方式接口编程，使用增强校验和，接口工作在 19200 bits/s，系统时钟为 24.5 MHz。

```
LINOCF  = 0x80;           // 使能接口
LINOCF |= 0x40;           // 设置节点为主方式

LINADDR = 0x0D;           // 指向LIN0MUL寄存器
// 初始化寄存器（预分频系数、倍频系数和分频系数的位8）
LINDATA = (0x01 << 6) + (0x00 << 1) + ((0x13F & 0x0100) >> 8);
LINADDR = 0x0C;           // 指向LIN0DIV寄存器
LINDATA = (unsigned char)_0x13F; // 初始化LIN0DIV
```

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

```
LINADDR = 0x0B;           // 指向LIN0SIZE寄存器
LINDATA |= 0x80;           // 初始化校验和为增强校验和
LINADDR = 0x08;           // 指向LIN0CTRL寄存器
LINDATA = 0xC0;           // 清除错误和中断
```

表 17.2 给出了典型系统时钟和波特率所需要的配置值：

表 17.2 手动波特率参数示例

波特率 (位/秒)	20K			19.2K			9.6K			4.8K			1K		
SYSCLK (MHz)	倍频 系数	预分 系数	分频 系数	倍频 系数	预分 系数	分频 系数	倍频 系数	预分 系数	分频 系数	倍频 系数	预分 系数	分频 系数	倍频 系数	预分 系数	分频 系数
25	0	1	312	0	1	325	1	1	325	3	1	325	19	1	312
24.5	0	1	306	0	1	319	1	1	319	3	1	319	19	1	306
24	0	1	300	0	1	312	1	1	312	3	1	312	19	1	300
22.1184	0	1	276	0	1	288	1	1	288	3	1	288	19	1	276
16	0	1	200	0	1	208	1	1	208	3	1	208	19	1	200
12.25	0	0	306	0	0	319	1	0	319	3	0	319	19	0	306
12	0	0	300	0	0	312	1	0	312	3	0	312	19	0	300
11.0592	0	0	276	0	0	288	1	0	288	3	0	288	19	0	276
8	0	0	200	0	0	208	1	0	208	3	0	208	19	0	200

17.2.4 波特率计算：自动方式

如果 LIN 外设被配置为从方式，只需按下面的公式计算预分频系数和分频系数：

$$\text{预分频系数} = \ln \left[\frac{\text{SYSCLK}}{4000000} \right] \times \frac{1}{\ln 2} - 1$$

$$\text{分频系数} = \frac{\text{SYSCLK}}{2^{\text{预分频系数}+1} \times 20000}$$

在下面的例子中，使用 24.5 MHz 的系统时钟计算这些系数的值：

$$\text{预分频系数} = \ln \left[\frac{24500000}{4000000} \right] \times \frac{1}{\ln 2} - 1 = 1.615 \cong 1$$

$$\text{分频系数} = \frac{24500000}{2^{(1+1)} \times 20000} = 306.25 \cong 306$$

表 17.3 给出了系统时钟和波特率以及相应系数的典型值

表 17.4 自动波特率参数示例

系统时钟 (MHz)	预分频率系数	分频率系数
25	1	312
24.5	1	306
24	1	300
22.1184	1	276
16	1	200
12.25	0	306
12	0	300
11.0592	0	276
8	0	200

17.3 LIN 主方式操作

主节点负责报文的时序和发送每个帧的首部，首部中包含 SYNCH BREAK FIELD、SYNCH FIELD 和 IDENTIFIER FIELD。安排报文时序的步骤如下所述：

- 1) 向 LIN0ID 寄存器装入 6 位标识符。
- 2) 向 LIN0SIZE 寄存器装入数据长度。将该值设置为数据字节数，或者设置为“1111b”（如果从标识符译码获得数据长度）。还要在 LIN0SIZE 寄存器中设置校验和类型（基本型或增强型）。
- 3) 通过设置 TXRX 位 (LIN0CTRL.5) 来设置数据方向。将该位置 1 执行主发送操作，将该位清 0 执行主接收操作。
- 4) 如果执行主发送操作，向数据缓冲区装入要发送的数据字节 (LIN0DT1 ~ LIN0DT8)。
- 5) 将 STREQ 位 (LIN0CTRL) 置 1 以启动报文传送。此后，LIN 外设安排报文帧时序，并在报文发送成功后或发生错误时申请中断。

下面的代码给出了执行发送操作时报文的发送过程。

```
LINADDR = 0x08;           // 指向LIN0CTRL
LINDATA |= 0x20;           // 选择发送数据
LINADDR = 0x0E;           // 指向LIN0ID
LINDATA = 0x11;           // 装载ID，本例中为 0x11
LINADDR = 0x0B;           // 指向LIN0SIZE
LINDATA = (LINDATA & 0xF0) | 0x08; // 装载长度值8

LINADDR = 0x00;           // 指向数据缓冲区第一个字节
for(i=0;i<8;i++)
{
    LINDATA = i + 0x41;     // 向缓冲区中装入'A', 'B', ...
    LINADDR++;             // 缓冲区地址加1
}

LINADDR = 0x08;           // 指向LIN0CTRL
LINDATA = 0x01;           // 启动请求
```

当有中断请求发生时，应用程序应执行下述步骤：

- 1) 检查DONE位 (LIN0ST.0) 和ERROR位 (LIN0ST.2)
- 2) 如果执行的是主接收操作，并且传送成功，从数据缓冲区读取接收的数据。
- 3) 如果传送不成功，检查错误寄存器，以确定错误类型。应用程序必须进行进一步的错误处理。
- 4) 置位RSTINT (LIN0CTRL.3) 和RSTERR (LIN0CTRL.2) 位，以复位中断请求和错误标志。

17.4 LIN 从方式操作

当器件被配置为工作在从方式时，它必须等待来自主节点的命令。只有在产生了数据请求 (DTREQ 位 (LIN0ST.4) 为 1) 并且 LIN 总线不活动 (ACTIVE 位 (LIN0ST.7) 为 0) 的情况下，应用才能访问 LIN 外设的数据缓冲区中的数据 and ID 寄存器。

从方式的 LIN 外设检测 LIN 主器件发送的报文帧的首部。如果从同步被使能 (自动波特率)，则 LIN 从器件将其内部位时间与主器件的位时间同步。

配置为从方式的 LIN 外设出现下面三种情况之一时会产生中断：

- 1) 接收到 IDENTIFIER FIELD 之后。
- 2) 检测到错误时。
- 3) 报文传送完成时。

当检测到中断请求时，应用必须执行下述步骤：

- 1) 检查DTREQ位 (LIN0ST.4) 的状态。该位在收到IDENTIFIER FIELD后被置1。
- 2) 如果DTREQ位 (LIN0ST.4) 被置1，则从LIN0ID寄存器中读标识符并处理。如果DTREQ位 (LIN0ST.4) 未置1，则执行步骤7。
- 3) 如果当前帧是从器件的发送操作，将 TXRX 位 (LIN0CTRL.5) 置 1；如果当前帧是从器件的接收操作，将 TXRX 位 (LIN0CTRL.5) 清 0。
- 4) 向LIN0SIZE寄存器装入数据长度。
- 5) 对于从发送操作，向数据缓冲区装入要发送的数据。
- 6) 将DTACK位 (LIN0CTRL.4) 置1，执行步骤10。
- 7) 如果DTREQ位 (LIN0ST.4) 未被置1，检查DONE位 (LIN0ST.0)。如果DONE位置1，表示发送成功。
- 8) 如果发送成功并且当前帧是从器件的接收操作，则从数据缓冲区接收数据字节。
- 9) 如果发送不成功，检查LIN0ERR，以确定错误发生的原因。应用必须进行进一步的错误处理。
- 10) 置位RSTINT (LIN0CTRL.3) 和RSTERR (LIN0CTRL.2) 位，以复位中断请求和错误标志。

除了上述步骤之外，应用程序还必须考虑下述问题：

- 1) 如果当前帧是一个从器件的发送操作，则步骤1~5必须在IN-FRAME RESPONSE

SPACE期间完成，否则主器件会检测到超时。

- 2) 如果当前帧是从器件的接收操作，则步骤1 ~ 5必须在接收到IDENTIFIER FIELD之后的第一个字节之前完成，否则LIN外设的内部接收缓冲区会被盖写，并且LIN外设会检测到超时错误。
- 3) LIN模块不直接支持LIN 1.3版扩展帧。如果应用检测到一个未知标识符（例如扩展标识符），它必须向STOP位（LIN0CTRL.7）写1而不是将DTACK位（LIN0CTRL.4）置1。此时，步骤2~5可被跳过。在这种情况下，LIN外设停止处理LIN通信，直到接收到下一个SYNC BREAK。
- 4) 在一次传输期间改变校验和配置（基本型到增强型或反之）会导致接口复位和传输丢失。所以当一次传输正在进行时，不应改变校验和设置。该原则同样适用于LIN接口方式的改变，即从从方式到主方式或从主方式到从方式。

17.5 休眠方式和唤醒

为了降低系统功耗，LIN 协议规范定义了一种休眠模式。LIN 主应用以与发送正常报文同样的方式广播一个休眠方式请求报文。LIN 从应用必须从标识符和数据字节对休眠模式帧译码。之后，它必须将 LIN 从节点置于休眠模式，这可通过将 SLEEP 位（LIN0CTRL.6）置 1 来实现。

如果 LIN 从应用的 SLEEP 位（LIN0CTRL.6）未置 1，并且总线处于不活动状态已超过 4 秒（规定的总线空闲超时），则 IDLTOUT 位（LIN0ST.6）被置 1，并会产生中断请求。此后应用程序可以认为 LIN 总线处于休眠模式并将 SLEEP 位（LIN0CTRL.6）置 1。

从主节点或任何一个从节点发送一个唤醒信号将终止 LIN 总线的休眠模式。为了发送唤醒信号，应用必须将 WUPREQ 位（LIN0CTRL.1）置 1。在唤醒信号发送成功后，主节点的 DONE 位（LIN0ST.0）被置 1 并产生一个中断请求。在唤醒信号发送成功后，LIN 从节点不产生中断请求；但如果主节点在 150 ms 内不响应该唤醒信号，则 LIN 从节点将产生中断请求。在这种情况下，ERROR 位（LIN0ST.2）和 TOUT 位（LIN0ERR.2）被置 1。应用必须决定是否发送另一个唤醒信号。

检测到一个唤醒信号的所有 LIN 节点都将 WAKEUP 位（LIN0ST.1）和 DONE 位（LIN0ST.0）置 1 并产生中断请求。此后，应用必须清除 LIN 从器件的控制寄存器中的 SLEEP 位（LIN0CTRL.6）。

17.6 错误检测和处理

LIN 外设检测到错误时会产生中断请求并停止对当前帧的处理。应用必须使用错误寄存器（LIN0ERR）检查错误类型。此后，它还必须通过向 RSTERR 位（LIN0CTRL.2）写 1 来复位错误寄存器和 ERROR 位（LIN0ST.2）。被选择为主器件的 LIN 外设启动一个新报文和被选择为主器件或从器件的 LIN 外设发送一个唤醒信号只在 ERROR 位（LIN0ST.2）被设置为 0 时才有可能。

17.7 LIN 寄存器

下面介绍可用的特殊功能寄存器。

17.7.1 LIN 直接访问 SFR 寄存器定义

SFR 定义 17.1 LINADDR: 间接地址寄存器

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: 0x92
<p>位 7-0: LINADDR[7:0]: LIN 间接地址寄存器位。 该寄存器保持用于间接访问 LIN0 寄存器的 8 位地址。 表 17.4 列出了所有 LIN0 寄存器及其间接地址。LINADDR 位指示读和写 LINDATA 寄存器时的目标寄存器。</p>								

SFR 定义 17.2 LINDATA: LIN 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: 0x93
<p>位 7-0: LINDATA[7:0]: LIN 间接数据寄存器位。 读该寄存器时，读出的是由 LINADDR 指向的 LIN0 寄存器的内容。 写该寄存器时，将值写入由 LINADDR 指向的 LIN0 寄存器。</p>								

SFR 定义 17.3 LINCf: 方式控制寄存器

R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	复位值
LINEN	MODE	ABAUD						00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x95
<p>位 7: LINEN: LIN 接口使能位 0: LIN0 被禁止。 1: LIN0 被使能。</p> <p>位 6: MODE: LIN 方式选择位 0: LIN0 工作在从方式。 1: LIN0 工作在主方式。</p> <p>位 5: ABAUD: LIN 方式自动波特率选择（仅从方式） 0: 手动波特率选择被使能。 1: 自动波特率选择被使能。</p>								

17.3.2 LIN 间接访问 SFR 寄存器定义

表 17.2 LIN 寄存器*（间接寻址）

名称	地址	位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0
LIN0DT1	0x00	DATA1[7:0]							
LIN0DT2	0x01	DATA 2[7:0]							
LIN0DT3	0x02	DATA 3[7:0]							
LIN0DT4	0x03	DATA 4[7:0]							
LIN0DT5	0x04	DATA 5[7:0]							
LIN0DT6	0x05	DATA 6[7:0]							
LIN0DT7	0x06	DATA 7[7:0]							
LIN0DT8	0x07	DATA 8[7:0]							
LIN0CTRL	0x08	STOP(s)	SLEEP(s)	TXRX	DTACK(s)	RSTINT	RSTERR	WUPREQ	STREQ(m)
LIN0ST	0x09	ACTIVE	IDLTOUT	ABORT(s)	DTREQ(s)	LININT	ERROR	WAKEUP	DONE
LIN0ERR	0x0A				SYNCH(s)	PRTY(s)	TOUT	CHK	BITERR
LIN0SIZE	0x0B	ENHCHK				LINSIZE[3:0]			
LIN0DIV	0x0C	DIVLSB[7:0]							
LIN0MUL	0x0D	PRESCL[1:0]		LINMUL[4:0]					DIV9
LIN0ID	0x0E			ID[5:0]					

*在主方式和从方式都要使用这些寄存器。标记(m)的寄存器只能在主方式访问，标记(s)的寄存器只能在从方式访问。所有其它寄存器在两种方式下都可访问。

SFR 定义 17.4 LIN0DT1: LIN0 数据字节 1

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	
								地址: 0x00 (间接)
位 7-0: LIN0DT1[7:0]: LIN 数据字节 1								
LIN 接口发送和接收的串行数据字节 1。								

SFR 定义 17.5 LIN0DT2: LIN0 数据字节 2

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	
								地址: 0x01 (间接)
位 7-0: LIN0DT2[7:0]: LIN 数据字节 2								
LIN 接口发送和接收的串行数据字节 2。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 17.6 LIN0DT3: LIN0 数据字节 3

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	

地址: 0x02 (间接)

位 7-0: LIN0DT3[7:0]: LIN 数据字节 3
LIN 接口发送和接收的串行数据字节 3。

SFR 定义 17.7 LIN0DT4: LIN0 数据字节 4

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	

地址: 0x03 (间接)

位 7-0: LIN0DT4[7:0]: LIN 数据字节 4
LIN 接口发送和接收的串行数据字节 4。

SFR 定义 17.8 LIN0DT5: LIN0 数据字节 5

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	

地址: 0x04 (间接)

位 7-0: LIN0DT5[7:0]: LIN 数据字节 5
LIN 接口发送和接收的串行数据字节 5。

SFR 定义 17.9 LIN0DT6: LIN0 数据字节 6

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	

地址: 0x05 (间接)

位 7-0: LIN0DT6[7:0]: LIN 数据字节 6
LIN 接口发送和接收的串行数据字节 6。

SFR 定义 17.10 LIN0DT7: LIN0 数据字节 7

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	
								地址: 0x06 (间接)
位 7-0: LIN0DT7[7:0]: LIN 数据字节 7								
LIN 接口发送和接收的串行数据字节 7。								

SFR 定义 17.11 LIN0DT8: LIN0 数据字节 8

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	
								地址: 0x07 (间接)
位 7-0: LIN0DT8[7:0]: LIN 数据字节 8								
LIN 接口发送和接收的串行数据字节 8。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 17.12 LIN0CTRL: LIN0 控制寄存器

W	W	W	R/W	R/W	R/W	R/W	R/W	复位值
STOP	SLEEP	TXRX	DTACK	RSTINT	RSTERR	WUPREQ	STREQ	00000000
位7	位6	位5	位4	位3	位2	位1	位0	
地址: 0x08 (间接)								
位 7:	STOP: 停止 LIN 通信处理位 (仅从方式)。 该位由应用置 1 以阻塞 LIN 通信处理, 直到下一个 SYNCH BREAK 信号。该位在应用正忙于处理数据请求中断而不能使用对应接收标识符的帧内容时使用(读出值总是为 0)。							
位 6:	SLEEP: 休眠方式告警。 该位由应用置 1 以告知外设收到了一个休眠方式帧并且总线处于休眠方式, 或者产生了总线空闲超时中断请求。当发生唤醒中断请求时, 应用必须将其复位。							
位 5:	TXRX: 发送/接收选择位。 该位决定当前帧是发送帧还是接收帧。 0: 当前帧是接收操作。 1: 当前帧是发送操作。							
位 4:	DTACK: 数据确认位 (仅从方式)。 该位在处理完一个数据请求中断后置 1, 对数据传送进行确认。该位由 LIN 控制器自动清 0。							
位 3:	RSTINT: 中断复位控制位。 该位的读出值总是为 0。 0: 不起作用。 1: 复位 LININT 位 (LIN0ST.3)。							
位 2:	RSTERR: 错误复位控制位。 该位的读出值总是为 0。 0: 不起作用。 1: 复位 LIN0ST 和 LIN0ERR 中的错误标志位。							
位 1:	WUPREQ: 唤醒请求位 该位置 1 结束休眠方式, 通过发送一个唤醒信号实现。该位由 LIN 控制器自动清 0。							
位 0:	STREQ: 开始请求位 (仅主方式)。 1: 开始一次 LIN 发送。该位只应在装载完标识符、数据长度和数据缓冲区 (如果需要) 之后被置 1。 在发送结束或检测到错误时, 该位被清 0。							

SFR 定义 17.13 LIN0ST: LIN0 状态寄存器

R	R	R	R	R/W	R	R	R	复位值
ACTIVE	IDLTOUIT	ABORT	DTREQ	LININT	ERROR	WAKEUP	DONE	00000000
位7	位6	位5	位4	位3	位2	位1	位0	
地址: 0x09 (间接)								
<p>位 7: ACTIVE: LIN 总线活动状态位。 0: 在 LIN 总线上未检测到传输活动。 1: 在 LIN 总线上检测到传输活动。</p> <p>位 6: IDLTOUT: 总线空闲超时位 (仅从方式)。 0: 总线空闲时间不足 4 秒。 1: 已经有 4 秒钟未检测到总线活动, 而且总线不处于休眠方式。</p> <p>位 5: ABORT: 放弃传输信号位 (仅从方式)。 0: 当前传输没有被中断或停止。该位在接收到一个不中断当前传输的 SYNCH BREAK 后被清 0。 1: 在最后一次传输结束之前或 STOP 位 (LIN0CTRL.7) 被置 1 之前检测到一个新的 SYNCH BREAK 信号。</p> <p>位 4: DTREQ: 数据请求位 (仅从方式)。 0: 未接收到数据标识符。 1: 已接收到数据标识符。</p> <p>位 3: LININT: 中断请求位。 0: 没有中断请求发生。该位通过置位 RSTINT 位 (LIN0CTRL.3) 清 0。 1: 有 LIN0 中断请求发生。</p> <p>位 2: ERROR: 通信错误位。 0: 没有检测到错误。该位通过置位 RSTERR 位 (LIN0CTRL.2) 清 0。 1: 检测到错误。</p> <p>位 1: WAKEUP: 唤醒位 0: 未发送也没接收到一个唤醒信号。 1: 正在发送或已经接收到一个唤醒信号。</p> <p>位 0: DONE: 发送完成位。 0: 不在发送或没有启动发送。该位在开始发送时被清 0。 1: 当前发送完成。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 17.14 LIN0ERR: LIN0 错误寄存器

R	R	R	R	R	R	R	R	复位值
			SYNCH	PRTY	TOUT	CHK	BITERR	00000000
位7	位6	位5	位4	位3	位2	位1	位0	
地址: 0x0A (间接)								
位 7-5: 未用。读 = 000b; 写 = 忽略。								
位 4: SYNCH: 同步错误位 (仅从方式)。								
0: 在 SYNCH FIELD 未检测到错误。								
1: SYNCH FIELD 的边沿超出了最大允许范围。								
位 3: PRTY: 校验位错误位 (仅从方式)。								
0: 未检测到校验位错误。								
1: 检测到校验位错误。								
位 2: TOUT: 超时错误位。								
0: 未检测到超时错误。								
1: 检测到超时错误。在出现下列条件之一时会检测到该错误:								
• 主器件等待来自从器件的数据, 但从器件没有回应。								
• 从器件期望从总线上接收数据, 但总线上没有数据。								
• 一帧数据未能在最大帧长度内结束传输。								
• 应用直到标识符后的第一个字节接收完都未将 DTACK 位 (LIN0CTRL.4) 或 STOP 位 (LIN0CTRL.7) 置 1。								
位 1: CHK: 校验和错误位。								
0: 未检测到校验和错误。								
1: 检测到校验和错误。								
位 0: BITERR: 位发送错误位。								
0: 未检测到发送错误。								
1: 在发送期间监视到的位值与发送的位值不同。								

SFR 定义 17.15 LIN0SIZE: LIN0 报文长度寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
ENHCHK	—	—	—	LINSIZE[3:0]				00000000
位7	位6	位5	位4	位3	位2	位1	位0	
地址: 0x0B (间接)								
位 7: ENHCHK: 校验和选择位。								
0: 使用符合规范 1.3 的基本校验和。校验和涵盖数据字节。								
1: 使用符合规范 2.0 的增强校验和。校验和涵盖数据字节及受保护的标识符。								
位 6-4: 未用。读 = 000b; 写 = 忽略。								
位 3-0: LINSIZE3-0: LIN 数据域长度。								
0000: 0 数据字节								
0001: 1 数据字节								
0010: 2 数据字节								
0011: 3 数据字节								
0100: 4 数据字节								
0101: 5 数据字节								
0110: 6 数据字节								
0111: 7 数据字节								
1000: 8 数据字节								
1001-1110: 保留								
1111: 使用 ID[1:0]位 (LIN0ID[5:4]) 确定数据长度。								

SFR 定义 17.16 LIN0DIV: LIN0 分频系数寄存器

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	
地址: 0x0C (间接)								
位 7-0: DIVLSB[7:0]: LIN 波特率分频系数低位。								
该寄存器包含波特率分频系数的低 8 位, 第 9 位或最高位是 DIV9 位 (LIN0MUL.0)。分频系数的合法范围是 200 ~ 511。								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 17.17 LIN0MUL: LIN0 倍频系数寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
PRESCL[1:0]		LINMUL[4:0]					DIV9	00000000
位7	位6	位5	位4	位3	位2	位1	位0	

地址: 0x0D (间接)

位 7-6: PRESCL[1:0]: LIN 波特率预分频位。
这些位是波特率预分频位。

位 5-1: LINMUL[4:0]: LIN 波特率倍频系数位。
这些位是波特率倍频系数位。从方式不使用这些位。

位 0: DIV9: LIN 波特率分频系数的最高位。
这是波特率分频系数的最高位。分频系数的低 8 位在 LIN0DIV 中。分频系数的合法范围是 200 ~ 511。

SFR 定义 17.18 LIN0ID: LIN0 ID 寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
		ID[5:0]						00000000
位7	位6	位5	位4	位3	位2	位1	位0	

地址: 0x0E (间接)

位 7-6: 未用。读 = 00b; 写 = 忽略。

位 5-0: ID[5:0]: LIN 标识符位。
如果 LINSIZE 位 (LIN0SIZE[3:0]) 是 1111b, 则位 ID[5:4]用于确定数据长度, 其意义如下:
00: 2 字节
01: 2 字节
10: 4 字节
11: 8 字节

18. 增强型串行外设接口（SPI0）

串行外设接口（SPI0）提供一个灵活的、全双工同步串行总线。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 SPI 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。

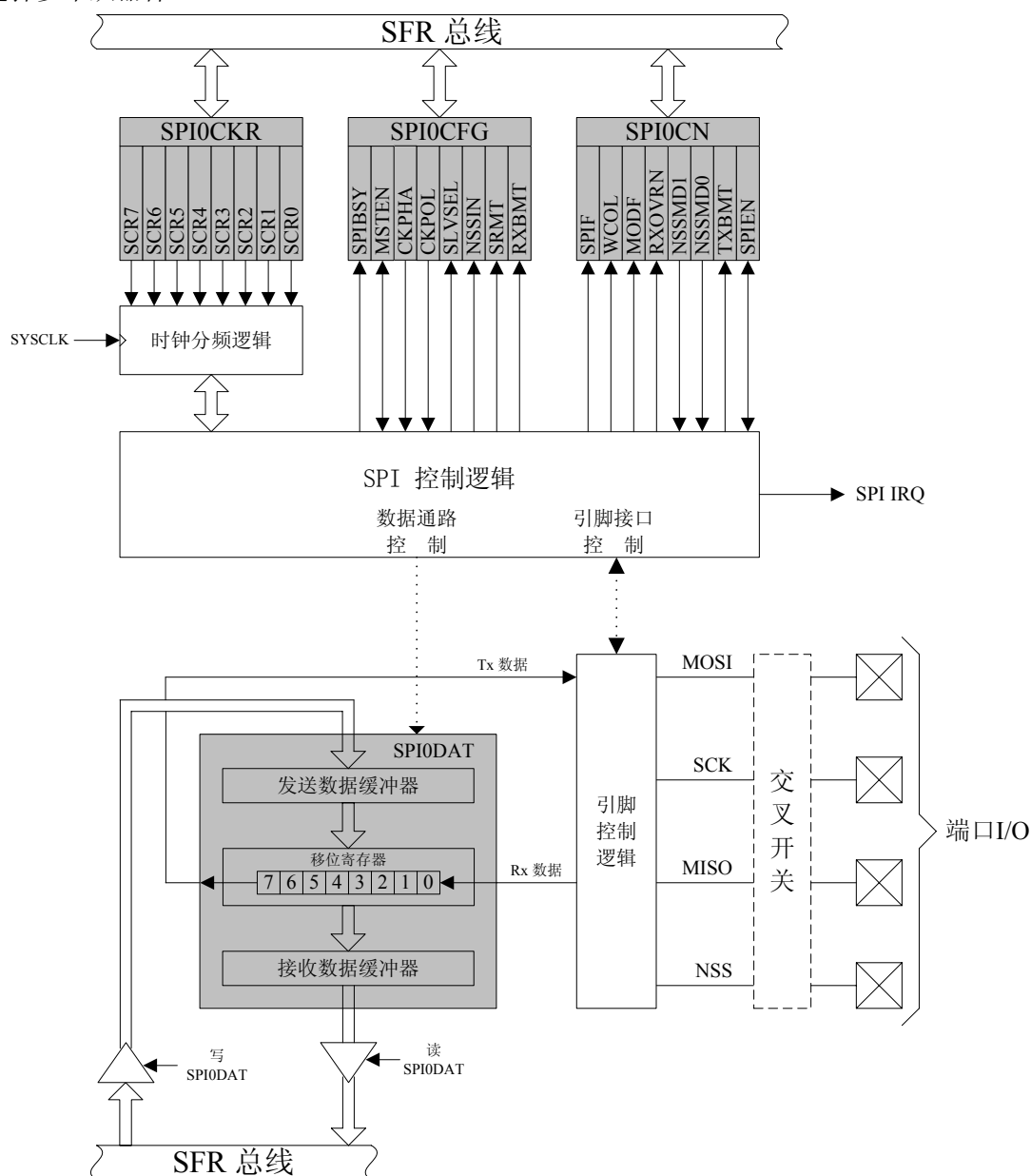


图 18.1 SPI 原理框图

18.1 信号说明

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

18.1.1 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的串行数据传输。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。

18.1.2 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的串行数据传输。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。

18.1.3 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。

18.1.4 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。有 3 种可能的方式：

1. NSSMD[1:0] = 00：3 线主方式或从方式：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 线方式时必须是在总线唯一的从器件。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 作为输入。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 作为输出。NSSMD0 的设置值决定 NSS 引脚的输出逻辑电平。这种配置只能在 SPI0 作为主器件时使用。

图 18.2 ~ 18.4 给出了不同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 必须被映射到器件引脚。

18.2 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, SPI0CN.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 SPI0 主器件在 MOSI 线向从器件发送数据时, 被寻址的 SPI 从器件可以同时从 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移位寄存器时, 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 (MODEF, SPI0CN.5) 被置 1。如果中断被允许, 将产生中断。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 引脚对从器件单独寻址 (如果需要)。图 18.2 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 18.3 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 18.4 给出了一个 4 线主方式主器件和两个从器件的连接图。

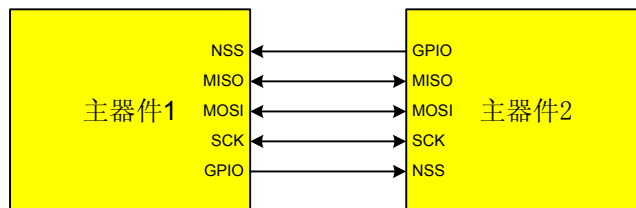


图 18.2 多主方式连接图

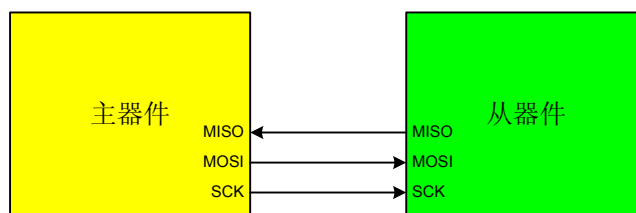


图 18.3 3 线单主方式和 3 线单从方式连接图

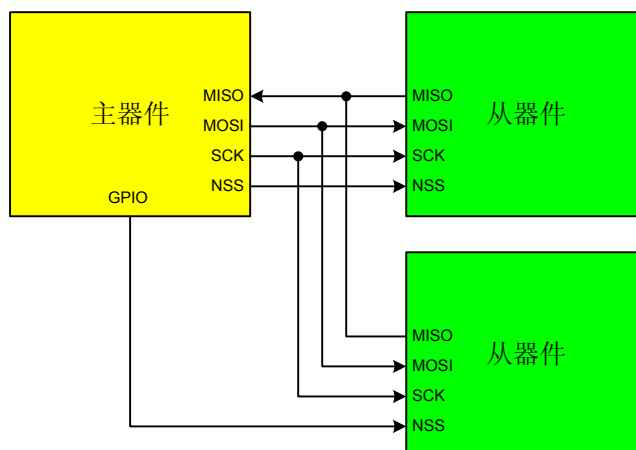


图 18.4 4 线单主方式和 4 线从方式连接图

18.3 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，接收到的字节被复制到接收缓冲器。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送。通过写 SPI0DAT 来预装要发送给主器件的数据到移位寄存器。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。

在从器件检测到第一个 SCK 边沿后，移位寄存器中的内容被锁定。在第一个 SCK 边沿之后写入 SPI0DAT 的内容被保持在发送锁存器中，直到当前传输结束。

当被配置为从器件时，SPI0 可以工作在 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 18.4 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 18.3 给出了一个 3 线从器件和一个主器件的连接图。

18.4 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 SPI0 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 SPI0 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。

18.5 串行时钟时序

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。注意: 在改变时钟相位和极性期间应禁止 SPI0 (通过清除 SPIEN 位, SPI0CN.0)。时钟和数据线的时序关系示于图 18.5。

SFR 定义 18.3 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。

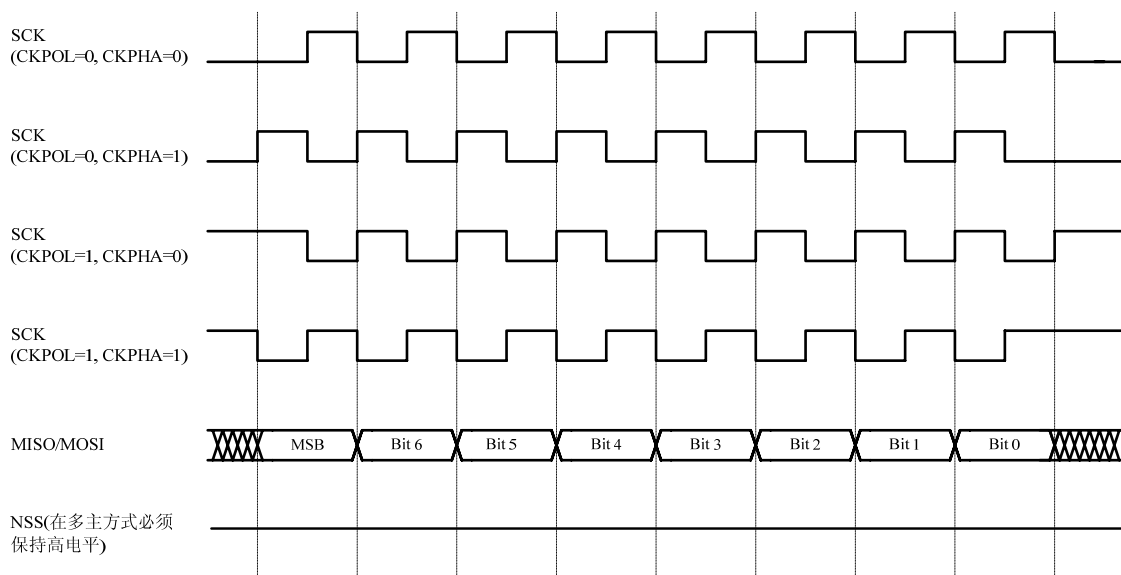


图 18.5 数据/时钟时序图

18.6 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 18.1 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: 主方式允许位

0: 禁止主方式, 工作在从方式。

1: 允许主方式, 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, 表示 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 引脚的瞬时值 (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 (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: 在主方式时, RXBMT = 1。

*注: 时序参数见表 18.1。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 18.2 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
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, 置 1 该位将会使 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。							
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), 表示数据传送期间对 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。							
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。							
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。							
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 总是器件的输入。 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。							
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。							
位 0:	SPIEN: SPI0 使能位 该位使能 / 禁止 SPI0。 0: SPI0 禁止。 1: SPI0 使能。							

SFR 定义 18.3 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: *SYSCCLK* 是系统时钟频率, *SPI0CKR* 是 SPI0CKR 寄存器中的 8 位值。

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

(0 ≤ SPI0CKR ≤ 255)

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

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

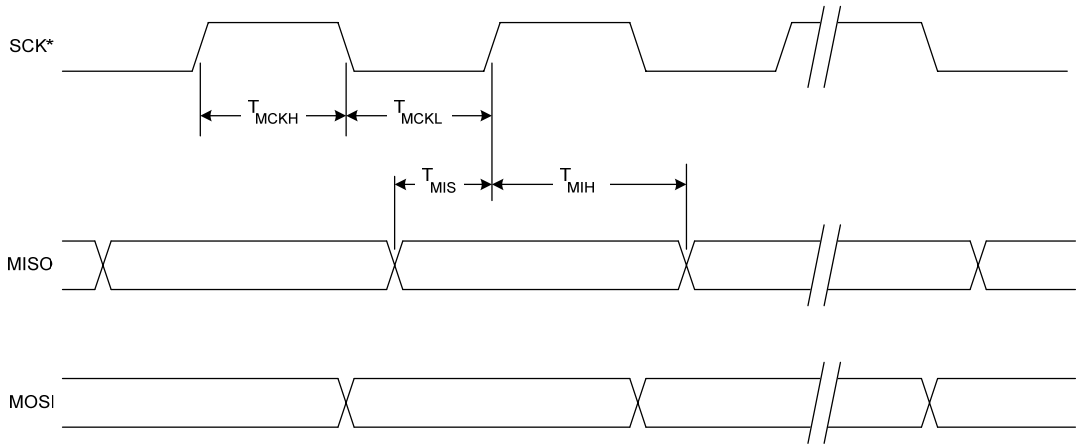
$$f_{SCK} = 200kHz$$

SFR 定义 18.4 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

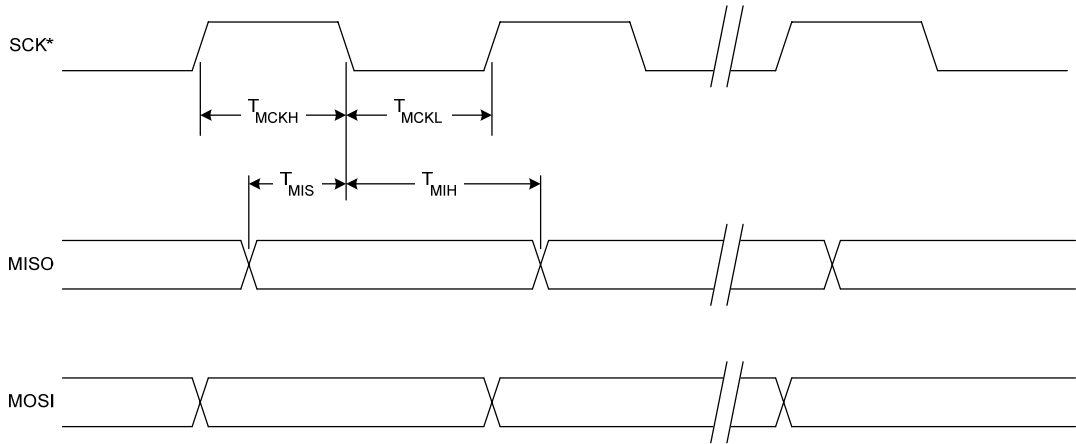
位 7-0: SPI0DAT: SPI0 发送和接收数据寄存器。

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。



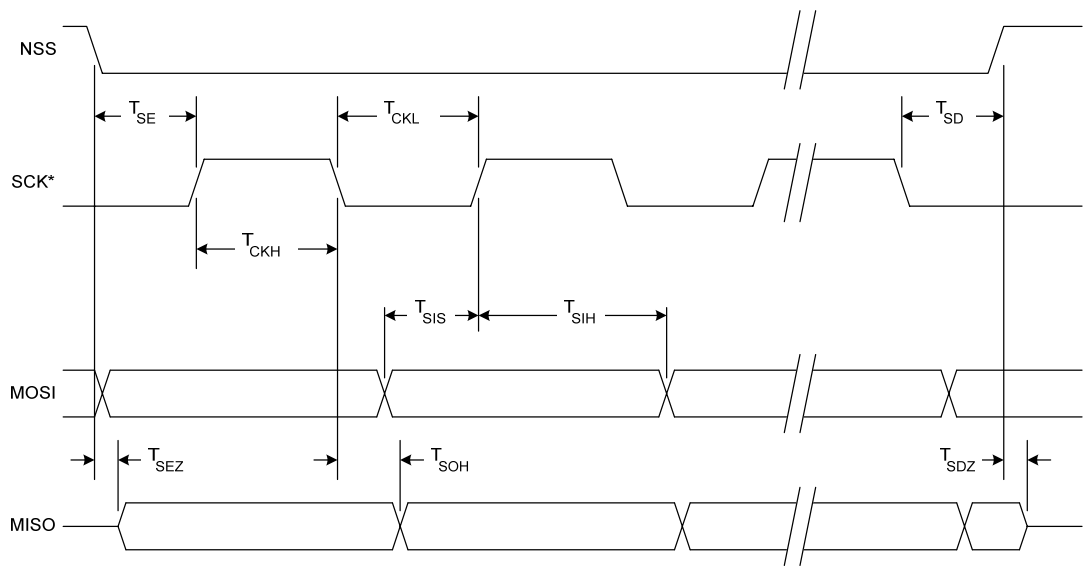
* 这是对应 CKPOL = 0时的 SCK 波形。对于 CKPOL = 1, SCK波形的极性反向。

图 18.6 SPI 主方式时序 (CKPHA = 0)



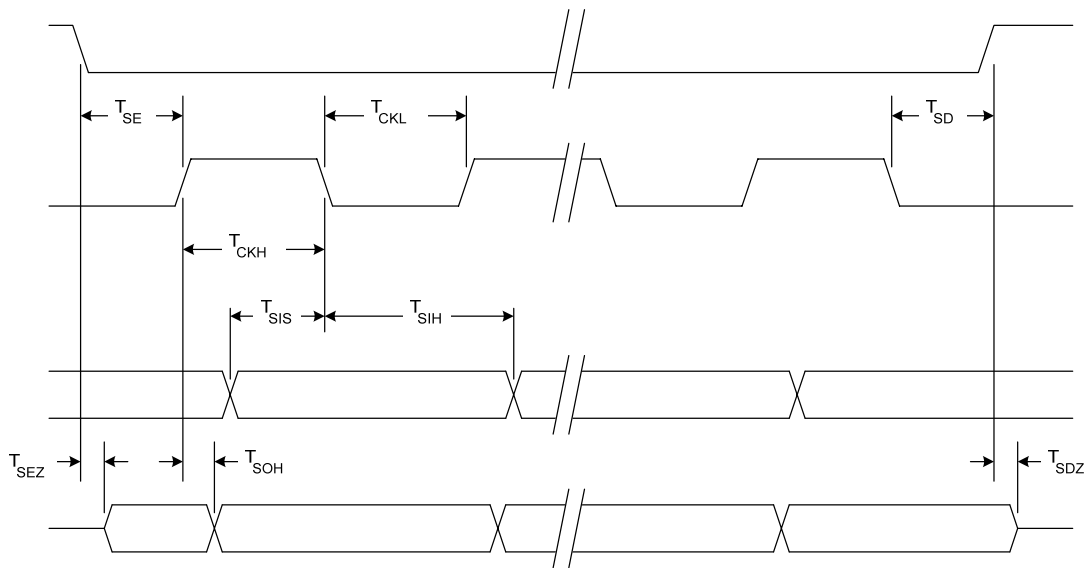
* 这是对应 CKPOL = 0时的 SCK 波形。对于 CKPOL = 1, SCK波形的极性反向。

图 18.7 SPI 主方式时序 (CKPHA = 1)



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

图 18.8 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 18.9 SPI 从方式时序 (CKPHA = 1)

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

表 18.1 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 18.6 和图 18.7）				
T _{MCKH}	SCK 高电平时间	1×T _{SYSCLK}	—	ns
T _{MCKL}	SCK 低电平时间	1×T _{SYSCLK}	—	ns
T _{MIS}	MISO 有效到 SCK 采样边沿	20	—	ns
T _{MIH}	SCK 采样边沿到 MISO 发生改变	0	—	ns
从方式时序*（见图 18.8 和图 18.9）				
T _{SE}	NSS 下降沿到第一个 SCK 边沿	2×T _{SYSCLK}	—	ns
T _{SD}	最后一个 SCK 边沿到 NSS 上升沿	2×T _{SYSCLK}	—	ns
T _{SEZ}	NSS 下降沿到 MISO 有效	—	4×T _{SYSCLK}	ns
T _{SDZ}	NSS 上升沿到 MISO 变为高阻态	—	4×T _{SYSCLK}	ns
T _{CKH}	SCK 高电平时间	5×T _{SYSCLK}	—	ns
T _{CKL}	SCK 低电平时间	5×T _{SYSCLK}	—	ns
T _{SIS}	MOSI 有效到 SCK 采样边沿	2×T _{SYSCLK}	—	ns
T _{SIH}	SCK 采样边沿到 MOSI 发生改变	2×T _{SYSCLK}	—	ns
T _{SOH}	SCK 移位边沿到 MISO 发生改变	—	4×T _{SYSCLK}	ns
*注：T _{SYSCLK} 为一个系统时钟（SYSCLK）周期（ns）。 SPI 的最大可能频率可计算如下： 发送：SYSCLK/2 接收：SYSCLK/10				

19. 定时器

C8051F52x/52xA/53x/53xA 内部有 3 个 16 位计数器/定时器：其中两个与标准 8051 中的计数器/定时器兼容，另外一个为 16 位自动重载定时器，可用于其他外设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 可作为一个 16 位或两个 8 位自动重载定时器。

定时器 0 和定时器 1 工作方式	定时器 2 工作方式
13 位计数器/定时器	16 位自动重载定时器
16 位计数器/定时器	
8 位自动重载的计数器/定时器	两个 8 位自动重载定时器
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 (见 SFR 定义 19.3)。

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

定时器 0 和定时器 1 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。

19.1 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。

19.1.1 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 19.1 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，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

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

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/T 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 19.3) 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 9.11), 置位 TR0 位 (TCON.4) 将允许定时器 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 11.5)。

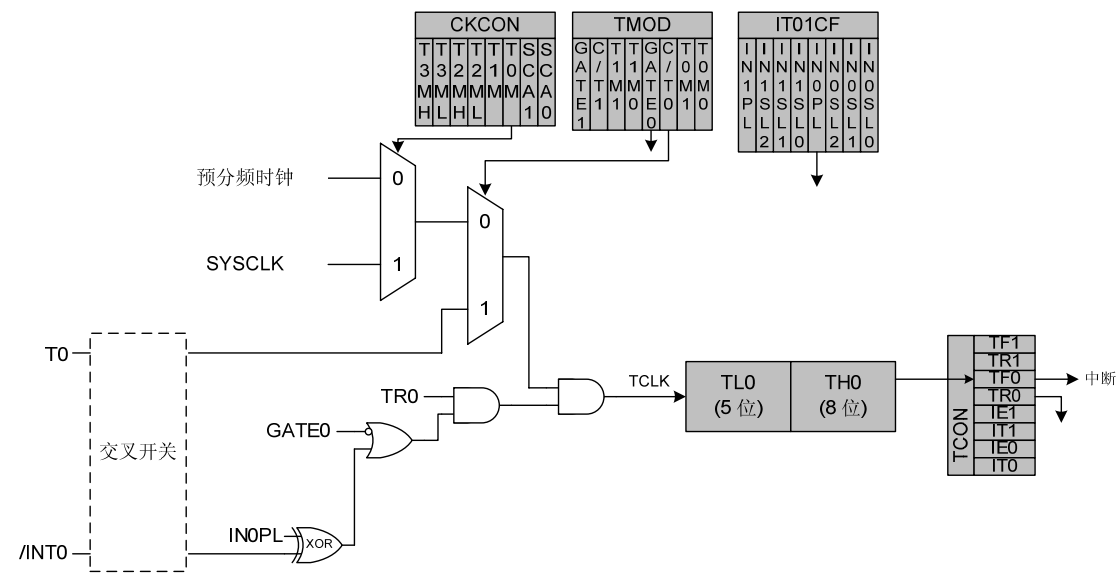


图 19.1 T0 方式 0 原理框图

19.1.2 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。

19.1.3 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，见“11.5 外部中断”），置位 TR0 位（TCON.4）将允许定时器 0 工作。

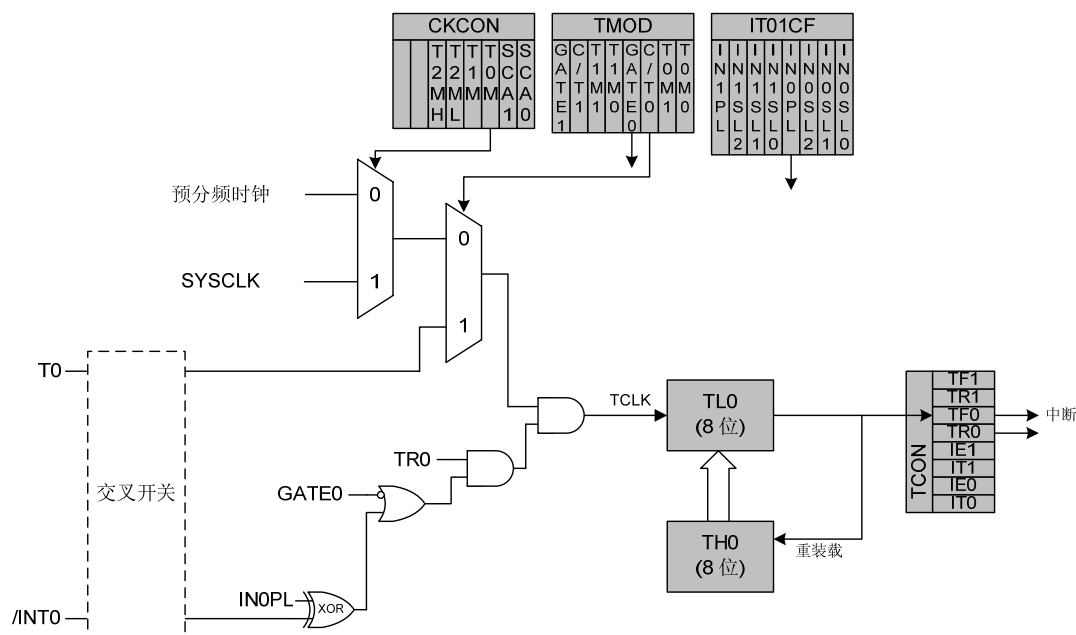


图 19.2 T0 方式 2 原理框图

19.1.4 方式3 — 两个8位计数器/定时器（仅定时器0）

在方式 3，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。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, 应使定时器 1 工作在方式 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。

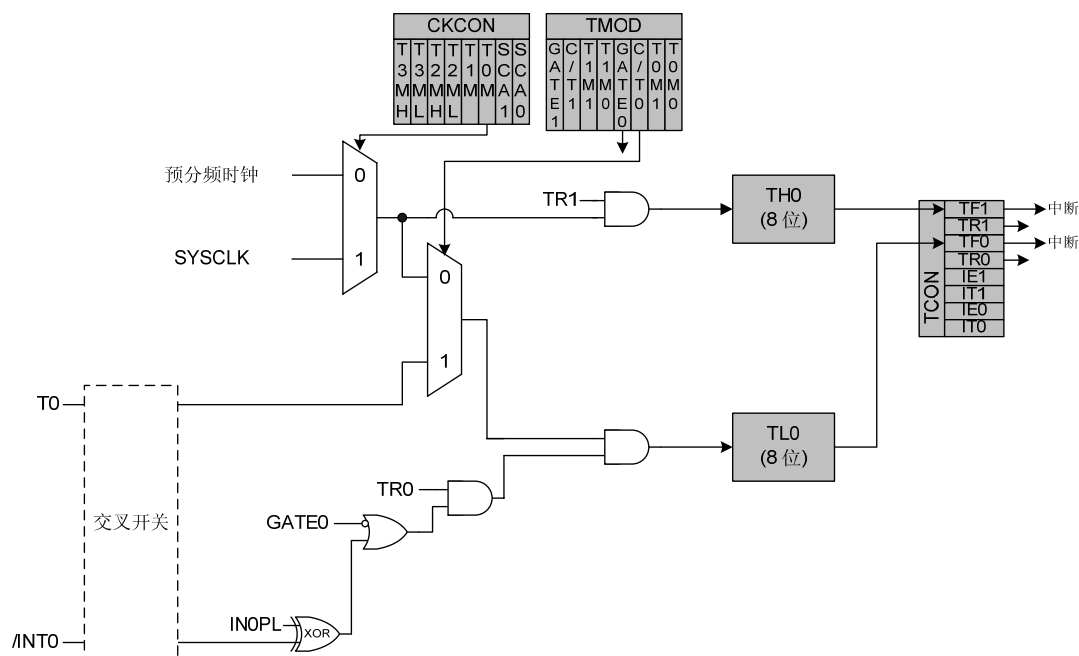


图 19.3 T0 方式 3 原理框图

SFR 定义 19.1 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
<p>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。 0: 未检测到定时器 1 溢出。 1: 定时器 1 发生溢出。</p> <p>位 6: TR1: 定时器 1 运行控制 0: 定时器 1 禁止。 1: 定时器 1 允许。</p> <p>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。 0: 未检测到定时器 0 溢出。 1: 定时器 0 发生溢出。</p> <p>位 4: TR0: 定时器 0 运行控制 0: 定时器 0 禁止。 1: 定时器 0 允许。</p> <p>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)。</p> <p>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。 0: /INT1 为电平触发。 1: /INT1 为边沿触发。</p> <p>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)。</p> <p>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。 0: /INT0 为电平触发。 1: /INT0 为边沿触发。</p>								

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 19.2 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。

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。

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

SFR 定义 19.3 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															
-	-	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															
<p>位 7-6: 保留。读 = 00b; 必须写 00b。</p>																							
<p>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。</p>																							
<p>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。</p>																							
<p>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 使用由预分频位 (SCA1-SAC0) 定义的时钟。 1: 定时器 1 使用系统时钟</p>																							
<p>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 使用由预分频位 (SCA1-SAC0) 定义的时钟。 1: 定时器 0 使用系统时钟</p>																							
<p>位 1-0: SCA1-SCA0: 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, 则这些位控制时钟分频数。</p>																							
<table><tr><th>SCA1</th><th>SCA0</th><th>预分频时钟</th></tr><tr><td>0</td><td>0</td><td>系统时钟/12</td></tr><tr><td>0</td><td>1</td><td>系统时钟/4</td></tr><tr><td>1</td><td>0</td><td>系统时钟/48</td></tr><tr><td>1</td><td>1</td><td>外部时钟/8</td></tr></table>									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
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																					
<p>注: 外部时钟 8 分频与系统时钟同步。</p>																							

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 19.4 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
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x8A
位 7-0: TL0: 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 19.5 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
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x8B
位 7-0: TL1: 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 19.6 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
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x8C
位 7-0: TH0: 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 19.7 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
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	SFR地址: 0x8D
位 7-0: TH1: 定时器 1 高字节								
TH1 寄存器是 16 位定时器 1 的高字节。								

19.2 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 还可被用于捕捉方式，以测量 RTC0 时钟频率或外部振荡器时钟频率。

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。外部振荡源时钟/8 与系统时钟同步。

19.2.1 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 19.4）。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，并将定时器 2 高字节溢出标志 TF2H（TMR2CN.7）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。

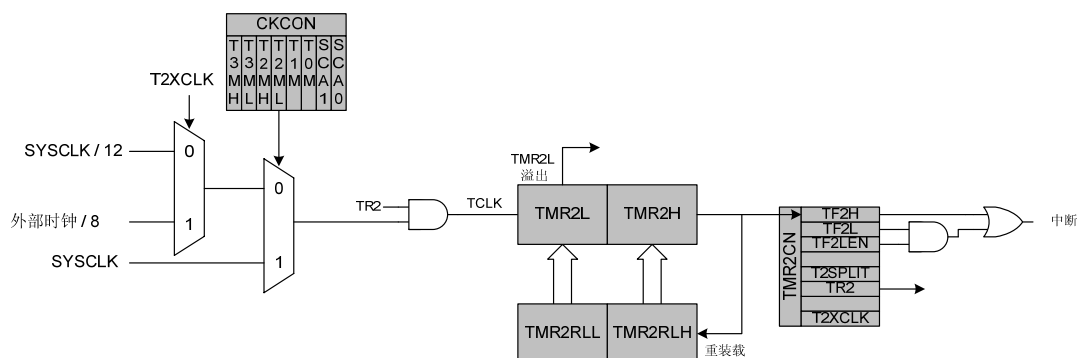


图 19.4 定时器 2 的 16 位方式原理框图

19.2.2 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（见图 19.5）。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，软件必须检查 TF2H 和 TF2L 标志，以确定定时器 2 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。

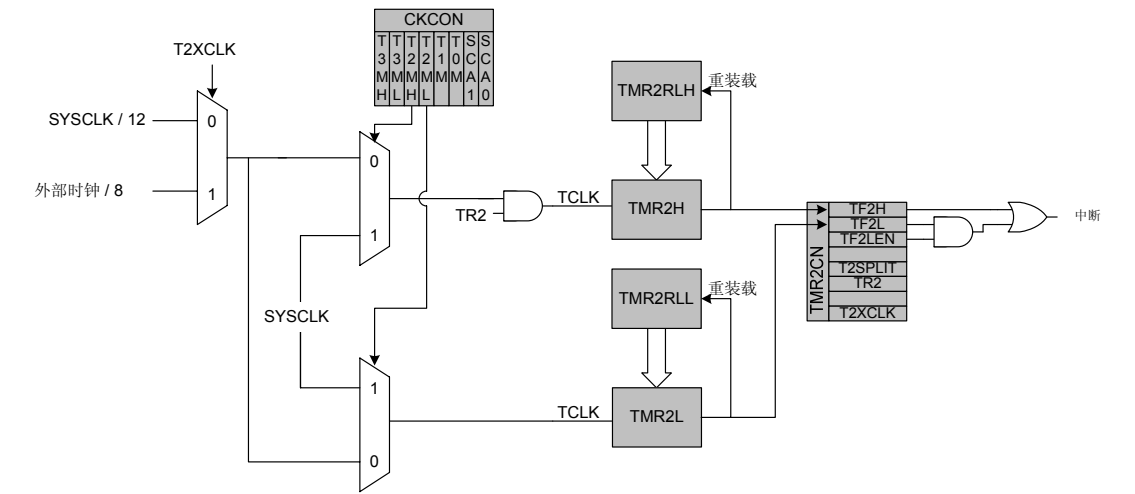


图 19.5 定时器 2 的 8 位方式原理框图

19.2.3 外部捕捉方式

外部捕捉方式允许使用系统时钟测量外部振荡器。定时器 2 可以使用系统时钟、系统时钟/12，由 T2ML (CKCON.4) 和 T2XCLK 的设置决定。当捕捉事件产生时，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，可以确定外部振荡器的频率（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捕捉时钟的频率。当使用捕捉方式时，定时器 2 应工作在 16 位自动重载方式。

例如，如果 T2ML = 1 并且 TF2CEN = 1，则定时器 2 每个 SYSCLK 加 1，每个外部时钟/8 发生一次捕捉。如果 SYSCLK 为 24.5 MHz，两次相邻捕捉的差值是 5984，则外部时钟频率为：

$$\frac{24.5MHz}{(5984/8)} = 0.032754MHz = 32.754kHz$$

当使用 RC 网络或电容产生时钟源时，该方式允许软件确定外部振荡器的频率。

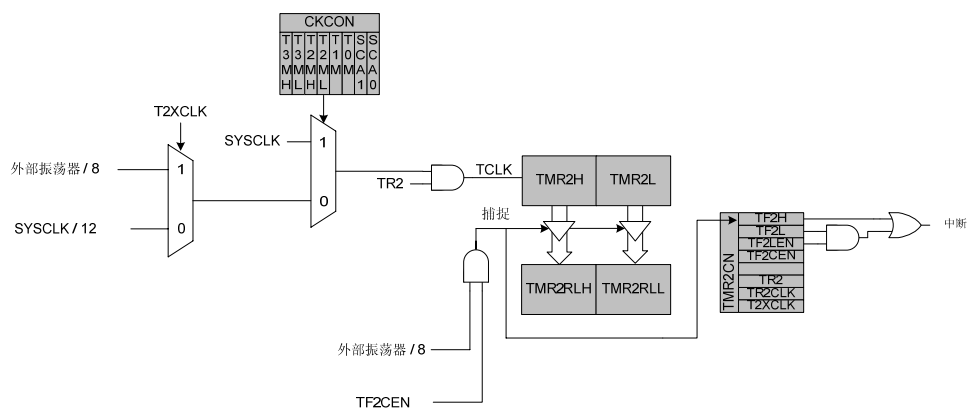


图 24.6 定时器 2 低频振荡器捕捉方式原理框图

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 19.8 TMR2CN: 定时器 2 控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	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
位 7:	TF2H: 定时器 2 高字节溢出标志 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。							
位 6:	TF2L: 定时器 2 低字节溢出标志 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。							
位 5:	TF2LEN: 定时器 2 低字节中断允许位 该位允许/禁止定时器 2 低字节中断。如果 TF2LEN 被置 1 并且定时器 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。 0: 禁止定时器 2 低字节中断。 1: 允许定时器 2 低字节中断。							
位 4:	TF2CEN: 定时器 2 捕捉使能位 0: 禁止定时器 2 捕捉方式。 1: 使能定时器 2 捕捉方式。							
位 3:	T2SPLIT: 定时器 2 双 8 位方式使能位 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。 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。							
位 2:	TR2: 定时器 2 运行控制 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。 0: 定时器 2 禁止。 1: 定时器 2 允许。							
位 1:	未用: 读 = 0b, 写 = 忽略。							
位 0:	T2XCLK: 定时器 2 外部时钟选择 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。 0: 定时器 2 外部时钟为系统时钟/12。 1: 定时器 2 外部时钟使用外部时钟/8。							

SFR 定义 19.9 TMR2RLL: 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: TMR2RLL: 定时器 2 重载寄存器的低字节 TMR2RLL 保持定时器 2 重载值的低字节。								

SFR 定义 19.10 TMR2RLH: 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: TMR2RLH: 定时器 2 重载寄存器的高字节 TMR2RLH 保持定时器 2 重载值的高字节。								

SFR 定义 19.11 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: 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 19.12 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: 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。								

20. 可编程计数器阵列（PCA0）

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线在被使能时可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6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。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20.1。

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“20.3 看门狗定时器方式”。

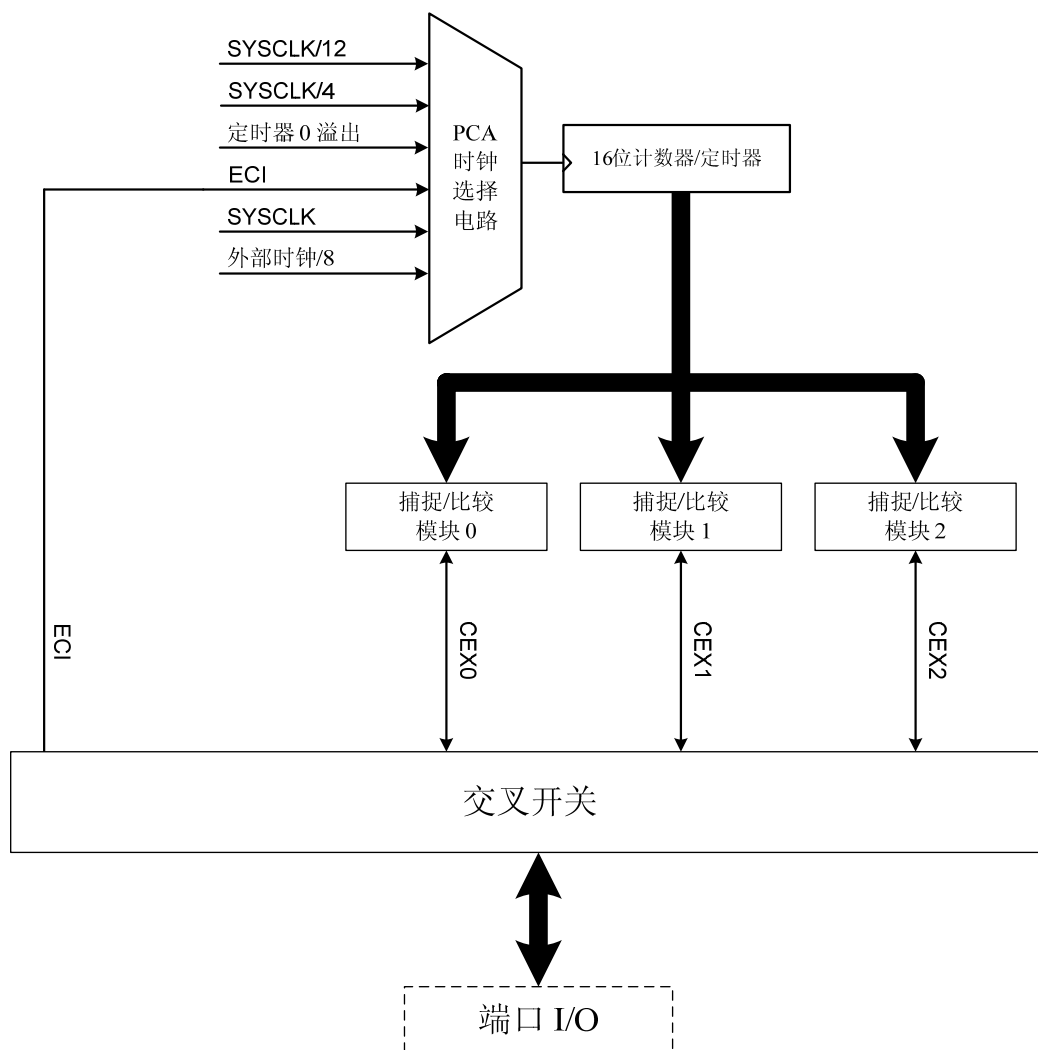


图 20.1 PCA 原理框图

20.1 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 20.1 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。注意，要使 CF 中断得到响应，必须先全局允许 PCA0 中断。通过将 EA 位（IE.7）和 EPCA0 位（EIE1.4）设置为逻辑 1 来全局允许 PCA0 中断。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 20.1 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*

*注：外部振荡器 8 分频与系统时钟同步。

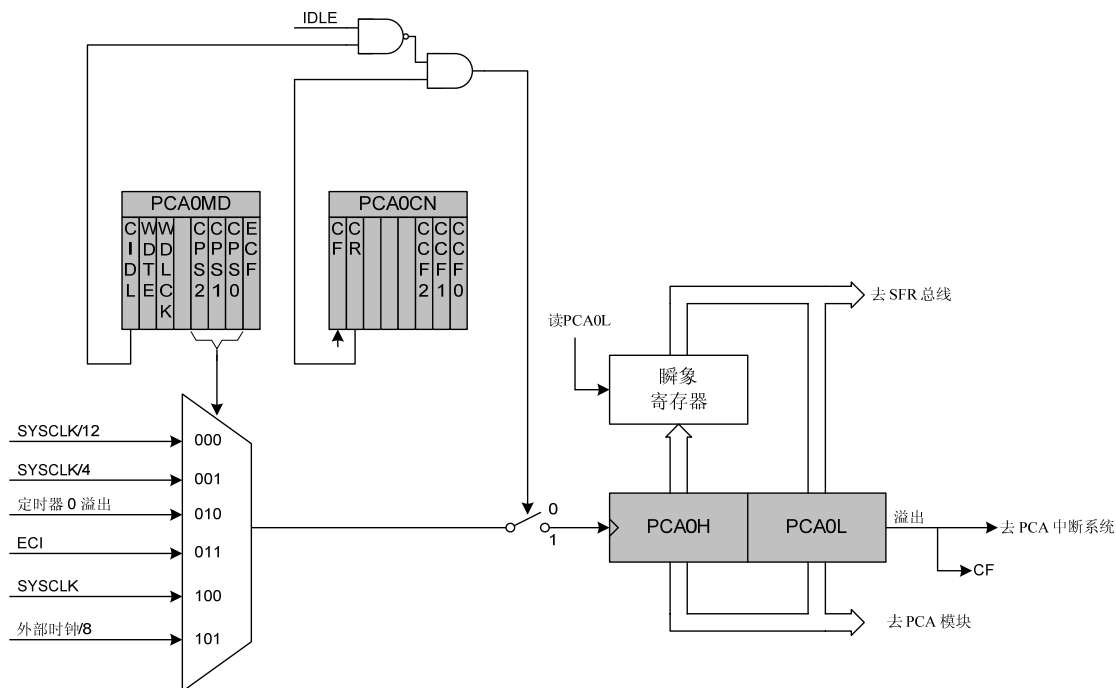


图 20.2 PCA 计数器/定时器原理框图

20.2.1 边沿触发的捕捉方式

在该方式，CEXn引脚上出现的电平跳变导致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并产生一个中断请求（如果CCF中断被允许）。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，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。

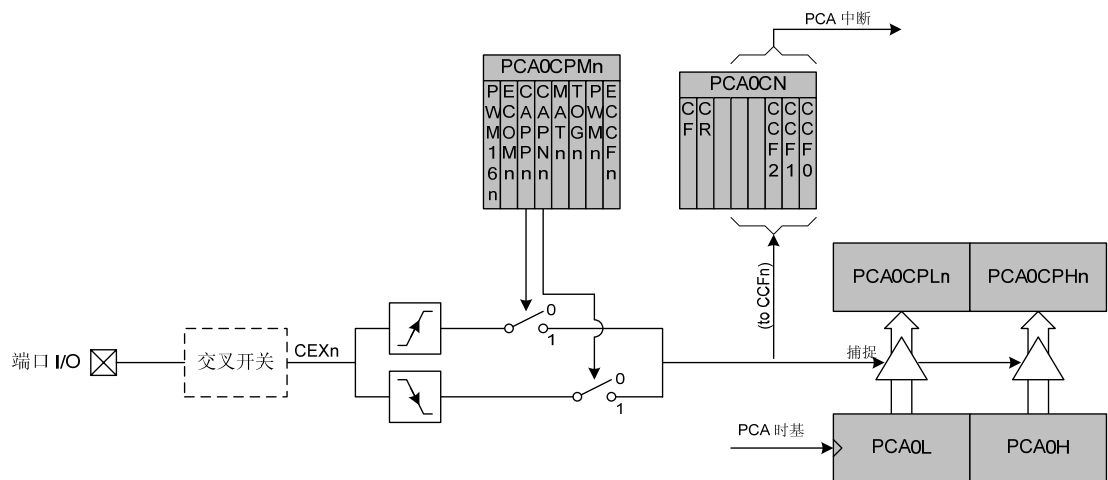


图20.4 PCA捕捉方式原理框图

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。

20.2.2 软件定时器（比较）方式

软件定时器方式也称为比较器方式。在该方式，PCA将计数器/定时器的计数值与模块的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并产生一个中断请求（如果CCF中断被允许）。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。

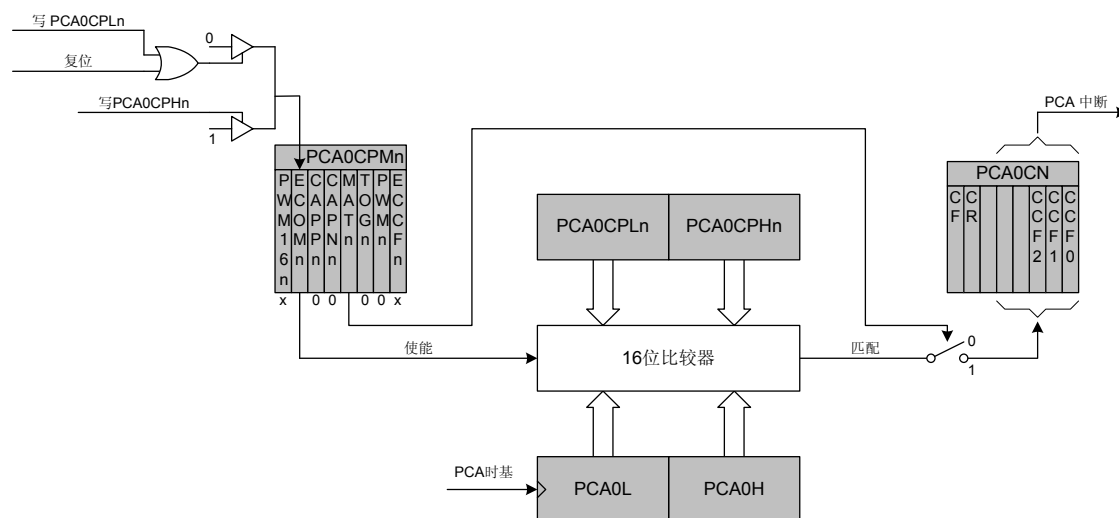


图 20.5 PCA 软件定时器方式原理框图

20.2.3 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。置位PCA0CPMn寄存器中的TOGn、MATn和ECOMn位将使能高速输出方式。

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。

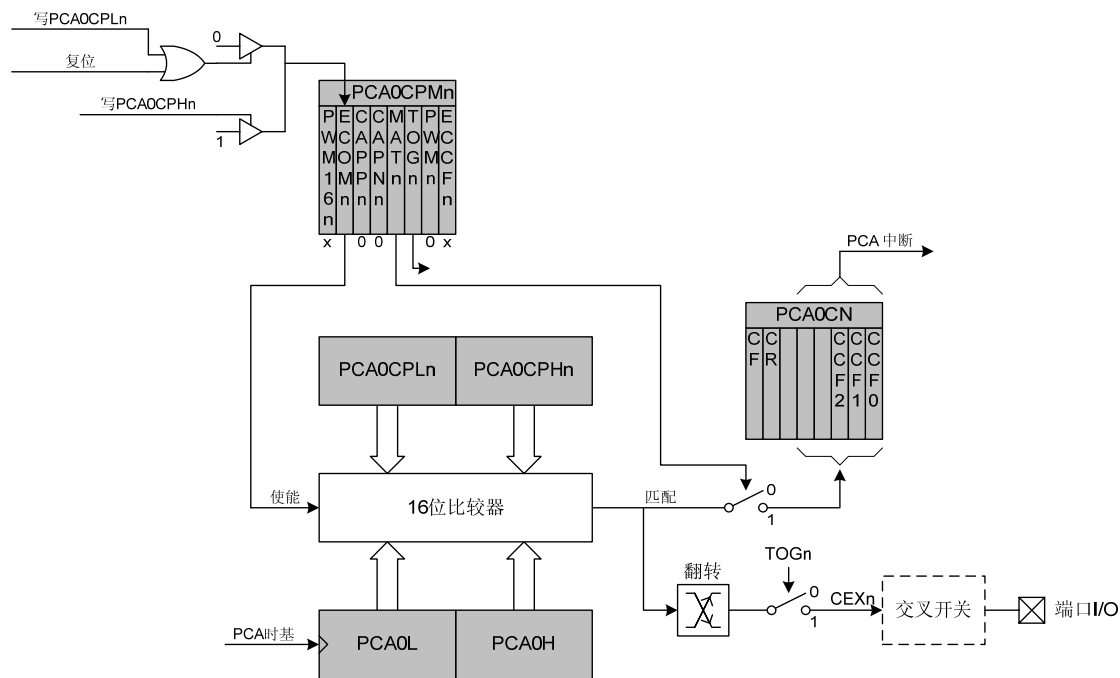


图 20.6 PCA 高速输出方式原理框图

注意：初始输出状态为逻辑1。当模块进入高速输出方式时，输出被初始化为该状态。

20.2.4 频率输出方式

频率输出方式可在CEX_n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20.1定义：

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

注：对于该方程，PCA0CPHn中的值为0x00时，相当于256。

方程20.1 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。通过将PCA0CPMn寄存器中ECOMn、TOGn和PWMn位置1来使能频率输出方式。

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。

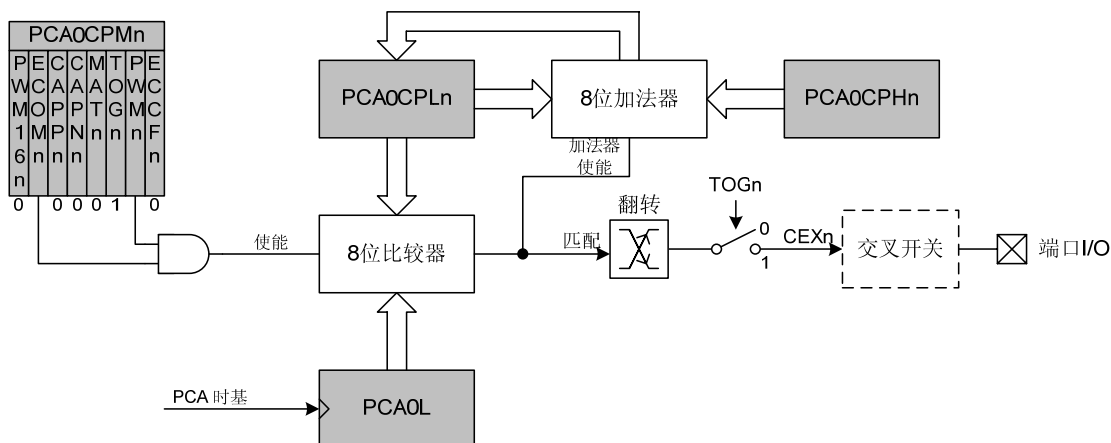


图 20.7 PCA 频率输出方式原理框图

20.2.5 8 位脉宽调制器方式

每个模块都可以被独立地用于在对应的CEXn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。使用模块的捕捉/比较寄存器PCA0CPHn改变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20.8）。当计数器/定时器的低字节PCA0L溢出时（从0xFF到0x00），保存在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来使能8位脉冲宽度调制器方式。8位PWM方式的占空比由方程20.2给出。

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。

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

方程20.2 8位PWM占空比

由方程20.2可知，最大占空比为100% (PCA0CPHn = 0)，最小占空比为0.39% (PCA0CPHn = 0xFF)。可以通过清除ECOMn位产生0%的占空比。

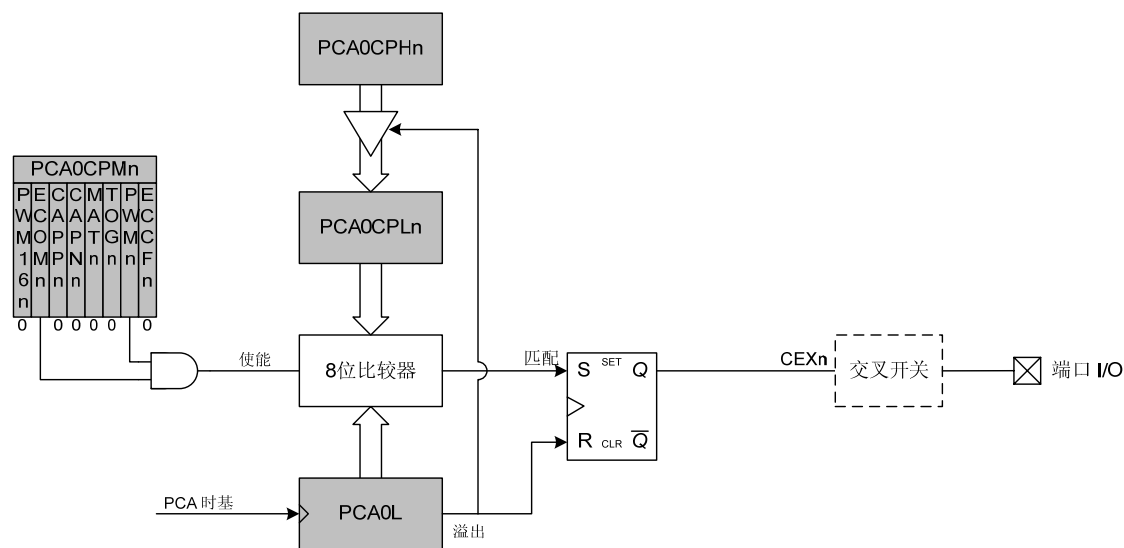


图 20.8 PCA 的 8 位 PWM 方式原理框图

20.2.6 16 位脉宽调制器方式

PCA模块还可被配置为工作在16位PWM方式。在该方式下，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计数器与模块的值匹配时，CEXn的输出被置为高电平；当计数器溢出时，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)，以同步对捕捉/比较寄存器的写操作。16位PWM方式的占空比由方程20.3给出。

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。

$$\text{占空比} = \frac{(65536 - \text{PCA0CPn})}{65536}$$

方程20.3 16位PWM占空比

由方程20.3可知，最大占空比为100% (PCA0CPn = 0)，最小占空比为0.0015% (PCA0CPn = 0xFFFF)。可以通过将ECOMn位清0产生0%的占空比。

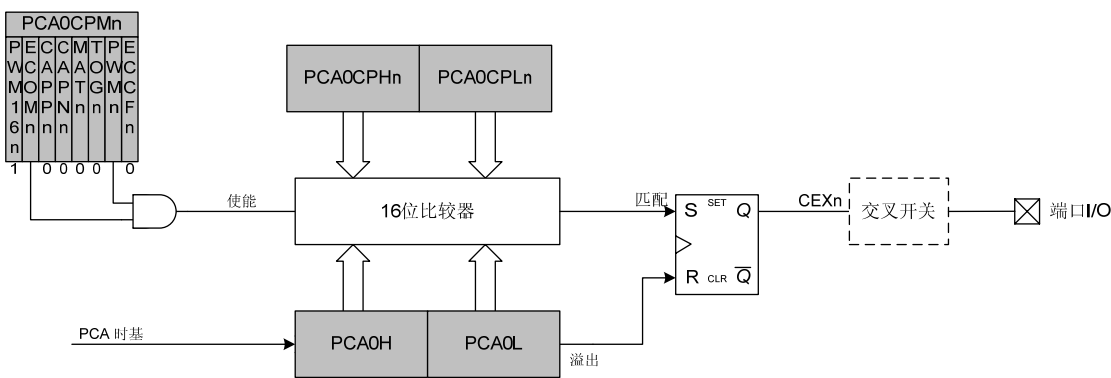


图 20.9 PCA 的 16 位 PWM 方式原理框图

20.3 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。

20.3.1 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，则系统将被复位。为了防止 WDT 复位，需要通过写 PCA0CPH2 来更新 WDT（写入值可以是任意值）。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 20.10）。

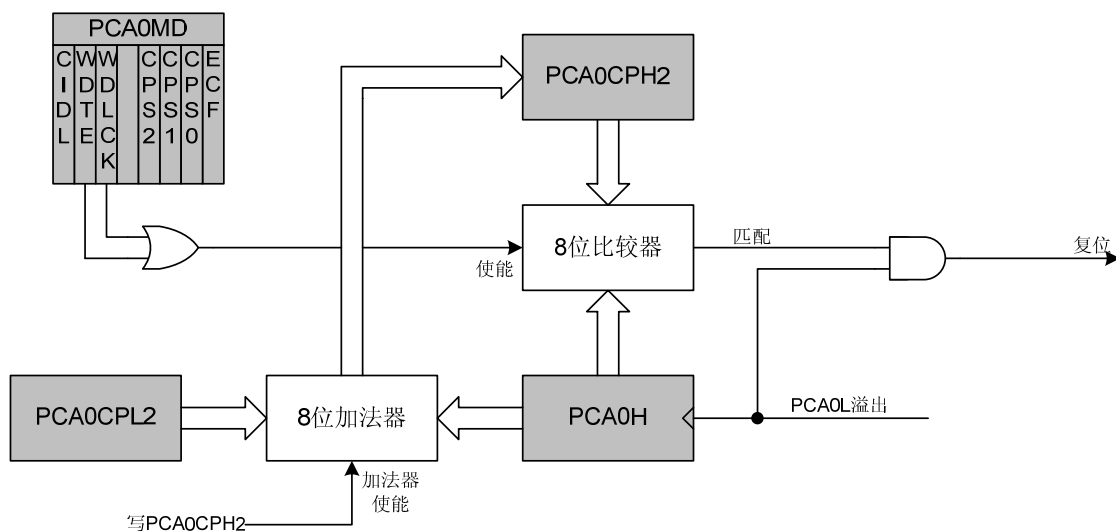


图 20.10 PCA 模块 2 的看门狗定时器方式

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 20.4 给出，其中 PCA0L 是执行更新操作时 PCA0L 寄存器的值。

$$\text{偏移值} = (256 \times \text{PCA0CPL2}) + (256 - \text{PCA0L})$$

方程20.4 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。

20.3.2 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。

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设置为使能状态。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，因此 WDT 的超时间隔为 3072 个系统时钟周期。表 20.3 列出了对应某些典型系统时钟频率的超时间隔。

表 20.3 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
18,432,000	255	42.7
18,432,000	128	21.5
18,432,000	32	5.5
11,059,200	255	71.1
11,059,200	128	35.8
11,059,200	32	9.2
3,062,500	255	257
3,062,500	128	129.5
3,062,500	32	33.1
191,406 ²	255	4109
191,406 ²	128	2070
191,406 ²	32	530
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. 内部振荡器复位频率。		

20.4 PCA 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 20.1 PCA0CN: PCA 控制寄存器

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值
CF	CR	保留	保留	保留	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
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。							
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。							
位 5-3:	保留							
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。							
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。							
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。							

SFR 定义 20.2 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 8 分频
1	1	0	保留
1	1	1	保留

注: 外部时钟 8 分频与系统时钟同步。

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 寄存器的值。若要改变 PCA0MD 的内容, 必须先禁止看门狗定时器。

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

SFR 定义 20.3 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 位 PWM。 1: 选择 16 位 PWM。							
位 6:	ECOMn: 比较器功能使能 该位使能/禁止 PCA 模块 n 的比较器功能。 0: 禁止。 1: 使能。							
位 5:	CAPPn: 正沿捕捉功能使能 该位使能/禁止 PCA 模块 n 的正边沿捕捉。 0: 禁止。 1: 使能。							
位 4:	CAPNn: 负沿捕捉功能使能 该位使能/禁止 PCA 模块 n 的负边沿捕捉。 0: 禁止。 1: 使能。							
位 3:	MATn: 匹配功能使能 该位使能/禁止 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。 0: 禁止。 1: 使能。							
位 2:	TOGn: 电平切换功能使能 该位使能/禁止 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。 0: 禁止。 1: 使能。							
位 1:	PWMn: 脉宽调制方式使能 该位使能/禁止 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 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。 0: 禁止。 1: 使能。							
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: 当 CCFn 位被置 1 时, 允许捕捉/比较标志的中断请求。							

SFR 定义 20.4 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: PCA0L: 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)。

SFR 定义 20.5 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: PCA0H: 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)。

SFR 定义 20.6 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, PCA0CPL1: 0xE9, PCA0CPL2: 0xEB

位 7-0: PCA0CPLn: 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)。

SFR 定义 20.7 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: PCA0CPHn: 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)。

21. 器件行为差别

本章介绍 C8051F52x/53x 器件和 C8051F52xA/53xA 器件之间的行为差别。这些差别不影响大多数系统的功能和性能。下面描述这些差别。

21.1 器件标识

器件封装上方的器件型号标识可以用于解译器件信息。在 C8051F52xA/53xA 器件上，器件型号以字母“A”结尾。在 C8051F52x/53x 器件上没有该字母。

图 21.1、21.2 和 21.3 介绍了如何找到器件封装上方的器件型号。



图 21.1 器件封装 — TSSOP20



图 21.2 器件封装 — QFN20



图 21.3 器件封装 — DFN10

21.2 复位引脚行为

C8051F52x/53x 的复位行为与 C8051F52xA/53xA 器件不同。这些差别影响 VDD 监视器复位期间 RST 引脚的状态。

对于 C8051F52x/53x 器件，VDD 监视器复位不影响 RST 引脚的状态。对于 C8051F52xA/53xA 器件，在欠压条件期间，VDD 监视器复位将 RST 引脚拉为低电平。

21.3 复位时延

C8051F52x/53x 的复位时延与 C8051F52xA/53xA 器件不同。

对于 C8051F52x/53x 器件，在上电复位时，复位时延会长达 80ms，这意味着它们可能需要等待多达 80ms 才能执行程序。后续的复位不会产生这么长的延时。对于 C8051F52xA/53xA 器件，启动时间在 350μs 左右。

21.4 UART 引脚

C8051F52x/53x 与 C8051F52xA/53xA 器件的串行 UART 接口使用的引脚不同。

对于 C8051F52x/53x 器件，UART 接口使用的 TX 和 RX 引脚被分配到 P0.3 (TX) 和 P0.4 (RX)。对于 C8051F52xA/53xA 器件，UART 接口使用的 TX 和 RX 引脚被分配到 P0.4 (TX) 和 P0.5 (RX)。

注意：对于 C8051F52xA/53xA 器件，如果 UART 被使能，UART 引脚必须被交叉开关跳过，以使外设能出现在 UART 以外的引脚。例如，在 SPI 和 UART 被使能，并且为 SPI 分配引脚 P1.0 ~ P1.3 的情况下，为了正确分配 SPI 引脚，必须用 P0SKIP 跳过 UART 引脚。

21.5 LIN

C8051F52x/53x 器件的 LIN 外设行为与 C8051F52xA/53xA 器件的 LIN 外设行为不同。其差别如下所述。

21.5.1 停止位检查

C8051F52x/53x 器件不检查 LIN 帧内各字段的停止位，在不能正确发送或接收停止位时也

C8051F52x/52xA/53x/53xA 混合信号 ISP FLASH 微控制器

不产生错误。C8051F52xA/53xA 器件检查停止位，在不能正确发送或接收停止位时产生错误。

21.5.2 SYNCH BREAK 和 SYNCH FIELD 长度检查

对于 C8051F52x/53x 器件，SYNCH BREAK 长度和 SYNCH FIELD 长度检查是不正确的。
对于 C8051F52xA/53xA 器件，SYNCH BREAK 长度必须大于 10 个位时间才能达到同步。

22. C2 接口

C8051F52x/52xA/53x/53xA 器件有一个 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。

22.1 C2 接口寄存器

下面对与 FLASH 编程有关的 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 22.1 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 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）（DEVICEID）。						
0x01		选择版本 ID 寄存器（数据读指令）（REVID）。						
0x02		选择 C2 FLASH 编程控制寄存器（数据读/写指令）（FPCTL）						
0xB4		选择 C2 FLASH 编程数据寄存器（数据读/写指令）（FPDAT）						

C2 寄存器定义 22.2 DEVICEID：C2 器件 ID 寄存器

								复位值
								00010001
位7	位6	位5	位4	位3	位2	位1	位0	
该只读寄存器返回 8 位的器件 ID 号：0x11（C8051F52x/52xA/53x/53xA）。								

C2 寄存器定义 22.3 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	
该只读寄存器返回 8 位的版本 ID 号。								
例如：0x00 = 版本 A。								

C2 寄存器定义 22.4 FPCTL: C2 FLASH 编程控制寄存器

								复位值
								00000000
位7	位6	位5	位4	位3	位2	位1	位0	
位 7-0: FPCTL: 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。一旦 C2 FLASH 编程被使能，必须进行一次系统复位才能恢复正常工作。								

C2 寄存器定义 22.5 FPDAT: C2 FLASH 编程数据寄存器

								复位值 00000000
位7	位6	位5	位4	位3	位2	位1	位0	

位 7-0: FPDAT: C2 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

22.2 C2 引脚共享

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停止工作，C2 接口可以安全地“借用”C2CK（正常方式为/RST）和 C2D（正常方式为 P2.0）引脚。在大多数情况下，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 22.1 所示。

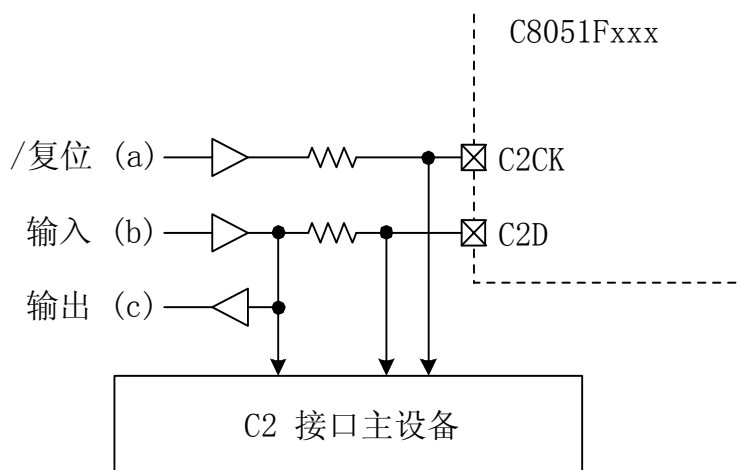


图 22.1 典型 C2 引脚共享电路

在图 22.1 的配置中，我们假设：

- 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。

对于具体的应用，可能还需要加入一些电阻。

联系信息

Silicon Laboratories Inc.

400 West Cesar Chavez
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