

### 数字转换器电容

- 支持按钮、滑动器、齿轮和电容接近感应
- 每通道 40  $\mu$ s 快速转换时间
- 16 位分辨率
- 多达 16 条输入通道
- 自动扫描和触碰唤醒
- 自动累积 4x、8x、16、32x 和 64x 样品

### 模拟外围设备

- **10 位 ADC**
  - 高达 500 ksp/s
  - 多达 16 个外部单端输入
  - 片上 VREF、外部引脚或  $V_{DD}$  的 VREF
  - 转换源的内部或外部启动
  - 内置温度传感器
- **比较器**
  - 可调节的迟滞性和响应时间
  - 可设置为中断或复位源

### 片上调试

- 片上调试电路提供全速的非侵入式系统内调试（无需仿真器）
- 提供断点、单步执行、检查 / 调节内存和寄存器
- 具有使用 ICE 芯片、目标特性板和插座仿真系统的卓越性能
- 低成本、完整的开发套件

### 高速 8051 $\mu$ C 内核

- 管线指令体系结构；执行 1 个或 2 个系统时钟中 70% 的指令
- 高达 25 MIPS 吞吐率和 25 兆赫时钟脉冲
- 扩展的中断处理程序

### 内存

- 高达 512 字节的内部数据 RAM (256 + 256)
- 高达 16 千字节闪存；系统内可在 512 字节段内编程

### 数字外围设备

- 17 或 13 个高灌电流端口 I/O
- 硬件增强型 UART、SMBus™ (I2C 兼容) 和增强型 SPI™ 串行端口
- 三个通用 16 位计数器 / 定时器
- 配置三个捕捉 / 比较模块和增强 PWM 功能的 16 位可编程计数器阵列 (PCA)
- 使用定时器和晶体的实时时钟模式

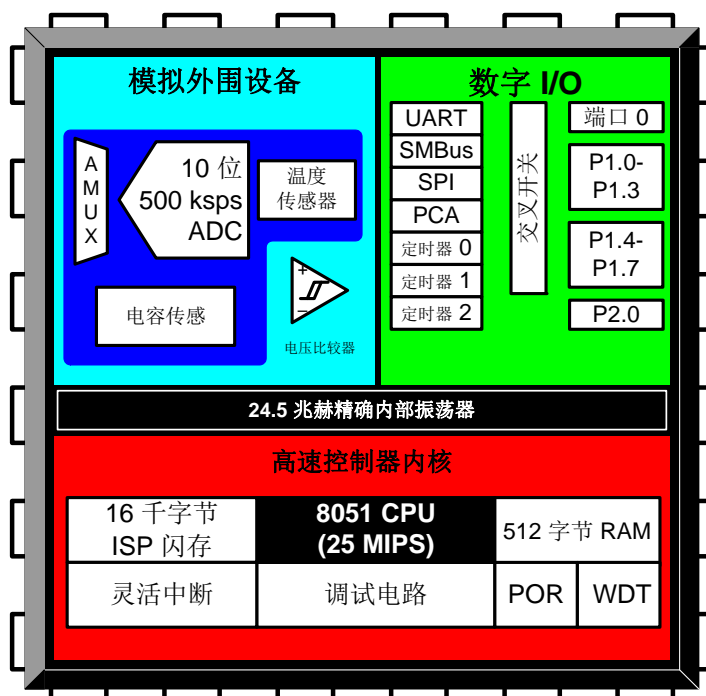
### 时钟源

- 24.5 兆赫  $\pm$ 2% 振荡器
  - 支持无晶 UART 运算
- 外部振荡器：晶体、RC、C 或时钟（1 或 2 个引脚模式）
- 可在时钟源运行间转换；可在省电模式中使用

### 供电电压 1.8 至 3.6 伏

- 内置供电电压监控器

**24 针 QSOP、20 针 QFN、16 针 SOIC 温度范围：-40 至 +85  $^{\circ}$ C**



# C8051F80x-83x

---

## 目录

1. 系统概览.....	15
2. 订购资料.....	25
3. 引脚定义.....	28
4. QFN-20 封装规格.....	33
5. QSOP-24 封装规格.....	35
6. SOIC-16 封装规格.....	37
7. 电气特性.....	39
7.1. 绝对值最大规格.....	39
7.2. 电气特性.....	40
8. 10 位 ADC (ADC0).....	46
8.1. 输出代码格式.....	47
8.2. 8 位模式.....	47
8.3. 操作模式.....	47
8.3.1. 启动转换.....	47
8.3.2. 追踪模式.....	48
8.3.3. 设置时间要求.....	49
8.4. 可编程窗口探测器.....	53
8.4.1. 窗口探测器范例.....	55
8.5. ADC0 模拟多路复用器.....	56
9. 温度传感器.....	58
9.1. 校准.....	58
10. 电压与接地基准选项.....	60
10.1. 外部电压基准.....	60
10.2. 内部电压基准选项.....	61
10.3. 模拟接地基准.....	61
10.4. 温度传感器启用.....	61
11. 稳压器 (REG0).....	63
12. 比较器 0.....	65
12.1. 比较器多路复用器.....	69
13. 电容传感 (CS0).....	71
13.1. 端口引脚配置为电容传感输入.....	72
13.2. 电容传感启动转换源.....	72
13.3. 自动扫描.....	72
13.4. CS0 比较器.....	73
13.5. CS0 转换累加器.....	74
13.6. 电容传感多路复用器.....	80
14. CIP-51 微控制器.....	82
14.1. 指令集.....	83
14.1.1. 指令和 CPU 定时.....	83
14.2. CIP-51 寄存器说明.....	88
15. 内存组织.....	92
15.1. 程序内存.....	93
15.1.1. MOVX 指令和程序内存.....	93

# C8051F80x-83x

15.2. 数据内存 .....	93
15.2.1. 内部 RAM .....	93
15.2.1.1. 通用寄存器 .....	94
15.2.1.2. 位寻址 .....	94
15.2.1.3. 堆栈 .....	94
<b>16. 系统内设备识别 .....</b>	<b>95</b>
<b>17. 特殊功能寄存器 .....</b>	<b>97</b>
<b>18. 中断 .....</b>	<b>102</b>
18.1. MCU 中断源和矢量 .....	103
18.1.1. 中断优先级 .....	103
18.1.2. 中断延时 .....	103
18.2. 中断寄存器说明 .....	104
18.3. INT0 和 INT1 外部中断 .....	111
<b>19. 闪存 .....</b>	<b>113</b>
19.1. 编程闪存 .....	113
19.1.1. 闪存锁定和主要功能 .....	113
19.1.2. 闪存擦除程序 .....	113
19.1.3. 闪存写入步骤 .....	114
19.2. 非易失性数据存储 .....	114
19.3. 安全选项 .....	114
19.4. 闪存写入和擦除指引 .....	115
19.4.1. VDD 维护和 VDD 监控器 .....	116
19.4.2. PSWE 维护 .....	116
19.4.3. 系统时钟 .....	117
<b>20. 电源管理模式 .....</b>	<b>120</b>
20.1. 空闲模式 .....	120
20.2. 停止模式 .....	121
20.3. 暂停模式 .....	121
<b>21. 复位源 .....</b>	<b>123</b>
21.1. 启动复位 .....	124
21.2. 掉电复位 / VDD 监视器 .....	125
21.3. 外部复位 .....	126
21.4. 缺失时钟探测器复位 .....	126
21.5. 比较器 0 复位 .....	127
21.6. PCA 看门狗复位 .....	127
21.7. 闪存故障复位 .....	127
21.8. 软件复位 .....	127
<b>22. 振荡器和时钟选择寄存器 .....</b>	<b>129</b>
22.1. 系统时钟选择 .....	129
22.2. 可编程内部高频 (H-F) 振荡器 .....	131
22.3. 外部振荡器驱动电路 .....	133
22.3.1. 外部晶体范例 .....	135
22.3.2. 外部 RC 范例 .....	136
22.3.3. 外部电容器范例 .....	137
<b>23. 端口输入 / 输出 .....</b>	<b>138</b>

23.1. 端口 I/O 模式操作 .....	139
23.1.1. 为模拟 I/O 配置端口针脚 .....	139
23.1.2. 为数字 I/O 配置端口针脚 .....	139
23.1.3. 连接端口 I/O 至 5 伏逻辑 .....	140
23.2. 分配端口 I/O 针脚至模拟和数字功能 .....	140
23.2.1. 分配端口 I/O 针脚至模拟功能 .....	140
23.2.2. 分配端口 I/O 针脚至数字功能 .....	141
23.2.3. 分配端口 I/O 针脚至外部数字事件捕捉功能 .....	142
23.3. 优先级交叉开关编码器 .....	143
23.4. 端口 I/O 初始化 .....	147
23.5. 端口匹配 .....	150
23.6. 访问和配置端口 I/O 的特殊功能寄存器 .....	152
<b>24. 循环冗余检验单位 (CRC0) .....</b>	<b>159</b>
24.1. 16 位 CRC 算法 .....	160
24.2. 32 位 CRC 算法 .....	161
24.3. 准备 CRC 计算 .....	162
24.4. 执行 CRC 计算 .....	162
24.5. 存取 CRC0 结果 .....	162
24.6. CRC0 位元反转特征 .....	166
<b>25. 增强型串行外围接口 (SPI0) .....</b>	<b>167</b>
25.1. 信号说明 .....	168
25.1.1. 主输出、从输入 (MOSI) .....	168
25.1.2. 主输入、从输出 (MISO) .....	168
25.1.3. 串行时钟 (SCK) .....	168
25.1.4. 从选择 (NSS) .....	168
25.2. SPI0 主模式操作 .....	168
25.3. SPI0 从模式操作 .....	170
25.4. SPI0 中断源 .....	171
25.5. 串行时钟相位和极性 .....	171
25.6. SPI 特殊功能寄存器 .....	173
<b>26. SMBus .....</b>	<b>180</b>
26.1. 辅助文件 .....	181
26.2. SMBus 配置寄存器 .....	181
26.3. SMBus 操作 .....	181
26.3.1. 发送器与接收器 .....	182
26.3.2. 仲裁 .....	182
26.3.3. 时钟低电平扩展 .....	182
26.3.4. SCL 低电平超时 .....	182
26.3.5. SCL 高电平 (SMBus 空闲) 超时 .....	183
26.4. 使用 SMBus .....	183
26.4.1. SMBus 配置寄存器 .....	183
26.4.2. SMB0CN 控制寄存器 .....	187
26.4.2.1. 软件 ACK 发生器 .....	187
26.4.2.2. 硬件 ACK 发生器 .....	187
26.4.3. 硬件从地址识别 .....	189

# C8051F80x-83x

---

26.4.4. 数据寄存器 .....	192
26.5. SMBus 传输模式.....	193
26.5.1. 写入顺序（主）.....	193
26.5.2. 读取顺序（主）.....	194
26.5.3. 写入顺序（从）.....	195
26.5.4. 读取顺序（从）.....	196
26.6. SMBus 状态解码.....	196
<b>27. UART0 .....</b>	<b>201</b>
27.1. 增强型波特率发生器 .....	202
27.2. 操作模式 .....	203
27.2.1. 8 位 UART.....	203
27.2.2. 9 位 UART.....	204
27.3. 多处理器通信 .....	205
<b>28. 定时器 .....</b>	<b>209</b>
28.1. 定时器 0 和定时器 1 .....	211
28.1.1. 模式 0: 13 位计数器 / 定时器 .....	211
28.1.2. 模式 1: 16 位计数器 / 定时器 .....	212
28.1.3. 模式 2: 8 位自动重载计数器 / 定时器 .....	212
28.1.4. 模式 3: 两个 8 位计数器 / 定时器（仅限定时器 0） .....	213
28.2. 定时器 2.....	219
28.2.1. 16 位自动重载定时器 .....	219
28.2.2. 8 位自动重载定时器 .....	220
<b>29. 可编程计数器阵列 .....</b>	<b>225</b>
29.1. PCA 计数器 / 定时器.....	226
29.2. PCA0 中断源 .....	227
29.3. 捕捉 / 比较模块 .....	228
29.3.1. 边缘触发捕捉模式 .....	229
29.3.2. 软件定时器（比较）模式.....	230
29.3.3. 高速输出模式.....	231
29.3.4. 频率输出模式.....	232
29.3.5. 8 位至 15 位脉冲宽度调制器模式 .....	232
29.3.5.1. 8 位脉冲宽度调制器模式.....	233
29.3.5.2. 9 位至 15 位脉冲宽度调制器模式.....	234
29.3.6. 16 位脉冲宽度调制器模式 .....	235
29.4. 看门狗模式.....	236
29.4.1. 看门狗操作 .....	236
29.4.2. 看门狗用法 .....	237
29.5. PCA0 寄存器说明 .....	237
<b>30. C2 接口 .....</b>	<b>244</b>
30.1. C2 接口寄存器 .....	244
30.2. C2CK 引脚共享.....	247
文件更新一览表.....	248
联络资料.....	250

---

## 表一览表

<b>1. 系统概览</b>	
<b>2. 订购资料</b>	
表 2.1. 产品选择指南 .....	26
<b>3. 引脚定义</b>	
表 3.1. <b>C8051F80x-83x</b> 引脚定义 .....	<b>28</b>
<b>4. QFN-20 封装规格</b>	
表 4.1. QFN-20 封装尺寸 .....	33
表 4.2. QFN-20 印制电路板焊盘图形尺寸 .....	34
<b>5. QSOP-24 封装规格</b>	
表 5.1. QSOP-24 封装尺寸 .....	35
表 5.2. QSOP-24 印制电路板焊盘图形尺寸 .....	36
<b>6. SOIC-16 封装规格</b>	
表 6.1. SOIC-16 封装尺寸 .....	37
表 6.2. SOIC-16 印制电路板焊盘图形尺寸 .....	38
<b>7. 电气特性</b>	
表 7.1. 绝对值最大等级 .....	39
表 7.2. 全球电气特性 .....	40
表 7.3. 端口 I/O DC 电气特性 .....	41
表 7.4. 复位电气特性 .....	41
表 7.5. 内部稳压器电气特性 .....	41
表 7.6. 闪存电气特性 .....	42
表 7.7. 内部高频振荡器电气特性 .....	42
表 7.8. 电容传感电气特性 .....	42
表 7.9. ADC0 电气特性 .....	43
表 7.10. 电源管理电气特性 .....	44
表 7.11. 温度传感器电气特性 .....	44
表 7.12. 电压基准电气特性 .....	44
表 7.13. 比较器电气特性 .....	45
<b>8. 10 位 ADC (ADC0)</b>	
<b>9. 温度传感器</b>	
<b>10. 电压与接地基准选项</b>	
<b>11. 稳压器 (REG0)</b>	
<b>12. 比较器 0</b>	
<b>13. 电容传感 (CS0)</b>	
表 13.1. 自动扫描和累加的操作 .....	74
<b>14. CIP-51 微控制器</b>	
表 14.1. CIP-51 指令集一览表 .....	84
<b>15. 内存组织</b>	
<b>16. 系统内设备识别</b>	
<b>17. 特殊功能寄存器</b>	
表 17.1. 特殊功能寄存器 (SFR) 存储器映像 .....	97
表 17.2. 特殊功能寄存器 .....	98

# C8051F80x-83x

<b>18. 中断</b>	
表 18.1: 中断概述 .....	104
<b>19. 闪存</b>	
表 19.1: 闪存安全概述 .....	115
<b>20. 电源管理模式</b>	
<b>21. 复位源</b>	
<b>22. 振荡器和时钟选择寄存器</b>	
<b>23. 端口输入 / 输出</b>	
表 23.1: 端口 I/O 模拟功能分配 .....	141
表 23.2: 端口 I/O 数字功能分配 .....	142
表 23.3: 端口 I/O 外部数字事件捕捉功能分配 .....	142
<b>24. 循环冗余检验单位 (CRC0)</b>	
表 24.1: 16 位 CRC 输出范例 .....	160
表 24.2: 32 位 CRC 输出范例 .....	161
<b>25. 增强型串行外围接口 (SPI0)</b>	
表 25.1: SPI 从定时参数 .....	179
<b>26. SMBus</b>	
表 26.1: SMBus 时钟源选择 .....	184
表 26.2: 最小 SDA 建立和保持时间 .....	185
表 26.3: SMB0CN 硬件更换源 .....	189
表 26.4: 硬件地址识别范例 (EHACK = 1) .....	190
表 26.5: SMBus 状态解码 (禁用硬件 ACK 发生器) (EHACK = 0) .....	197
表 26.6: SMBus 状态解码 (启用硬件 ACK 发生器) (EHACK = 1) .....	199
<b>27. UART0</b>	
表 27.1: 使用内部 24.5 MHz 振荡器进行标准波特率的定时器设置 .....	208
表 27.2: 使用外部 22.1184 MHz 振荡器进行标准波特率的定时器设置 .....	208
<b>28. 定时器</b>	
<b>29. 可编程计数器阵列</b>	
表 29.1: PCA 时基输入选项 .....	226
表 29.2: PCA 捕捉 / 比较模块的 PCA0CPM 和 PCA0PWM 位设置 <sup>1,2,3,4,5,6</sup> .....	228
表 29.3: 看门狗超时间隔 1 .....	237
<b>30. C2 接口</b>	



## 图一览表

### 1. 系统概览

图 1.1. C8051F800、C8051F806、C8051F812、C8051F818 结构图 .....	16
图 1.2. C8051F801、C8051F807、C8051F813、C8051F819 结构图 .....	17
图 1.3. C8051F802、C8051F808、C8051F814、C8051F820 结构图 .....	18
图 1.4. C8051F803、C8051F809、C8051F815、C8051F821 结构图 .....	19
图 1.5. C8051F804、C8051F810、C8051F816、C8051F822 结构图 .....	20
图 1.6. C8051F805、C8051F811、C8051F817、C8051F823 结构图 .....	21
图 1.7. C8051F824、C8051F827、C8051F830、C8051F833 结构图 .....	22
图 1.8. C8051F825、C8051F828、C8051F831、C8051F834 结构图 .....	23
图 1.9. C8051F826、C8051F829、C8051F832、C8051F835 结构图 .....	24

### 2. 订购资料

### 3. 引脚定义

图 3.1. QFN-20 引脚分布图（俯视图） .....	30
图 3.2. QSOP-24 引脚分布图（俯视图） .....	31
图 3.3. SOIC-16 引脚分布图（俯视图） .....	32

### 4. QFN-20 封装规格

图 4.1. QFN-20 封装图 .....	33
图 4.2. QFN-20 推荐的印制电路板焊盘图形 .....	34

### 5. QSOP-24 封装规格

图 5.1. QSOP-24 封装图 .....	35
图 5.2. QSOP-24 印制电路板焊盘图形 .....	36

### 6. SOIC-16 封装规格

图 6.1. SOIC-16 封装图 .....	37
图 6.2. SOIC-16 印制电路板焊盘图形 .....	38

### 7. 电气特性

### 8. 10 位 ADC (ADC0)

图 8.1. ADC0 功能结构图 .....	46
图 8.2. 10 位 ADC 追踪和转换范例定时 .....	48
图 8.3. ADC0 等效输入电路 .....	49
图 8.4. ADC 窗口比较范例：右对齐数据 .....	55
图 8.5. ADC 窗口比较范例：左对齐数据 .....	55
图 8.6. ADC0 多路复用器功能结构图 .....	56

### 9. 温度传感器

图 9.1. 温度传感器转移函数 .....	58
图 9.2. 在 0 °C 附近 1 点校准的温度传感器误差 .....	59

### 10. 电压与接地基准选项

图 10.1. 电压基准功能结构图 .....	60
-------------------------	----

### 11. 稳压器 (REG0)

### 12. 比较器 0

图 12.1. 比较器 0 功能结构图 .....	65
图 12.2. 比较器迟滞数据表 .....	66
图 12.3. 比较器输入多路复用器结构图 .....	69

# C8051F80x-83x

<b>13. 电容传感 (CS0)</b>	
图 13.1. CS0 结构图 .....	71
图 13.2. 自动扫描范例 .....	73
图 13.3. <b>CS0 多路复用器结构图</b> .....	<b>80</b>
<b>14. CIP-51 微控制器</b>	
图 14.1. CIP-51 结构图 .....	82
<b>15. 内存组织</b>	
图 15.1. C8051F80x-83x 存储器映像 .....	<b>92</b>
图 15.2. 闪存程序存储器映像 .....	93
<b>16. 系统内设备识别</b>	
<b>17. 特殊功能寄存器</b>	
<b>18. 中断</b>	
<b>19. 闪存</b>	
<b>20. 电源管理模式</b>	
<b>21. 复位源</b>	
图 21.1. 复位源 .....	123
图 21.2. 启动和 VDD 监视器复位定时 .....	124
<b>22. 振荡器和时钟选择寄存器</b>	
图 22.1. 振荡器选项 .....	129
图 22.2. 外部 32.768 千赫石英晶体振荡器接线图 .....	136
<b>23. 端口输入 / 输出</b>	
图 23.1. 端口 I/O 功能结构图 .....	138
图 23.2. 端口 I/O 单元结构图 .....	139
图 23.3. 端口 I/O 超速电流 .....	140
图 23.4. 优先级交叉开关编码器潜在引脚分配 .....	144
图 23.5. 优先级交叉开关编码器范例 1 — 无跳过引脚 .....	145
图 23.6. 优先级交叉开关编码器范例 2 — 跳过引脚 .....	146
<b>24. 循环冗余检验单位 (CRC0)</b>	
图 24.1. CRC0 结构图 .....	159
<b>25. 增强型串行外围接口 (SPI0)</b>	
图 25.1. SPI 结构图 .....	167
图 25.2. 多主模式接线图 .....	169
图 25.3. 3 线单主模式和 3 线单从模式接线图 .....	169
图 25.4. 4 线单主模式和 4 线从模式接线图 .....	170
图 25.5. 主模式数据 / 时钟定时 .....	172
图 25.6. 从模式数据 / 时钟定时 (CKPHA = 0) .....	172
图 25.7. 从模式数据 / 时钟定时 (CKPHA = 1) .....	173
图 25.8. SPI 主定时 (CKPHA = 0) .....	177
图 25.9. SPI 主定时 (CKPHA = 1) .....	177
图 25.10. SPI 从定时 (CKPHA = 0) .....	178
图 25.11. SPI 从定时 (CKPHA = 1) .....	178
<b>26. SMBus</b>	
图 26.1. SMBus 结构图 .....	180
图 26.2. 典型 SMBus 配置 .....	181
图 26.3. SMBus 传输 .....	182

---

图 26.4. 典型 SMBus SCL 发生器 .....	184
图 26.5. 典型主发送器写入顺序 .....	193
图 26.6. 典型主接收器读取顺序 .....	194
图 26.7. 典型从接收器写入顺序 .....	195
图 26.8. 典型从发送器读取顺序 .....	196
<b>27. UART0</b>	
图 27.1. UART0 结构图 .....	201
图 27.2. UART0 波特率逻辑 .....	202
图 27.3. UART 互连图 .....	203
图 27.4. 8 位 UART 定时图 .....	203
图 27.5. 9 位 UART 定时图 .....	204
图 27.6. UART 多处理器模式互连图 .....	205
<b>28. 定时器</b>	
图 28.1. T0 模式 0 结构图 .....	212
图 28.2. T0 模式 2 结构图 .....	213
图 28.3. T0 模式 3 结构图 .....	214
图 28.4. 定时器 2 16 位模式结构图 .....	219
图 28.5. 定时器 2 8 位模式结构图 .....	220
<b>29. 可编程计数器阵列</b>	
图 29.1. PCA 结构图 .....	225
图 29.2. PCA 计数器 / 定时器结构图 .....	226
图 29.3. PCA 中断结构图 .....	227
图 29.4. PCA 捕捉模式图 .....	229
图 29.5. PCA 软件定时器模式图 .....	230
图 29.6. PCA 高速输出模式图 .....	231
图 29.7. PCA 频率输出模式 .....	232
图 29.8. PCA 8 位 PWM 模式图 .....	233
图 29.9. PCA 9 位至 15 位 PWM 模式图 .....	234
图 29.10. PCA 16 位 PWM 模式图 .....	235
图 29.11. 启用看门狗的 PCA 模块 2 .....	236
<b>30. C2 接口</b>	
图 30.1: 典型 C2 针脚共享 .....	247

# C8051F80x-83x

---

## 寄存器一览表

SFR 定义 8.1. ADC0CF: ADC0 配置寄存器 .....	50
SFR 定义 8.2. ADC0H: ADC0 数据字 MSB 寄存器 .....	51
SFR 定义 8.3. ADC0L: ADC0 数据字 LSB 寄存器 .....	51
SFR 定义 8.4. ADC0CN: ADC0 控制寄存器 .....	52
SFR 定义 8.5. ADC0GTH: ADC0 下限数据高字节寄存器 .....	53
SFR 定义 8.6. ADC0GTL: ADC0 下限数据低字节寄存器 .....	53
SFR 定义 8.7. ADC0LTH: ADC0 上限数据高字节寄存器 .....	54
SFR 定义 8.8. ADC0LTL: ADC0 上限数据低字节寄存器 .....	54
SFR 定义 8.9. ADC0MX: AMUX0 通道选择寄存器 .....	57
SFR 定义 10.1. REF0CN: 电压基准控制寄存器 .....	62
SFR 定义 11.1. REG0CN: 稳压器控制寄存器 .....	64
SFR 定义 12.1. CPT0CN: 比较器 0 控制寄存器 .....	67
SFR 定义 12.2. CPT0MD: 比较器 0 模式选择 .....	68
SFR 定义 12.3. CPT0MX: 比较器 0 MUX 选择寄存器 .....	70
SFR 定义 13.1. CS0CN: 电容传感控制寄存器 .....	75
SFR 定义 13.2. CS0CF: 电容传感配置寄存器 .....	76
SFR 定义 13.3. CS0DH: 电容传感数据高字节寄存器 .....	77
SFR 定义 13.4. CS0DL: 电容传感数据低字节寄存器 .....	77
SFR 定义 13.5. CS0SS: 电容传感自动扫描启动通道寄存器 (Capacitive Sense Auto-Scan Start Channel) .....	78
SFR 定义 13.6. CS0SE: 电容传感自动扫描结束通道寄存器 (Capacitive Sense Auto-Scan End Channel) .....	78
SFR 定义 13.7. CS0THH: 电容传感比较器阈值高字节寄存器 (Capacitive Sense Comparator Threshold High Byte) .....	79
SFR 定义 13.8. CS0THL: 电容传感比较器阈值低字节寄存器 (Capacitive Sense Comparator Threshold Low Byte) .....	79
SFR 定义 13.9. <b>CS0MX: 电容传感 Mux 通道选择</b> .....	<b>81</b>
SFR 定义 14.1. DPL: 数据指针低字节寄存器 .....	88
SFR 定义 14.2. DPH: 数据指针高字节寄存器 .....	88
SFR 定义 14.3. SP: 堆栈指针寄存器 .....	89
SFR 定义 14.4. ACC: 累加器 .....	89
SFR 定义 14.5. B: B 寄存器 .....	90
SFR 定义 14.6. PSW: 程序状态字寄存器 .....	91
SFR 定义 16.1. HWID: 硬件 ID 字节寄存器 .....	95
SFR 定义 16.2. DERIVID: 硬件 ID 字节寄存器 .....	96
SFR 定义 16.3. REVID: 硬件版本 ID 字节寄存器 .....	96
SFR 定义 18.1. IE: 中断允许寄存器 .....	105
SFR 定义 18.2. IP: 中断优先级寄存器 .....	106
SFR 定义 18.3. EIE1: 扩展中断允许 1 寄存器 .....	107
SFR 定义 18.4. EIE2: 扩展中断允许 2 寄存器 .....	108
SFR 定义 18.5. EIP1: 扩展中断优先级 1 寄存器 .....	109
SFR 定义 18.6. EIP2: 扩展中断优先级 2 寄存器 .....	110
SFR 定义 18.7. IT01CF: INT0 / INT1 配置寄存器 .....	112

# C8051F80x-83x

SFR 定义 19.1:	PSCTL: 程序存储读写控制寄存器	118
SFR 定义 19.2:	FLKEY: 闪存锁定和关键码寄存器	119
SFR 定义 20.1:	PCON: 电源控制寄存器	122
SFR 定义 21.1:	VDM0CN: VDD 监视器控制寄存器	126
SFR 定义 21.2:	RSTSRC: 复位源	128
SFR 定义 22.1:	CLKSEL: 时钟选择	130
SFR 定义 22.2:	OSCICL: 内部 H-F 振荡器校准	131
SFR 定义 22.3:	OSICN: 内部 H-F 振荡器控制	132
SFR 定义 22.4:	OSXCN: 外部振荡器控制寄存器	134
SFR 定义 23.1:	XBR0: 端口 I/O 交叉开关寄存器 0	148
SFR 定义 23.2:	XBR1: 端口 I/O 交叉开关寄存器 1	149
SFR 定义 23.3:	P0MASK: 端口 0 屏蔽寄存器	151
SFR 定义 23.4:	P0MAT: 端口 0 匹配寄存器	151
SFR 定义 23.5:	P1MASK: 端口 1 屏蔽寄存器	152
SFR 定义 23.6:	P1MAT: 端口 1 匹配寄存器	152
SFR 定义 23.7:	P0: 端口 0	153
SFR 定义 23.8:	P0MDIN: 端口 0 输入模式	154
SFR 定义 23.9:	P0MDOUT: 端口 0 输出模式	154
SFR 定义 23.10:	P0SKIP: 端口 0 跳过寄存器	155
SFR 定义 23.11:	P1: 端口 1	155
SFR 定义 23.12:	P1MDIN: 端口 1 输入模式	156
SFR 定义 23.13:	P1MDOUT: 端口 1 输出模式	156
SFR 定义 23.14:	P1SKIP: 端口 1 跳过寄存器	157
SFR 定义 23.15:	P2: 端口 2	157
SFR 定义 23.16:	P2MDOUT: 端口 2 输出模式	158
SFR 定义 24.1:	CRC0CN: CRC0 控制寄存器	163
SFR 定义 24.2:	CRC0IN: CRC 数据输入寄存器	164
SFR 定义 24.3:	CRC0DATA: CRC 数据输出寄存器	164
SFR 定义 24.4:	CRC0AUTO: CRC 自动控制寄存器	165
SFR 定义 24.5:	CRC0CNT: CRC 自动闪存扇区总数寄存器	165
SFR 定义 24.6:	CRC0FLIP: CRC 位翻转寄存器	166
SFR 定义 25.1:	SPI0CFG: SPI0 配置寄存器	174
SFR 定义 25.2:	SPI0CN: SPI0 控制寄存器	175
SFR 定义 25.3:	SPI0CKR: SPI0 时钟频率寄存器	176
SFR 定义 25.4:	SPI0DAT: SPI0 数据寄存器	176
SFR 定义 26.1:	SMB0CF: SMBus 时钟 / 配置寄存器	186
SFR 定义 26.2:	SMB0CN: SMBus 控制寄存器	188
SFR 定义 26.3:	SMB0ADR: SMBus 从地址寄存器	191
SFR 定义 26.4:	SMB0ADM: SMBus 从地址屏蔽寄存器	191
SFR 定义 26.5:	SMB0DAT: SMBus 数据寄存器	192
SFR 定义 27.1:	SCON0: 串行端口 0 控制寄存器	206
SFR 定义 27.2:	SBUF0: 串行 (UART0) 端口数据缓冲器	207
SFR 定义 28.1:	CKCON: 时钟控制寄存器	210
SFR 定义 28.2:	TCON: 定时器控制寄存器	215
SFR 定义 28.3:	TMOD: 定时器模式	216

---

SFR 定义 28.4:	TL0: 定时器 0 低字节寄存器	217
SFR 定义 28.5:	TL1: 定时器 1 低字节寄存器	217
SFR 定义 28.6:	TH0: 定时器 0 高字节寄存器	218
SFR 定义 28.7:	TH1: 定时器 1 高字节寄存器	218
SFR 定义 28.8:	TMR2CN: 定时器 2 控制寄存器	222
SFR 定义 28.9:	TMR2RLL: 定时器 2 重载寄存器低字节	223
SFR 定义 28.10:	TMR2RLH: 定时器 2 重载寄存器高字节	223
SFR 定义 28.11:	TMR2L: 定时器 2 低字节寄存器	224
SFR 定义 28.12:	TMR2H: 定时器 2 高字节寄存器	224
SFR 定义 29.1:	PCA0CN: PCA0 控制寄存器	238
SFR 定义 29.2:	PCA0MD: PCA0 模式	239
SFR 定义 29.3:	PCA0PWM: PCA0 PWM 配置寄存器	240
SFR 定义 29.4:	PCA0CPMn: PCA0 捕捉 / 比较模式寄存器	241
SFR 定义 29.5:	PCA0L: PCA0 计数器 / 定时器低字节寄存器	242
SFR 定义 29.6:	PCA0H: PCA0 计数器 / 定时器高字节	242
SFR 定义 29.7:	PCA0CPLn: PCA0 捕捉模块低字节寄存器	243
SFR 定义 29.8:	PCA0CPHn: PCA0 捕捉模块高字节寄存器	243
C2 寄存器定义 30.1:	C2ADD: C2 地址寄存器	244
C2 寄存器定义 30.2:	DEVICEID: C2 设备 ID	244
C2 寄存器定义 30.3:	REVID: C2 版本 ID	245
C2 寄存器定义 30.4:	FPCTL: C2 闪存编程控制寄存器	246
C2 寄存器定义 30.5:	FPDAT: C2 闪存编程数据寄存器	246

# C8051F80x-83x

---



## 1. 系统概览

C8051F80x-83x 设备为完全集成、混合信号、片上系统电容传感的 MCU。主要部件列示如下。有关特定产品部件选择和部件订购编号，请参阅表 2.1。

- 高速流水线 8051 兼容微控制器内核（高达 25 MIPS）
- 系统内、全速、非侵入式调试接口（片上）
- 具有 16 条输入通道的电容传感接口
- 具有 16 条通道模拟多路复用器和集成温度传感器的 10 位 500 ksps 单端 ADC
- 精密的 24.5 兆赫内部振荡器
- 16 千字节片上闪存
- 512 字节片上 RAM
- 硬件中应用的 ?SMBus/I2C、增强型 UART 和增强型 SPI 串行接口
- 三个通用 16 位定时器
- 具有三个捕捉 / 比较模块的可编程计数器 / 定时器阵列 (PCA)
- 片上内部电压基准
- 片上看门狗
- 片上启动复位和电源监控器
- 片上电压比较器
- 17 个通用 I/O

由于带有片上启动复位、 $V_{DD}$  监视器、看门狗和时钟振荡器，C8051F80x-83x 设备成为一个真正独立的片上系统解决方案。该设备甚至可在线改编闪存，前提是非易失数据存储，且允许 8051 固件现场升级。用户软件已完成所有外围设备的控制，并可单独关闭任何或所有外围设备以节省电力。

C8051F80x-83x 处理器包括 Silicon Laboratories 的 2 线 C2 调试和编程接口，可利用安装到最终应用中的产品 MCU，进行非侵入性（不使用片上资源）、全速、在线调试。该调试逻辑支持进行内存检查、查看和修改特殊功能寄存器、设置断点、单步执行以及运行和停止命令。使用 C2 调试时，所有模拟和数字外围设备均具有完善的功能。两个 C2 接口引脚可通过用户功能共享，在不占用封装引脚的情况下允许系统调试。

每个设备在整个工业温度范围（-45 至 +85 °C）内均被指定为以 1.8–3.6 伏运转。内部 LDO 稳压器用于提供 1.8 伏的处理器内核电压。端口 I/O 和 RST 引脚能接受达 5 伏的输入信号。有关订购资料，请参阅表 2.1。C8051F80x-83x 系列的设备结构图列示于图 1.1 至图 1.9。

# C8051F80x-83x

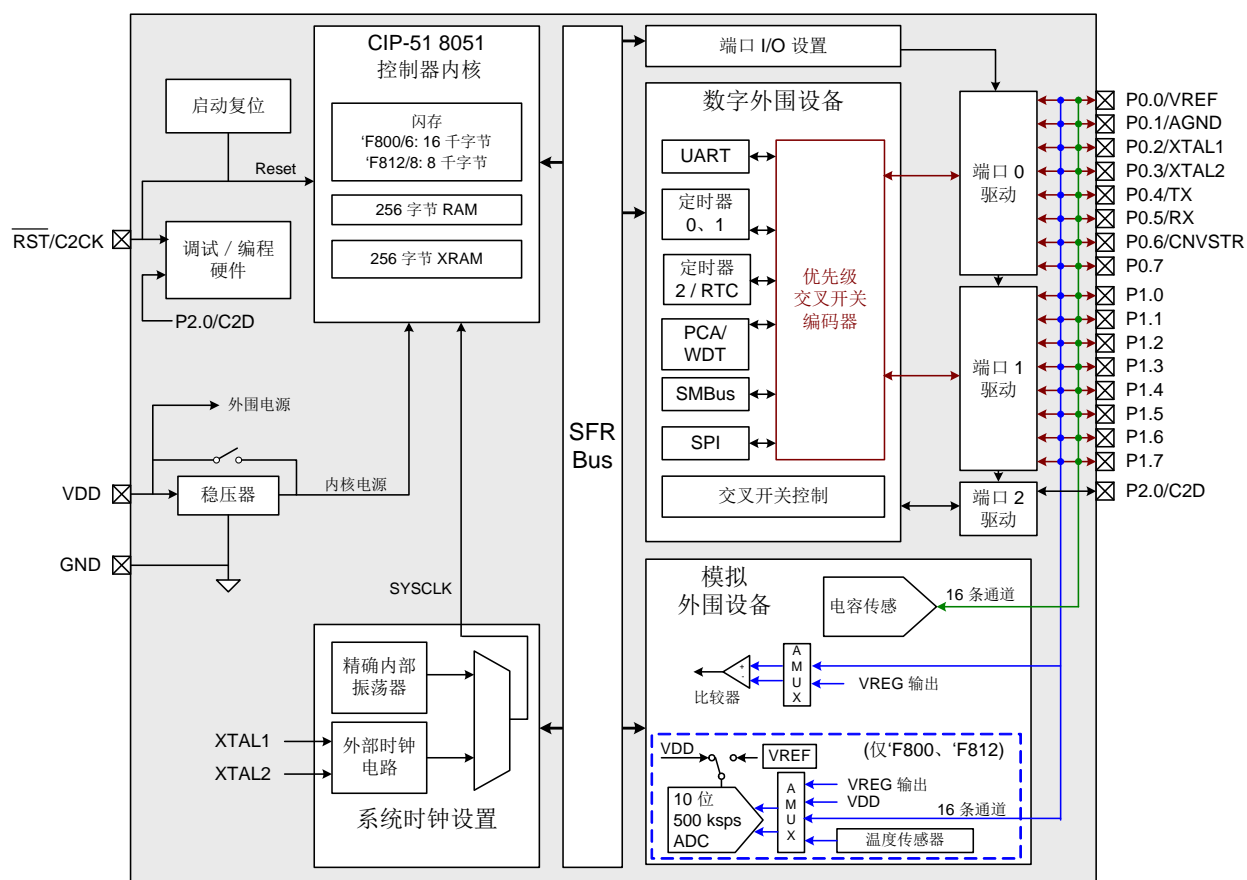


图 1.1. C8051F800、C8051F806、C8051F812、C8051F818 结构图

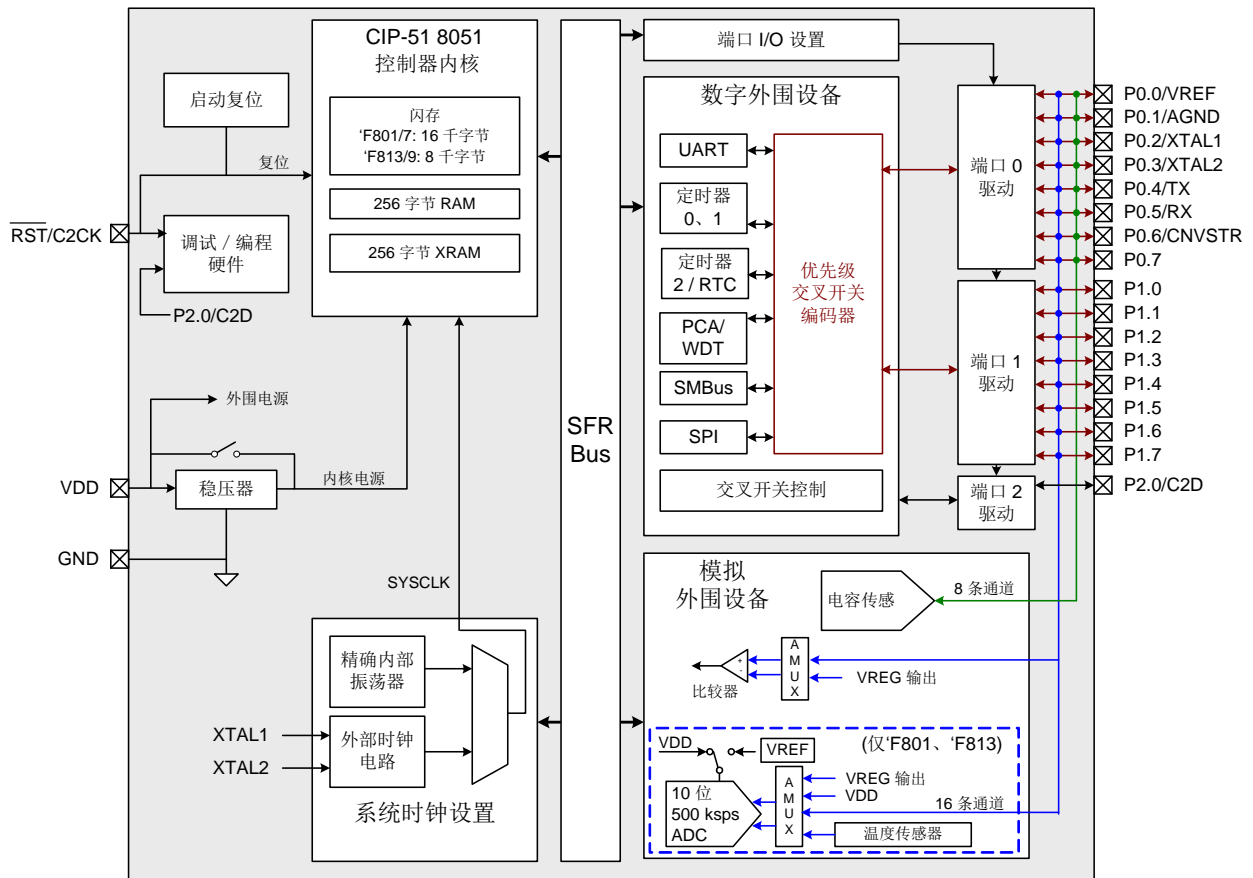


图 1.2. C8051F801、C8051F807、C8051F813、C8051F819 结构图

# C8051F80x-83x

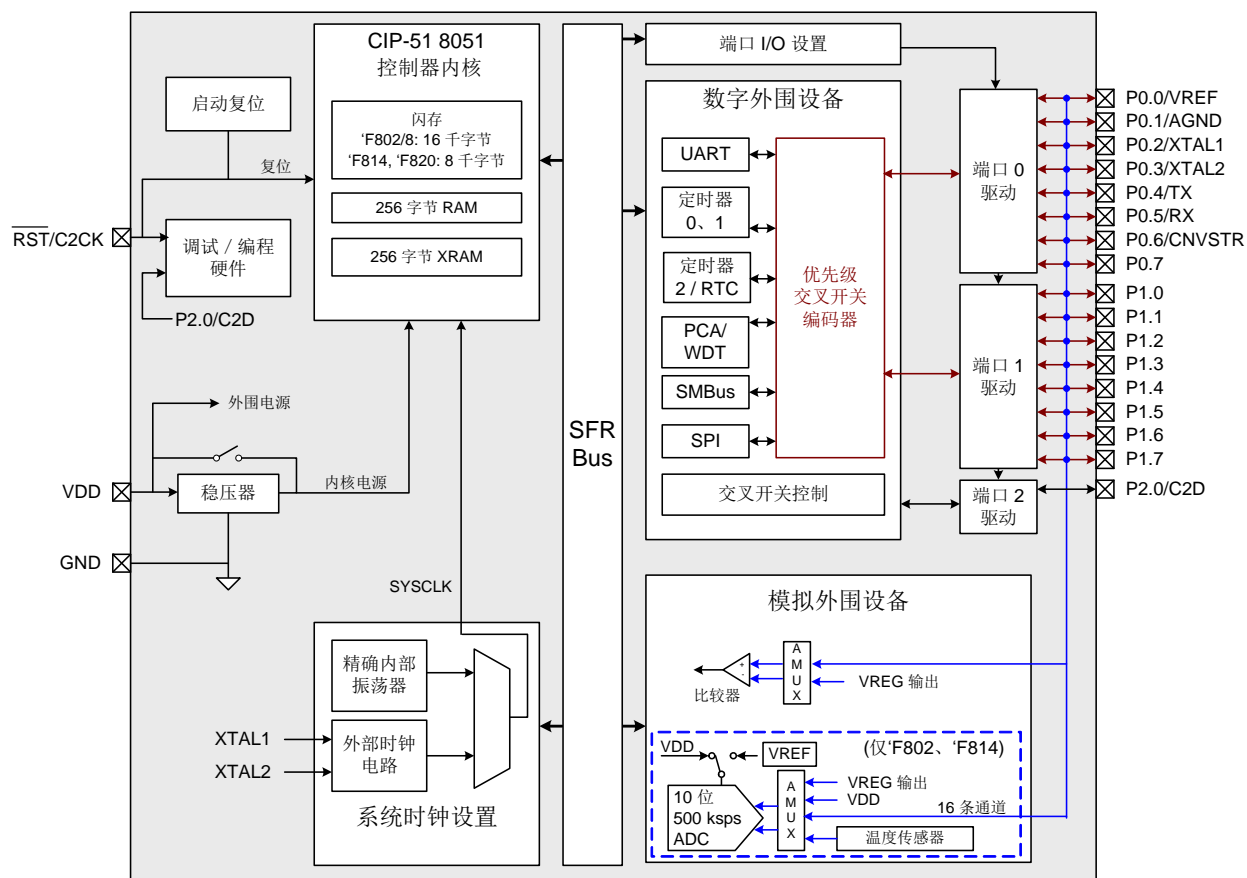


图 1.3. C8051F802、C8051F808、C8051F814、C8051F820 结构图

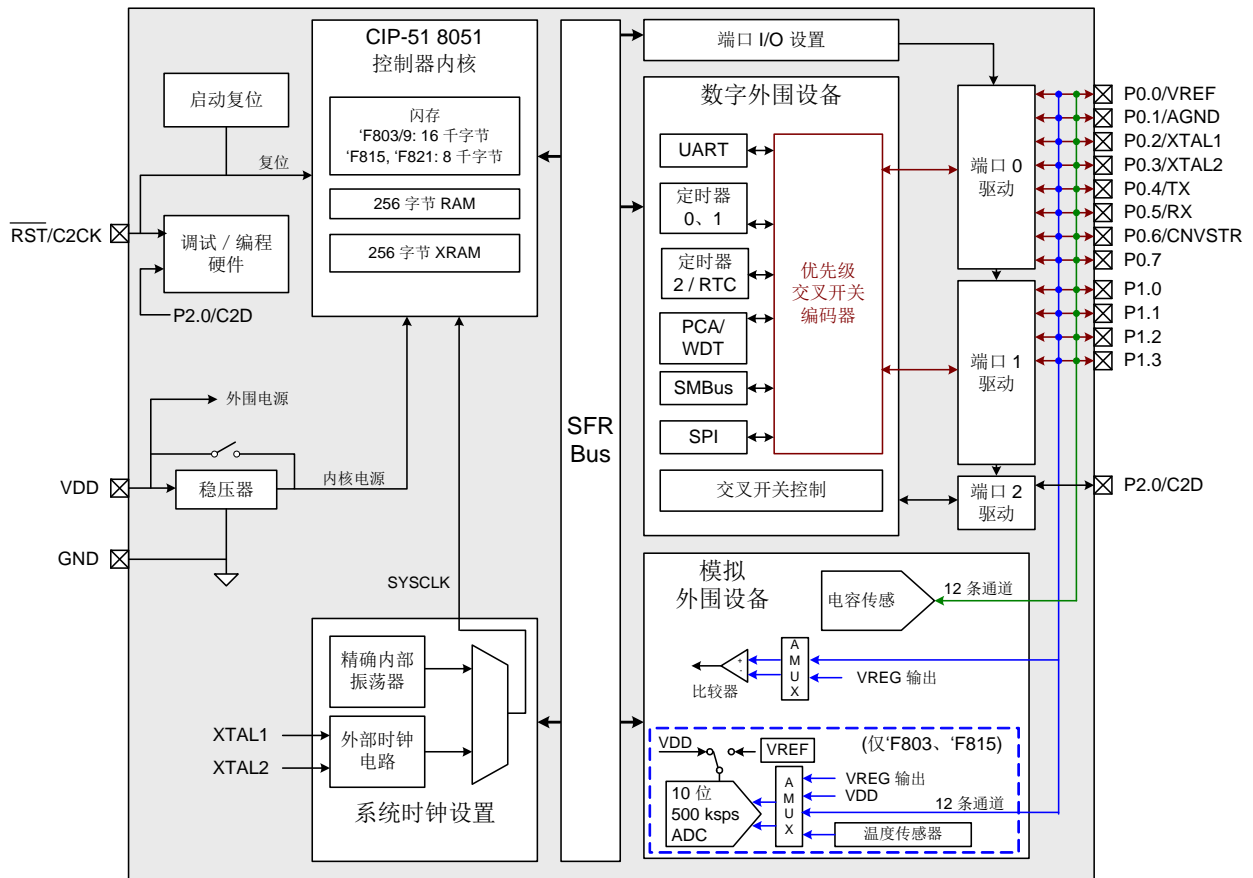


图 1.4. C8051F803、C8051F809、C8051F815、C8051F821 结构图

# C8051F80x-83x

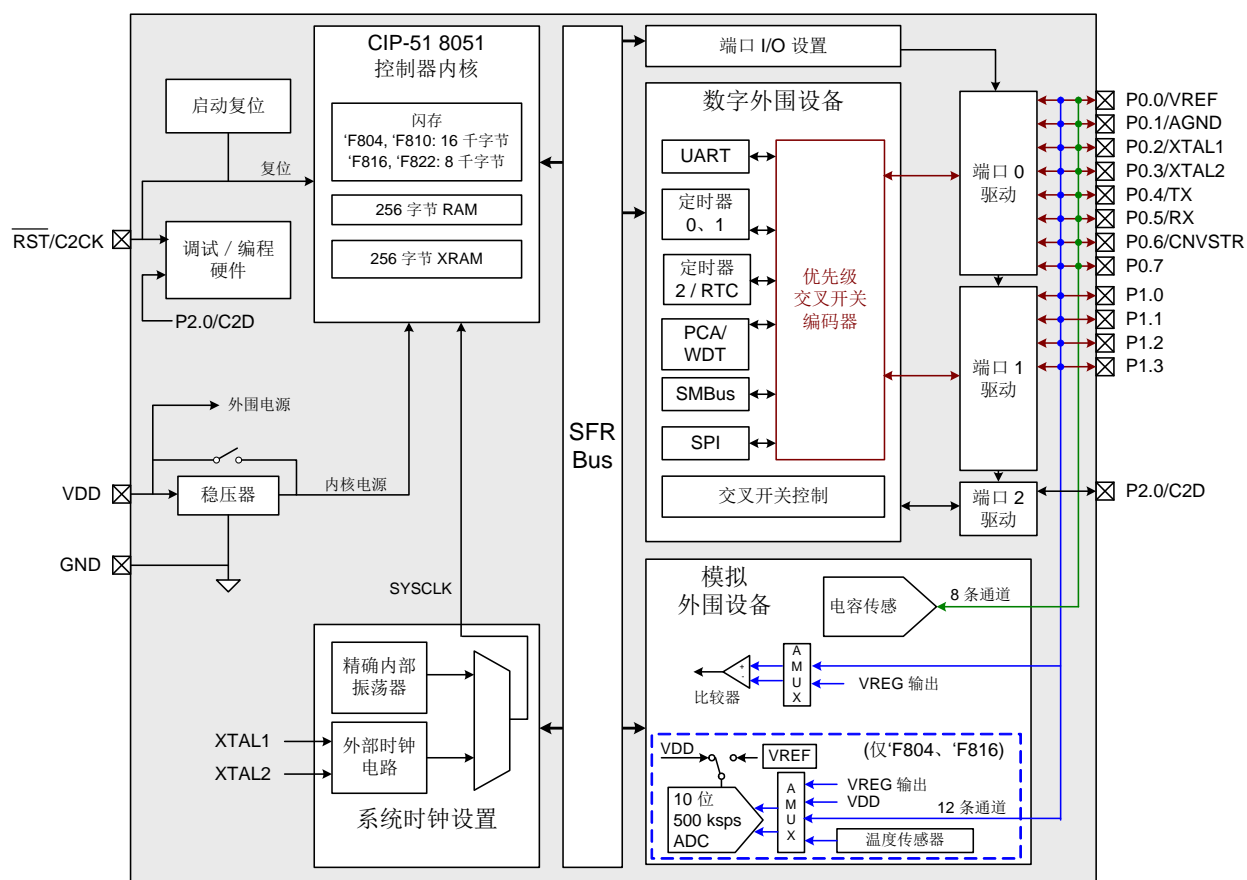


图 1.5. C8051F804、C8051F810、C8051F816、C8051F822 结构图

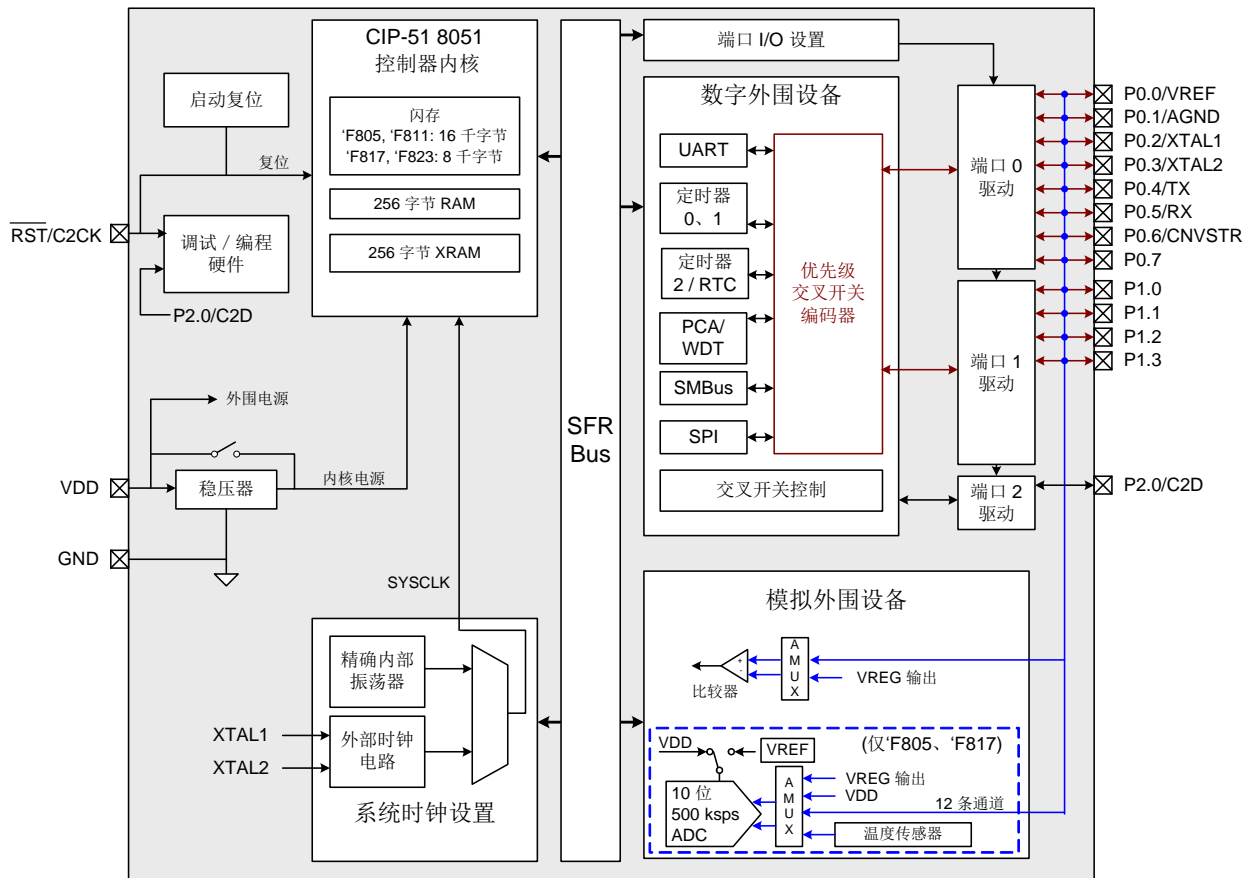


图 1.6. C8051F805、C8051F811、C8051F817、C8051F823 结构图

# C8051F80x-83x

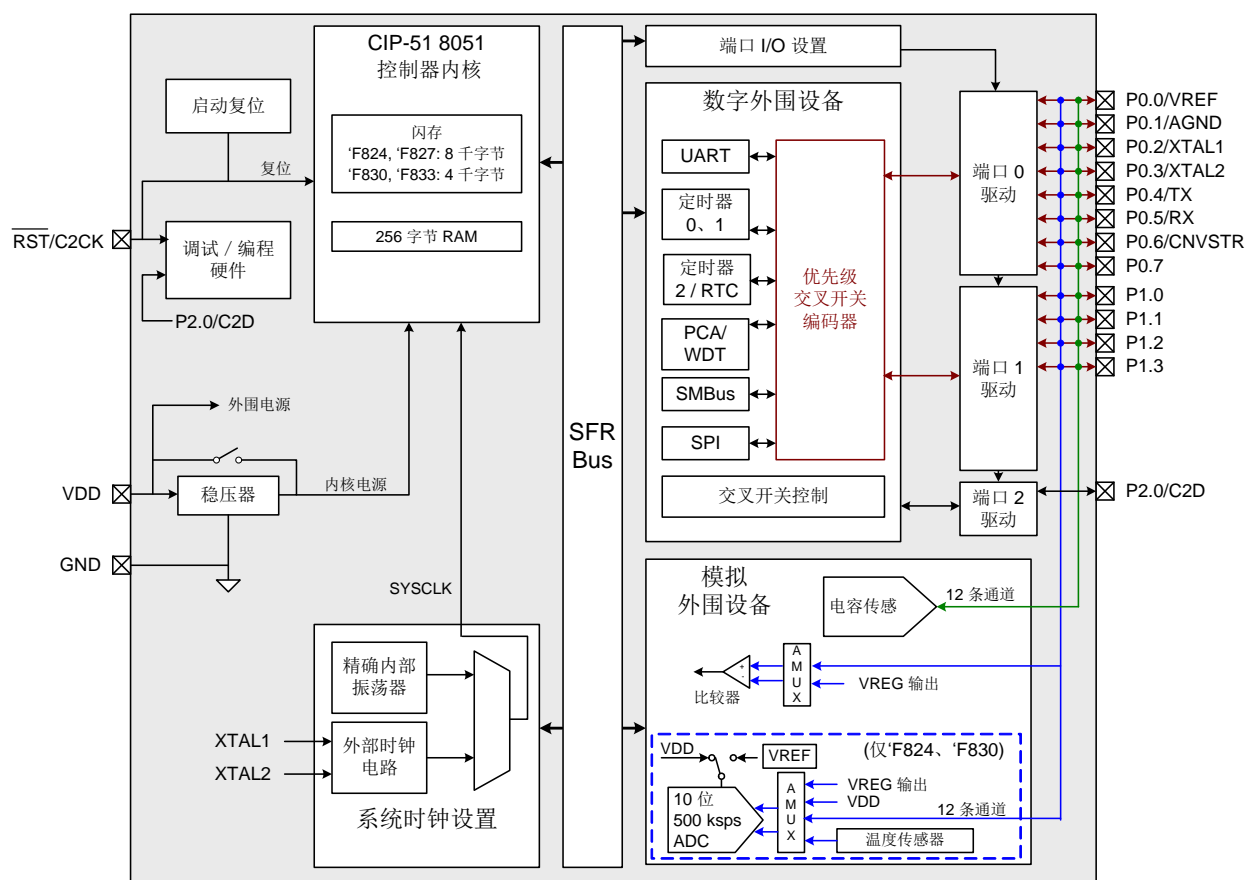


图 1.7. C8051F824、C8051F827、C8051F830、C8051F833 结构图



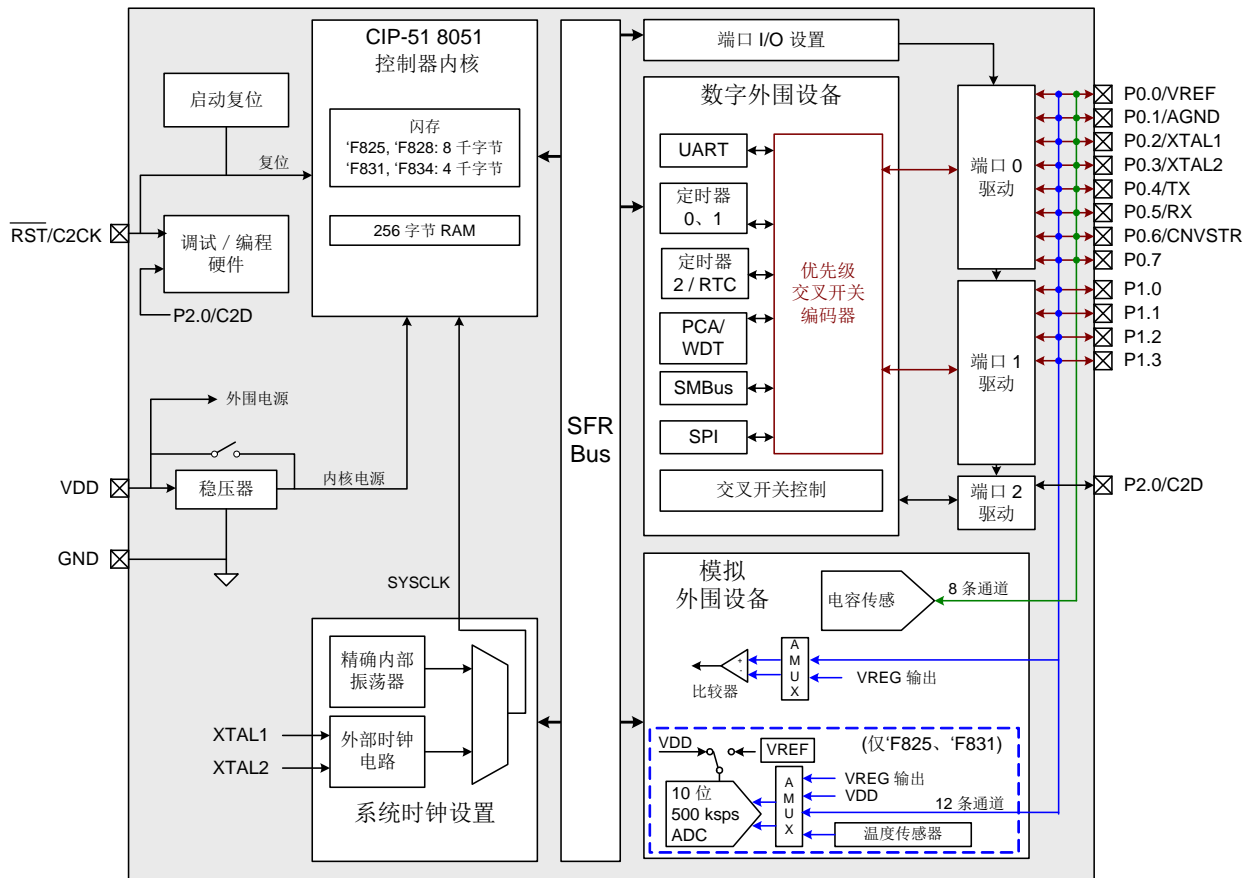


图 1.8. C8051F825、C8051F828、C8051F831、C8051F834 结构图

# C8051F80x-83x

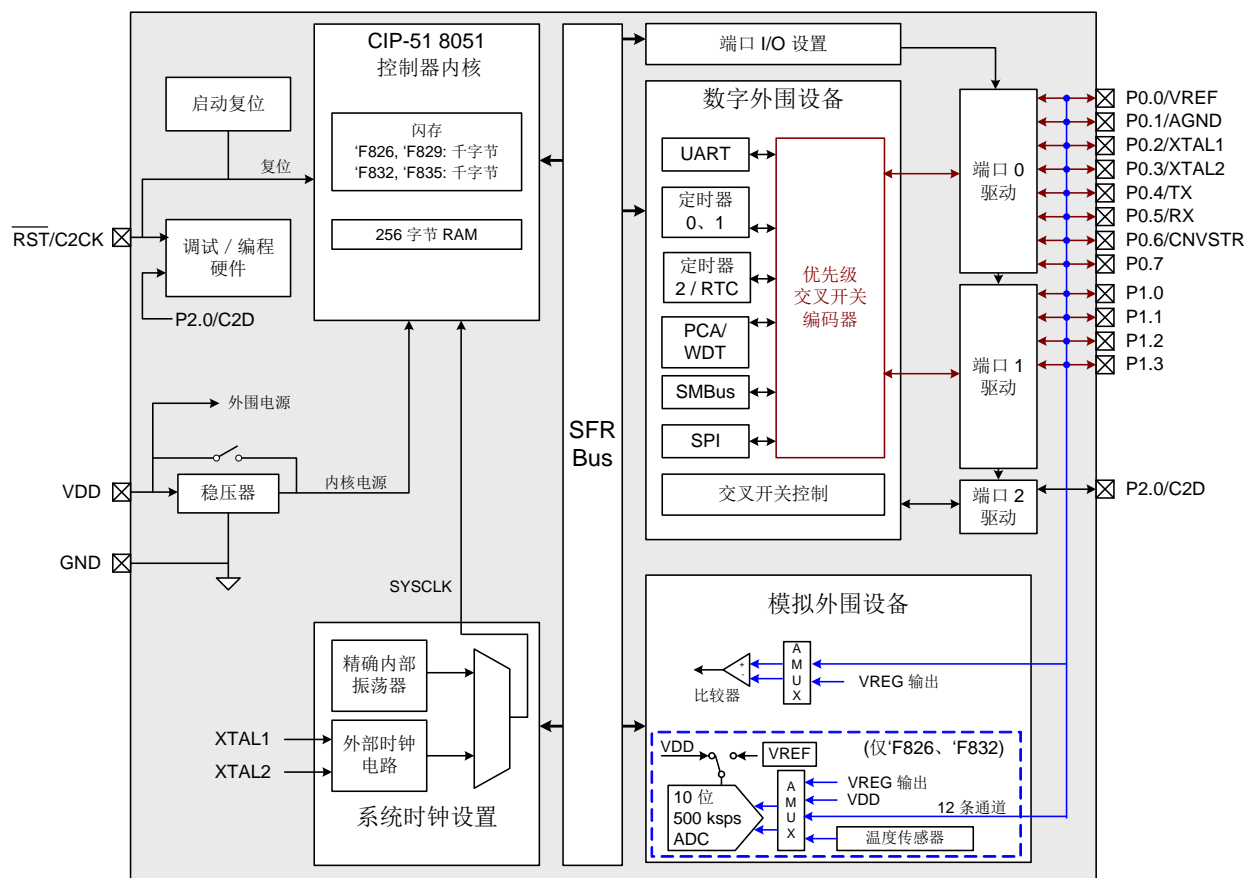


图 1.9. C8051F826、C8051F829、C8051F832、C8051F835 结构图

## 2. 订购资料

所有 C8051F80x-83x 设备均具备以下特征：

- 25 MIPS（峰值）
- 已校准内部振荡器
- SMBus / I2C
- 增强型 SPI
- UART
- 可编程计数器阵列（3 条通道）
- 3 个定时器（16 位）
- 1 个比较器
- 无铅（通过 RoHS 认证）封装

除上述特征外，C8051F80x-83x 系列中的各设备还具有与该产品系列不同的一系列特征。请参见表 2.1，了解该系列中各设备具有的独有特征一览表。

# C8051F80x-83x

表 2.1. 产品选择指南

部件编号	数字端口 I/O	电容传感 通道	闪存 (千字节)	RAM (字节)	10 位 500 ksp ADC	ADC 通道	温度传感器	封装 (RoHS)
C8051F800-GU	17	16	16	512	✓	16	✓	QSOP-24
C8051F801-GU	17	8	16	512	✓	16	✓	QSOP-24
C8051F802-GU	17	—	16	512	✓	16	✓	QSOP-24
C8051F800-GM	17	16	16	512	✓	16	✓	QFN-20
C8051F801-GM	17	8	16	512	✓	16	✓	QFN-20
C8051F802-GM	17	—	16	512	✓	16	✓	QFN-20
C8051F803-GS	13	12	16	512	✓	12	✓	SOIC-16
C8051F804-GS	13	8	16	512	✓	12	✓	SOIC-16
C8051F805-GS	13	—	16	512	✓	12	✓	SOIC-16
C8051F806-GU	17	16	16	512	—	—	—	QSOP-24
C8051F807-GU	17	8	16	512	—	—	—	QSOP-24
C8051F808-GU	17	—	16	512	—	—	—	QSOP-24
C8051F806-GM	17	16	16	512	—	—	—	QFN-20
C8051F807-GM	17	8	16	512	—	—	—	QFN-20
C8051F808-GM	17	—	16	512	—	—	—	QFN-20
C8051F809-GS	13	12	16	512	—	—	—	SOIC-16
C8051F810-GS	13	8	16	512	—	—	—	SOIC-16
C8051F811-GS	13	—	16	512	—	—	—	SOIC-16
C8051F812-GU	17	16	8	512	✓	16	✓	QSOP-24
C8051F813-GU	17	8	8	512	✓	16	✓	QSOP-24
C8051F814-GU	17	—	8	512	✓	16	✓	QSOP-24
C8051F812-GM	17	16	8	512	✓	16	✓	QFN-20
C8051F813-GM	17	8	8	512	✓	16	✓	QFN-20
C8051F814-GM	17	—	8	512	✓	16	✓	QFN-20
C8051F815-GS	13	12	8	512	✓	12	✓	SOIC-16
C8051F816-GS	13	8	8	512	✓	12	✓	SOIC-16
C8051F817-GS	13	—	8	512	✓	12	✓	SOIC-16
C8051F818-GU	17	16	8	512	—	—	—	QSOP-24
C8051F819-GU	17	8	8	512	—	—	—	QSOP-24
C8051F820-GU	17	—	8	512	—	—	—	QSOP-24
C8051F818-GM	17	16	8	512	—	—	—	QFN-20
C8051F819-GM	17	8	8	512	—	—	—	QFN-20
C8051F820-GM	17	—	8	512	—	—	—	QFN-20
C8051F821-GS	13	12	8	512	—	—	—	SOIC-16
C8051F822-GS	13	8	8	512	—	—	—	SOIC-16

表 2.1. 产品选择指南 (Continued)

部件编号	数字端口 I/O	电容传感 通道	闪存 (千字节)	RAM (字节)	10 位 500 ksp ADC	ADC 通道	温度传感器	封装 (RoHS)
C8051F823-GS	13	—	8	512	—	—	—	SOIC-16
C8051F824-GS	13	12	8	256	✓	12	✓	SOIC-16
C8051F825-GS	13	8	8	256	✓	12	✓	SOIC-16
C8051F826-GS	13	—	8	256	✓	12	✓	SOIC-16
C8051F827-GS	13	12	8	256	—	—	—	SOIC-16
C8051F828-GS	13	8	8	256	—	—	—	SOIC-16
C8051F829-GS	13	—	8	256	—	—	—	SOIC-16
C8051F830-GS	13	12	4	256	✓	12	✓	SOIC-16
C8051F831-GS	13	8	4	256	✓	12	✓	SOIC-16
C8051F832-GS	13	—	4	256	✓	12	✓	SOIC-16
C8051F833-GS	13	12	4	256	—	—	—	SOIC-16
C8051F834-GS	13	8	4	256	—	—	—	SOIC-16
C8051F835-GS	13	—	4	256	—	—	—	SOIC-16

所有设备上的含铅材料均为 100% 雾锡 (Sn)。

# C8051F80x-83x

---

## 3. 引脚定义

表 3.1. C8051F80x-83x 引脚定义

名称	引脚 QSOP-24	引脚 QFN-20	引脚 SOIC-16	类型	说明
GND	5	2	4		接地。 必需进行该接地连接。中心板可连接至地面或连接在 QFN-20 封装上。
V <sub>DD</sub>	6	3	5		电源电压。
$\overline{\text{RST}}$ / C2CK	7	4	6	数字 I/O  数字 I/O	设备复位。内部 POR 或 V <sub>DD</sub> 监控器开路输出。外部信源可以通过使引脚降低至少 10 微秒来启动系统复位。 C2 调试接口时钟信号。
P2.0/ C2D	8	5	7	数字 I/O  数字 I/O	C2 调试接口双向数据信号。与 P2.0 共享 20 针封装及与 P2.4 共享 24 针封装。 C2 调试接口双向数据信号。与 P2.0 共享 20 针封装及与 P2.4 共享 24 针封装。
P0.0/ VREF	4	1	3	数字 I/O 或音频 输入  音频输入	端口 0.0。  外部 VREF 输入。
P0.1	3	20	2	数字 I/O 或音频 输入	端口 0.1。
P0.2/ XTAL1	2	19	1	数字 I/O 或音频 输入  音频输入	端口 0.2。  外部时钟输入。此引脚为晶体或谐振器的外部振荡器回路。
P0.3/ XTAL2	23	18	16	数字 I/O 或音频 输入  音频 I/O 或数字 输入	端口 0.3。  外部时钟输出。就外部晶体或谐振器而言，此引脚为自激驱动。此引脚为 CMOS、电容器或 RC 振荡器设置的外部时钟输入。
P0.4	22	17	15	数字 I/O 或音频 输入	端口 0.4。

# C8051F80x-83x

表 3.1. C8051F80x-83x 针脚定义 (Continued)

名称	针脚 QSOP-24	针脚 QFN-20	针脚 SOIC-16	类型	说明
P0.5	21	16	14	数字 I/O 或音频 输入	端口 0.5。
P0.6/ CNVSTR	20	15	13	数字 I/O 或音频 输入  数字输入	端口 0.6。  ADC0 外部转化启动或 IDA0 更新信源输入。
P0.7	19	14	12	数字 I/O 或音频 输入	端口 0.7。
P1.0	18	13	11	数字 I/O 或音频 输入	端口 1.0。
P1.1	17	12	10	数字 I/O 或音频 输入	端口 1.1。
P1.2	16	11	9	数字 I/O 或音频 输入	端口 1.2。
P1.3	15	10	8	数字 I/O 或音频 输入	端口 1.3。
P1.4	14	9		数字 I/O 或音频 输入	端口 1.4。
P1.5	11	8		数字 I/O 或音频 输入	端口 1.5。
P1.6	10	7		数字 I/O 或音频 输入	端口 1.6。
P1.7	9	6		数字 I/O 或音频 输入	端口 1.7。
NC	1, 12, 13, 24				无连接。



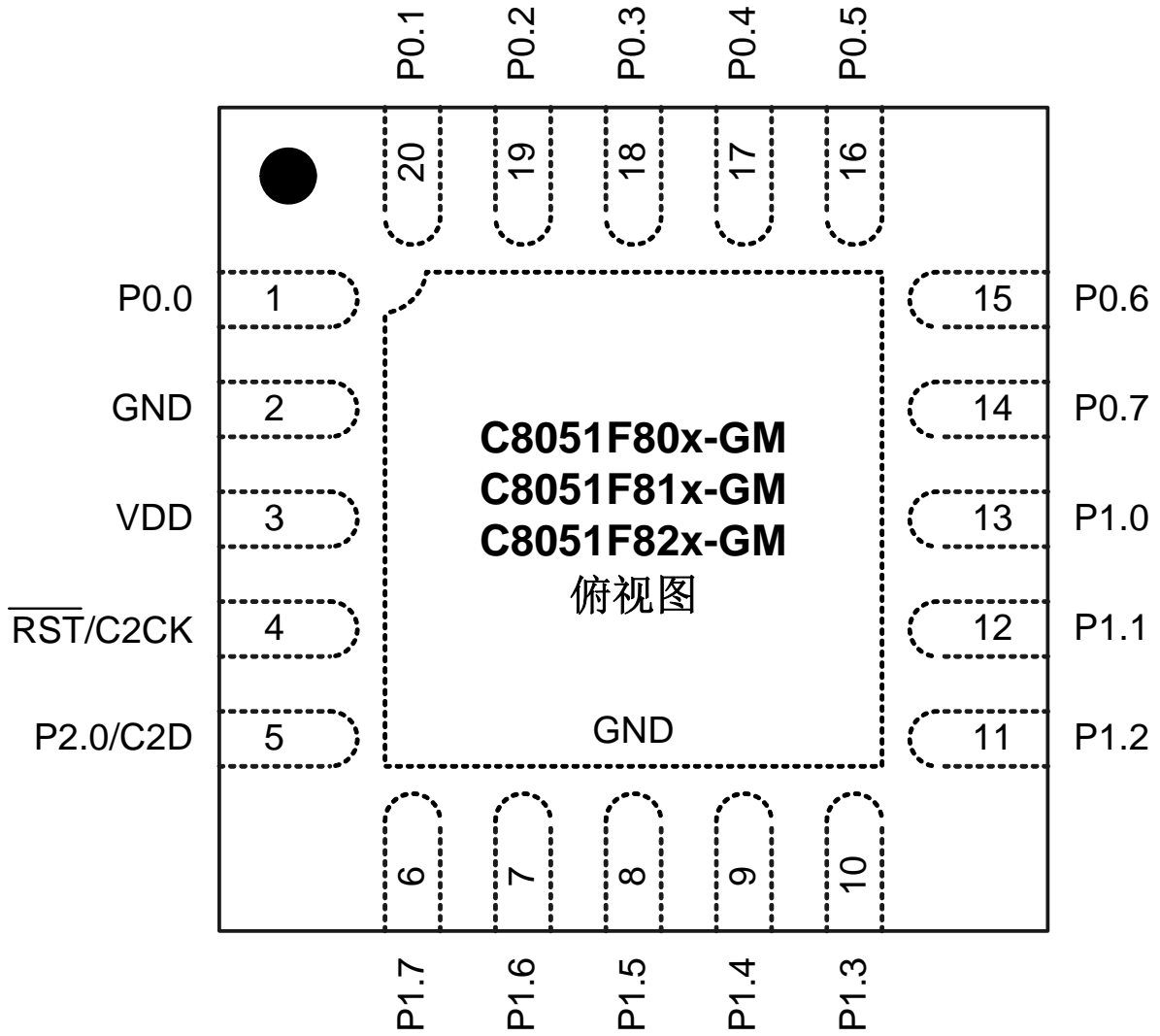


图 3.1. QFN-20 针脚分布图（俯视图）

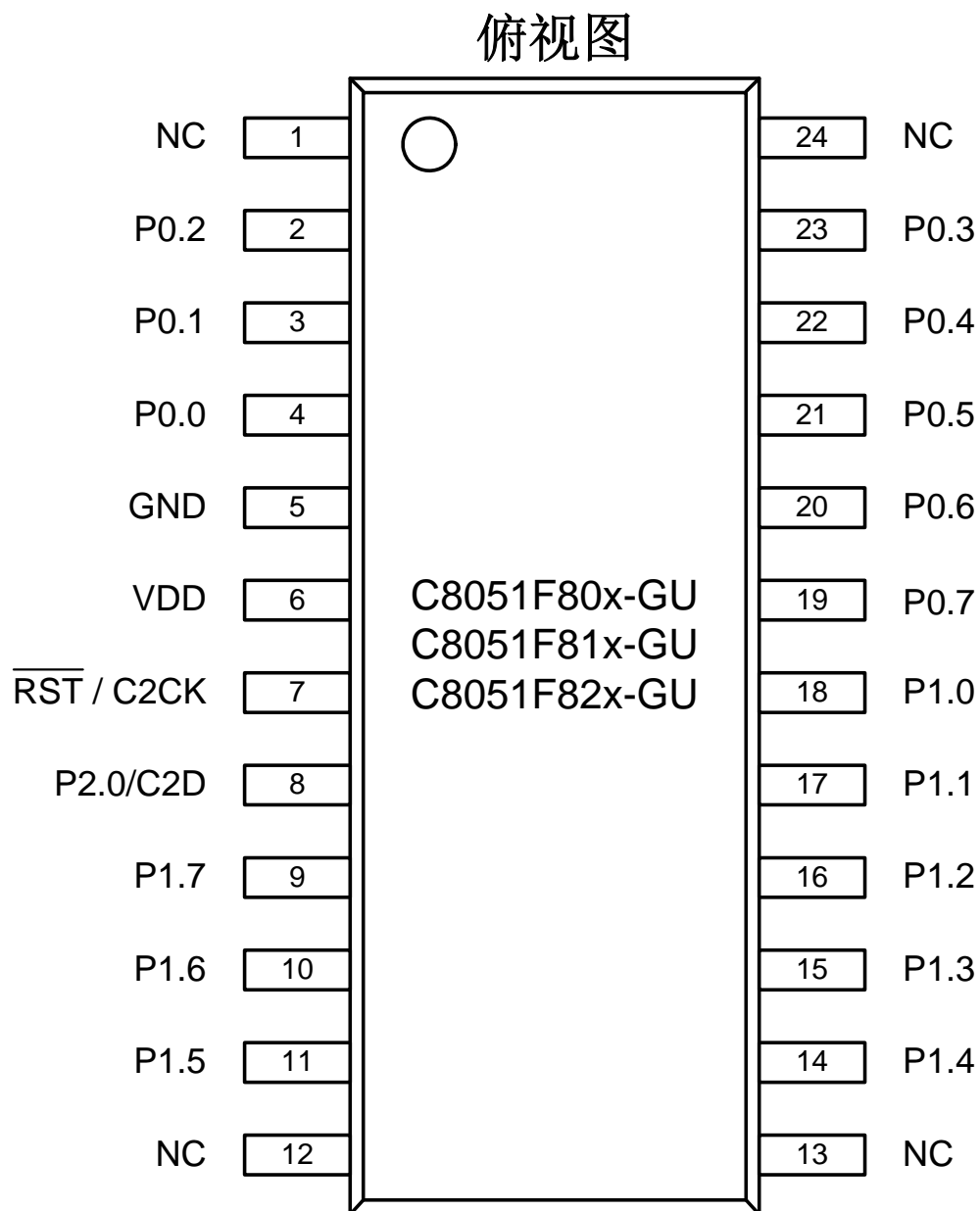


图 3.2. QSOP-24 引脚分布图（俯视图）

俯视图

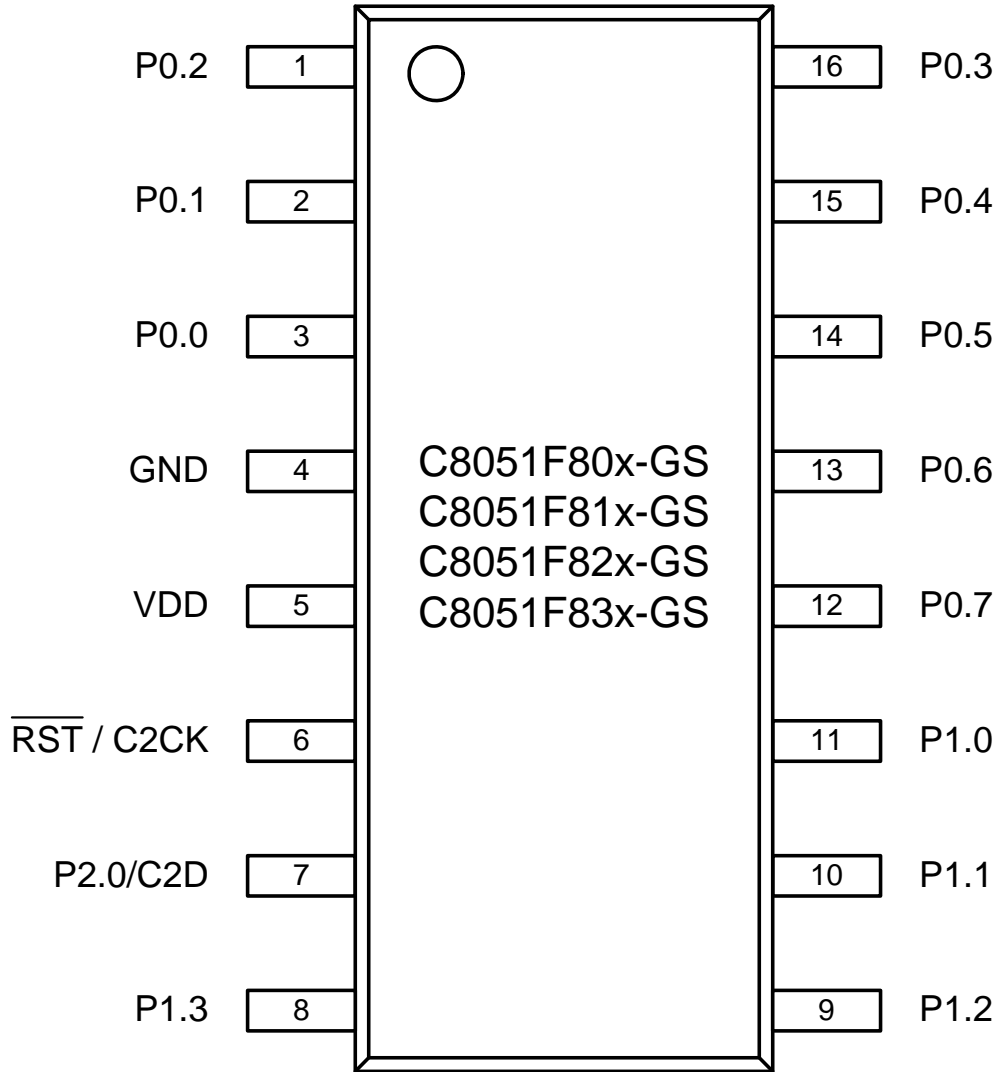


图 3.3. SOIC-16 引脚分布图（俯视图）

# C8051F80x-83x

---

## 4. QFN-20 封装规格

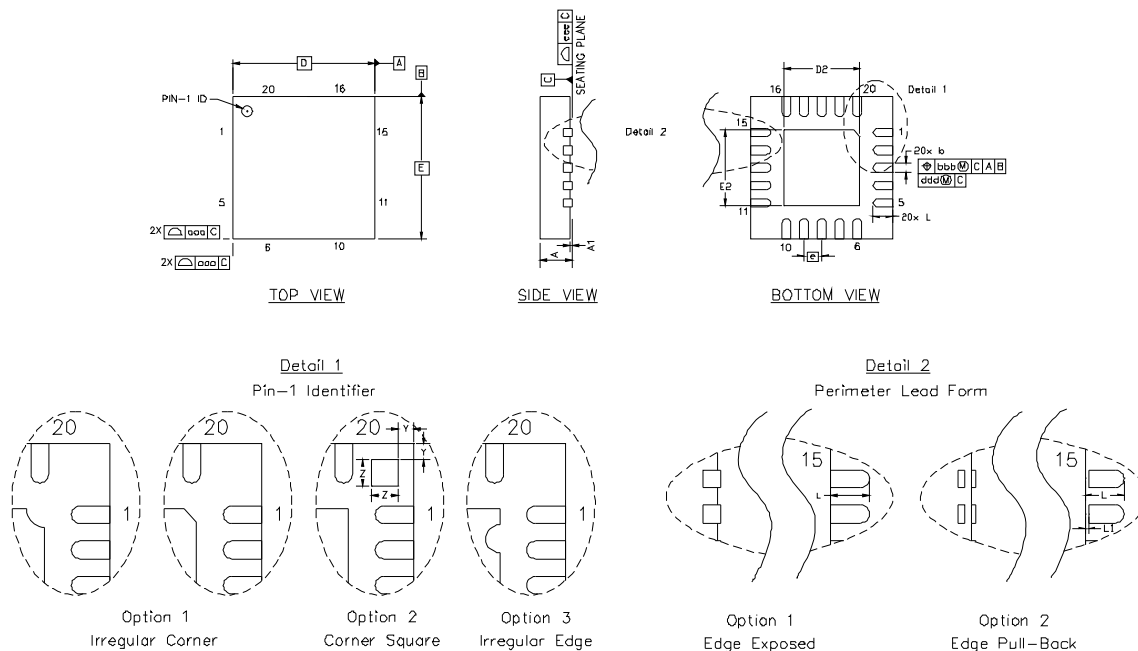


图 4.1. QFN-20 封装图

表 4.1. QFN-20 封装尺寸

尺寸	最小值	特有值	最大值	尺寸	最小值	特有值	最大值
A	0.80	0.90	1.00	L	0.45	0.55	0.65
A1	0.00	0.02	0.05	L1	0.00	—	0.15
b	0.18	0.25	0.30	aaa	—	—	0.15
D	4.00 BSC.			bbb	—	—	0.10
D2	2.00	2.15	2.25	ddd	—	—	0.05
e	0.50 BSC.			eee	—	—	0.08
E	4.00 BSC.			Z	—	0.43	—
E2	2.00	2.15	2.25	Y	—	0.18	—

**附注:**

1. 除非另有注明，否则，所有尺寸都以毫米 (mm) 列示。
2. 尺寸和容差参照 ANSI Y14.5M-1994。
3. 本图纸符合 JEDEC 固态外形图 MO-220、公差 VGGD，但自定义特征 D2、E2、Z、Y 和 L 的容差则依循供应商指示。
4. 获推荐的卡回流资料符合 JEDEC/IPC J-STD-020 有关小型元件的规格。

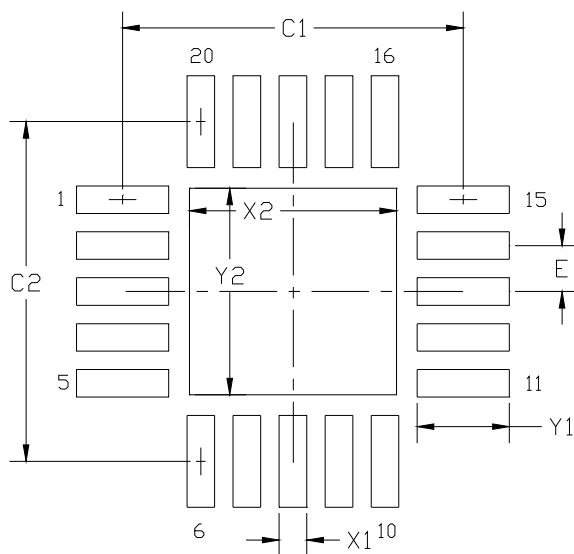


图 4.2. QFN-20 推荐的印制电路板焊盘图形

表 4.2. QFN-20 印制电路板焊盘图形尺寸

尺寸	最小值	最大值	尺寸	最小值	最大值
C1	3.70		X2	2.15	2.25
C2	3.70		Y1	0.90	1.00
E	0.50		Y2	2.15	2.25
X1	0.20	0.30			

**附注:**

一般情况

1. 除非另有注明，否则，所有尺寸都以毫米 (mm) 列示。
2. 尺寸和容差参照 ANSI Y14.5M-1994 的规格。
3. 本焊盘图形根据 IPC-7351 的指引设计。

阻焊层设计

4. 所有金属板都为非阻焊层限定 (NSMD)。阻焊层与金属板之间的间距至少为 60  $\mu\text{m}$ ，一直沿着金属板排列。

模板设计

5. 应使用不锈钢、激光切割并已电镀的模板以及梯形墙，以确保适当释放焊膏。
6. 模板厚度应为 0.125 mm (千分之五寸)。
7. 所有外围针脚的模板开孔与焊盘尺寸的比例应为 1:1。
8. 中心板应在 1.1 mm 的间距上 2x2 阵列的 0.95 mm 开口，以确保焊膏数量恰当。

卡组装

9. 建议使用免清洗的 3 型焊膏。
10. 获推荐的卡回流资料符合 JEDEC/IPC J-STD-020 有关小型元件的规格。

## 5. QSOP-24 封装规格

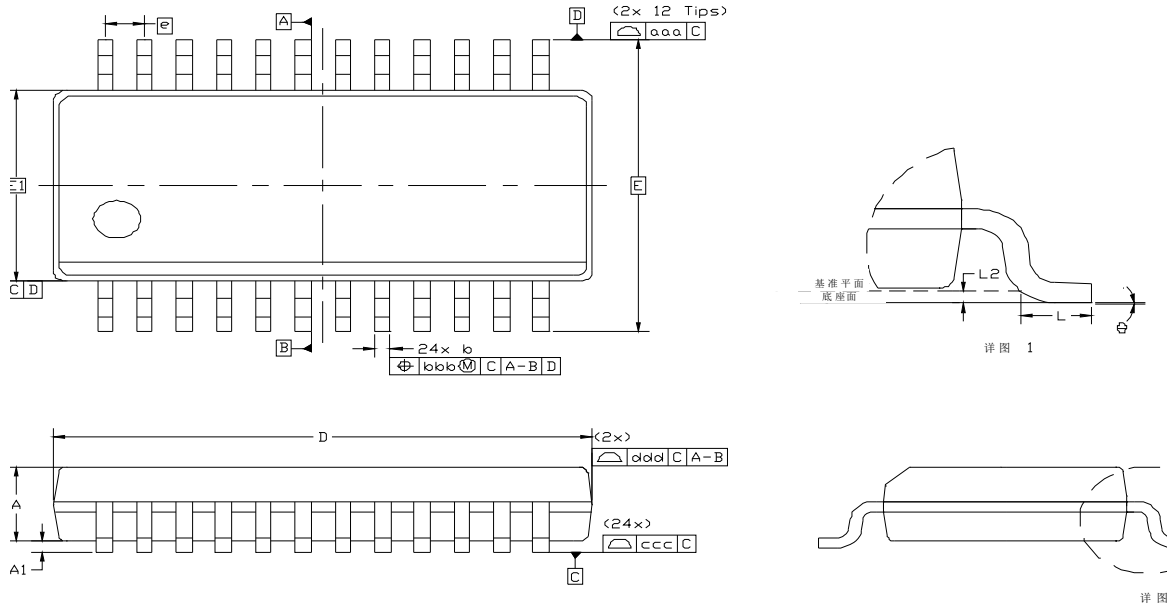


图 5.1. QSOP-24 封装图

表 5.1. QSOP-24 封装尺寸

尺寸	最小值	正常值	最大值	尺寸	最小值	正常值	最大值
A	—	—	1.75	L	0.40	—	1.27
A1	0.10	—	0.25	L2	0.25 BSC		
b	0.20	—	0.30	θ	0°	—	8°
c	0.10	—	0.25	aaa	0.20		
D	8.65 BSC			bbb	0.18		
E	6.00 BSC			ccc	0.10		
E1	3.90 BSC			ddd	0.10		
e	0.635 BSC						

**附注:**

1. 除非另有注明, 否则, 所有尺寸都以毫米 (mm) 列示。
2. 尺寸和容差参照 ANSI Y14.5M-1994。
3. 本图纸符合 JEDEC 外形图 MO-137、公差 AE。
4. 获推荐的卡回流资料符合 JEDEC/IPC J-STD-020 有关小型元件的规格。

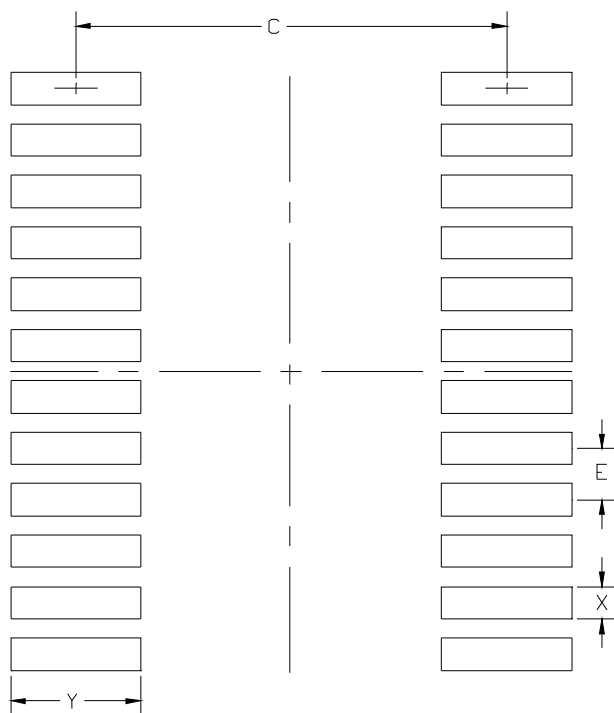


图 5.2. QSOP-24 印制电路板焊盘图形

表 5.2. QSOP-24 印制电路板焊盘图形尺寸

尺寸	最小值	最大值
C	5.20	5.30
E	0.635 BSC	
X	0.30	0.40
Y	1.50	1.60

**附注:**

一般情况

1. 除非另有注明，否则，所有尺寸都以毫米 (mm) 列示。
2. 本焊盘图形根据 IPC-7351 的指引设计。

阻焊层设计

3. 所有金属板都为非阻焊层限定 (NSMD)。阻焊层与金属板之间的间距至少为 60  $\mu\text{m}$ ，一直沿着金属板排列。

模板设计

4. 应使用不锈钢、激光切割并已电镀的模板以及梯形墙，以确保适当释放焊膏。
5. 模板厚度应为 0.125 mm (千分之五寸)。
6. 所有外围焊盘的模板开孔与焊盘尺寸的比例应为 1:1。

卡组装

7. 建议使用免清洗的 3 型焊膏。
8. 获推荐的卡回流资料符合 JEDEC/IPC J-STD-020 有关小型元件的规格。



## 6. SOIC-16 封装规格

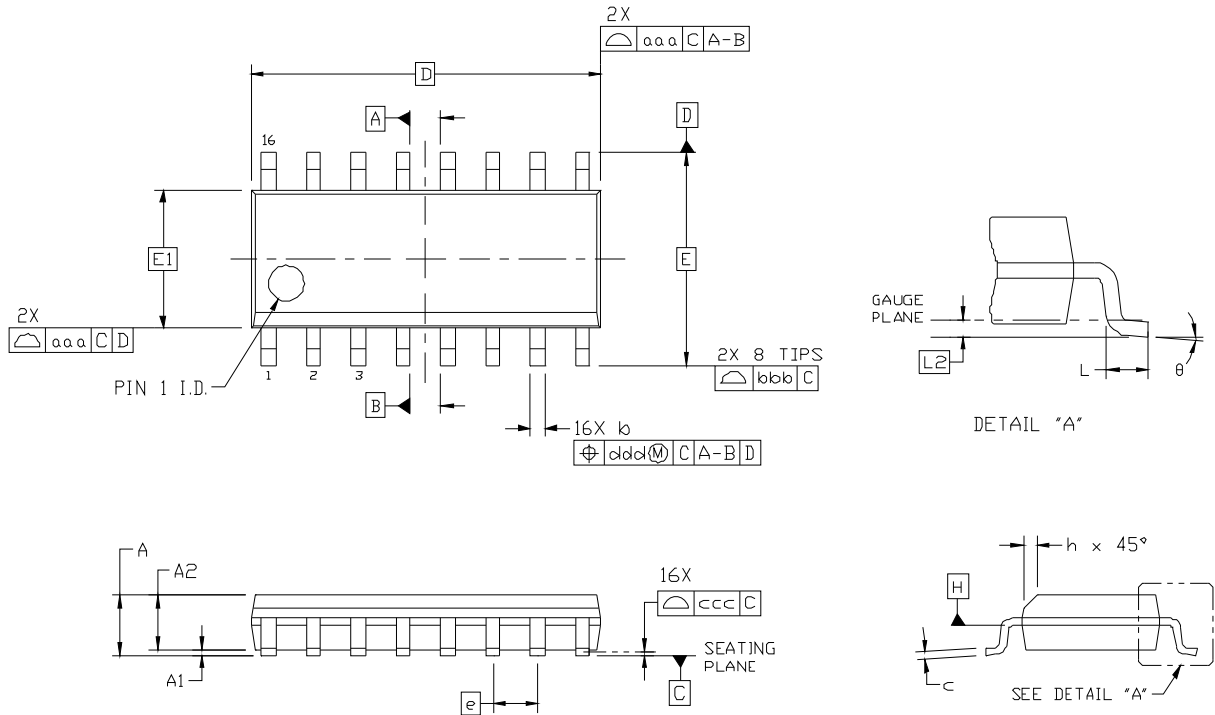


图 6.1. SOIC-16 封装图

表 6.1. SOIC-16 封装尺寸

尺寸	最小值	正常值	最大值	尺寸	最小值	正常值	最大值
A	—		1.75	L	0.40		1.27
A1	0.10		0.25	L2	0.25 BSC		
A2	1.25		—	h	0.25		0.50
b	0.31		0.51	theta	0°		8°
c	0.17		0.25	aaa	0.10		
D	9.90 BSC			bbb	0.20		
E	6.00 BSC			ccc	0.10		
E1	3.90 BSC			ddd	0.25		
e	1.27 BSC						

**附注:**

1. 除非另有注明，否则，所有尺寸都以毫米 (mm) 列示。
2. 尺寸和容差参照 ANSI Y14.5M-1994。
3. 本图纸符合 JEDEC 固态外形图 MS-012、公差 AC。
4. 获推荐的卡回流资料符合 JEDEC/IPC J-STD-020 有关小型元件的规格。

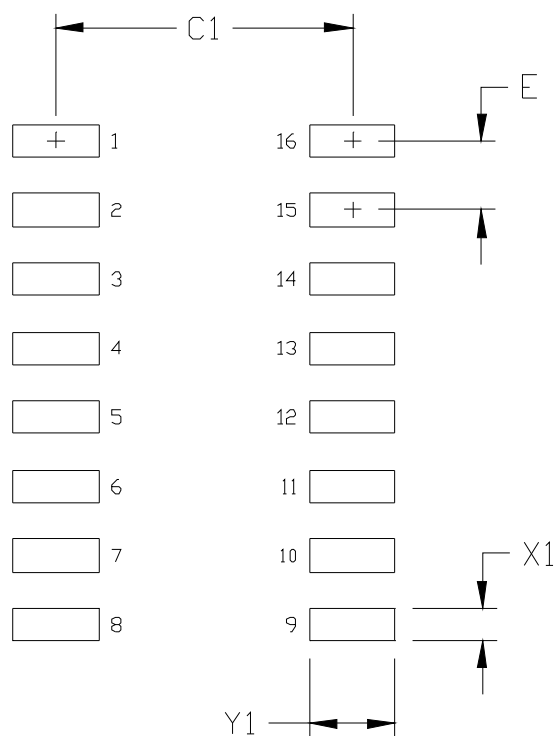


图 6.2. SOIC-16 印制电路板焊盘图形

表 6.2. SOIC-16 印制电路板焊盘图形尺寸

尺寸	特征	(mm)
C1	焊盘列间距	5.40
E	焊盘行距	1.27
X1	焊盘宽度	0.60
Y1	焊盘长度	1.55

**附注:**

一般情况

1. 除非另有注明, 否则, 所有尺寸都以毫米 (mm) 列示。
2. 本焊盘图形根据 IPC-7351 图形 SOIC127P600X165-16N 的密度等级 B (中等焊盘突出) 设计。
3. 所有形体尺寸都按最大实体条件 (MMC) 列示, 且已假设卡制作公差为 0.05 mm。

## 7. 电气特性

### 7.1. 绝对值最大规格

表 7.1. 绝对值最大等级

参数	条件	最小值	特有值	最大值	单位
偏差内环境温度		-55	—	125	°C
存储温度		-65	—	150	°C
$\overline{RST}$ 或任何端口 I/O 针脚上的 GND 电压		-0.3	—	5.8	V
$V_{DD}$ 上的 GND 电压		-0.3	—	4.2	V
通过 $V_{DD}$ 及 GND 的最大总电流		—	—	500	mA
流入 $\overline{RST}$ 或任何端口针脚的最大输出电流		—	—	100	mA
<p><b>附注：</b> 如使用负荷超过“绝对值最大等级”所列数值，可能对设备造成永久损害。上表仅列出负荷等级，并未说明设备在上述或高于规格操作列表所指的任何其它条件下的功能性操作。如设备长期处于最大的等级条件下，则其可靠性可能受到损害。</p>					

# C8051F80x-83x

## 7.2. 电气特性

表 7.2. 全球电气特性

除非另有指明，否则，温度为 -40 至 +85 °C，系统时钟为 25 MHz。

参数	条件	最小值	特有值	最大值	单位
供电电压		1.8	3.0	3.6	V
CPU 激活时（正常模式 <sup>1</sup> ）的 数字电源电流	V <sub>DD</sub> 为 1.8 V，时钟为 25 MHz	—	4.6	6.0	mA
	V <sub>DD</sub> 为 1.8 V，时钟为 1 MHz	—	1.2	—	mA
	V <sub>DD</sub> 为 1.8 V，时钟为 32 kHz	—	135	—	μA
	V <sub>DD</sub> 为 3.0 V，时钟为 25 MHz	—	5.5	6.5	mA
	V <sub>DD</sub> 为 3.0 V，时钟为 1 MHz	—	1.3	—	mA
	V <sub>DD</sub> 为 3.0 V，时钟为 32 kHz	—	150	—	μA
CPU 未激活时（空闲模式 <sup>1</sup> ） 的数字电源电流	V <sub>DD</sub> 为 1.8 V，时钟为 25 MHz	—	2	2.6	mA
	V <sub>DD</sub> 为 1.8 V，时钟为 1 MHz	—	190	—	μA
	V <sub>DD</sub> 为 1.8 V，时钟为 32 kHz	—	100	—	μA
	V <sub>DD</sub> 为 3.0 V，时钟为 25 MHz	—	2.3	2.8	mA
	V <sub>DD</sub> 为 3.0 V，时钟为 1 MHz	—	335	—	μA
	V <sub>DD</sub> 为 3.0 V，时钟为 32 kHz	—	115	—	μA
数字电源电流（关机）	振荡器未运行（停止模式），内部 稳压器关闭，25 °C	—	0.5	2	μA
	振荡器未运行（停止或暂停模式）， 内部稳压器启动，25 °C	—	105	140	μA
数字电源 RAM 数据保留电压		—	1.3	—	V
指定操作温度范围		-40	—	+85	°C
SYSCCLK（系统时钟频率）	请参见附注 2	0	—	25	MHz
T <sub>sysl</sub> （SYSCCLK 低时间）		18	—	—	ns
T <sub>sysl</sub> （SYSCCLK 高时间）		18	—	—	ns
附注： 1. 包括内部稳压器的偏差电流。 2. SYSCCLK 必须低于 32 kHz，以启动调试。					

表 7.3. 端口 I/O DC 电气特性

除非另有指明，否则， $V_{DD}$  为 1.8 至 3.6 V，温度为  $-40$  至  $+85$  °C。

参数	条件	最小值	特有值	最大值	单位
输出高压	$I_{OH}$ 为 $-3$ mA，端口 I/O 为推拉式	$V_{DD} - 0.7$	—	—	V
	$I_{OH}$ 为 $-10$ $\mu$ A，端口 I/O 为推拉式	$V_{DD} - 0.1$	—	—	V
	$I_{OH}$ 为 $-10$ mA，端口 I/O 为推拉式	—	$V_{DD} - 0.8$	—	V
输出低压	$I_{OL}$ 为 8.5 mA	—	—	0.6	V
	$I_{OL}$ 为 10 $\mu$ A	—	—	0.1	V
	$I_{OL}$ 为 25 mA	—	1.0	—	V
输入高压		$0.75 \times V_{DD}$	—	—	V
输入低压		—	—	0.6	V
输入泄露电流	弱上拉关闭	-1	—	1	$\mu$ A
	弱上拉开启， $V_{IN}$ 为 0 V	—	15	50	$\mu$ A

表 7.4. 复位电气特性

除非另有指明，否则， $V_{DD}$  为 1.8 至 3.6 V，温度为  $-40$  至  $+85$  °C。

参数	条件	最小值	特有值	最大值	单位
$\overline{RST}$ 输出低压	$I_{OL}$ 为 8.5 mA， $V_{DD}$ 为 1.8 V 至 3.6 V	—	—	0.6	V
$\overline{RST}$ 输入高压		$0.75 \times V_{DD}$	—	—	V
$\overline{RST}$ 输入低压		—	—	$0.3 \times V_{DD}$	$V_{DD}$
$\overline{RST}$ 输入上拉电流	$\overline{RST}$ 为 0.0 V	—	25	50	$\mu$ A
$V_{DD}$ POR 启动时间		—	—	1	ms
$V_{DD}$ 监视器阈值 ( $V_{RST}$ )		1.7	1.75	1.8	V
缺失时钟探测器超时	从最后一个系统时钟上升沿到产生复位的时间	100	500	1000	$\mu$ s
复位时间延迟	从退出任何复位源到开始执行位于 $0x0000$ 地址的代码之间的延时	—	—	30	$\mu$ s
产生系统复位的最小 $\overline{RST}$ 低电平时间		15	—	—	$\mu$ s
$V_{DD}$ 监视器启动时间	$V_{DD} = V_{RST} - 0.1$ V	—	50	—	$\mu$ s
$V_{DD}$ 监视器电源电流		—	20	30	$\mu$ A

表 7.5. 内部稳压器电气特性

除非另有指明，否则， $V_{DD}$  为 3.0 V，温度为  $-40$  至  $+85$  °C。

参数	条件	最小值	特有值	最大值	单位
输入电压范围		1.8	—	3.6	V
偏差电流		—	50	65	$\mu$ A

# C8051F80x-83x

表 7.6. 闪存电气特性

参数	条件	最小值	特有值	最大值	单位
闪存尺寸 (附注 1)	C8051F80x 和 C8051F810/1 C8051F812/3/4/5/6/7/8/9 和 C8051F82x C8051F830/1/2/3/4/5		16384 8192 4096		字节 字节 字节
擦写寿命		10000	—	—	周期
擦除时间	25 MHz 时钟	15	20	26	ms
写入时间	25 MHz 时钟	15	20	26	μs
闪存写入 / 擦除操作的时钟速度		1	—	—	MHz
附注: 内含安全锁定字节。					

表 7.7. 内部高频振荡器电气特性

除非另有指明, 否则,  $V_{DD}$  为 1.8 至 3.6 V;  $T_A$  为 -40 至 +85 °C。采用出厂校准的设置。

参数	条件	最小值	特有值	最大值	单位
振荡器频率	IFCN = 11b	24	24.5	25	MHz
振荡器电源电流	25 °C, $V_{DD} = 3.0$ V, OSCICN.7 = 1, OCSICN.5 = 0	—	350	650	μA

表 7.8. 电容传感电气特性

除非另有指明, 否则,  $V_{DD}$  为 1.8 至 3.6 V;  $T_A$  为 -40 至 +85 °C。

参数	条件	最小值	特有值	最大值	单位
转换时间	单次转换	26	38	50	μs
每种代码的电容		—	1	—	fF
外部电容负载		—	—	45	pF
量化噪音 <sup>1</sup>	RMS	—	3	—	fF
	峰间	—	20	—	fF
电源电流	25 °C 下 CS 模块偏置电流	—	40	60	μA
	25 °C 下, 单独使用 CS 模块时的最大 代码输出	—	75	105	μA
	CS 阈值时唤醒, 25 °C <sup>2</sup>	—	150	165	μA
附注:					
1. RMS 噪音等于一个标准值偏离。峰间噪音包含 ±3.3 个标准值偏离。					
2. 仅包括从稳压器、CS 模块和暂停模式下的 MCU 通过的电流。					

表 7.9. ADC0 电气特性

除非另有指明，否则， $V_{DD}$  为 3.0 V， $V_{REF}$  为 2.40 V ( $REFSL=0$ )，温度为  $-40$  至  $+85$  °C。

参数	条件	最小值	特有值	最大值	单位
<b>DC 精度</b>					
分辨率		10			bits
积分非线性		—	$\pm 0.5$	$\pm 1$	LSB
微分非线性	保证单调	—	$\pm 0.5$	$\pm 1$	LSB
偏移误差		-2	0	2	LSB
满度误差		-2	0	2	LSB
偏差温度系数		—	45	—	ppm/°C
<b>动态性能（10 kHz 正弦波单端输入，满度值的 1 dB 以下，200 ksps）</b>					
信噪比加失真		54	60	—	dB
总谐波失真	多达 5 次谐波	—	75	—	dB
无失真动态范围		—	-90	—	dB
<b>转换速率</b>					
SAR 转换时钟		—	—	8.33	MHz
转换时间（SAR 时钟数）	10 位模式	13	—	—	clocks
	8 位模式	11	—	—	clocks
跟踪 / 保持捕获时间	$V_{DD} \geq 2.0$ V	300	—	—	ns
	$V_{DD} < 2.0$ V	2.0	—	—	$\mu$ s
吞吐速率		—	—	500	ksps
<b>模拟输入</b>					
ADC 输入电压范围		0	—	$V_{REF}$	V
采样容量	1x 增益	—	5	—	pF
	0.5x 增益	—	3	—	pF
输入多路复用器电阻		—	5	—	k $\Omega$
<b>电源规格</b>					
电源电流	操作模式，500 ksps	—	630	1000	$\mu$ A
电源抑制比		—	-70	—	dB

# C8051F80x-83x

**表 7.10. 电源管理电气特性**

除非另有指明，否则， $V_{DD}$  为 1.8 至 3.6 V； $T_A$  为  $-40$  至  $+85$  °C。采用出厂校准的设置。

参数	条件	最小值	特有值	最大值	单位
空闲模式唤醒时间		2	—	3	SYCLKs
暂停模式唤醒时间		—	500	—	ns

**表 7.11. 温度传感器电气特性**

除非另有指明，否则， $V_{DD}$  为 3.0 V，温度为  $-40$  至  $+85$  °C。

参数	条件	最小值	特有值	最大值	单位
线性度		—	1	—	°C
斜率		—	2.43	—	mV/°C
斜率误差 *		—	$\pm 45$	—	$\mu V/°C$
偏移	温度为 0 °C	—	873	—	mV
偏移误差 *	温度为 0 °C	—	$\pm 14.5$	—	mV

\* 附注：代表偏离平均值一个标准差。

**表 7.12. 电压基准电气特性**

除非另有指明，否则， $V_{DD}$  为 1.8 至 3.6 V；温度为  $-40$  至  $+85$  °C。

参数	条件	最小值	特有值	最大值	单位
<b>内部高速基准 (REFSL[1:0] = 11)</b>					
输出电压	环境温度 25 °C	1.55	1.65	1.75	V
启动时间		—	—	1.7	$\mu s$
电源电流		—	180	—	$\mu A$
<b>外部基准 (REF0E = 0)</b>					
输入电压范围		0	—	$V_{DD}$	
输入电流	采样速率 = 500 ksp/s； $V_{REF} = 3.0$ V	—	7	—	$\mu A$



表 7.13. 比较器电气特性

除非另有指明，否则， $V_{DD}$  为 3.0 V，温度为  $-40$  至  $+85$  °C。

参数	条件	最小值	特有值	最大值	单位
响应时间： 模式 0, $V_{cm}^* = 1.5$ V	CP0+ – CP0– = 100 mV	—	220	—	ns
	CP0+ – CP0– = –100 mV	—	225	—	ns
响应时间： 模式 1, $V_{cm}^* = 1.5$ V	CP0+ – CP0– = 100 mV	—	340	—	ns
	CP0+ – CP0– = –100 mV	—	380	—	ns
响应时间： 模式 2, $V_{cm}^* = 1.5$ V	CP0+ – CP0– = 100 mV	—	510	—	ns
	CP0+ – CP0– = –100 mV	—	945	—	ns
响应时间： 模式 3, $V_{cm}^* = 1.5$ V	CP0+ – CP0– = 100 mV	—	1500	—	ns
	CP0+ – CP0– = –100 mV	—	5000	—	ns
共模抑制比		—	1	4	mV/V
正向回差电压 1	模式 2, CP0HYP1–0 = 00b	—	0	1	mV
正向回差电压 2	模式 2, CP0HYP1–0 = 01b	2	5	10	mV
正向回差电压 3	模式 2, CP0HYP1–0 = 10b	7	10	20	mV
正向回差电压 4	模式 2, CP0HYP1–0 = 11b	10	20	30	mV
负向回差电压 1	模式 2, CP0HYN1–0 = 00b	—	0	1	mV
负向回差电压 2	模式 2, CP0HYN1–0 = 01b	2	5	10	mV
负向回差电压 3	模式 2, CP0HYN1–0 = 10b	7	10	20	mV
负向回差电压 4	模式 2, CP0HYN1–0 = 11b	10	20	30	mV
反相或同相输入电压范围		–0.25	—	$V_{DD} + 0.25$	V
输入偏移电压		–7.5	—	7.5	mV
<b>电源规格</b>					
电源抑制比		—	0.1	—	mV/V
上电时间		—	10	—	$\mu$ s
电源电流 (DC)	模式 0	—	20	—	$\mu$ A
	模式 1	—	8	—	$\mu$ A
	模式 2	—	3	—	$\mu$ A
	模式 3	—	0.5	—	$\mu$ A
附注： $V_{cm}$ 是 CP0+ 和 CP0– 上的共模电压。					

# C8051F80x-83x

---



## 8. 10 位 ADC (ADC0)

C8051F800/1/2/3/4/5、C8051F812/3/4/5/6/7、C8051F824/5/6 和 C8051F830/1/2 上的 ADC0 是一个 500 kps 10 位逐次逼近寄存器 (SAR) ADC，ADC 中集成了追踪保持电路、可编程至 1x 或 0.5x 的增益阶段和可编程窗口探测器。ADC 可在软件控制下通过特殊功能寄存器完全设置。ADC 可被设置为用于使用模拟多路复用器（请参阅 56 “8.5. ADC0 模拟多路复用器”）测量各种不同的信号。ADC 的电压基准按第 9 条所述选择。58 “9. 温度传感器” 描述。只有当 ADC0 控制寄存器 (ADC0CN) 中的 AD0EN 位设置为逻辑 1 时，ADC0 子系统才能启用。当 AD0EN 位为逻辑 0 时，ADC0 子系统处于低功耗关断状态。

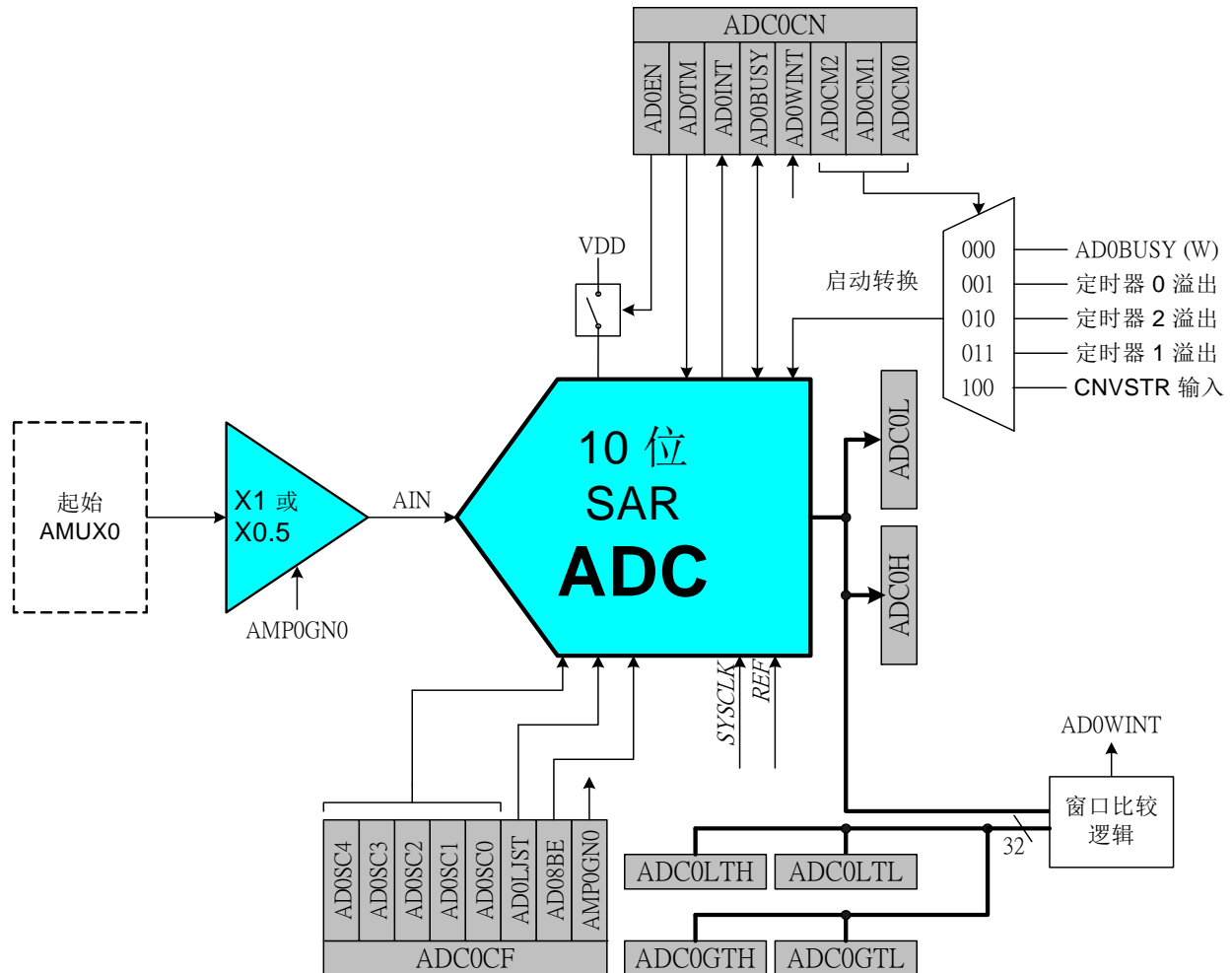


图 8.1. ADC0 功能结构图

# C8051F80x-83x

## 8.1. 输出代码格式

ADC 测量有关 GND 的输入电压。完成每次转换时，寄存器 ADC0H 和 ADC0L 包含高字节和低字节 ADC 输出转换码。数据可根据 AD0LJST 位的设置设为右对齐或左对齐。转换码为 10 位无符号整数。所测的输入范围为 0 到  $V_{REF} \times 1023/1024$ 。下面列举右对齐和左对齐数据代码示例。ADC0H 和 ADC0L 寄存器中未使用的位设为 0。

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

## 8.2. 8 位模式

将寄存器 ADC0CF 中的 ADC08BE 位设为 1 将使 ADC 处于 8 位模式。在 8 位模式中，只有 8 个数据 MSB 被转换，且 ADC0H 寄存器保存相关结果。AD0LJST 位在 8 位模式中被忽略。8 位转换比 10 位转换少两个 SAR 时钟周期，因此，该转换可更快完成，且使用较慢的 SAR 时钟也可达到 500 ksp/s 的采样速率。

## 8.3. 操作模式

ADC0 的最高转换速度为 500 ksp/s。ADC0 转换时钟由系统时钟分频得到，分频数由 ADC0CF 寄存器中的 AD0SC 位决定。

### 8.3.1. 启动转换

转换启动有六种方式，由寄存器 ADC0CN 中的 ADC0 转换启动模式位 (AD0CM2-0) 的设定状态决定采用哪一种方式。转换触发源有：

1. 写 1 到寄存器 ADC0CN 的 AD0BUSY 位
2. 定时器 0 溢出（即定时的连续转换）
3. 定时器 2 溢出
4. 定时器 1 溢出
5. CNVSTR 输入信号的上升沿

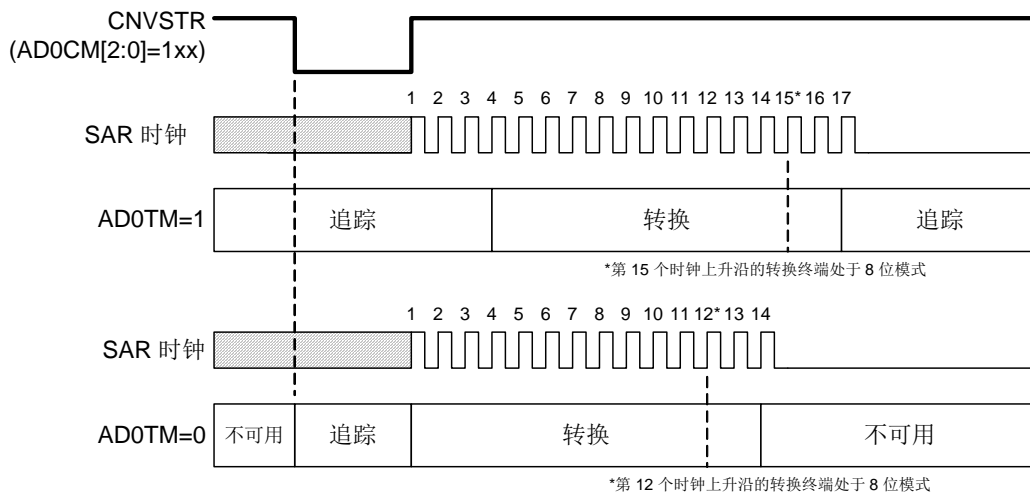
向 AD0BUSY 写 1 的方式提供了用软件控制 ADC0 的能力，借此“应要求”执行转换。转换期间，AD0BUSY 位被设为逻辑 1，转换结束后复位为逻辑 0。AD0BUSY 位的下降沿触发中断（允许时）并设置 ADC0 中断标志 (AD0INT)。应使用 ADC0 中断标志 (AD0INT) 来检测 ADC 转换是否完成。当 AD0INT 位为逻辑 1 时，ADC0 数据寄存器 ADC0H:ADC0L 中的转换数据有效。当转换源是定时器 2 溢出时，如果定时器 2/3 处于 8 位模式，则使用低字节溢出；如果定时器 2 处于 16 位模式，则使用高字节溢出。有关定时器设置，请参见第 209 页“28. 定时器”。

**关于使用 CNVSTR 的重要事项：**CNVSTR 输入引脚也是端口 I/O 引脚。当使用 CNVSTR 输入作为 ADC0 转换源时，应使数字交叉开关跳过相关引脚。有关端口 I/O 设置的详细信息，请参见第 138 页“23. 端口输入 / 输出”。

## 8.3.2. 追踪模式

寄存器 ADC0CN 中的 AD0TM 位能启动“延时的转换”，且将使实际转换延迟三个 SAR 时钟周期，在这三个 SAR 时钟周期中，ADC 将继续追踪输入。若 AD0TM 被置逻辑 0，转换将立即开始，而无额外追踪时间。若内部启动转换源，则 ADC 将在未执行转换时始终追踪。如果使用 CNVSTR 信号启动转换，则当 AD0TM 为逻辑 1 或当 AD0TM 为逻辑 0 时，ADC0 将进行追踪，且 CNVSTR 保持较低水平。有关追踪和转换定时的详细信息，请参见图 8.2。当 AMUX 设置因设置时间要求（请参阅第 49 页“8.3.3. 设置时间要求”）而频繁改变时，延时转换模式十分有用。

### A. 使用外部触发源的 ADC 定时



### B. 使用内部触发源的 ADC 定时

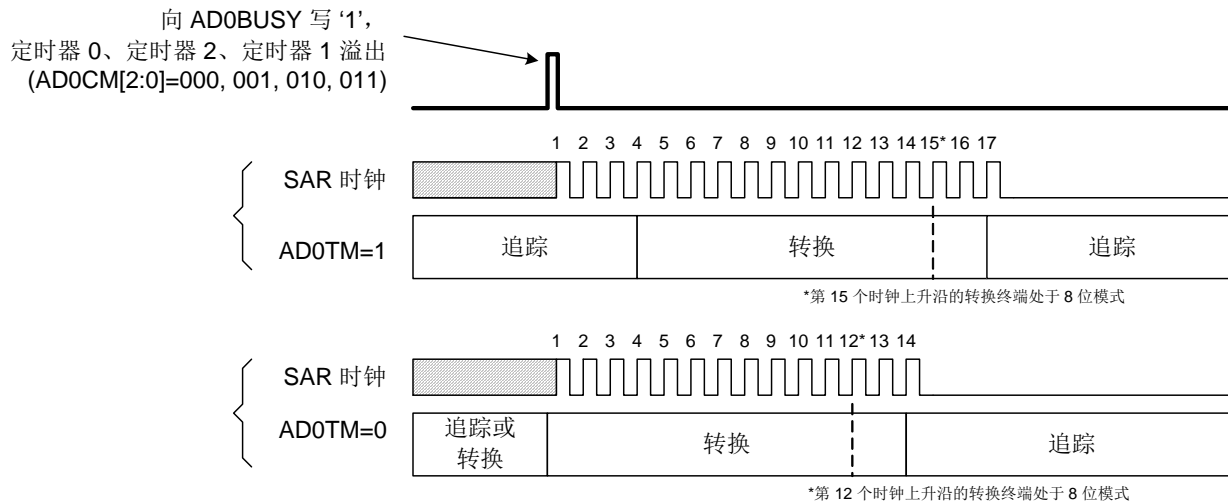


图 8.2. 10 位 ADC 追踪和转换范例定时

# C8051F80x-83x

## 8.3.3. 设置时间要求

在每次转换前，必需有一个最小追踪时间，以确保准确执行转换。该追踪时间由任何串联电阻（包括 AMUX0 电阻）、ADC0 采样容量和所要求的转换精度决定。在延时追踪模式下，每次启动转换时，均使用三个 SAR 时钟追踪。对于大多数应用，这三个 SAR 时钟可以满足最小追踪时间要求。

图 8.3 列示等效 ADC0 输入电路。对于一个给定的设置精度 (SA)，所需 ADC0 设置时间可用方程 8.1 估算。有关 ADC0 最小设置时间要求以及 mux 电阻和采样电容器数值，请参见表 7.9。

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

### 方程 8.1. ADC0 设置时间要求

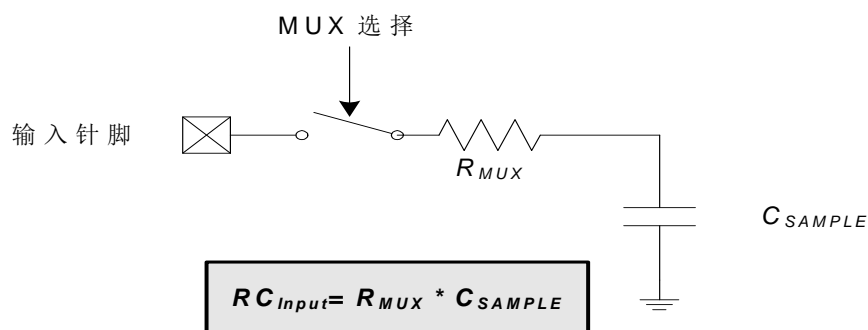
其中：

SA 是设置精度，用一个 LSB 的分数表示（例如，设置精度 0.25 对应 1/4 LSB）

t 为所需的设置时间，以秒为单位

$R_{TOTAL}$  为 AMUX0 电阻与任何外部信号源电阻之和。

n 为 ADC 分辨率，用比特表示 (10)。



附注：有关  $R_{MUX}$  和  $C_{SAMPLE}$  参数，请参见电气规格表。

图 8.3. ADC0 等效输入电路

## SFR 定义 8.1. ADC0CF: ADC0 配置寄存器

位	7	6	5	4	3	2	1	0
名称	AD0SC[4:0]					AD0LJST	AD08BE	AMP0GN0
类型	R/W					R/W	R/W	R/W
复位	1	1	1	1	1	0	0	1

SFR 地址 = 0xBC

位	名称	功能
7:3	AD0SC[4:0]	<p><b>ADC0 SAR 转换时钟周期控制位。</b></p> <p>SAR 转换时钟来源于系统时钟，由下面的方程式得出，其中 AD0SC 表示 AD0SC 4-0 中保存的 5 位数值。SAR 转换时钟要求已列于 ADC 规格表中。</p> $AD0SC = \frac{SYSCLK}{CLK_{SAR}} - 1$
2	AD0LJST	<p><b>ADC0 左对齐选择。</b></p> <p>0: ADC0H:ADC0L 寄存器中的数据均为右对齐。            1: ADC0H:ADC0L 寄存器中的数据均为左对齐。            附注: AD0LJST 位仅在 10 位模式 (AD08BE = 0) 下有效。</p>
1	AD08BE	<p><b>8 位模式启用。</b></p> <p>0: ADC 在 10 位模式 (正常) 下运行。            1: ADC 在 8 位模式下运行。            附注: 当 AD08BE 设为 1 时, AD0LJST 位被忽略。</p>
0	AMP0GN0	<p><b>ADC 增益控制位。</b></p> <p>0: 增益 = 0.5            1: 增益 = 1</p>

# C8051F80x-83x

## SFR 定义 8.2. ADC0H: ADC0 数据字 MSB 寄存器

位	7	6	5	4	3	2	1	0
名称	ADC0H[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xBE

位	名称	功能
7:0	ADC0H[7:0]	<b>ADC0 数据字高序位。</b> 当 AD0LJST = 0 时: 位 7-2 的读数将为 000000b。位 1-0 为 10 位 ADC0 数据字的高 2 位。 当 AD0LJST = 1 时: 位 7-0 为 10 位 ADC0 数据字的最高位。 <b>附注:</b> 在 8 位模式下, AD0LJST 被忽略, 且 ADC0H 保存 8 位数据字。

## SFR 定义 8.3. ADC0L: ADC0 数据字 LSB 寄存器

位	7	6	5	4	3	2	1	0
名称	ADC0L[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xBD

位	名称	功能
7:0	ADC0L[7:0]	<b>ADC0 数据字低序位。</b> 当 AD0LJST = 0 时: 位 7-0 为 10 位数据字的低 8 位。 当 AD0LJST = 1 时: 位 7-6 为 10 位数据字的低 2 位。位 5-0 的读数将一直为 0。 <b>附注:</b> 在 8 位模式下, AD0LJST 被忽略, 且 ADC0L 的读数将复 00000000b。



## SFR 定义 8.4. ADC0CN: ADC0 控制寄存器

位	7	6	5	4	3	2	1	0
名称	AD0EN	AD0TM	AD0INT	AD0BUSY	AD0WINT	AD0CM[2:0]		
类型	R/W	R/W	R/W	R/W	R/W	R/W		
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xE8 ; 可位寻址

位	名称	功能	
7	AD0EN	<b>ADC0 启用位。</b> 0: ADC0 禁用。ADC0 处于低耗关闭状态。 1: ADC0 启用。ADC0 处于活动状态, 并准备转换数据。	
6	AD0TM	<b>ADC0 追踪模式位。</b> 0: 正常追踪模式: 当 ADC0 被启用时, 除了转换期间之外, 一直处于追踪模式。如 AD0CM[2:0] 所定义, 启动转换事件发生后, 转换立即开始。 1: 延时追踪模式: 当 ADC0 被启用时, 如转换未进行, 则会追踪输入。启动转换信号启动三个额外追踪 SAR 时钟, 然后开始转换。	
5	AD0INT	<b>ADC0 转换结束中断标志。</b> 0: 从 AD0INT 最后一次将该位清 0 后, ADC0 还没有完成一次数据转换。 1: ADC0 完成了一次数据转换。	
4	AD0BUSY	<b>ADC0 忙标志位。</b>	<b>读:</b> 0: ADC0 转换未进行。 1: ADC0 转换正在进行。 <b>写:</b> 0: 无作用。 1: 若 AD0CM[2:0] = 000b 则启动 ADC0 转换
3	AD0WINT	<b>ADC0 窗口比较中断标志。</b> 0: 该标志最后一次被清除后, 未发生 ADC0 窗口比较数据匹配。 1: 发生 ADC0 窗口比较数据匹配。	
2:0	AD0CM[2:0]	<b>ADC0 转换启动模式选择。</b> 000: ADC0 启动转换源为向 AD0BUSY 写 1。 001: ADC0 启动转换源为定时器 0 溢出。 010: ADC0 启动转换源为定时器 2 溢出。 011: ADC0 启动转换源为定时器 1 溢出。 100: ADC0 启动转换源为外部 CNVSTR 上升沿。 101-111: 备用。	

# C8051F80x-83x

## 8.4. 可编程窗口探测器

ADC 可编程窗口探测器不断地将 ADC0 输出寄存器与用户编程的极限值进行比较，并在检测到越限条件时通知系统控制器。这在一个中断驱动系统中尤其有效，既可以节省代码空间和 CPU 带宽，又能提供快速系统响应时间。窗口探测器中断标志（ADC0CN 寄存器中的 AD0WINT）也可以用于查询模式。ADC0 下限（大于）寄存器（ADC0GTH、ADC0GTL）和上限（小于）寄存器（ADC0LTH、ADC0LTL）中保持比较值。窗口探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 ADC0 上限和 ADC0 下限寄存器的编程值。

### SFR 定义 8.5. ADC0GTH: ADC0 下限数据高字节寄存器

位	7	6	5	4	3	2	1	0
名称	ADC0GTH[7:0]							
类型	R/W							
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0xC4

位	名称	功能
7:0	ADC0GTH[7:0]	ADC0 下限数据字高序位。

### SFR 定义 8.6. ADC0GTL: ADC0 下限数据低字节寄存器

位	7	6	5	4	3	2	1	0
名称	ADC0GTL[7:0]							
类型	R/W							
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0xC3

位	名称	功能
7:0	ADC0GTL[7:0]	ADC0 下限数据字低序位。

**SFR 定义 8.7. ADC0LTH: ADC0 上限数据高字节寄存器**

位	7	6	5	4	3	2	1	0
名称	ADC0LTH[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xC6

位	名称	功能
7:0	ADC0LTH[7:0]	ADC0 上限数据字高序位。

**SFR 定义 8.8. ADC0LTL: ADC0 上限数据低字节寄存器**

位	7	6	5	4	3	2	1	0
名称	ADC0LTL[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xC5

位	名称	功能
7:0	ADC0LTL[7:0]	ADC0 上限数据字低序位。

# C8051F80x-83x

## 8.4.1. 窗口探测器范例

图 8.4 列示两个右对齐数据窗口比较的范例，其中  $ADC0LTH:ADC0LTL = 0x0080$  (128d) 及  $ADC0GTH:ADC0GTL = 0x0040$  (64d)。GND 输入电压范围可为 0 到  $VREF \times (1023/1024)$ ，用一个 10 位无符号整数值表示。对于左边的范例，如果  $ADC0$  转换结果数据字 ( $ADC0H:ADC0L$ ) 位于由  $ADC0GTH:ADC0GTL$  和  $ADC0LTH:ADC0LTL$  定义的范围之内（即  $0x0040 < ADC0H:ADC0L < 0x0080$ ），则会产生一个  $AD0WINT$  中断。对于右边的范例，如果  $ADC0$  转换结果数据字位于由  $ADC0GT$  和  $ADC0LT$  寄存器定义的范围之外（即  $ADC0H:ADC0L < 0x0040$  或  $ADC0H:ADC0L > 0x0080$ ），则会产生一个  $AD0WINT$  中断。图 8.5 列示使用具有相同比较值的左对齐数据的范例。

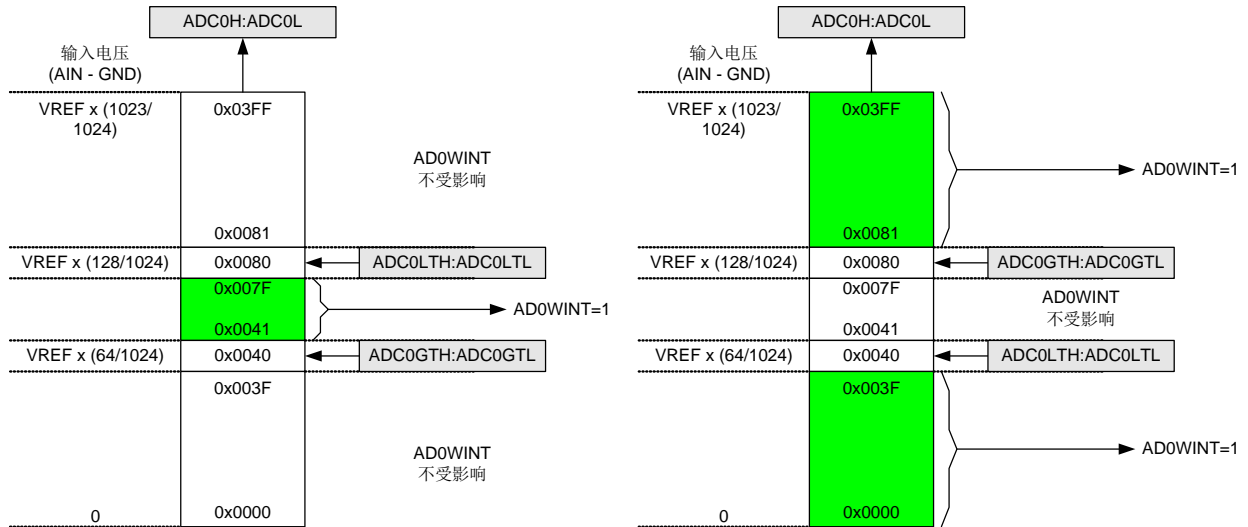


图 8.4. ADC 窗口比较范例：右对齐数据

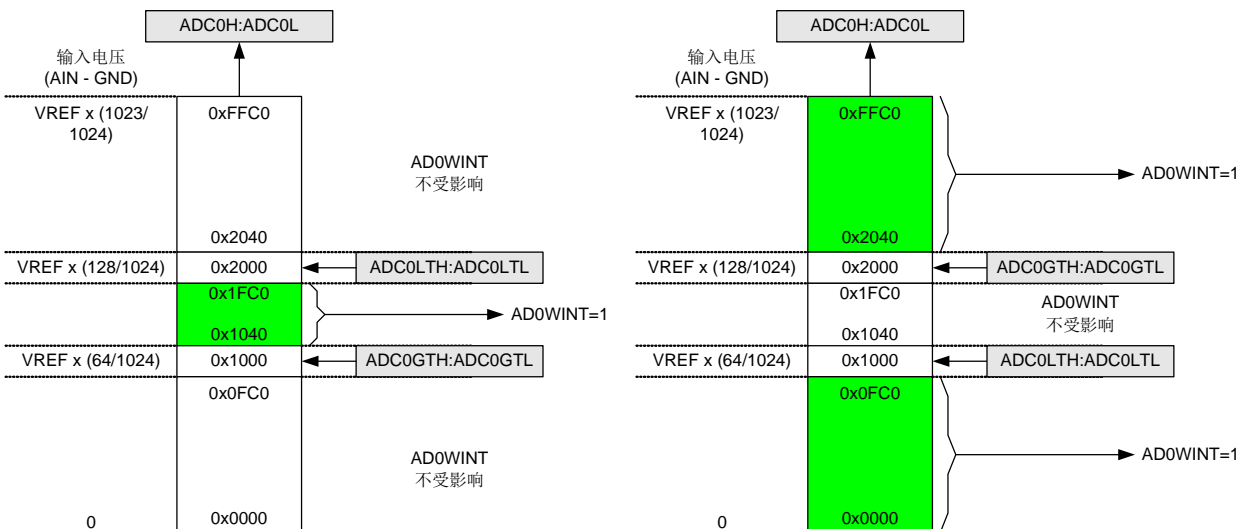


图 8.5. ADC 窗口比较范例：左对齐数据

## 8.5. ADC0 模拟多路复用器

C8051F800/1/2/3/4/5, C8051F812/3/4/5/6/7, C8051F824/5/6, and C8051F830/1/2 上的 ADC0 使用一个模拟输入多路复用器选择 ADC 的正输入。以下任一项都可以被选择为正输入：端口 0 或端口 1 I/O 针脚、片上温度传感器或正电源 ( $V_{DD}$ )。ADC0 输入通道在 ADC0MX 寄存器中选择（见 SFR 定义 8.9）。

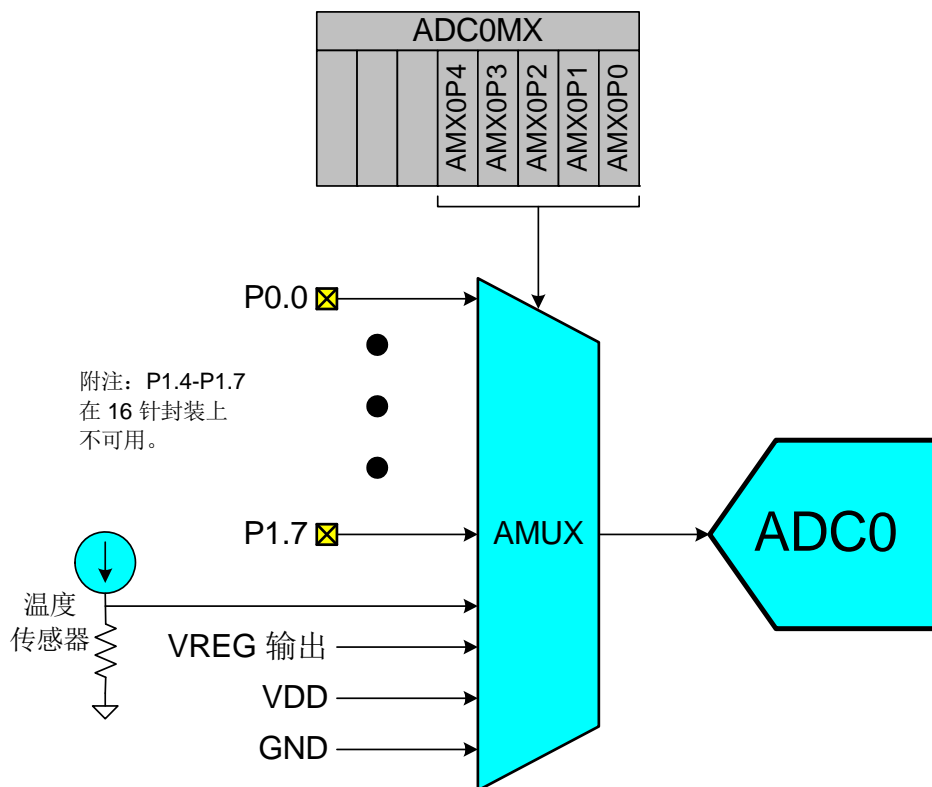


图 8.6. ADC0 多路复用器功能结构图

**有关 ADC0 输入设置的重要事项：**被选择为 ADC0 输入的端口针脚应被设置为模拟输入，并且应被数字交叉开关跳过。要将一个端口针脚设置为模拟输入，应将  $PnMDIN$  寄存器中的相应位设为 0。要使交叉开关跳过一个端口针脚，应将寄存器  $PnSKIP$  中的相应位设为 1。有关端口 I/O 设置的更多详细资料，请参见“23. 端口输入/输出”。

# C8051F80x-83x

## SFR 定义 8.9. ADC0MX: AMUX0 通道选择寄存器

位	7	6	5	4	3	2	1	0
名称				AMX0P[3:0]				
类型	R	R	R	R/W				
复位	0	0	0	1	1	1	1	1

SFR 地址 = 0xBB

位	名称	功能		
7:5	未使用	读 = 000b ; 写 = 无需在意。		
4:0	AMX0P[4:0]	<b>AMUX0 正输入选择寄存器。</b>		
			<b>20 针和 24 针设备</b>	<b>16 针设备</b>
		00000:	P0.0	P0.0
		00001:	P0.1	P0.1
		00010:	P0.2	P0.2
		00011:	P0.3	P0.3
		00100:	P0.4	P0.4
		00101:	P0.5	P0.5
		00110:	P0.6	P0.6
		00111:	P0.7	P0.7
		01000	P1.0	P1.0
		01001	P1.1	P1.1
		01010	P1.2	P1.2
		01011	P1.3	P1.3
		01100	P1.4	备用。
		01101	P1.5	备用。
		01110	P1.6	备用。
		01111	P1.7	备用。
		10000:	温度传感器	温度传感器
10001:	VREG 输出	VREG 输出		
10010:	VDD	VDD		
10011:	GND	GND		
10100 – 11111:	未选择任何输入			

## 9. 温度传感器

C8051F800/1/2/3/4/5, C8051F812/3/4/5/6/7, C8051F824/5/6, and C8051F830/1/2 上已配备一个片上温度传感器，可通过单端设置中的 ADC 多路复用器直接访问。要使用 ADC 测量温度传感器，ADC mux 通道应设置为连接至温度传感器。温度传感器转移函数如图 9.1 所示。当 ADC 多路复用器设置正确时，输出电压 ( $V_{TEMP}$ ) 为正 ADC 输入。寄存器 REF0CN 中的 TEMPE 位启用 / 禁用温度传感器（见 SFR 定义 10.1）。当禁用时，温度传感器默认为高电阻状态，此时传感器执行的任何 ADC 测量均为无效数据。有关温度传感器的斜率和偏移参数，请参阅表 7.11。

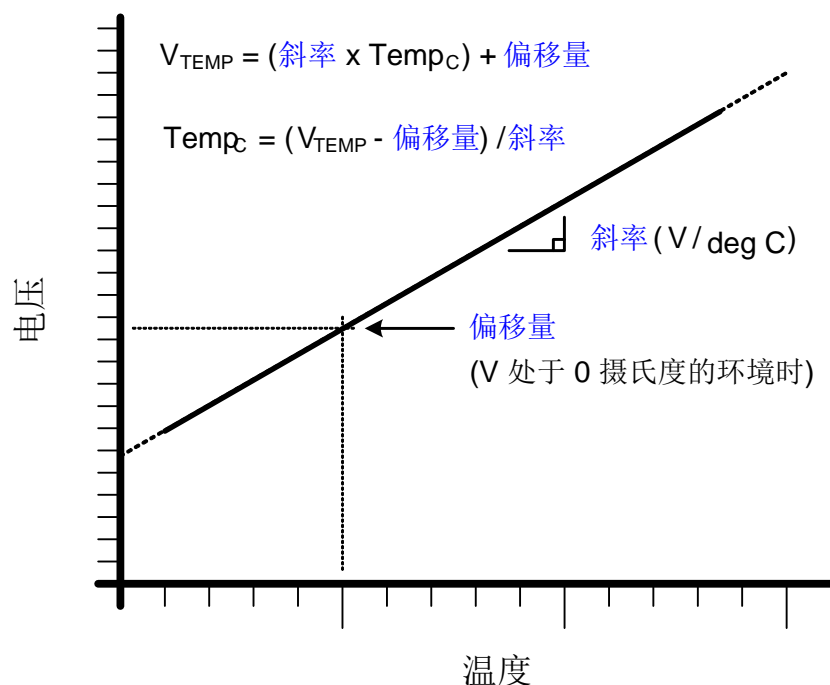


图 9.1. 温度传感器转移函数

### 9.1. 校准

未经校准的温度传感器输出的线性度非常好，适合于相对温度测量（有关线性度规格，请参见表 5.1）。对于绝对温度测量，建议对偏移和 / 或增益进行校准。典型的 1 点（偏移）校准包括下列步骤：

1. 控制 / 测量环境温度（该温度必须是已知的）。
2. 给设备上电，等待几秒钟使设备自热。
3. 将温度传感器选为 ADC 输入，执行一次 ADC 转换。
4. 计算偏移值，并将这些计算值存入非易失性内存中，以备以后使用温度传感器进行温度测量时使用。

图 5.3 列示典型温度传感器误差曲线（假设在 0°C 1 点校准）。

影响 ADC 测量的参数，尤其是电压基准值，也同样会影响温度测量。

# C8051F80x-83x

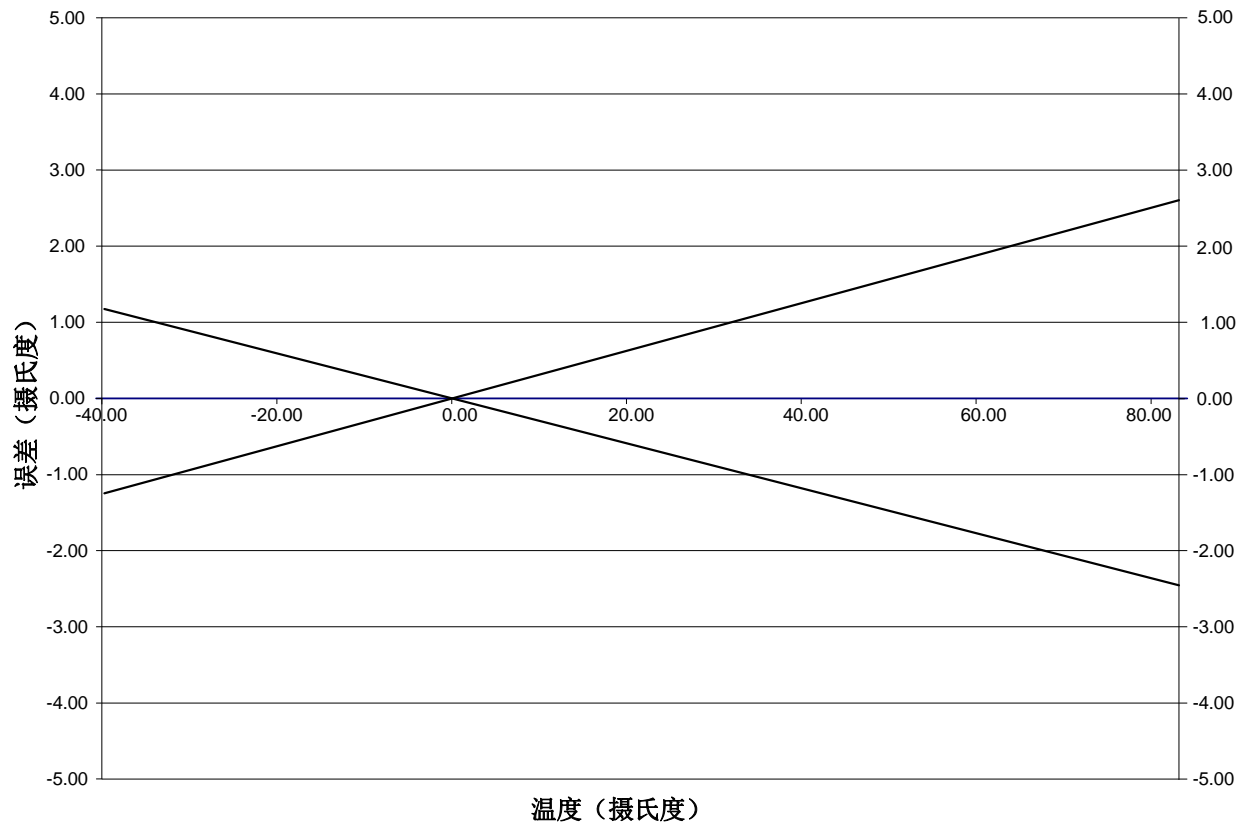


图 9.2. 在 0 °C 附近 1 点校准的温度传感器误差



## 10. 电压与接地基准选项

电压基准 MUX 可设置为使用连接到外部电压基准、片上电压基准或两种电源电压中的一种（请参见图 10.1）。接地基准 MUX 允许在接地引脚 (GND) 或专用模拟接地端口引脚 (P0.1/AGND) 之间选择 ADC0 接地基准。

电压与接地基准选项均使用 REF0CN SFR（见第 62 页）进行设置。电气规格见电气规格一章。

**有关  $V_{REF}$  和 AGND 输入的重要事项：** 端口引脚被用作外部  $V_{REF}$  和 AGND 输入。当使用外部电压基准时，P0.0/ $V_{REF}$  应被设置为模拟输入，并被数字交叉开关跳过。当使用 AGND 作为 ADC0 接地基准时，P0.1/AGND 应被设置为模拟输入，并被数字交叉开关跳过。有关端口 I/O 设置的所有详细信息，请参阅第 138 页“23. 端口输入/输出”。外部基准电压必须满足  $0 \leq V_{REF} \leq V_{DD}$ ，外部接地基准 DC 电压必须与 GND 相同。

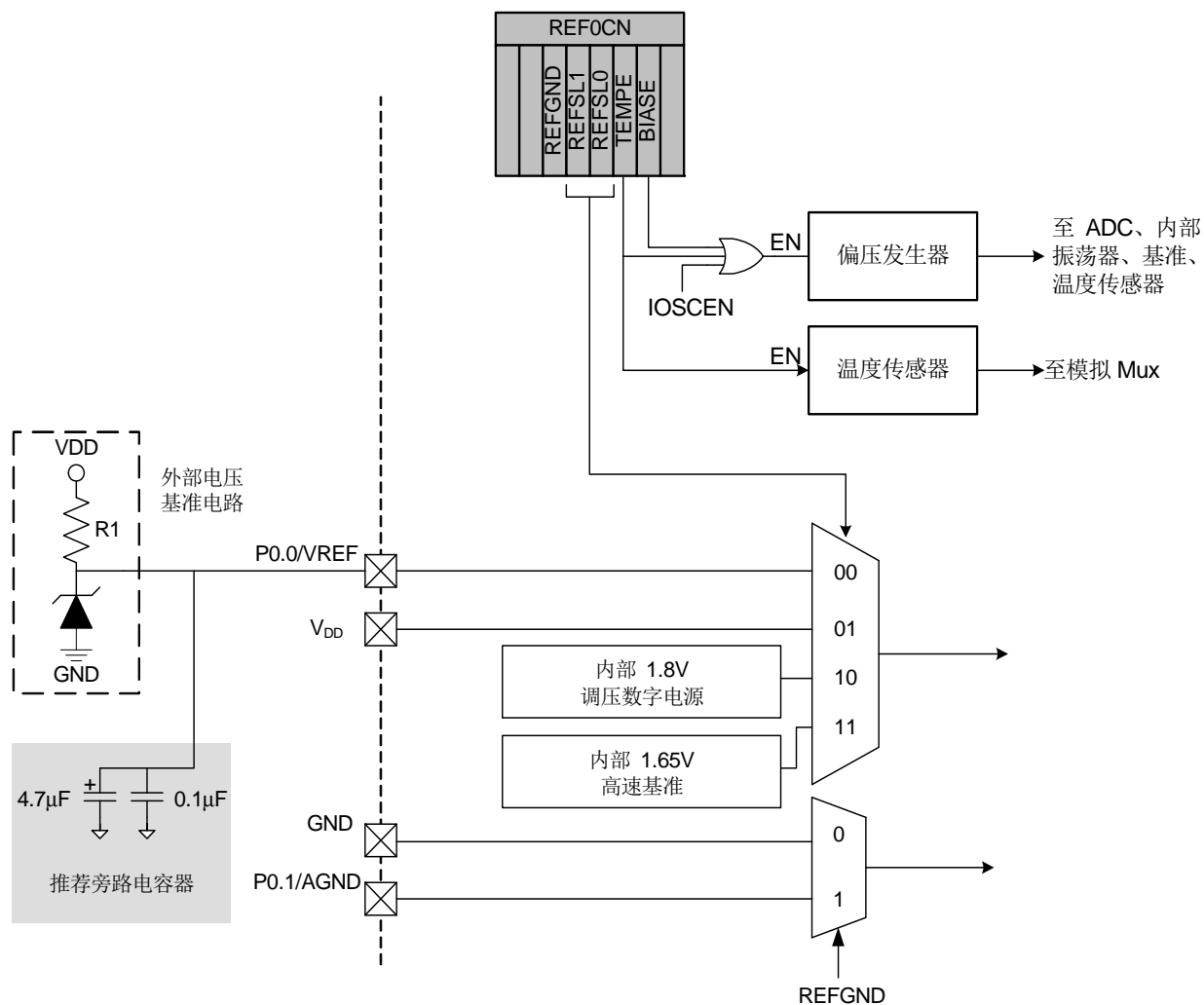


图 10.1. 电压基准功能结构图

## 10.1. 外部电压基准

要使用外部电压基准，REFSL[1:0] 应设为 00。应根据外部电压基准制造商的推荐添加旁路电容器。

## 10.2. 内部电压基准选项

片上含有 1.65 V 高速基准。已通过将 REFSL[1:0] 设为 11 来选择高速内部基准。选择后，ADC0 将视需要自动启用高速内部基准。

对于采用稳定电源电压的应用，如将电源作为电压基准，ADC0 可以降低电源噪声抑制为代价，来扩大动态范围。要使用 1.8 到 3.6 V 的电源电压 ( $V_{DD}$ ) 或使用 1.8 V 的调压数字电源电压作为基准源，REFSL[1:0] 应分别设为 01 或 10。

## 10.3. 模拟接地基准

为防止转换数字逻辑产生的接地噪声影响敏感模拟测量，有单独的模拟接地基准选项可供选择。允许时，ADC0 基底基准取自 P0.1/AGND 引脚。P0.1/AGND 引脚应参考 ADC0 采集的任何外部传感器样本。单独模拟接地基准选线已通过将 REFGND 设为 1 启用。请注意，使用该选项时，P0.1/AGND 必须连接至与 GND 相同的电压。

## 10.4. 温度传感器启用

寄存器 REF0CN 中的 TEMPE 位启用温度传感器。当禁用时，温度传感器默认为高电阻状态，此时传感器执行的任何 ADC0 测量均为无效数据。

## SFR 定义 10.1. REF0CN: 电压基准控制寄存器

位	7	6	5	4	3	2	1	0
名称			REFGND	REFSL		TEMPE	BIASE	
类型	R	R	R/W	R/W	R/W	R/W	R/W	R
复位	0	0	0	1	0	0	0	0

SFR 地址 = 0xD1

位	名称	功能
7:6	未使用	读 = 00b ; 写 = 无需在意。
5	REFGND	<b>模拟接地基准。</b> 选择 ADC0 接地基准。 0: ADC0 接地基准为 GND 针脚。 1: ADC0 接地基准为 P0.1/AGND 针脚。
4:3	REFSL	<b>电压基准选择。</b> 选择 ADC0 电压基准。 00: ADC0 电压基准为 P0.0/VREF 针脚。 01: ADC0 电压基准为 VDD 针脚。 10: ADC0 电压基准为内部 1.8 V 数字电源电压。 11: ADC0 电压基准为内部 1.65 V 高速电压基准。
2	TEMPE	<b>温度传感器启用。</b> 启用 / 禁用内部温度传感器。 0: 温度传感器禁用。 1: 温度传感器启用。
1	BIASE	<b>内部模拟偏压发生器启用位。</b> 0: 内部偏压发生器关闭。 1: 内部偏压发生器打开。
0	未使用	读 = 0b ; 写 = 无需在意。

# C8051F80x-83x

---

---

## 11. 稳压器 (REG0)

C8051F80x-83x 设备配有一个内部稳压器 (REG0)，以将内部内核电源从 1.8 到 3.6 V 的  $V_{DD}$  电源调整为 1.8 V。稳压器内置省电模式，以帮助在低耗应用中减少电流消耗。该模式可通过 REG0CN 寄存器（见 SFR 定义 11.1）访问。片上稳压器的电气特性请参见第 41 页表 7.5。

在默认状况下，当设备进入停止模式时，内部稳压器仍然运行。这使任何已启用的复位源为设备进行复位，并使设备退出停止模式。要额外省电，可在设备进入停止模式时，使用 STOPCF 位关闭稳压器和设备的内部电网。当 STOPCF 设为 1 时， $\overline{RST}$  针脚或设备的一个完整供电周期是使设备复位的唯一方法。

# C8051F80x-83x

## SFR 定义 11.1. REG0CN: 稳压器控制寄存器

位	7	6	5	4	3	2	1	0
名称	STOPCF							
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xC9

位	名称	功能
7	STOPCF	<b>停止模式设置。</b> 当设备进入停止模式时，此位设置稳压器的运行方式。 0: 在停止模式下，稳压器仍然处于活动状态。任何启用的复位源都将使设备复位。 1: 在停止模式下，稳压器关闭。只有 $\overline{\text{RST}}$ 引脚或供电周期能使设备复位。
6:0	备用	必须写入 0000000b。

## 12. 比较器 0

C8051F80x-83x 设备含有一个片上可编程电压比较器（比较器 0），如图 12.1 所示。

该比较器提供可编程的响应时间和回差电压、一个模拟输入多路复用器和两个输出（可在端口引脚选配）：一个同步“锁存”输出 (CP0) 或一个异步“直接”输出 (CP0A)。即使在系统时钟未激活时，异步 CP0A 信号仍然可用。这使比较器可以在设备处于停机模式时运行并产生输出。当被分配端口引脚时，比较器输出可被设置为漏极开路或推拉式（请参见 ?147?“23.4. 端口 I/O 初始化”）。比较器 0 也可以被用作复位源（请参见 ?127?“21.5. 比较器 0 复位”）。

比较器 0 的输入通过比较器输入多路复用器（见第 69 页“12.1. 比较器多路复用器”）选择。

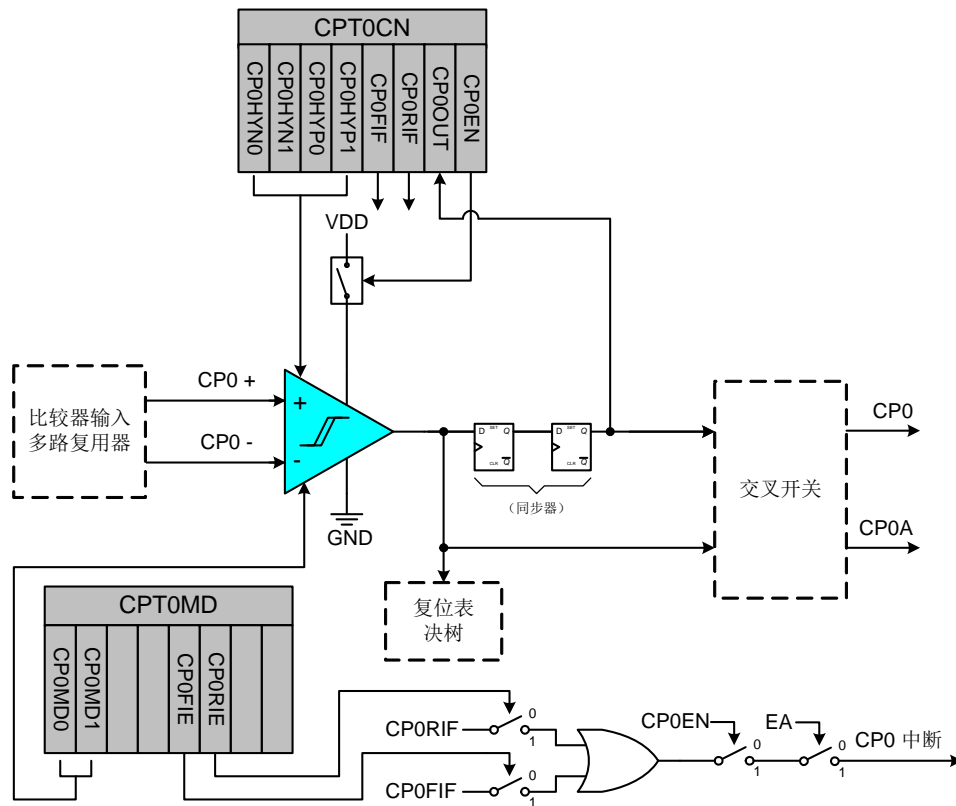


图 12.1. 比较器 0 功能结构图

该比较器输出可以由软件查询，作为中断源和 / 或被连到端口引脚。当被连到端口引脚时，比较器输出可与系统时钟同步或异步；即使在停止模式下（系统时钟停止），异步输出仍然可用。当被禁用时，比较器输出（如果已通过交叉开关分配到端口 I/O 引脚）默认为逻辑低状态，比较器电源关闭。有关通过数字交叉开关设置比较器输出的详细信息，请参见 ?143?“23.3. 优先级交叉开关编码器”。比较器输入可以承受  $-0.25\text{ V}$  到  $(V_{DD}) + 0.25\text{ V}$  的外部驱动电压而不至损坏或发生运行错误。完整的比较器电气规格，请参阅 ?39?“7. 电气特性”。

# C8051F80x-83x

比较器响应时间可通过 CPT0MD 寄存器（见 SFR 定义 12.2）用软件设置。选择较长的响应时间可以减少比较器电源电流。

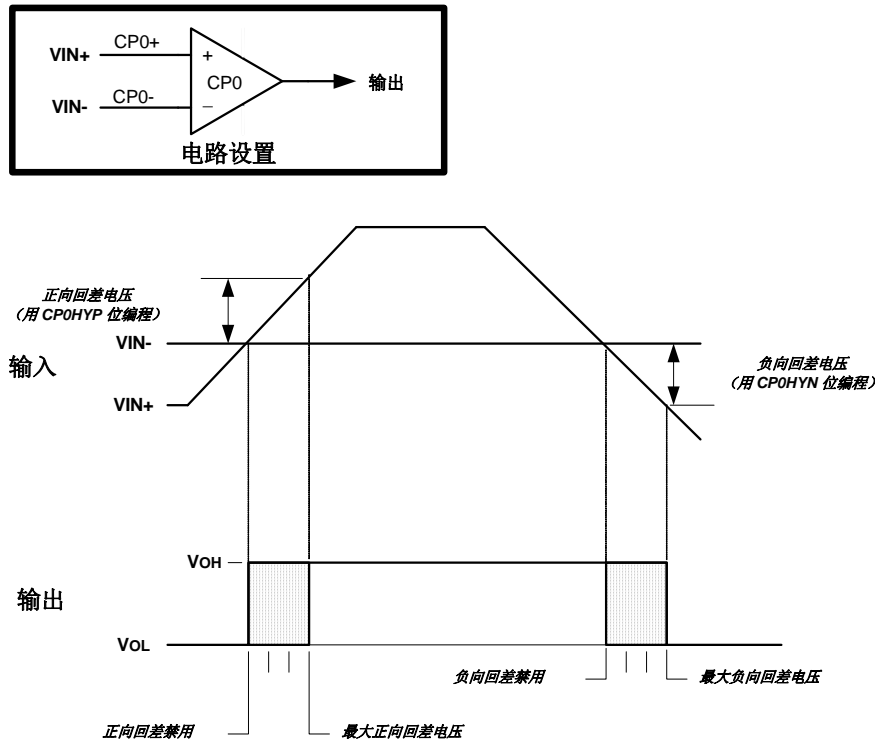


图 12.2. 比较器迟滞数据表

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（这里指输入电压）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。

比较器回差使用比较器控制寄存器 CPT0CN（见 SFR 定义 12.1）中的位 3:0 进行编程。负向回差电压值由 CP0HYN 位的设置决定。如图 12.2 所示，可以设置 20 mV、10 mV 或 5 mV 的负向回差值，或者禁用负向回差电压。类似地，通过 CP0HYP 位设置正向回差电压值在比较器输出的上升沿和下降沿都可以产生中断。（有关中断允许和优先级控制，请参见“18.1. MCU 中断源和矢量”）。比较器下降沿发生时，CP0FIF 中断标志被设为逻辑 1，比较器上升沿发生时，CP0RIF 中断标志被设为逻辑 1。一旦设置，这些位将一直保持设定状态，直到被软件清除。比较器上升沿中断屏蔽可通过将 CP0RIE 设为逻辑 1 启用。比较器 0 下降沿中断屏蔽可通过将 CP0FIE 设为逻辑 1 启用。

比较器输出状态可以随时通过读取 CP0OUT 位获得。通过将 CP0EN 位设为逻辑 1 可以启用比较器，通过将该位清 0 可禁用比较器。

请注意，当比较器开始接通电源或对回差电压或响应时间控制位做出改动时，可发现错误的上升沿和下降沿。因此，建议在比较器启用或其模式位改变后不久，将上升沿和下降沿标志明确清 0。



## SFR 定义 12.1. CPT0CN: 比较器 0 控制寄存器

位	7	6	5	4	3	2	1	0
名称	CP0EN	CP0OUT	CP0RIF	CP0FIF	CP0HYP[1:0]		CP0HYN[1:0]	
类型	R/W	R	R/W	R/W	R/W		R/W	
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x9B

位	名称	功能
7	CP0EN	比较器 0 启用位。 0: 比较器 0 禁用。 1: 比较器 0 启用。
6	CP0OUT	比较器 0 输出状态标志。 0: 电压值 CP0+ < CP0-。 1: 电压值 CP0+ > CP0-。
5	CP0RIF	比较器 0 上升沿标志。必须被软件清除。 0: 自该标志位被清除后, 并无出现比较器 0 上升沿。 1: 出现过比较器 0 上升沿。
4	CP0FIF	比较器 0 下降沿标志。必须被软件清除。 0: 自该标志位被清除后, 没有出现过比较器 0 下降沿。 1: 出现过比较器 0 下降沿。
3:2	CP0HYP[1:0]	比较器 0 正向回差电压控制位。 00: 正向回差电压禁用。 01: 正向回差电压 = 5 mV。 10: 正向回差电压 = 10 mV。 11: 正向回差电压 = 20 mV。
1:0	CP0HYN[1:0]	比较器 0 负向回差电压控制位。 00: 负向回差电压禁用。 01: 负向回差电压 = 5 mV。 10: 负向回差电压 = 10 mV。 11: 负向回差电压 = 20 mV。

# C8051F80x-83x

## SFR 定义 12.2. CPT0MD: 比较器 0 模式选择

位	7	6	5	4	3	2	1	0
名称			CP0RIE	CP0FIE			CP0MD[1:0]	
类型	R	R	R/W	R/W	R	R	R/W	
复位	0	0	0	0	0	0	1	0

SFR 地址 = 0x9D

位	名称	功能
7:6	未使用	读 = 00b, 写 = 无需在意。
5	CP0RIE	<b>比较器 0 上升沿中断允许寄存器。</b> 0: 比较器 0 上升沿中断禁用。 1: 比较器 0 上升沿中断启用。
4	CP0FIE	<b>比较器 0 下降沿中断允许寄存器。</b> 0: 比较器 0 下降沿中断禁用。 1: 比较器 0 下降沿中断启用。
3:2	未使用	读 = 00b, 写 = 无需在意。
1:0	CP0MD[1:0]	<b>比较器 0 模式选择。</b> 这些位影响比较器 0 的响应时间和电耗。 00: 模式 0 (最快响应时间, 最高电耗) 01: 模式 1 10: 模式 2 11: 模式 3 (最慢响应时间, 最低电耗)

## 12.1. 比较器多路复用器

C8051F80x-83x 设备包含一个模拟输入多路复用器，将端口 I/O 引脚连接至比较器输入。比较器 0 输入在 CPT0MX 寄存器（见 SFR 定义 12.3）中选择。CMX0P3– CMX0P0 位选择比较器 0 正输入；CMX0N3– CMX0N0 位选择比较器 0 负输入。有关比较器输入的重要事项：应将选作比较器输入的端口引脚设置为其相关端口设置寄存器中的模拟输入，且设置为被交叉开关跳过（有关端口设置的详细信息，请参见“23.6. 访问和配置端口 I/O 的特殊功能寄存器”）。

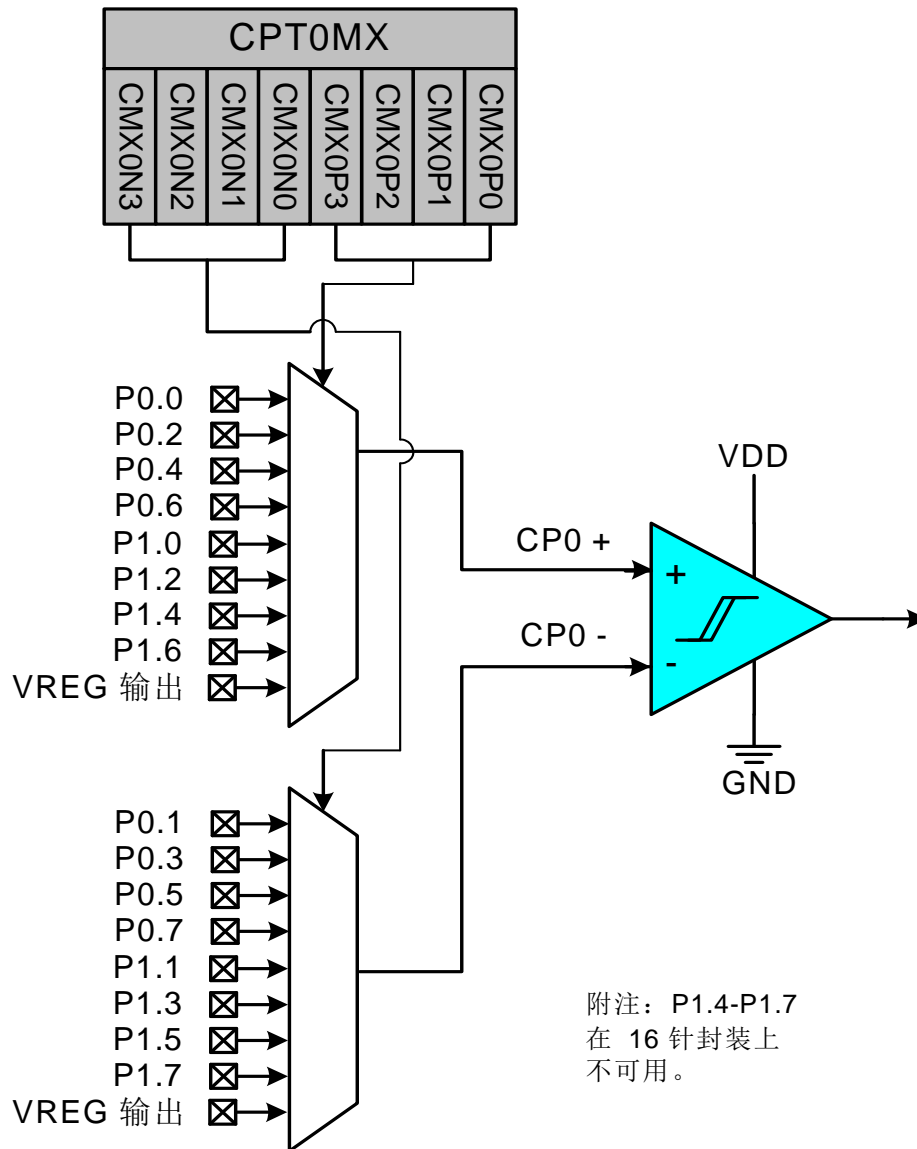


图 12.3. 比较器输入多路复用器结构图

# C8051F80x-83x

## SFR 定义 12.3. CPT0MX: 比较器 0 MUX 选择寄存器

位	7	6	5	4	3	2	1	0
名称	CMX0N[3:0]				CMX0P[3:0]			
类型	R/W				R/W			
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0x9F

位	名称	功能		
7:4	CMX0N[3:0]	比较器 0 负输入 MUX 选择寄存器。		
			<b>20 针和 24 针设备</b>	<b>16 针设备</b>
		0000	P0.1	P0.1
		0001	P0.3	P0.3
		0010	P0.5	P0.5
		0011	P0.7	P0.7
		0100	P1.1	P1.1
		0101	P1.3	P1.3
		0110	P1.5	备用。
		0111	P1.7	备用。
		1000	VREG 输出。	VREG 输出。
		1001-1111	未选择任何输入。	未选择任何输入。
		3:0	CMX0P[3:0]	比较器 0 正输入 MUX 选择寄存器。
	<b>20 针和 24 针设备</b>			<b>16 针设备</b>
0000	P0.0			P0.0
0001	P0.2			P0.2
0010	P0.4			P0.4
0011	P0.6			P0.6
0100	P1.0			P1.0
0101	P1.2			P1.2
0110	P1.4			备用。
0111	P1.6			备用。
1000	VREG 输出。			VREG 输出。
1001-1111	未选择任何输入。			未选择任何输入。

## 13. 电容传感 (CS0)

C8051F800/1/3/4/6/7/9, C8051F810/2/3/5/6/8/9, C8051F821/2/4/5/7/8, C8051F830/1/3/4 上的电容传感子系统使用数字电容电路确定端口针脚上的电容。该模块可以使用模块的模拟多路复用器测量不同端口针脚的电容。模拟多路复用器支持多达 16 条通道。有关特定部件编号的通道可用性, 请参见第 81 页 SFR 定义 13.9. “CS0MX: 电容传感 Mux 通道选择”。该模块仅在 CS0EN 位 (CS0CN) 被设置为 1 时启用。在其它情况下, 该模块处于低功耗关闭状态。该模块可被设置为使用自动扫描对一个端口针脚或一组端口针脚进行测量。可配置一个累加器, 累计一个输入通道上的多次转换。当 CS0 完成一次转换或测量值超过 CS0THH:L 中界定的阈值时, 会发生中断。

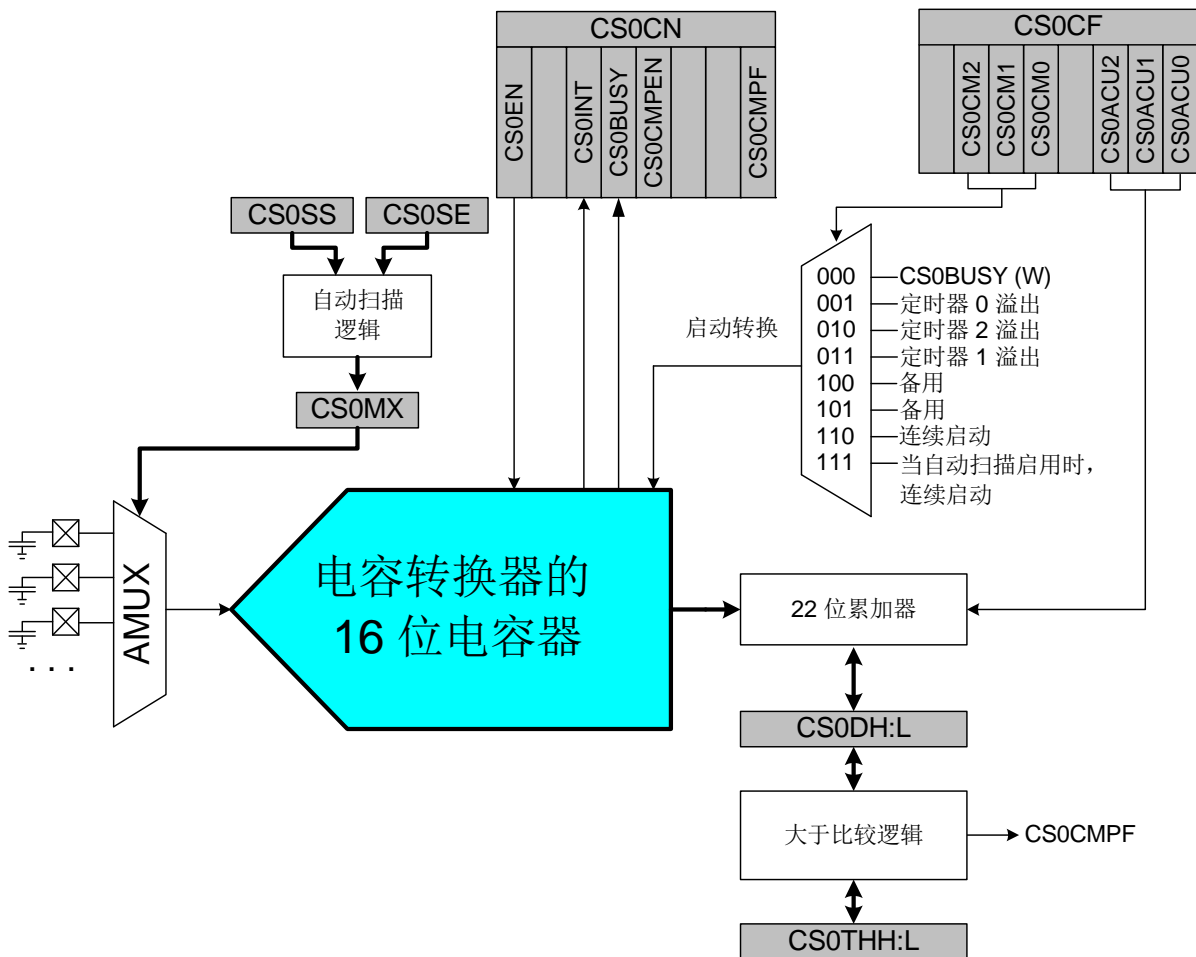


图 13.1. CS0 结构图

## 13.1. 端口引脚配置为电容传感输入

为用 CS0 测量端口引脚，该端口引脚必须配置为模拟输入（请参见“23. 端口输入 / 输出”）。将输入多路复用器配置为未配置为模拟输入的端口引脚将引致电容传感比较器输出有误的测量结果。

## 13.2. 电容传感启动转换源

电容传感转换根据 CS0 启动转换位的设计模式 (CS0CF6:4)，可用下列七种方式中任一种启动。转换触发源有：

1. 写 1 到寄存器 CS0CN 的 CS0BUSY 位
2. 定时器 0 溢出
3. 定时器 2 溢出
4. 定时器 1 溢出
5. 连续转换
6. 当自动扫描启用时，连续转换

可通过下列两种方法中任一种将转换配置为连续启动。CS0 可配置为单一通道连续转换，或配置为自动扫描启用时连续转换。配置为连续转换时，CS0CF 中的 CS0BUSY 位经设置后将启动转换。

如通过设置 ECSCPT 位 (EIE2.0) 允许 CS0 转换结束中断，则将产生一个中断。

**附注：** CS0 转换结束中断的产生取决于 CS0 累加器的设置。如 CS0 配置为累加一个输入通道的多次转换，则 CS0 转换结束中断仅在最后一次转换结束后产生。

## 13.3. 自动扫描

CS0 可通过配置和启用自动扫描，配置为自动扫描相邻 CS0 输入通道的顺序。使用 CS0 比较器中断被允许的自动扫描无需任何额外专用的 MCU 资源，即可使系统探测到测量电容的变更。

通过将 CS0 启动转换位 (CS0CF6:4) 设置为 111b 启用自动扫描。启用自动扫描后，启动和结束通道应分别在 CS0SS 和 CS0SE 中设置为适当的值。启用自动扫描时写 CS0SS 将引致写入 CS0SS 的值被写入 CS0MX。启用之后，向 CS0BUSY 写 1 将启动自动扫描转换。当自动扫描完成 CS0 累加器位 (CS0CF1:0) 中界定的转换次数（请参见“13.5. CS0 转换累加器”），自动扫描将 CS0MX 配置为下一个最高的配置为模拟输入的端口引脚，并在该通道启动转换。此扫描顺序将保持，直至 CS0MX 达到 CS0SE 中界定的结束输入通道值。此通道进行一次或多次转换之后，自动扫描将 CS0MX 配置为原来的启动输入通道。请参见第 73 页的图“13.2 自动扫描范例”，查看配置为使用自动扫描的系统范例。

**附注：** 不论 CS0MX 通道的端口引脚是否已配置为模拟输入，自动扫描都将在该通道上进行一次转换。如在设备进入暂停模式时启用自动扫描，则自动扫描将保持启用和运行状态。通过纳入自动扫描输入顺序的任何配置的电容器传感输入中的 CS0 下限比较器事件，此功能将设备从暂停模式中唤醒。

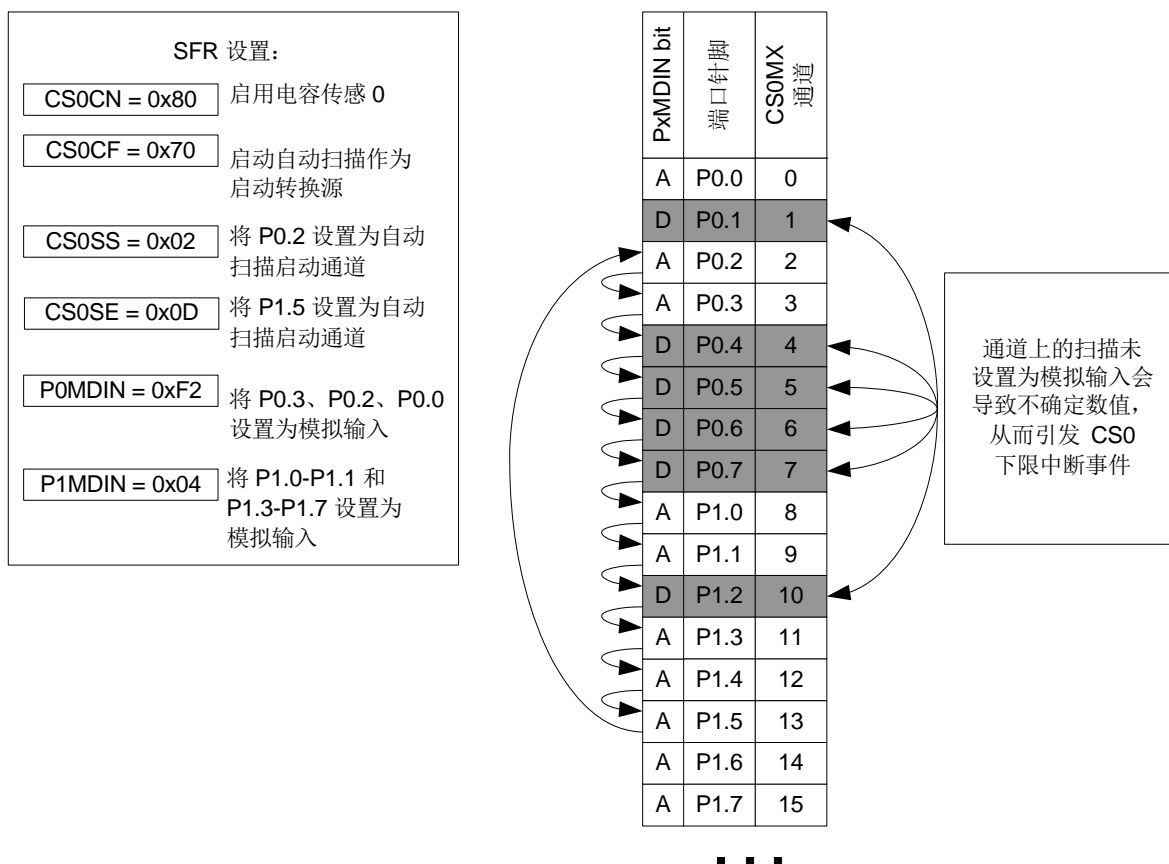


图 13.2. 自动扫描范例

### 13.4. CS0 比较器

CS0 比较器比较最新电容传感转换结果和储存在 CS0THH:CS0THL 中的数值。如果结果小于或等于储存数值，则 CS0CMPF 位 (CS0CN:0) 被设置为 0。如果结果大于储存数值，则 CS0CMPF 被设置为 1。

如果已配置 CS0 转换累加器累加多次转换，则直到最后一次转换被累加才会进行比较。

比较器将 CS0CMPF 设置为 1 时，若通过设置 ECSGRT 位 (EIE2.1) 允许 CS0 下限比较器中断，则会发生中断。

如果比较器设置 CS0CMPF 位时，自动扫描正在运行，则直到固件将 CS0BUSY 设置为 1 才会开始进一步自动扫描启动转换。

CS0 下限比较器事件可将设备从暂停模式唤醒。这一功能在设置用于连续采样一个或多个电容传感通道的系统中十分有用。设备将一直处于低耗暂停状态，直到一个扫描通道的捕获数值导致发生 CS0 下限比较器事件。为了利用下限事件将设备从暂停模式唤醒，必需允许 CS0 比较器发生中断。

**附注:** 因 CS0 下限比较器事件而从暂停模式唤醒后，CS0CN 寄存器至少在两个系统时钟周期后方可被访问。有关不同 CS0 比较器、自动扫描和自动累加器设置的行为概要，请参见表 13.1。

# C8051F80x-83x

## 13.5. CS0 转换累加器

可配置 CS0，累加一个输入通道上的多次转换。将被累加的样品数目使用 CS0ACU2:0 位 (CS0CF2:0) 进行设置。累加器可累加 1、4、8、16、32 或 64 个样品。在界定的样品数目被累加后，有关结果通过将 22 位累加器除以 1、4、8、16、32 或 64（取决于 CS0ACU[2:0] 设置）被转换成一个 16 位数值，并被复制到 CS0DH:CS0DL SFR。

表 13.1. 自动扫描和累加的操作

自动扫描启动	累加器启用	CS0 转换完成 中断行为	CS0 下限中断行为	CS0MX 行为
N	N	1 次转换完成后 发生 CS0INT 中断	如果 CS0DH:CS0DL 中的数值 大于 CS0THH:CS0THL，则 1 次转换完成后发生中断	CS0MX 无变化。
N	Y	M 次转换完成后 发生 CS0INT 中断	如果 16 位累加器中的数值大 于 CS0THH:CS0THL，则 M 次转换完成后发生中断	CS0MX 无变化。
Y	N	1 次转换完成后 发生 CS0INT 中断	如果 CS0DH:CS0DL 中的数值 大于 CS0THH:CS0THL， 则转换完成后发生中断， 自动扫描停止	如果下限比较器发现转换值大于 CS0THH:CS0THL，则 CMUX0 保持 不变；否则，CMUX0 更新至下一个 通道 (CS0MX + 1) 并在通过 CS0SE 后 绕回 CS0SS
Y	Y	M 次转换完成后 发生 CS0INT 中断	如果 16 位累加器中的数值 大于 CS0THH:CS0THL， 则 M 次转换完成后发生中断； 自动扫描停止	如果下限比较器发现转换值大于 CS0THH:CS0THL，则 CS0MX 保持 不变；否则，CS0MX 更新至下一个 通道 (CS0MX + 1) 并在通过 CS0SE 后 绕回 CS0SS
<b>M = 累加器设置 (1x、4x、8x、16x、32x、64x)</b>				



## SFR 定义 13.1. CS0CN: 电容传感控制寄存器

位	7	6	5	4	3	2	1	0
名称	CS0EN		CS0INT	CS0BUSY	CS0CMPEN			CS0CMPF
类型	R/W	R	R/W	R/W	R/W	R	R	R
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xB0 ; 可位寻址

位	名称	功能
7	CS0EN	<b>CS0 启用。</b> 0: CS0 禁用并且处于低耗模式。 1: CS0 启用并且准备就绪, 可进行转换。
6	未使用	读 = 0b ; 写 = 无需在意
5	CS0INT	<b>CS0 中断标志。</b> 0: 从最后一次 CS0INT 被清零后, CS0 尚未完成一次数据转换。 1: CS0 已完成一次数据转换。 此位不能由硬件自动清除。
4	CS0BUSY	<b>CS0 忙。</b> 读: 0: CS0 转换已完成或现在未进行转换。 1: CS0 转换正在进行。 写: 0: 无作用。 1: 如果 CS0CM[2:0] = 000b、110b 或 111b, 则启动 CS0 转换。
3	CS0CMPEN	<b>CS0 数字比较器启动位。</b> 启动数字比较器, 数字比较器将累加的 CS0 转换输出与储存在 CS0THH:CS0THL 中的数值进行比较。 0: CS0 数字比较器禁用。 1: CS0 数字比较器启用。
2:1	未使用	读 = 00b ; 写 = 无需在意
0	CS0CMPF	<b>CS0 数字比较器中断标志。</b> 0: 自最后一次 CS0CMPF 被清零后, CS0 结果小于 CS0THH 和 CS0THL 设置的数值。 1: 自最后一次 CS0CMPF 被清零后, CS0 结果大于 CS0THH 和 CS0THL 设置的数值。
附注: 因 CS0 下限比较器事件而从暂停模式唤醒后, CS0CN 寄存器至少在两个系统时钟周期后方可被访问。		

# C8051F80x-83x

## SFR 定义 13.2. CS0CF: 电容传感配置寄存器

位	7	6	5	4	3	2	1	0
名称	CS0CM[2:0]				CS0ACU[2:0]			
类型	R	R/W	R/W	R/W	R	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x9E

位	名称	功能
7	未使用	读 = 0b; 写 = 无需在意
6:4	CS0CM[2:0]	<b>CS0 转换启动模式选择。</b> 000: 每次向 CS0BUSY 写 1 时, 转换启动。 001: 定时器 0 溢出时, 转换启动。 010: 定时器 2 溢出时, 转换启动。 011: 定时器 1 溢出时, 转换启动。 100: 备用。 101: 备用。 110: 向 CS0BUSY 写 1 后, 转换连续启动。 111: 自动扫描启用, 向 CS0BUSY 写 1 后, 转换连续启动。
3	未使用	读 = 0b; 写 = 无需在意
2:0	CS0ACU[2:0]	<b>CS0 累加器模式选择。</b> 000: 累加 1 样品。 001: 累加 4 样品。 010: 累加 8 样品。 011: 累加 16 样品。 100: 累加 32 样品。 101: 累加 64 样品。 11x: 备用。

**SFR 定义 13.3. CS0DH: 电容传感数据高字节寄存器**

位	7	6	5	4	3	2	1	0
名称	CS0DH[7:0]							
类型	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xAC

位	名称	功能
7:0	CS0DH	<b>CS0 数据高字节寄存器。</b> 储存最后完成的 16 位电容传感转换的高字节。

**SFR 定义 13.4. CS0DL: 电容传感数据低字节寄存器**

位	7	6	5	4	3	2	1	0
名称	CS0DL[7:0]							
类型	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xAB

位	名称	功能
7:0	CS0DL	<b>CS0 数据低字节寄存器。</b> 储存最后完成的 16 位电容传感转换的低字节。

# C8051F80x-83x

## SFR 定义 13.5. CS0SS: 电容传感自动扫描启动通道寄存器 (Capacitive Sense Auto-Scan Start Channel)

位	7	6	5	4	3	2	1	0
名称				CS0SS[4:0]				
类型	R	R	R	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xB9

位	名称	功能
7:5	未使用	读 = 000b ; 写 = 无需在意
4:0	CS0SS[4:0]	<b>自动扫描启动通道寄存器。</b> 当自动扫描启动和激活时，设置第一条 CS0 通道，以供 mux 选择用于电容传感转换。  当自动扫描启动时，向 CS0SS 写入值也会更新 CS0MX。

## SFR 定义 13.6. CS0SE: 电容传感自动扫描结束通道寄存器 (Capacitive Sense Auto-Scan End Channel)

Bit	7	6	5	4	3	2	1	0
Name				CS0SE[4:0]				
Type	R	R	R	R/W	R/W	R/W	R/W	R/W
Reset	0	0	0	0	0	0	0	0

SFR 地址 = 0xBA

位	名称	功能
7:5	未使用	读 = 000b ; 写 = 无需在意
4:0	CS0SE[4:0]	<b>自动扫描结束通道寄存器。</b> 当自动扫描启动和激活时，设置最后一条 CS0 通道，以供 mux 选择用于电容传感转换。

**SFR 定义 13.7. CS0THH: 电容传感比较器阈值高字节寄存器 (Capacitive Sense Comparator Threshold High Byte)**

位	7	6	5	4	3	2	1	0
名称	CS0THH[7:0]							
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x97

位	名称	功能
7:0	CS0THH[7:0]	<b>CS0 比较器阈值高字节寄存器。</b> 将 16 位数值高字节与电容传感转换结果进行比较。

**SFR 定义 13.8. CS0THL: 电容传感比较器阈值低字节寄存器 (Capacitive Sense Comparator Threshold Low Byte)**

位	7	6	5	4	3	2	1	0
名称	CS0THL[7:0]							
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x96

位	名称	功能
7:0	CS0THL[7:0]	<b>CS0 比较器阈值低字节寄存器。</b> 将 16 位数值低字节与电容传感转换结果进行比较。

# C8051F80x-83x

---

## 13.6. 电容传感多路复用器

输入多路复用器可以通过两种方法控制。CS0MX 寄存器可以通过固件写入，或该寄存器可以使用模块自动扫描功能自动设置（请参见“13.3. 自动扫描”）。

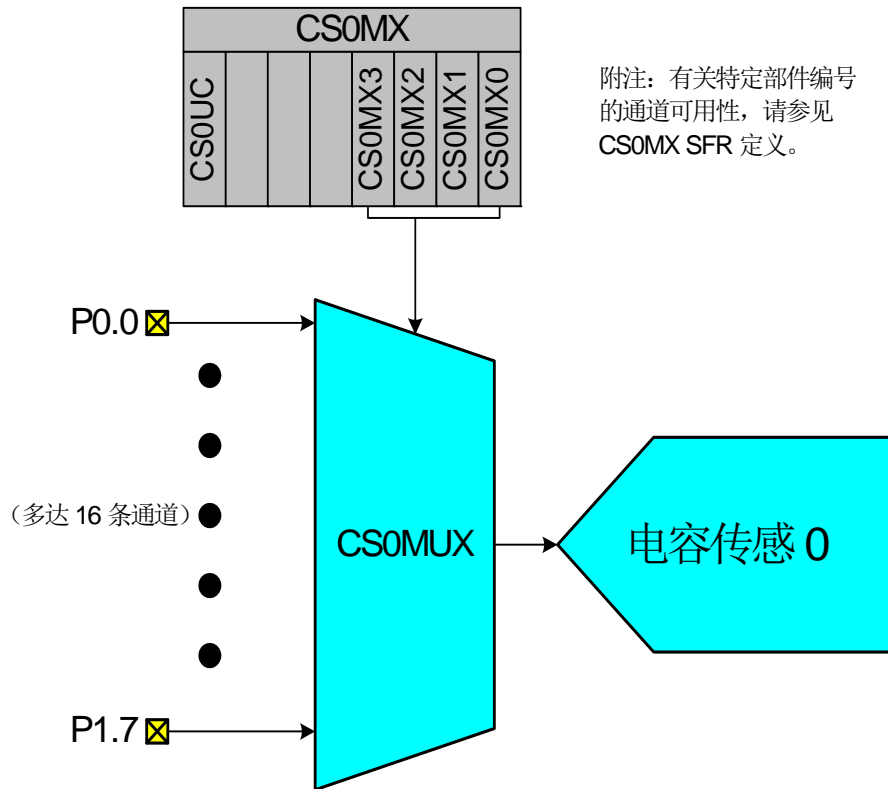


图 13.3. CS0 多路复用器结构图

# C8051F80x-83x

## SFR 定义 13.9. CS0MX: 电容传感 Mux 通道选择

位	7	6	5	4	3	2	1	0
名称	CS0UC				CS0MX[3:0]			
类型	R/W	R	R	R	R/W	R/W	R/W	R/W
复位	1	0	0	1	1	1	1	1

SFR 地址 = 0x9C

位	名称	功能			
7	CS0UC	<b>CS0 未连接。</b> 将 CS0 从所有端口引脚断开，无论所选通道为何。 0: CS0 已连接至端口引脚 1: CS0 已从端口引脚断开连接			
6:4	备用	读 = 000b ; 写 = 000b			
3:0	CS0MX[3:0]	<b>CS0 Mux 通道选择寄存器。</b> 选择 16 条通道中的一条进行电容传感转换。			
		数值	C8051F800/6、 C8051F812/8	C8051F803/9、 C8051F815、 C8051F821/4/7、 C8051F830/3	C8051F801/4/7、 C8051F810/3/6/9、 C8051F822/5/8、 C8051F831/4
		0000	P0.0	P0.0	P0.0
		0001	P0.1	P0.1	P0.1
		0010	P0.2	P0.2	P0.2
		0011	P0.3	P0.3	P0.3
		0100	P0.4	P0.4	P0.4
		0101	P0.5	P0.5	P0.5
		0110	P0.6	P0.6	P0.6
		0111	P0.7	P0.7	P0.7
		1000	P1.0	P1.0	备用。
		1001	P1.1	P1.1	备用。
		1010	P1.2	P1.2	备用。
		1011	P1.3	P1.3	备用。
		1100	P1.4	备用。	备用。
		1101	P1.5	备用。	备用。
		1110	P1.6	备用。	备用。
		1111	P1.7	备用。	备用。
附注: 在未于上表列出的所有设备上, CS0MX 均备用。					



## 14. CIP-51 微控制器

MCU 系统控制器内核是 CIP-51 微控制器。CIP-51 与 MCS-51™ 指令集完全兼容；可以使用标准 803x/805x 汇编器和编译器进行软件开发。该 MCU 系列具有标准 8051 配备的所有外围设备超集合。该 CIP-51 还包含片上调试硬件（请参见 ?30? 说明）和与 MCU 直接接口的模拟和数字子系统，在一个集成电路内提供完整的数据采集或控制系统解决方案。

CIP-51 微控制器内核除具有标准 8051 组织结构和外围设备外，另有增加的定制外围设备和功能，大大增强了它的处理能力（见图 14.1 结构图）。CIP-51 具有以下特点：

- 与 MCS-51 指令集完全兼容
- 配备 25 MHz 时钟时，最大吞吐率达 25 MIPS
- 0 到 25 MHz 的时钟频率
- 扩展的中断处理程序
- 复位输入
- 电源管理模式
- 片上调试逻辑
- 程序和数据内存安全

### 性能

CIP-51 采用流水线结构，与标准 8051 结构相比指令吞吐率有很大提高。在一个标准 8051 中，除 MUL 和 DIV 以外，所有指令都需要 12 或 24 个系统时钟周期执行，并且通常最大系统时钟频率为 12 MHz。而对于 CIP-51 内核，70% 的指令都是在一个或两个系统时钟周期执行，没有执行时间超过八个系统时钟周期的指令。

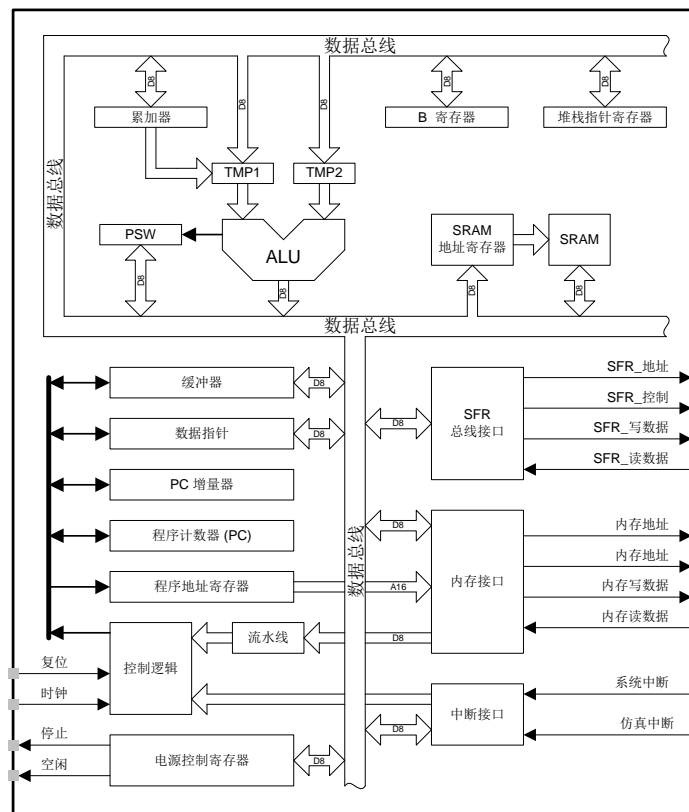


图 14.1. CIP-51 结构图

# C8051F80x-83x

CIP-51 处于最大系统时钟频率 25 MHz 时，最大吞吐率达 25 MIPS。CIP-51 共有 109 条指令。下表列示指令总条数与执行时所需系统时间周期数的关系。

执行周期数	1	2	2/3	3	3/4	4	4/5	5	8
指令条数	26	50	5	14	6	3	2	2	1

## 14.1. 指令集

CIP-51 系统控制器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 开发工具开发 CIP-51 软件。所有 CIP-51 指令在二进制码和功能上（包括操作码、寻址模式和对 PSW 标志的影响）均与同类 MCS-51™ 产品相等。但是，指令定时与标准 8051 不同。

### 14.1.1. 指令和 CPU 定时

在很多 8051 产品中，机器周期与时钟周期不同，机器周期的长度介于 2 到 12 个时钟周期。但 CIP-51 执行只基于时钟周期定时。所有指令定时都以时钟周期计算。

由于 CIP-51 采用流水线结构，大多数指令执行所需的时钟周期数与指令的字节数一致。条件转移指令在不发生转移时的执行周期数比发生转移时少一个。表 14.1 为 CIP-51 指令集一览表，其中包括每条指令的助记符、字节数和时钟周期数。

表 14.1. CIP-51 指令集一览表

助记符	说明	字节	时钟周期
<b>算术操作类指令</b>			
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 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
<b>逻辑操作类指令</b>			
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

# C8051F80x-83x

表 14.1. CIP-51 指令集一览表 (Continued)

助记符	说明	字节	时钟周期
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
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
<b>数据传输类指令</b>			
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 位常数载入 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	外部数据 (8 位地址) 传送到累加器	1	3
MOVX @Ri, A	累加器传送到外部数据 (8 位地址)	1	3
MOVX A, @DPTR	外部数据 (16 位地址) 传送到累加器	1	3
MOVX @DPTR, A	累加器传送到外部数据 (16 位地址)	1	3
PUSH direct	直接寻址字节压入堆栈	2	2
POP direct	堆栈数据弹出到直接寻址字节	2	2
XCH A, Rn	寄存器与累加器交换	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 与累加器交换低半字节	1	2
<b>布尔调处指令</b>			
CLR C	清除进位位	1	1
CLR bit	清除直接寻址位	2	2
SETB C	设置进位位	1	1
SETB bit	设置直接寻址位	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

**表 14.1. CIP-51 指令集一览表 (Continued)**

助记符	说明	字节	时钟周期
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	若进位位已设置则跳转	2	2/3
JNC rel	若进位位未设置则跳转	2	2/3
JB bit, rel	若直接寻址位已设置则跳转	3	3/4
JNB bit, rel	若直接寻址位未设置则跳转	3	3/4
JBC bit, rel	若直接寻址位已设置则跳转，并清除改位	3	3/4
<b>程序分支指令</b>			
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	累加器为零则转移	2	2/3
JNZ rel	累加器不为零则转移	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	4/5
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，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

# C8051F80x-83x

## 寄存器、操作数和寻址模式说明：

**Rn**— 当前选择的寄存器区的寄存器 R0–R7。

**@Ri**— 通过寄存器 R0 或 R1 间接寻址的数据 RAM 地址。

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

**direct**— 8 位内部数据位置地址。可以是直接访问数据 RAM 地址 (0x00–0x7F) 或一个 SFR 地址 (0x80–0xFF)。

**#data**— 8 位常数

**#data16**— 16 位常数

**bit**— 数据 RAM 或 SFR 中的直接寻址位

**addr11**—ACALL 和 AJMP 使用的 11 位目的地址。目的地址必须与下一条指令第一个字节处于同一个 2 kB 的程序内存页。

**addr16**—LCALL 和 LJMP 使用的 16 位目的地址。目的地址可以是 8 kB 程序内存空间内的任何位置。

有一个未使用操作码 (0xA5)，该操作码执行与 NOP 指令相同的功能。

所有助记符归 © Intel Corporation 1980 版权所有。

## 14.2. CIP-51 寄存器说明

下面对与 CIP-51 系统控制器操作有关的 SFR 加以说明。保留位应始终写 SFR 说明中指示的数值。将来的产品版本可能会使用这些位执行新功能，在这种情况下，各位的复位值将为指示的数值，以选择功能的默认状态。有关其它 SFR 的详细说明载于本数据表与其对应的系统功能相关的章节。

### SFR 定义 14.1. DPL: 数据指针低字节寄存器

位	7	6	5	4	3	2	1	0
名称	DPL[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x82

位	名称	功能
7:0	DPL[7:0]	数据指针低字节。 DPL 寄存器是 16 位 DPTR 的低字节寄存器。

### SFR 定义 14.2. DPH: 数据指针高字节寄存器

位	7	6	5	4	3	2	1	0
名称	DPH[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x83

位	名称	功能
7:0	DPH[7:0]	数据指针高字节。 DPH 寄存器是 16 位 DPTR 的高字节寄存器。

# C8051F80x-83x

## SFR 定义 14.3. SP: 堆栈指针寄存器

位	7	6	5	4	3	2	1	0
名称	SP[7:0]							
类型	R/W							
复位	0	0	0	0	0	1	1	1

SFR 地址 = 0x81

位	名称	功能
7:0	SP[7:0]	堆栈指针。 堆栈指针保持栈顶位置。在每次执行 PUSH 操作前，堆栈指针增加。SP 寄存器复位后的默认值为 0x07。

## SFR 定义 14.4. ACC: 累加器

位	7	6	5	4	3	2	1	0
名称	ACC[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xE0；可位寻址

位	名称	功能
7:0	ACC[7:0]	累加器。 该寄存器是算术操作累加器。



## SFR 定义 14.5. B: B 寄存器

位	7	6	5	4	3	2	1	0
名称	B[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xF0；可位寻址

位	名称	功能
7:0	B[7:0]	<b>B 寄存器。</b> 该寄存器为某些算术操作的第二累加器。

# C8051F80x-83x

## SFR 定义 14.6. PSW: 程序状态字寄存器

位	7	6	5	4	3	2	1	0
名称	CY	AC	F0	RS[1:0]		OV	F1	PARITY
类型	R/W	R/W	R/W	R/W		R/W	R/W	R
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xD0; 可位寻址

位	名称	功能
7	CY	<b>进位标志。</b> 当最后一次算术操作产生进位（加法）或借位（减法）时，该位被设置。其它算术操作将其清为逻辑 0。
6	AC	<b>辅助进位标志。</b> 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，该位被设置。其它算术操作将其清为逻辑 0。
5	F0	<b>用户标志 0。</b> 这是一个可位寻址、受软件控制的通用标志位。
4:3	RS[1:0]	<b>寄存器区选择。</b> 这些位在寄存器访问时用于选择寄存器区。 00: 寄存器区 0, 地址 0x00-0x07 01: 寄存器区 1, 地址 0x08-0x0F 10: 寄存器区 2, 地址 0x10-0x17 11: 寄存器区 3, 地址 0x18-0x1F
2	OV	<b>溢出标志。</b> 在下列情况下，该位设置为 1： <ul style="list-style-type: none"> <li>• 一个 ADD、ADDC 或 SUBB 指令导致符号改变溢出。</li> <li>• 一个 MUL 指令导致溢出（结果大于 255）。</li> <li>• 一个 DIV 指令导致出现除以 0 的情况。</li> </ul> 在所有其它情况下，OV 位被 ADD、ADDC、SUBB、MUL 和 DIV 指令清 0。
1	F1	<b>用户标志 1。</b> 这是一个可位寻址、受软件控制的通用标志位。
0	PARITY	<b>奇偶标志。</b> 若累加器中八个位的和为奇数，则该位设置为逻辑 1，为偶数时则清零。

## 15. 内存组织

CIP-51 系统控制器的内存组织类似于标准 8051 的内存组织，它分为两个独立的内存空间：程序内存和数据内存。程序和数据内存共享同一地址空间，但需经不同的指令类型存取。C8051F80x-83x 设备系列的内存组织见图 15.1。

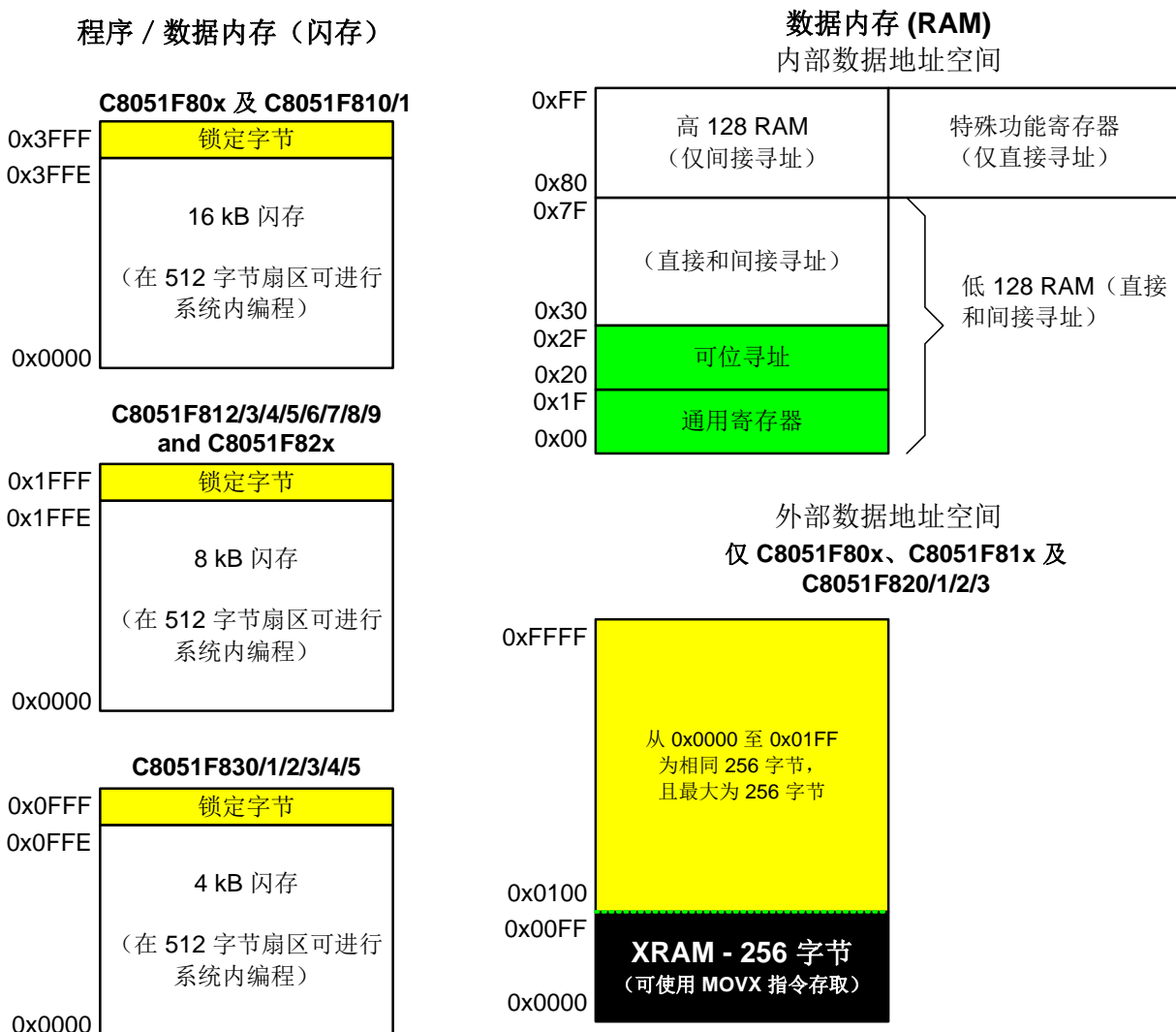


图 15.1. C8051F80x-83x 存储器映像

# C8051F80x-83x

## 15.1. 程序内存

C8051F80x-83x 设备系列包含 16 kB（C8051F80x and C8051F810/1）、8 kB（C8051F812/3/4/5/6/7/8/9 and C8051F82x），或 4 kB（C8051F830/1/2/3/4/5）可再编程闪存，可存储非易失性程序或数据。用户代码空间的最后一字节用作安全锁定字节（16 kB 设备的 0x3FFF、8 kB 设备的 0x1FFF 及 4 kB 设备的 0x0FFF）。

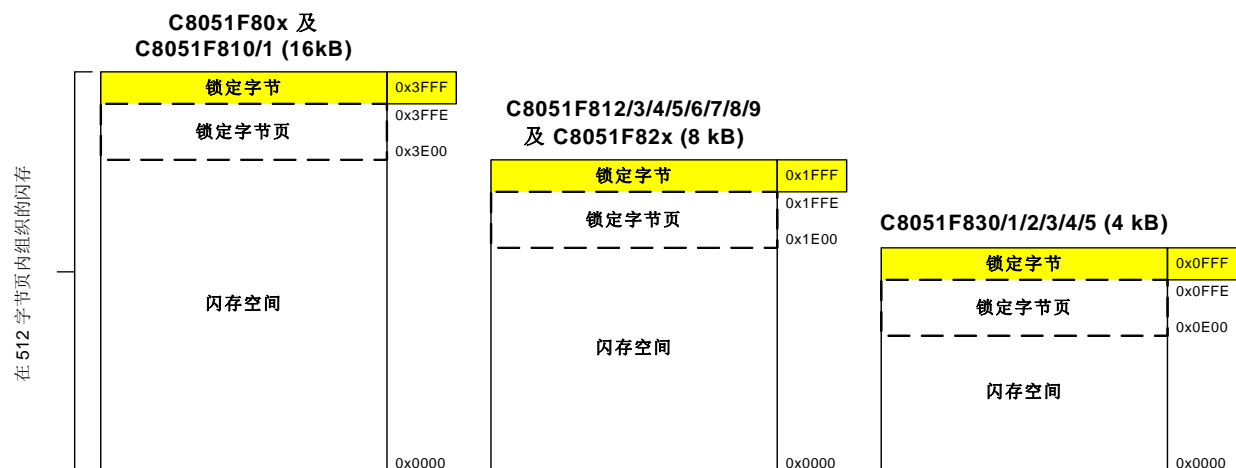


图 15.2. 闪存程序存储器映像

### 15.1.1. MOVX 指令和程序内存

8051 设备的 MOVX 指令通常用于存取外部数据内存。在 C8051F80x-83x 设备上，MOVX 指令通常用于读写片上 XRAM，但可重新配置以读取和擦除片上闪存空间。MOVC 指令一直用于读取闪存，而 MOVX 写入指令用于擦除和写入闪存。该闪存存取功能提供一种便于 C8051F80x-83x 更新程序代码并利用程序内存空间存储非易失性数据的机制。详情请参阅 113 “19. c 闪存”??。

## 15.2. 数据内存

C8051F80x-83x 设备系列产品包含 512 字节（C8051F80x, C8051F81x, and C8051F820/1/2/3）或 256 字节（C8051F824/5/6/7/8/9 and C8051F830/1/2/3/4/5）RAM 数据内存。在所有 C8051F80x-83x 设备中，该 256 字节的内存被映射到 8051 的内部 RAM 空间。在拥有 512 字节 RAM 的设备中，剩余的 256 字节内存属片上“外部”内存。数据内存图见图 15.1，以供参考。

### 15.2.1. 内部 RAM

从 0x00 到 0xFF 有 256 字节内部 RAM 被映射到数据内存空间。低 128 字节数据内存用于通用寄存器及高速暂存器。不论直接或间接寻址均可用于存取低 128 字节数据内存。因有通用寄存器的四个地址芯片，每个包括八字节宽的寄存器，从 0x00 到 0x1F 的单元均可寻址。其次从 0x20 到 0x2F 单元的 16 字节，可按字节寻址或以直接寻址模式按 128 位单元寻址。

高 128 字节数据内存仅通过间接寻址存取。该区占有与特殊功能寄存器 (SFR) 相同的地址空间，但实体上与其空间相分离。存取 0x7F 以上的单元的指令使用的寻址模式决定 CPU 存取高 128 字节数据内存空间或 SFR。使用直接寻址的指令将存取 SFR 空间。间接寻址 0x7F 以上的指令存取高 128 字节数据内存。图 15.1 描述了 C8051F80x-83x 的数据内存组织。

## 15.2.1.1. 通用寄存器

低 32 字节数据内存位于 0x00 至 0x1F，可依照通用寄存器的四个地址芯片被寻址。每一地址芯片包含八字节宽寄存器，分别命名为 R0 至 R7。这些地址芯片一次仅能激活一个。程序状态字中的两位元 RS0 (PSW.3) 和 RS1 (PSW.4)，选择激活的寄存器地址芯片（参见 SFR 定义 14.6 的 PSW 说明）。这使进入子程序并中断服务例行程序时可进行快速上下文转接。间接寻址模式使用寄存器 R0 和 R1 作为变址寄存器。

## 15.2.1.2. 位寻址

除直接存取字节形式的数据内存外，位于 0x20 至 0x2F 的 16 个数据内存单元也可作为 128 个独立的可寻址位存取。每位均有一个从 0x00 至 0x7F 的位地址。位于 0x20 的字节 0 位地址为 0x00，而位于 0x20 的字节 7 位地址为 0x07。位于 0x2F 的字节 7 位地址为 0x7F。按照使用的指令类型，位存取有别于全字节存取（位源或目的运算元而非字节源或目的）。

MCS-51™ 汇编语言允许使用为区域 XX.B（XX 为字节地址，B 为该字节内的位位置）位寻址的替代符。例如，指令：

```
MOV    C, 22.3h
```

将 0x13（0x22 单元的字节 3 位）的布尔值移至进位标志。

## 15.2.1.3. 堆栈

一个程序员的堆栈在 256 字节的数据内存的任何地点均可被定位。堆栈区指定使用堆栈指针寄存器 (SP) SFR。SP 将指向最后使用的单元。插入堆栈的下一值位于 SP+1，而后 SP 即会增加。复位将使堆栈指针初始化至单元 0x07。因此，插入堆栈的首值置于单元 0x08，也是寄存器地址芯片 1 的首个寄存器 (R0)。由此，如使用的寄存器地址芯片不止一个，SP 应被初始化至数据内存内，而非用于数据存储。堆栈深度可扩展至 256 字节。

# C8051F80x-83x

---

## 16. 系统内设备识别

C8051F80x-83x 拥有 SFR，可识别设备系列及衍生产品。此类 SFR 可在运行时被固件读取以确定执行代码的 MCU 能力。这样相同的固件形象可在不同的内存容量和外设的 MCU 上运行，并动态改变功能以适应该 MCU 能力。

为使固件识别 MCU，必须读取三个 SFR。HWID 描述 MCU 系列，DERIVID 描述该设备系列中的具体衍生产品，而 REVID 描述 MCU 的硬件版本。

### SFR 定义 16.1. HWID: 硬件 ID 字节寄存器

位	7	6	5	4	3	2	1	0
名称	HWID[7:0]							
类型	R	R	R	R	R	R	R	R
复位	0	0	1	0	0	0	1	1

SFR 地址 = 0xB5

位	名称	功能
7:0	HWID[7:0]	<p>硬件 ID 字节寄存器。</p> <p>描述 MCU 系列。</p> <p>0x23: 本文件所载设备 (C8051F80x-83x)</p>

# C8051F80x-83x

## SFR 定义 16.2. DERIVID: 硬件 ID 字节寄存器

位	7	6	5	4	3	2	1	0
名称	DERIVID[7:0]							
类型	R	R	R	R	R	R	R	R
复位	变化	变化	变化	变化	变化	变化	变化	变化

SFR 地址 = 0xAD

位	名称	功能
7:0	DERIVID[7:0]	<p><b>硬件 ID 字节寄存器。</b></p> <p>显示使用的 C8051F80x-83x 衍生产品。</p> <p>0xD0: C8051F800; 0xD1: C8051F801; 0xD2: C8051F802; 0xD3: C8051F803            0xD4: C8051F804; 0xD5: C8051F805; 0xD6: C8051F806; 0xD7: C8051F807            0xD8: C8051F808; 0xD9: C8051F809; 0xDA: C8051F810; 0xDB: C8051F811            0xDC: C8051F812; 0xDD: C8051F813; 0xDE: C8051F814; 0xDF: C8051F815            0xE0: C8051F816; 0xE1: C8051F817; 0xE2: C8051F818; 0xE3: C8051F819            0xE4: C8051F820; 0xE5: C8051F821; 0xE6: C8051F822; 0xE7: C8051F823            0xE8: C8051F824; 0xE9: C8051F825; 0xEA: C8051F826; 0xEB: C8051F827            0xEC: C8051F828; 0xED: C8051F829; 0xEE: C8051F830; 0xEF: C8051F831            0xF0: C8051F832; 0xF1: C8051F833; 0xF2: C8051F834; 0xF3: C8051F835</p>

## SFR 定义 16.3. REVID: 硬件版本 ID 字节寄存器

位	7	6	5	4	3	2	1	0
名称	REVID[7:0]							
类型	R	R	R	R	R	R	R	R
复位	变化	变化	变化	变化	变化	变化	变化	变化

SFR 地址 = 0xB6

位	名称	功能
7:0	REVID[7:0]	<p><b>硬件版本 ID 字节寄存器。</b></p> <p>显示使用的 C8051F80x-83x 硬件版本。            例如, 0x00 = 版本 A。</p>



## 17. 特殊功能寄存器

位于 0x80 至 0xFF 的直接存取数据内存构成特殊功能寄存器（SFR）。SFR 可控制 C8051F80x-83x 的资源及外围设备，并与其进行数据交换。CIP-51 控制器内核可复制典型的 8051 设备内的 SFR，并能安装额外的 SFR，以配置及访问 C8051F80x-83x 独有的子系统。这样在保持与 MCS-51™ 指令设备兼容的情况下，即可添加新的功能。表 17.1 列载了装载入 C8051F80x-83x 设备系列的 SFR。

SFR 寄存器是一种可随时直接访问的寻址模式，可访问 0x80 至 0xFF 的内存位置。配有 0x0 或 0x8 寻址终端的 SFR（如 P0、TCON、SCON0 及 IE 等）可通过位及字节寻址。而所有其他的 SFR 则仅能通过字节寻址。SFR 中未占用的地址空间将留作日后使用。访问这类区域所造成的后果不详，因此应避免此类操作。请参阅数据表中的相应页面（载于表 17.2），以了解各台寄存器的详细情况。

表 17.1. 特殊功能寄存器 (SFR) 存储器映像

F8	SPI0CN	PCA0L	PCA0H	PCA0CPL0	PCA0CPH0	P0MAT	P0MASK	VDM0CN
F0	B	P0MDIN	P1MDIN	EIP1	EIP2			PCA0PWM
E8	ADC0CN	PCA0CPL1	PCA0CPH1	PCA0CPL2	PCA0CPH2	P1MAT	P1MASK	RSTSRC
E0	ACC	XBR0	XBR1		IT01CF		EIE1	EIE2
D8	PCA0CN	PCA0MD	PCA0CPM0	PCA0CPM1	PCA0CPM2	CRC0IN	CRC0DATA	
D0	PSW	REF0CN	CRC0AUTO	CRC0CNT	P0SKIP	P1SKIP	SMB0ADM	SMB0ADR
C8	TMR2CN	REG0CN	TMR2RLL	TMR2RLH	TMR2L	TMR2H	CRC0CN	CRC0FLIP
C0	SMB0CN	SMB0CF	SMB0DAT	ADC0GTL	ADC0GTH	ADC0LTL	ADC0LTH	
B8	IP	CS0SS	CS0SE	ADC0MX	ADC0CF	ADC0L	ADC0H	
B0	CS0CN	OSCXCN	OSCI CN	OSCI CL		HWID	REVID	FLKEY
A8	IE	CLKSEL		CS0DL	CS0DH	DERVID		
A0	P2	SPI0CFG	SPI0CKR	SPI0DAT	P0MDOUT	P1MDOUT	P2MDOUT	
98	SCON0	SBUF0		CPT0CN	CS0MX	CPT0MD	CS0CF	CPT0MX
90	P1						CS0THL	CS0THH
88	TCON	TMOD	TL0	TL1	TH0	TH1	CKCON	PSCTL
80	P0	SP	DPL	DPH				PCON
	0(8)	1(9)	2(A)	3(B)	4(C)	5(D)	6(E)	7(F)

附注：以 0x0 或 0x8 结尾的 SFR 地址可通过位进行寻址，也可接受逐位指令。

**表 17.2. 特殊功能寄存器**

按照字母顺序排列 SFR。保留所有未定义的 SFR 位置。

寄存器	地址	说明	页码
<b>ACC</b>	0xE0	累加器	89
<b>ADC0CF</b>	0xBC	ADC0 配置寄存器	50
<b>ADC0CN</b>	0xE8	ADC0 控制寄存器	52
<b>ADC0GTH</b>	0xC4	ADC0 下限比较高寄存器	53
<b>ADC0GTL</b>	0xC3	ADC0 下限比较低寄存器	53
<b>ADC0H</b>	0xBE	ADC0 高寄存器	51
<b>ADC0L</b>	0xBD	ADC0 低寄存器	51
<b>ADC0LTH</b>	0xC6	ADC0 上限比较高字寄存器	54
<b>ADC0LTL</b>	0xC5	ADC0 上限比较低字寄存器	54
<b>ADC0MX</b>	0xBB	AMUX0 多路复用器通道选择器	57
<b>B</b>	0xF0	B 寄存器	90
<b>CKCON</b>	0x8E	时钟控制寄存器	210
<b>CLKSEL</b>	0xA9	时钟选择寄存器	210
<b>CPT0CN</b>	0x9B	比较器 0 控制寄存器	67
<b>CPT0MD</b>	0x9D	比较器 0 模式选择寄存器	68
<b>CPT0MX</b>	0x9F	比较器 0 MUX 选择寄存器	70
<b>CRC0AUTO</b>	0xD2	CRC0 自动控制寄存器	165
<b>CRC0CN</b>	0xCE	CRC0 控制寄存器	163
<b>CRC0CNT</b>	0xD3	CRC0 自动闪存扇区总数寄存器	165
<b>CRC0DATA</b>	0xDE	CRC0 数据输出寄存器	164
<b>CRC0FLIP</b>	0xCF	CRC0 位翻转寄存器	166
<b>CRC0IN</b>	0xDD	CRC 数据输入寄存器	164
<b>CS0THH</b>	0x97	CS0 数字比较阈值高寄存器	79
<b>CS0THL</b>	0x96	CS0 数字比较阈值高寄存器	79
<b>CS0CN</b>	0xB0	CS0 控制寄存器	75
<b>CS0DH</b>	0xAC	CS0 数据高寄存器	77
<b>CS0DL</b>	0xAB	CS0 数据低寄存器	77

表 17.2. 特殊功能寄存器 (Continued)

按照字母顺序排列 SFR。保留所有未定义的 SFR 位置。

寄存器	地址	说明	页码
CS0CF	0x9E	CS0 配置寄存器	76
CS0MX	0x9C	CS0 Mux	81
CS0SE	0xBA	自动扫描结束通道寄存器	78
CS0SS	0xB9	自动扫描启动通道寄存器	78
DERIVID	0xAD	硬件 ID 寄存器	96
DPH	0x83	数据指针高寄存器	88
DPL	0x82	数据指针低寄存器	88
EIE1	0xE6	扩展中断允许 1 寄存器	107
EIE2	0xE7	扩展中断允许 2 寄存器	108
EIP1	0xF3	扩展中断优先级 1 寄存器	109
EIP2	0xF4	扩展中断优先级 2 寄存器	110
FLKEY	0xB7	闪存锁定和关键码寄存器	119
HWID	0xB5	硬件 ID 寄存器	95
IE	0xA8	中断允许寄存器	105
IP	0xB8	中断优先级寄存器	106
IT01CF	0xE4	INT0/INT1 配置寄存器	112
OSCICL	0xB3	内部振荡器校准寄存器	131
OSCICN	0xB2	内部振荡器控制寄存器	132
OSCXCN	0xB1	外部振荡器控制寄存器	134
P0	0x80	端口 0 门锁寄存器	153
P0MASK	0xFE	端口 0 屏蔽寄存器	151
P0MAT	0xFD	端口 0 匹配寄存器	151
P0MDIN	0xF1	端口 0 输入模式配置寄存器	154
P0MDOUT	0xA4	端口 0 输出模式配置寄存器	154
P0SKIP	0xD4	端口 0 跳过寄存器	155
P1	0x90	端口 1 门锁寄存器	155
P1MASK	0xEE	P0 匹配寄存器	152

表 17.2. 特殊功能寄存器 (Continued)

按照字母顺序排列 SFR。保留所有未定义的 SFR 位置。

寄存器	地址	说明	页码
P1MAT	0xED	P1 匹配寄存器	152
P1MDIN	0xF2	端口 1 输入模式配置寄存器	156
P1MDOUT	0xA5	端口 1 输出模式配置寄存器	156
P1SKIP	0xD5	端口 1 跳过寄存器	157
P2	0xA0	端口 2 闩锁寄存器	157
P2MDOUT	0xA6	端口 2 输出模式配置寄存器	158
PCA0CN	0xD8	PCA 控制寄存器	238
PCA0CPH0	0xFC	PCA 捕捉 0 高寄存器	243
PCA0CPH1	0xEA	PCA 捕捉 0 高寄存器	243
PCA0CPH2	0xEC	PCA 捕捉 2 高寄存器	243
PCA0CPL0	0xFB	PCA 捕捉 0 低寄存器	243
PCA0CPL1	0xE9	PCA 捕捉 1 低寄存器	243
PCA0CPL2	0xEB	PCA 捕捉 2 低寄存器	243
PCA0CPM0	0xDA	PCA 模块 0 模式寄存器	241
PCA0CPM1	0xDB	PCA 模块 1 模式寄存器	241
PCA0CPM2	0xDC	PCA 模块 2 模式寄存器	241
PCA0H	0xFA	PCA 计数器高寄存器	242
PCA0L	0xF9	PCA 计数器低寄存器	242
PCA0MD	0xD9	PCA 模式寄存器	239
PCA0PWM	0xF7	PCA PWM 配置寄存器	240
PCON	0x87	电源控制寄存器	122
PSCTL	0x8F	程序存储读写控制寄存器	118
PSW	0xD0	程序状态字寄存器	91
REF0CN	0xD1	电压基准控制寄存器	62
REG0CN	0xC9	稳压器控制寄存器	64
REVID	0xB6	ID 版本寄存器	96
RSTSRC	0xEF	复位源配置 / 状态	128

表 17.2. 特殊功能寄存器 (Continued)

按照字母顺序排列 SFR。保留所有未定义的 SFR 位置。

寄存器	地址	说明	页码
<b>SBUF0</b>	0x99	UART0 数据缓冲器寄存器	207
<b>SCON0</b>	0x98	UART0 控制寄存器	206
<b>SMB0ADM</b>	0xD6	SMBus 从地址屏蔽寄存器	191
<b>SMB0ADR</b>	0xD7	SMBus 从地址寄存器	191
<b>SMB0CF</b>	0xC1	SMBus 配置寄存器	186
<b>SMB0CN</b>	0xC0	SMBus 控制寄存器	188
<b>SMB0DAT</b>	0xC2	SMBus 数据寄存器	192
<b>SP</b>	0x81	堆栈指针寄存器	89
<b>SPI0CFG</b>	0xA1	SPI0 配置寄存器	174
<b>SPI0CKR</b>	0xA2	SPI0 时钟频率控制寄存器	176
<b>SPI0CN</b>	0xF8	SPI0 控制寄存器	175
<b>SPI0DAT</b>	0xA3	SPI0 数据寄存器	176
<b>TCON</b>	0x88	定时器 / 计数器控制寄存器	215
<b>TH0</b>	0x8C	定时器 / 计数器 0 高寄存器	218
<b>TH1</b>	0x8D	定时器 / 计数器 1 高寄存器	218
<b>TL0</b>	0x8A	定时器 / 计数器 0 低寄存器	217
<b>TL1</b>	0x8B	定时器 / 计数器 1 低寄存器	217
<b>TMOD</b>	0x89	定时器 / 计数器模式寄存器	216
<b>TMR2CN</b>	0xC8	定时器 / 计数器 2 控制寄存器	222
<b>TMR2H</b>	0xCD	定时器 / 计数器 2 高寄存器	224
<b>TMR2L</b>	0xCC	定时器 / 计数器 2 低寄存器	224
<b>TMR2RLH</b>	0xCB	定时器 / 计数器 2 重载高寄存器	223
<b>TMR2RLL</b>	0xCA	定时器 / 计数器 2 重载低寄存器	223
<b>VDM0CN</b>	0xFF	VDD 监视器控制寄存器	126
<b>XBR0</b>	0xE1	端口 I/O 交叉开关控制 0 寄存器	148
<b>XBR1</b>	0xE2	端口 I/O 交叉开关控制 1 寄存器	149
所有其他 SFR 的位置		备用	

# C8051F80x-83x

---

## 18. 中断

C8051F80x-83x 包括一个扩展中断系统，该系统共支持两种优先级别的 15 种中断源。片上外围设备与外输入引脚间的中断源分布依据设备的特定版本而有所差异。每项中断源在 SFR 中均有一个或多个对应的未决中断标志。当外围设备或外部源头遇到有效的中断条件时，相关的未决中断标志即会设定为逻辑 1。

如对源头开启中断，则在设置未决中断标志时会产生一个中断请求。每当执行完当前指令，CPU 会立即在预定地址生成一个 LCALL，即可开始执行中断服务例行程序 (ISR)。每个 ISR 必须以 RETI 指令结尾，这样即可使程序执行回复至下一项指令并执行该指令，犹如中断命令尚未发出。如未启用中断，则硬件及程序将忽略未决中断标志，按常规执行。（无论启用 / 禁用中断状态，未决中断标志均将设置为逻辑 1。）

通过控制 SFR (IE-EIE1) 中与每个中断源相应的中断启用位，即可单独启用或禁用每个中断源。不过，必须事先将 EA 位 (IE.7) 设置为逻辑 1，以便启用所有中断，方可识别启用的单项中断。将 EA 位设置为逻辑 0 将禁用所有中断源，无论是否启用单独的中断设置。

当 CPU 矢量转至 ISR 时，硬件将自动清除若干未决中断标志。不过，多数标志并不会被硬件清除，而必须自 ISR 返回前通过软件清除。如在 CPU 完成中断回归 (RETI) 指令后，未决中断标志的设置仍有效，则将立即生成一项新的中断要求，CPU 在完成下一次指令后将重新进入 ISR。

## 18.1. MCU 中断源和矢量

C8051F80x-83x MCU 支持 15 种中断源。可通过设置未决中断标志为逻辑 1，使软件模拟中断。如对标志启用中断，则将生成一项中断要求，CPU 将转至与未决中断标志相关的 ISR 矢量地址。表 18.1 概述了与矢量地址、优先级顺序及控制位相关的 MCU 中断源。有关外围设备有效的中断条件及未决中断标志操作的资料，请参阅有关特定片上外围设备的数据表部分。

### 18.1.1. 中断优先级

每种中断源均可单独编程入一种或两种优先级：低级别和高级别。较低优先级的中断服务例程可能会被高优先级的中断抢先。而高优先级中断则不会被抢先。每种中断在 SFR（IP 或 EIP1）中均有其对应的中断优先级位，用以配置其优先等级。默认为低优先级。如同时识别两个中断，则将优先适用具有较高优先级别的中断。如两个中断的优先级别相当，则将根据已设定的优先顺序（载于表 18.1）进行判断。

### 18.1.2. 中断延时

中断发生时，其反应时间依据 CPU 的状态而定。未决中断在每个系统时钟循环中进行抽样及优先解码。因此，最快的可能反应时间为 5 个系统时钟循环：其中 1 个时钟循环用于检测中断，4 个时钟循环用于完成 ISR 中的 LCALL。如 RETI 执行时出现一个未决中断，则在 LCALL 处理未决中断前将执行一项单独的指令。因此，当 CPU 正在执行 RETI 指令，且下一项指令为 DIV 时，（如当时并未处理其他的中断或新的中断具有更高的优先级）中断将需要最长的反应时间。在这种情况下，反应时间为 18 个系统时钟循环：其中 1 个时钟循环用于检测中断，5 个时钟循环用于执行 RETI，8 个时钟循环用于完成 DIV 指令，而剩下的 4 个时钟循环则用于执行 ISR 中的 LCALL。如新的中断与 CPU 所执行在 ISR 中所执行的中断具有相同或更低的优先级，则仅当当前的 ISR（包括 RETI 及随后的指令）完成后（包括 RETI 及随后的指令），才会执行新的中断。



表 18.1: 中断概述

中断源	中断矢量	优先顺序	未决标志	是否可通过位寻址?	是否可由 HW 清除?	启用标志	优先级控制
复位	0x0000	最高	无	不可用	不可用	始终启用	始终为最高级
外部中断 0 (INT0)	0x0003	0	IE0 (TCON.1)	是	是	EX0 (IE.0)	PX0 (IP.0)
定时器 0 溢出	0x000B	1	TF0 (TCON.5)	是	是	ET0 (IE.1)	PT0 (IP.1)
外部中断 1 (INT1)	0x0013	2	IE1 (TCON.3)	是	是	EX1 (IE.2)	PX1 (IP.2)
定时器 1 溢出	0x001B	3	TF1 (TCON.7)	是	是	ET1 (IE.3)	PT1 (IP.3)
UART0	0x0023	4	RI0 (SCON0.0) TI0 (SCON0.1)	是	否	ES0 (IE.4)	PS0 (IP.4)
定时器 2 溢出	0x002B	5	TF2H (TMR2CN.7) TF2L (TMR2CN.6)	是	否	ET2 (IE.5)	PT2 (IP.5)
SPI0	0x0033	6	SPIF (SPI0CN.7) WCOL (SPI0CN.6) MODF (SPI0CN.5) RXOVRN (SPI0CN.4)	是		ESPI0 (IE.6)	PSPI0 (IP.6)
SMB0	0x003B	7	SI (SMB0CN.0)	是	否	ESMB0 (EIE1.0)	PSMB0 (EIP1.0)
端口匹配	0x0043	8	无	不可用	不可用	EMAT (EIE1.1)	PMAT (EIP1.1)
ADC0 窗口比较	0x004B	9	AD0WINT (ADC0CN.3)	是	否	EWADC0 (EIE1.2)	PWADC0 (EIP1.2)
ADC0 转换完成	0x0053	10	AD0INT (ADC0CN.5)	是	否	EADC0 (EIE1.3)	PADC0 (EIP1.3)
可编程计数器阵列	0x005B	11	CF (PCA0CN.7) CCFn (PCA0CN.n)	是	否	EPCA0 (EIE1.4)	PPCA0 (EIP1.4)
比较器 0	0x0063	12	CP0FIF (CPT0CN.4) CP0RIF (CPT0CN.5)	否	否	ECP0 (EIE1.5)	PCP0 (EIP1.5)
备用							
备用							
CS0 转换完成	0x007B	15	CS0INT (CS0CN.5)	否	否	ECSCPT (EIE2.0)	PSCCPT (EIP2.0)
CS0 下限	0x0083	16	CS0CMPF (CS0CN.0)	否	否	ECSGRT (EIE2.1)	PSCGRT (EIP2.1)

## 18.2. 中断寄存器说明

本节介绍用于启用中断源及设置其优先级别的 SFR。有关外围设备有效的中断条件及未决中断标志操作的资料，请参阅有关特定片上外围设备的数据表部分。

# C8051F80x-83x

## SFR 定义 18.1: IE: 中断允许寄存器

位	7	6	5	4	3	2	1	0
名称	EA	ESPI0	ET2	ES0	ET1	EX1	ET0	EX0
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xA8; 可位寻址

位	名称	功能
7	EA	<p><b>启用所有中断。</b>            全部启用 / 禁用所有中断。覆盖单独的中断屏蔽设置。            0: 禁用所有的中断源。            1: 根据单独的屏蔽设置启用每项中断。</p>
6	ESPI0	<p><b>启用的串行外围接口 (SPI0) 中断。</b>            此位设置 SPI0 中断的屏蔽。            0: 禁用所有的 SPI0 中断。            1: 启用 SPI0 生成的中断要求。</p>
5	ET2	<p><b>启用定时器 2 中断。</b>            此位设置定时器 2 中断的屏蔽。            0: 禁用定时器 2 中断。            1: 启用 TF2L 或 TF2H 标志生成的中断要求。</p>
4	ES0	<p><b>启用 UART0 中断。</b>            此位设置 UART0 中断的屏蔽。            0: 禁用 UART0 中断。            1: 启用 UART0 中断。</p>
3	ET1	<p><b>启用定时器 1 中断。</b>            此位设置定时器 1 中断的屏蔽。            0: 禁用所有定时器 1 中断。            1: 启用 TF1 标志生成的中断要求。</p>
2	EX1	<p><b>启用外部中断 1。</b>            此位设置外部中断 1 的屏蔽。            0: 禁用外部中断 1。            1: 启用 INT1 输入生成的中断要求。</p>
1	ET0	<p><b>启用定时器 0 中断。</b>            此位设置定时器 0 中断的屏蔽。            0: 禁用所有定时器 0 中断。            1: 启用 TF0 标志生成的中断要求。</p>
0	EX0	<p><b>启用外部中断 0。</b>            此位设置外部中断 0 的屏蔽。            0: 禁用外部中断 0。            1: 启用 INTO 输入生成的中断要求。</p>

## SFR 定义 18.2: IP: 中断优先级寄存器

位	7	6	5	4	3	2	1	0
名称		PSPI0	PT2	PS0	PT1	PX1	PT0	PX0
类型	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	1	0	0	0	0	0	0	0

SFR 地址 = 0xB8; 可位寻址

位	名称	功能
7	未使用	读 = 1b, 写 = 无需在意。
6	PSPI0	<b>串行外围接口 (SPI0) 中断优先级控制。</b> 此位设置 SPI0 中断的优先级。 0: SPI0 中断设置可降低优先级别。 1: SPI0 中断设置可升高优先级别。
5	PT2	<b>定时器 2 中断优先级控制。</b> 此位设置定时器 2 中断的优先级。 0: 定时器 2 中断设置可降低优先级别。 1: 定时器 2 中断设置可升高优先级别。
4	PS0	<b>UART0 中断优先级控制。</b> 此位设置 UART0 中断的优先级。 0: UART0 中断设置可降低优先级别。 1: UART0 中断设置可升高优先级别。
3	PT1	<b>定时器 1 中断优先级控制。</b> 此位设置定时器 1 中断的优先级。 0: 定时器 1 中断设置可降低优先级别。 1: 定时器 1 中断设置可升高优先级别。
2	PX1	<b>外部中断 1 优先级控制。</b> 此位设置外部中断 1 中断的优先级。 0: 外部中断 1 设置可降低优先级别。 1: 外部中断 1 设置可升高优先级别。
1	PT0	<b>定时器 0 中断优先级控制。</b> 此位设置定时器 0 中断的优先级。 0: 定时器 0 中断设置可降低优先级别。 1: 定时器 0 中断设置可升高优先级别。
0	PX0	<b>外部中断 0 优先级控制。</b> 此位设置外部中断 0 中断的优先级。 0: 外部中断 0 设置可降低优先级别。 1: 外部中断 0 设置可升高优先级别。

# C8051F80x-83x

## SFR 定义 18.3: EIE1: 扩展中断允许 1 寄存器

位	7	6	5	4	3	2	1	0
名称	备用	备用	ECP0	EADC0	EPCA0	EWADC0	EMAT	ESMB0
类型	W	W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xE6

位	名称	功能
7	备用	必须写入 0。
6	备用	备用。 必须写入 0。
5	ECP0	<b>启用比较器 0 (CP0) 中断。</b> 此位设置 CP0 上升沿或下降沿中断的屏蔽。 0: 禁用 CP0 中断。 1: 启用 CP0RIF 及 CP0FIF 标志生成的中断要求。
4	EADC0	<b>启用 ADC0 转换完成中断。</b> 此位设置 ADC0 转换完成中断的屏蔽。 0: 禁用 ADC0 转换完成中断。 1: 启用 AD0INT 标志生成的中断要求。
3	EPCA0	<b>启用可编程计数器阵列 (PCA0) 中断。</b> 此位设置 PCA0 中断的屏蔽。 0: 禁用所有 PCA0 中断。 1: 启用 PCA0 生成的中断要求。
2	EWADC0	<b>启用窗口比较 ADC0 中断。</b> 此位设置 ADC0 窗口比较中断的屏蔽。 0: 禁用 ADC0 窗口比较中断。 1: 启用 ADC0 窗口比较标志 (AD0WINT) 生成的中断要求。
1	EMAT	<b>启用端口匹配中断。</b> 此位设置端口匹配事件中中断的屏蔽。 0: 禁用所有端口匹配中断。 1: 启用端口匹配生成的中断要求。
0	ESMB0	<b>启用 SMBus (SMB0) 中断。</b> 此位设置 SMB0 中断的屏蔽。 0: 禁用所有 SMB0 中断。 1: 启用 SMB0 生成的中断要求。

## SFR 定义 18.4: EIE2: 扩展中断允许 2 寄存器

位	7	6	5	4	3	2	1	0
名称							ECSGRT	ECSCPT
类型	R	R	R	R	R	R	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xE7

位	名称	功能
7:2	未使用	读取 = 000000b; 写 = 无需在意。
1	ECSGRT	启用电容传感下限比较器中断。 0: 禁用电容传感下限比较器中断。 1: 启用 CS0CMPF 生成的中断要求。
0	ECSCPT	启用电容传感转换完成中断。 0: 禁用电容传感转换完成中断。 1: 启用 CS0INT 生成的中断要求。

# C8051F80x-83x

## SFR 定义 18.5: EIP1: 扩展中断优先级 1 寄存器

位	7	6	5	4	3	2	1	0
名称	备用	备用	PCP0	PPCA0	PADC0	PWADC0	PMAT	PSMB0
类型	W	W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xF3

位	名称	功能
7:6	备用	必须写入 0。
5	PCP0	<b>比较器 0 (CP0) 中断优先级控制。</b> 此位设置 CP0 上升沿或下降沿中断的优先级。 0: CP0 中断设置可降低优先级别。 1: CP0 中断设置可升高优先级别。
4	PPCA0	<b>可编程计数器阵列 (PCA0) 中断优先级控制。</b> 此位设置 PCA0 中断的优先级。 0: PCA0 中断设置可降低优先级别。 1: PCA0 中断设置可升高优先级别。
3	PADC0	<b>ADC0 转换完成中断优先级控制。</b> 此位设置 ADC0 转换完成中断的优先级。 0: ADC0 转换完成中断设置可降低优先级别。 1: ADC0 转换完成中断设置可升高优先级别。
2	PWADC0	<b>ADC0 窗口比较器中断优先级控制。</b> 此位设置 ADC0 窗口中断的优先级。 0: ADC0 窗口中断设置可降低优先级别。 1: ADC0 窗口中断设置可升高优先级别。
1	PMAT	<b>端口匹配中断优先级控制。</b> 此位设置端口匹配事件中中断的优先级。 0: 端口匹配中断设置可降低优先级别。 1: 端口匹配中断设置可升高优先级别。
0	PSMB0	<b>SMBus (SMB0) 中断优先级控制。</b> 此位设置 SMB0 中断的优先级。 0: SMB0 中断设置可降低优先级别。 1: SMB0 中断设置可升高优先级别。

**SFR 定义 18.6: EIP2: 扩展中断优先级 2 寄存器**

位	7	6	5	4	3	2	1	0
名称	备用	备用	备用	备用	备用	备用	PSCGRT	PSCCPT
类型	R	R	R	R	R	R	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xF4

位	名称	功能
7:2	备用	
1	PSCGRT	<b>电容传感下限比较器优先级控制。</b> 此位设置电容传感下限比较器中断的优先级。 0: CS0 下限比较器中断设置可降低优先级别。 1: CS0 下限比较器设置可升高优先级别。
0	PSCCPT	<b>电容传感转换完成优先级控制。</b> 此位设置电容传感转换完成中断的优先级。 0: CS0 转换完成设置可降低优先级别。 1: CS0 转换完成设置可提高优先级别。

## 18.3. $\overline{\text{INT0}}$ 和 $\overline{\text{INT1}}$ 外部中断

$\overline{\text{INT0}}$  和  $\overline{\text{INT1}}$  外部中断源可配置为高态或低态有效，边沿或水平敏感。 $\text{IT01CF}$  寄存器中规定  $\text{IN0PL}$  ( $\overline{\text{INT0}}$  极性) 和  $\text{IN1PL}$  ( $\overline{\text{INT1}}$  极性) 位选择高态或低态有效， $\text{TCON}$  中的  $\text{IT0}$  和  $\text{IT1}$  位 (见 211 页第“28.1 条定时器 0 和定时器 1”) 选择水平或边沿敏感。下表列载出可能的配置。

$\text{IT0}$	$\text{IN0PL}$	$\overline{\text{INT0}}$ 中断
1	0	低态有效，边沿敏感
1	1	高态有效，边沿敏感
0	0	低态有效，水平敏感
0	1	高态有效，水平敏感

$\text{IT1}$	$\text{IN1PL}$	$\overline{\text{INT1}}$ 中断
1	0	低态有效，边沿敏感
1	1	高态有效，边沿敏感
0	0	低态有效，水平敏感
0	1	高态有效，水平敏感

$\overline{\text{INT0}}$  和  $\overline{\text{INT1}}$  在端口针脚的分布情况详见  $\text{IT01CF}$  寄存器定义 (见  $\text{SFR}$  定义 18.7)。注意， $\overline{\text{INT0}}$  和  $\overline{\text{INT1}}$  端口针脚分配与任何交叉开关分配无关。 $\overline{\text{INT0}}$  和  $\overline{\text{INT1}}$  将监控各自所分配的端口针脚，而不会影响通过交叉开关分配予端口针脚的外围设备。如仅向  $\overline{\text{INT0}}$  及 / 或  $\overline{\text{INT1}}$  分配端口针脚，则可设定交叉开关以跳过所选的针脚。此操作可通过在寄存器  $\text{XBR0}$  中设置相关位来完成 (有关交叉开关配置的完整详情，请参阅第 143 页“23.3. 优先级交叉开关编码器”一节)。

$\text{IE0}$  ( $\text{TCON.1}$ ) 和  $\text{IE1}$  ( $\text{TCON.3}$ ) 分别作为  $\overline{\text{INT0}}$  和  $\overline{\text{INT1}}$  外部中断的未决中断标志。如  $\overline{\text{INT0}}$  或  $\overline{\text{INT1}}$  的外部中断配置属边沿敏感，则当 CPU 矢量转至  $\text{ISR}$  时，相应的未决中断标志将会被硬件自动清除。当配置属水平敏感，则未决中断标志仍将保持为逻辑 1，而输入将根据相应极性位 ( $\text{IN0PL}$  或  $\text{IN1PL}$ ) 的定义处于激活状态，当输入处于非激活状态时，标志将保持为逻辑 0。外部中断源须保持输入处于激活状态，直至中断要求被识别。然后须先释放中断要求，方可完成  $\text{ISR}$  的执行或产生另一项中断要求。



SFR 定义 18.7: IT01CF:  $\overline{\text{INT0}}$  /  $\overline{\text{INT1}}$  配置寄存器

位	7	6	5	4	3	2	1	0
名称	IN1PL	IN1SL[2:0]			INOPL	IN0SL[2:0]		
类型	R/W	R/W			R/W	R/W		
复位	0	0	0	0	0	0	0	1

SFR 地址 = 0xE4

位	名称	功能
7	IN1PL	<b><math>\overline{\text{INT1}}</math> 极性。</b> 0: $\overline{\text{INT1}}$ 输入处于低态有效。 1: $\overline{\text{INT1}}$ 输入处于高态有效。
6:4	IN1SL[2:0]	<b><math>\overline{\text{INT1}}</math> 端口引脚选择位。</b> 这些位选择分配予 $\overline{\text{INT1}}$ 的端口引脚。注意，这种引脚分配独立于交叉开关； $\overline{\text{INT1}}$ 将监控分配的端口引脚，而不会影响通过交叉开关分配予端口引脚的外围设备。如配置为跳过所选引脚，则交叉开关不会将端口引脚分配予外围设备。 000: 选择 P0.0 001: 选择 P0.1 010: 选择 P0.2 011: 选择 P0.3 100: 选择 P0.4 101: 选择 P0.5 110: 选择 P0.6 111: 选择 P0.7
3	INOPL	<b><math>\overline{\text{INT0}}</math> 极性。</b> 0: $\overline{\text{INT0}}$ 输入处于低态有效。 1: $\overline{\text{INT0}}$ 输入处于高态有效。
2:0	IN0SL[2:0]	<b><math>\overline{\text{INT0}}</math> 端口引脚选择位。</b> 这些位选择分配予 $\overline{\text{INT0}}$ 的端口引脚。注意，这种引脚分配独立于交叉开关； $\overline{\text{INT0}}$ 将监控分配的端口引脚，而不会影响通过交叉开关分配予端口引脚的外围设备。如配置为跳过所选引脚，则交叉开关不会将端口引脚分配予外围设备。 000: 选择 P0.0 001: 选择 P0.1 010: 选择 P0.2 011: 选择 P0.3 100: 选择 P0.4 101: 选择 P0.5 110: 选择 P0.6 111: 选择 P0.7

# C8051F80x-83x

---

## 19. c闪存

内部含有片上、可再编程闪存，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 写入指令对闪存进行系统编程。清除到逻辑 0 时，必须擦除一个闪存位方能再返回逻辑 1 状态。在进行重新编程之前，一般要将闪存字节擦除（设置为 0xFF）。为了保障操作正确，写入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入 / 擦除操作何时结束。在闪存写入 / 擦除操作期间，代码停止执行。请参阅表 7.6，了解闪存的完整电气特性。

### 19.1. 编程闪存

对闪存进行编程的最简单的方法是使用 Silicon Laboratories 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设备的唯一编程方法。有关对闪存进行编程的 C2 命令的详情，请见第 244 页“30. C2 接口”一条。

软件使用 MOVX 写入指令，将待编程的地址和数据字节当作正常操作数，可对闪存进行编程。使用 MOVX 进行闪存编程前，必须通过以下步骤启用闪存编程操作：(1) 将 PSWE 程序存储写入允许位（PSCTL.0）设置为逻辑 1（指示 MOVX 写入目标闪存）；及 (2) 按顺序将闪存关键码写入闪存锁定寄存器（FLKEY）。在被软件清除之前，PSWE 位一直保持设定。有关固件闪存编程的详细指引请见第 115 页“19.4. 闪存写入和擦除指引”一条。

**注意：**写入或擦除闪存具有最小 SYSCLK 频率要求，详细信息请参照 <“电气特性”一章中的表 7.6>

为了保证闪存内容的正确性，必须在包含代码（该代码可以从软件中写入及 / 或擦除闪存）的任何系统中启用片上 VDD 监视器，并将其作为复位源。此外，启用 VDD 监视器和允许 VDD 监视器担任复位源之间不得有延时。在 VDD 监视器被禁用或未被允许作为复位源时，任何意图写入或擦除闪存的举动都会导致闪存故障设备复位。

#### 19.1.1. 闪存锁定和主要功能

从用户软件编写和擦除闪存受锁定和关键码功能的保护。在进行闪存操作之前，必须按顺序向闪存锁定和关键码寄存器（FLKEY）写入正确的关键码。关键码为：0xA5、0xF1。写入关键码的时序并不重要，但必须按顺序填写。如果关键码的写入顺序不对，或写入了错误的键码，闪存写入和擦除将被禁止，直至下一次系统复位。如果在正确写入关键码之前进行了闪存写入和擦除操作，闪存写入和擦除也将被禁止。每次写入或擦除操作之后，闪存锁定功能复位；在进行下一次闪存操作之前，必须重写关键码。SFR 定义 19.2. 中给出了 FLKEY 寄存器的详细说明。

#### 19.1.2. 闪存擦除程序

闪存是以 512 字节的页面来组织的。一次擦除操作将应用到整个页面（将该页的所有字节设置为 0xFF）。要擦除整个 512 字节的页面，执行以下步骤：

1. 跳过电流中断状态并禁止中断。
2. 设置 PSEE 位（寄存器 PSCTL）。
3. 设置 PSWE 位（寄存器 PSCTL）。
4. 将第一个关键码写入 FLKEY: 0xA5。
5. 将第二个关键码写入 FLKEY: 0xF1。
6. 用 MOVX 指令向待擦除的 512 字节页面内的任何一个地址写入一个数据字节。
7. 清除 PSWE 和 PSEE 位。
8. 恢复先前的中断状态。

# C8051F80x-83x

---

每擦除一页 512 字节页面，必须重复 4-6 步。

**附注：** 闪存安全设置可预防一些闪存页的擦除，如保留区和含有锁定字节的页面。有关闪存安全设置和限制影响闪存擦除操作的概述，请参阅 114 页的“19.3. 安全选项”章节。

## 19.1.3. 闪存写入步骤

对闪存的写入可清除位元至逻辑 0，但无法设置它们；惟有擦除操作可将闪存内的位元设为内存的逻辑 1。  
**写入新值前，应擦除待编写的字节位置。**

建议按如下步骤向内存中写入单字节：

1. 跳过电流中断状态并禁止中断。
2. 确保闪存字节已擦除（赋值 0xFF）。
3. 设置 PSWE 位（寄存器 PSCTL）。
4. 清除 PSEE 位（寄存器 PSCTL）。
5. 将第一个密钥写入 FLKEY: 0xA5。
6. 将第二个密钥写入 FLKEY: 0xF1。
7. 使用 MOVX 指令，将单数据字节写入 512 字节扇区内的指定位置。
8. 清除 PSWE 位。
9. 恢复先前的中断状态。

每写入一字节，必须重复步骤 5-7。

**附注：** 闪存安全设置可预防写入保留区等若干闪存区。有关闪存安全设置和限制影响闪存写入操作的概述，请参阅 114 页的“19.3. 安全选项”章节。

## 19.2. 非易失性数据存储

闪存可用于存储非易失性数据和程序代码。这允许在运行时计算并存储如校准系数之类的数据。使用 MOVX 写入指令写入数据，使用 MOVC 指令读取数据。

**附注：** MOVX 读取指令始终用于 XRAM。

## 19.3. 安全选项

CIP-51 提供安全选项以防备闪存被软件不慎修改及专利程序代码和常量泄露。程序存储写入启用位（寄存器 PSCTL 内的 PSWE 位）和程序存储擦除启用位（寄存器 PSCTL 内的 PSEE 位）可防备闪存遭软件意外修改。在软件可修改闪存之前，PSWE 必须被明确设置为 1；在软件可擦除闪存之前，PSWE 和 PSEE 必须被设置为 1。其他安全特征防备专利程序代码和数据常量通过 C2 接口遭读取或更改。

位于闪存用户空间最后一字节的安全锁定字节可防备程序闪存被未受保护的代码或 C2 接口存取（读取、写入和擦除）。通过将非 0xFF 数值写入锁定字节，闪存安全机制允许用户锁定从 0 页开始的所有闪存页。**请注意，在锁定字节中写入非 0xFF 数值将会锁定所有闪存页，使其无法被读取、写入或擦除（包括含有锁定字节的页面）。**

闪存安全级别取决于闪存存取方式。读取、写入和擦除的三种闪存存取方式是通过 C2 调试接口，在未锁定页上执行的用户固件和在锁定页上执行的用户固件。表 19.1 概述 C8051F80x-83x 设备的闪存安全特征。

表 19.1: 闪存安全概述

操作	C2 调试接口	用户固件执行对象:	
		未锁定页	锁定页
读取、写入或擦除未锁定页 (含有锁定字节的页除外)	允许	允许	允许
读取、写入或擦除锁定页 (含有锁定字节的页除外)	禁止	FEDR	允许
读取或写入含有锁定字节的页 (如无锁定页)	允许	允许	允许
读取或写入含有锁定字节的页 (如均为锁定页)	禁止	FEDR	允许
读取锁定字节内容 (如无锁定页)	允许	允许	允许
读取锁定字节内容 (如均为锁定页)	禁止	FEDR	允许
擦除含有锁定字节的页 (如无锁定页)	允许	FEDR	FEDR
擦除含有锁定字节的页 - 解除所有页的锁定 (如均为锁定页)	仅通过 C2DE	FEDR	FEDR
锁定其他页 (锁定字节的 1s 改为 0s)	禁止	FEDR	FEDR
解锁独立页 (锁定字节的 0s 改为 1s)	禁止	FEDR	FEDR
读取、写入或擦除保留区	禁止	FEDR	FEDR
<p>C2DE—C2 设备擦除 (擦除包括含有锁定字节页在内的所有闪存页)            FEDR—禁止; 促使闪存故障设备复位 (复位后 RSTSRC 内的 FERROR 位为 1)</p> <ul style="list-style-type: none"> <li>■ 无需在意通过 C2 接口执行的所有违规操作 (不会引起设备复位)。</li> <li>■ 锁定任何闪存页及含有锁定字节的页。</li> <li>■ 锁定字节一经写入, 则仅可通过执行 C2 设备擦除来修改。</li> <li>■ 如用户代码写入锁定字节, 则直至下次设备复位锁定方可生效。</li> </ul>			

#### 19.4. 闪存写入和擦除指引

包含按照软件写入或擦除闪存的例行程序的任何系统均有风险, 即如 CPU 运行时超出其 VDD 规定运行范围、系统时钟频率或温度, 写入或擦除例行程序将自行启用。该闪存修改代码的意外操作会导致闪存内容改变, 造成系统故障, 惟有重新设置设备代码方可修复。

为帮助预防闪存被固件意外修改, 必须启用 C8051F80x-83x 设备上的 VDD 监控器并作为复位源以成功修改闪存。如无法启用 VDD 监控器或 VDD 监控器复位源, 当固件试图修改闪存时将造成闪存故障设备复位。

# C8051F80x-83x

对于包含按照代码写入或擦除闪存的例行程序的任何系统，建议遵循以下指引。

## 19.4.1. VDD 维护和 VDD 监控器

1. 如系统供电受电压或电流“峰值”影响，请为电源配置足够的瞬变保护设备以确保电源电压低于绝对最大额定值表所列。
2. 确保最小 **VDD** 上升时间达到规定的 **1 ms**。如系统未能满足上升时间规定，则将一外部 **VDD** 节电电路加至设备 **RST** 针，该 **RST** 针保持设备复位直至 **VDD** 达到最低设备工作电压，而当 **VDD** 低于最低设备工作电压时再次使用 **RST**。
3. 保持片上 **VDD** 监控器处于启用状态，并启用 **VDD** 监控器作为复位源，尽早使用代码。这应是执行复位向量后的第一批指令。对 **C** 语言系统而言，这还包括更改 **C** 语言编译程序施加的启动代码。欲知详情，请参阅编译程序文件。确保软件及时启用 **VDD** 监控器并将其作为复位源。指导该操作的代码实例，请登陆 **Silicon Laboratories** 网站查找“AN201：固件写入闪存”。

**附注：** 在 **C8051F80x-83x** 设备上，**VDD** 监控器及其复位源均须被启用以写入或擦除闪存，而不产生闪存故障设备复位。

在 **C8051F80x-83x** 设备上，**VDD** 监控器及其复位源均在上电复位后由硬件启用。

4. 作为附加的预防措施，明确启用 **VDD** 监控器并使其在写入和擦除闪存的功能内作为复位源。**VDD** 监控器启用指令应紧接 **PSWE** 设置为 **a 1** 的指令后发出，但早于闪存写入或擦除操作指令发出。
5. 确保对 **RSTSRC**（复位源）寄存器的所有写入使用直接赋值算符并明确禁用位运算符（如 **AND** 或 **OR**）。例如，“**RSTSRC = 0x02**”为正确格式，而“**RSTSRC |= 0x02**”则属错误。
6. 确保对 **RSTSRC** 寄存器的所有写入明确将 **PORSF** 位设置为 **a 1**。需检查启用其他复位源的初始化代码，如缺失时钟探测器或比较器，以及强制软件复位的指令。一项关于“**RSTSRC**”的全球调查可很快证实这一点。

## 19.4.2. PSWE 维护

1. 当 **PSWE** 位（**b0** 在 **PSCTL**）被设置为 **a 1** 时减少代码位数。应有一个例行代码将 **PSWE** 设置为 **a 1** 以写入闪存字节而有另一例行代码将 **PSWE** 和 **PSEE** 均设为 **a 1** 以擦除闪存页。
2. 当 **PSWE** 设为 **a 1** 时，将可变存取数目减至最低。更新指针地址并在“**PSWE = 1; ...PSWE = 0;**”区外环回维护。指导该操作的代码实例，请登陆 **Silicon Laboratories** 网站查找“AN201：固件写入闪存”。
3. 在将 **PSWE** 设为 **a 1** 前不可中断运行，直至 **PSWE** 被复位至 **0** 后方可中断。在闪存操作已完成及软件再次中断运行后，在闪存写入或擦除操作过程中施加的任何中断将加入优先级别。
4. 确报闪存写入和擦除指针变量未位于 **XRAM** 内。有关如何明确定位不同内存区变量的指令，请参阅您的编译程序文件。
5. 对写入或擦除闪存的例行程序进行地址边界检查以确保例行程序访问非法地址不会导致闪存修改。

### 19.4.3. 系统时钟

1. 如在外部晶体环境下操作，请注意晶体性能易受电干扰，且对布局和温度的改变十分敏感。若系统在电噪声环境下运行，请使用内部振荡器或使用外部 CMOS 时钟。
2. 若在外部振荡器下运行，请在闪存写入或擦除操作过程中切换至内部振荡器。外部振荡器可持续运行，闪存操作结束后可将 CPU 切回外部振荡器。

更多闪存建议和代码实例，请参阅 Silicon Laboratories 网站的“AN201：由固件写入闪存”。

# C8051F80x-83x

## SFR 定义 19.1: PSCTL: 程序存储读写控制寄存器

位	7	6	5	4	3	2	1	0
名称							PSEE	PSWE
类型	R	R	R	R	R	R	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 =0x8F

位	名称	功能
7:2	未使用	读 = 000000b; 写 = 无需在意。
1	PSEE	<b>程序存储擦除允许。</b> 设置此位（以及 PSWE）允许删除一整页闪存程序内存。如此位为逻辑 1 且允许闪存写入（PSWE 为逻辑 1），使用 MOVX 指令写入的闪存将擦除包含以 MOVX 指令定址的一整页。写入数据字节的值无关紧要。 0: 闪存程序内存擦除禁用。 1: 闪存程序内存擦除启用。
0	PSWE	<b>程序存储写入允许。</b> 设置此位允许使用 MOVX 写入指令向闪存程序内存写入一个字节。写入数据前应擦除闪存位置。 0: 写入闪存程序内存禁用。 1: 写入闪存程序内存启用； MOVX 写入指令定位于闪存。



## SFR 定义 19.2: FLKEY: 闪存锁定和关键码寄存器

位	7	6	5	4	3	2	1	0
名称	FLKEY[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xB7

位	名称	功能
7:0	FLKEY[7:0]	<p><b>Fash 锁定和关键码寄存器。</b></p> <p><b>写:</b> 此寄存器提供闪存擦除和写入的锁定和关键码功能。闪存写入和擦除是通过向 FLKEY 寄存器中写入 0xA5 再写入 0xF1 启用。下一步写入或擦除完成后，闪存写入和擦除被自动禁用。如向 FLKEY 中的任何写入操作不正确，或在有关功能被禁用时执行闪存写入或擦除操作，闪存将被永久锁定，不能写入或擦除，直至下次设备复位。如某项应用无法写入闪存，则会通过自软件向 FLKEY 写入非 0xA5 值有意锁定闪存。</p> <p><b>读:</b> 读取时，1-0 位表明当前的闪存锁定状态。 00: 闪存为写入 / 擦除锁定。 01: 首位关键代码已被写入 (0xA5)。 10: 闪存解除锁定 (允许写入 / 擦除)。 11: 下次复位前，闪存写入 / 擦除被禁用。</p>

# C8051F80x-83x

---



## 20. 电源管理模式

C8051F80x-83x 设备有三种软件可编程的电源管理模式：空闲、停止及暂停。空闲模式和停止模式是标准 8051 结构的一部分，而暂停模式是由高度振荡器外围设备实施的一种增强型省电模式。

空闲模式下，CPU 暂停而外围设备和时钟激活。停止模式下，CPU 暂停、所有中断处理程序和定时器（缺失时钟探测器除外）均处于非激活状态，内部振荡器停止（模拟外围设备依然处于所选状态；外部振荡器不受影响）。暂停模式与停止模式类似，内部振荡器和 CPU 暂停，但在出现端口不匹配、比较器输出过低或定时器 3 溢出事件时便能唤醒有关设备。由于时钟在空闲模式下运行，决定电耗的是系统时钟的频率和进出空闲状态前处于激活状态的外围设备数目。停止模式和暂停模式耗电最少，因为大多数设备都被关闭，且没有时钟处于激活状态。SFR 定义 20.1 描述用于控制 C8051F80x-83x 停止和空闲电源管理模式的电源控制寄存器（PCON）。暂停模式由 OSCICN 寄存器中的暂停位控制（SFR 定义 22.3）。

尽管 C8051F80x-83x 有空闲、停止和暂停模式可供使用，可按需要通过启用 / 禁用单个外围设备，实现对设备电源的更多控制。每一个模拟外围设备在未被使用及处于低能模式时均可被禁用。数字外围设备（如定时器或串行总线）不使用时用电极少。关闭振荡器可显著降低电耗，其代价就是功能减少。

### 20.1. 空闲模式

设置空闲模式选择位（PCON.0），硬件让 CPU 暂停并在设置位的指令执行完成后立即进入空闲模式。所有内部寄存器和存储器均维持其原始数据。在空闲模式下，所有模拟和数字外围设备均可保持激活状态。

允许中断被判定或出现复位时，空闲模式即告终止。判定允许中断将引起空闲模式选择位（PCON.0）被清除及 CPU 重新运行。未决中断将得到执行，而从中断中回复后要执行的下个指令将是紧随设置空闲模式选择位之后的指令。如空闲模式是因内部或外部复位而终止，则 CIP-51 执行正常的复位流程，并在 0x0000 位置开始程序执行。

**附注：**如写入空闲位之后的指令是一个单字节指令，且中断出现在设置空闲位的指令的执行阶段，未来出现中断时 CPU 可能无法从空闲模式中唤醒。因此，设置空闲位的指令后面应该紧随一个两字节或更多字节的指令，例如：

```
// 用 'C' 语言：
PCON |= 0x01;           // 设置 IDLE 位
PCON = PCON;           // ... 跟随一条 3 周期伪指令

; 用汇编语言：
ORL PCON, #01h         ; 设置 IDLE 位
MOV PCON, PCON         ; ... 跟随一条 3 周期伪指令
```

如启用，看门狗（WDT）将最终引起内部看门狗复位，从而终止空闲模式。此特点保护系统在出现不慎写入 PCON 寄存器事件时免于被意外永久关机。如不需要此行为，而 WDT 最初被配置为允许此操作，可在进入空闲模式前通过软件禁用 WDT。这提供附加的省电机会，无限期允许系统保持空闲模式，等待外部刺激将其唤醒。参见第 236 页“29.4. 看门狗模式”一条，了解 WDT 使用及配置方面的更多资料。

## 20.2. 停止模式

设置停止模式选择位 (PCON.1)，在设置位的指令执行完成后立即让控制器内核进入停止模式。在停止模式下，内部振荡器、CPU 及所有数字外围设备均停止；外部振荡器电路的状态不受影响。每一个模拟外围设备（包括外部振荡器电路）在进入停止模式前可能分别关闭。停止模式仅能通过内部或外部复位终止。复位时，设备执行正常的复位流程，并在 0x0000 位置开始程序执行。

如启用，缺失时钟探测器将引起内部复位，从而终止停止模式。如 CPU 处于停止模式的时间将长于 100  $\mu$ s 的 MCD 超时，则应禁用缺失时钟探测器。

## 20.3. 暂停模式

暂停模式允许系统从内部振荡器进入类似停止模式的极低电耗状态，但若干状态下可无需将设备复位而唤醒处理器。选择暂停位 (OSCICN.5)，硬件让 CPU 及高频内部振荡器暂停，并在设置位的指令执行完成后立即进入暂停模式。所有内部寄存器和存储器均维持其原始数据。暂停模式下，多数数字外围设备均未激活。但端口匹配特征和定时器 3 从外部振荡器源运行则除外。

进入暂停模式时，在寄存器 CLKSEL 中的时钟除法器位 CLKDIV[2:0] 必须设置为“除以 1”。

可通过五类事件终止暂停模式：端口匹配（见第 150 页“23.5. 端口匹配”一条所述）、定时器 2 溢出（见第 219 页“28.2. 定时器 2”一条所述）、比较器输出过低（如启用）、电容传感下限比较器事件或设备复位事件。为在暂停模式下运行定时器 3，必须自外部时钟源将定时器配置为时钟。暂停模式终止后，该设备将继续按设置暂停位的后一个指令运行。如唤醒事件（端口匹配或定时器 2 溢出）被配置为产生中断，则唤醒设备时该中断获执行。如暂停模式是因内部或外部复位而终止，则 CIP-51 执行正常的复位流程，并在 0x0000 位置开始程序执行。

**注意：**如果唤醒源“挂起”，则设备仍会进入暂停模式，而设备不会唤醒这些挂起的唤醒源。必须确保设备进入暂停模式后，可以触发设定的唤醒源，这一点至关重要。例如，如果在设备进入暂停模式之前 CS0 转换结束且产生中断，则该中断不会触发唤醒事件。由于端口匹配事件有优先级之分，因此只要在设备进入暂停模式时仍存在匹配条件，先发生的端口匹配事件就会触发唤醒。

**SFR 定义 20.1: PCON: 电源控制寄存器**

位	7	6	5	4	3	2	1	0
名称	GF[5:0]						停止	空闲
类型	R/W						R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x87

位	名称	功能
7:2	GF[5:0]	<b>通用标志 5-0。</b> 存在在软件控制下使用的通用标志。
1	停止	<b>停止模式选择。</b> 设置此位会将 CIP-51 置于停止模式。此位将一直读作 0。 1: CPU 进入停止模式（内部振荡器停止）。
0	空闲	<b>空闲：空闲模式选择。</b> 设置此位会将 CIP-51 置于空闲模式。此位将一直读作 0。 1: CPU 进入空闲模式。（CPU 时钟关闭，但定时器、中断、串行端口和模拟外围设备的时钟依然处于激活状态。）

# C8051F80x-83x

---

## 21. 复位源

复位电路让控制器轻松位于预先定义的默认条件中。进入此复位状态时，会出现以下情况：

- CIP-51 暂停程序执行
- 特殊功能寄存器（SFR）初始化为设定的复位值
- 外部端口针脚被迫进入已知状态
- 中断和定时器被禁用。

所有 SFR 均复位到 SFR 详细说明中的预先定义值。复位时内部数据内存的内容不受影响；任何先前存储的数据均被保留。但由于堆栈指针特殊功能寄存器被复位，尽管堆栈上的数据未变更，堆栈实际上已经丢失。

端口 I/O 锁存器被复位到开路模式下的 0xFF（全部为逻辑 1）。复位期间和复位之后，弱上拉模式启用。对于 V<sub>DD</sub> 监视器和启动复位，RST<sup>–</sup> 针脚变为低电平，直至设备退出复位状态时为止。

退出重置状态后，程序计数器（PC）被重置，系统时钟默认到内部振荡器。用系统时钟除以 12 作为时钟源，便能启用看门狗。程序执行始于位置 0x0000。

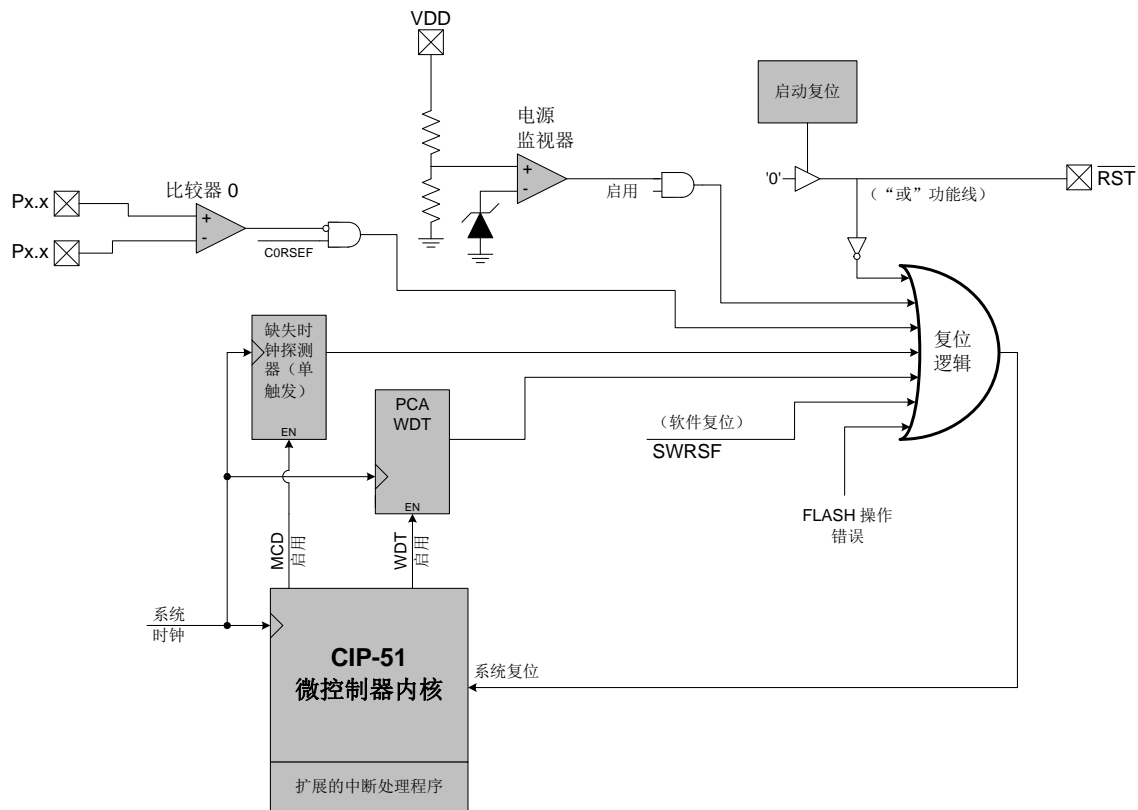


图 21.1. 复位源

## 21.1. 启动复位

在上电期间，设备处于复位状态， $\overline{\text{RST}}$  针脚处于低电平，直至  $V_{\text{DD}}$  上升至  $V_{\text{RST}}$  以上时为止。设备从复位状态退出复位状态前会出现延时，延时随  $V_{\text{DD}}$  上升时间的增加而减少（ $V_{\text{DD}}$  上升时间被定义为  $V_{\text{DD}}$  从 0 V 上升到  $V_{\text{RST}}$  的时间）。图 21.2. 是启动和  $V_{\text{DD}}$  监视器复位定时最大  $V_{\text{DD}}$  上升时间为 1 ms；上升时间过慢会导致设备在  $V_{\text{DD}}$  达到  $V_{\text{RST}}$  电平之前退出复位状态。当上升时间小于 1 ms 时，启动复位延时 ( $T_{\text{PORDelay}}$ ) 通常小于 10 ms。

在退出启动复位状态时，PORSF 标志 (RSTSRC.1) 被硬件设置为逻辑 1。当 PORSF 被置位时，RSTSRC 寄存器中的所有其他复位标志都是不确定的（PORSF 被所有其他复位清零）。由于所有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引起的复位。在一次启动复位后，内部数据内存中的内容应被认为是未定义的。将启用并选择  $V_{\text{DD}}$  监视器作为上电复位后的复位源。

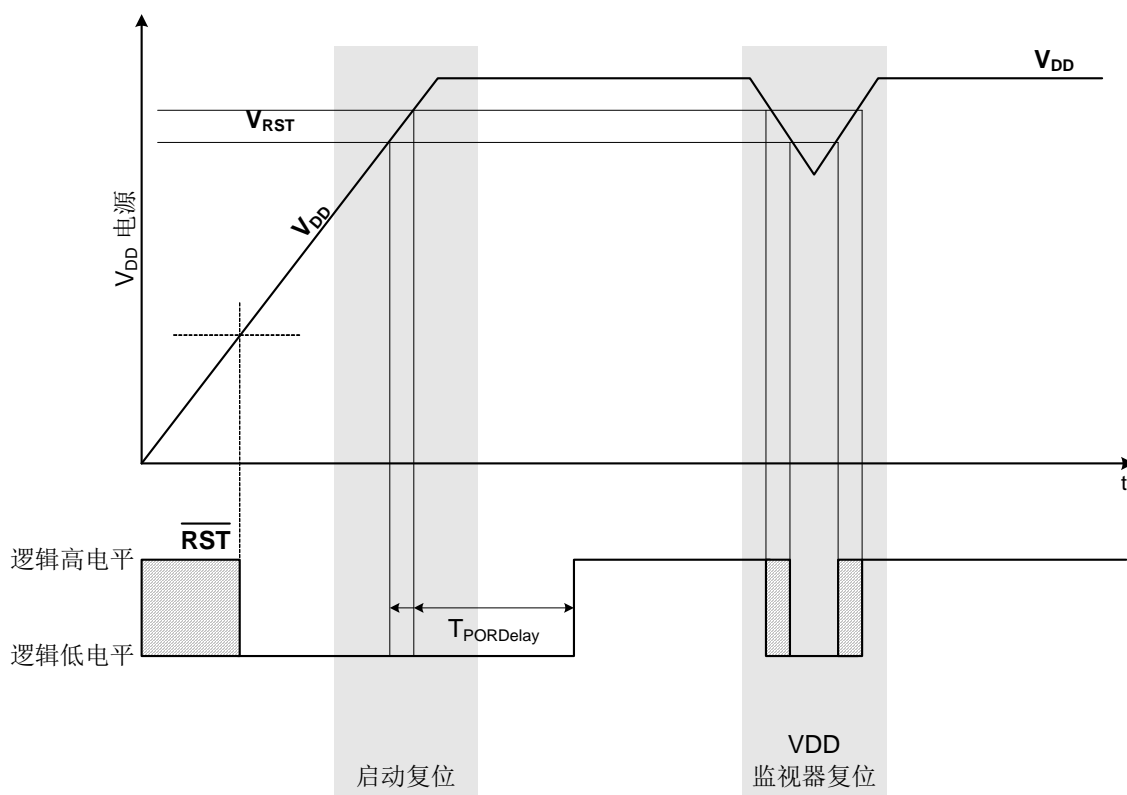


图 21.2. 启动和  $V_{\text{DD}}$  监视器复位定时



## 21.2. 掉电复位 / $V_{DD}$ 监视器

当发生掉电或因电源波动导致  $V_{DD}$  降到  $V_{RST}$  以下时，电源监视器将  $\overline{RST}$  针脚驱动为低电平并使 CIP-51 保持复位状态（见图 21.2）。当  $V_{DD}$  又返回高于  $V_{RST}$  的电平时，CIP-51 将退出复位状态。尽管内部数据内存的内容可能没有因掉电复位而发生改变，但无法确定  $V_{DD}$  是否降到了数据保持所要求的最低电平以下。如果  $PORSF$  标志的读数为 1，则数据可能不再有效。启动复位后， $V_{DD}$  监视器被启用。但其设定状态（启用 / 禁用）不受任何其他复位源影响。将启用并选择  $V_{DD}$  监视器作为上电复位后的复位源。

**重要事项：**如  $V_{DD}$  监视器被从禁用状态打开，则在选择其作为复位源之前，必须启用  $V_{DD}$  监视器。在  $V_{DD}$  监视器被启用或稳定之前选其为复位源可能导致系统复位。某些应用中，此复位可能属不良现象：如在有关应用中属于不良现象，则应在启用监视器和选其作为复位源之间插入延时。启用  $V_{DD}$  监视器并将其从禁用状态配置为复位源的操作步骤如下：

1. 启用  $V_{DD}$  监视器（ $VDMOCN$  中的  $VDMEN$  位 = 1）。
2. 如必要，等待  $V_{DD}$  监视器稳定。
3. 选择  $V_{DD}$  监视器作为复位源（ $RSTSRC$  中的  $PORSF$  位 = 1）。

图 21.2 给出了  $V_{DD}$  监视器的时序；注意，在  $V_{DD}$  监视器复位后没有上电复位延时。见第 39 页"7. 电气特性"一条，了解  $V_{DD}$  监视器的完整电气特性。

# C8051F80x-83x

## SFR 定义 21.1: VDM0CN: VDD 监视器控制寄存器

位	7	6	5	4	3	2	1	0
名称	VDMEN	VDDSTAT						
类型	R/W	R	R	R	R	R	R	R
复位	Varies	Varies	Varies	Varies	Varies	Varies	Varies	Varies

SFR 地址 = 0xFF

位	名称	功能
7	VDMEN	<b>V<sub>DD</sub> 监视器启用。</b> 此位控制 V <sub>DD</sub> 监视器电路的通断。V <sub>DD</sub> 监视器在寄存器 RSTSRC (SFR 定义 21.2) 中被选择为复位源之前不可能产生系统复位。在 V <sub>DD</sub> 监视器稳定之前选其为复位源可能导致系统复位。在此复位可能属于不良状态的系统中, 则应在启用 V <sub>DD</sub> 监视器和选其作为复位源之间插入延时。上电复位后, 将启用 V <sub>DD</sub> 监视器, 此位读数为 1。此位状态不受任何其他复位源影响。 0: V <sub>DD</sub> 监视器禁用。 1: V <sub>DD</sub> 监视器启用。
6	VDDSTAT	<b>V<sub>DD</sub> 状态。</b> 此位指示当前电源状态 (V <sub>DD</sub> 监视器输出)。 0: V <sub>DD</sub> 等于或低于 V <sub>DD</sub> 监视器阈值。 1: V <sub>DD</sub> 高于 V <sub>DD</sub> 监视器阈值。
5:0	未使用	读 = Varies; 写 = 无需在意。

### 21.3. 外部复位

外部  $\overline{\text{RST}}$  针脚提供了使用外部电路强制设备进入复位状态的手段。在  $\overline{\text{RST}}$  针脚上加一个低电平有效信号产生复位; 可能需要外部上拉及 / 或对  $\overline{\text{RST}}$  去耦以避免过强噪声引起复位。见第 39 页"7. 电气特性"一条, 了解完整  $\overline{\text{RST}}$  针脚规格。从外部复位退出后, PINRSF 标志 (RSTSRC.0) 被设置。

### 21.4. 缺失时钟探测器复位

缺失时钟探测器 (MCD) 是由系统时钟触发的单触法电路。如系统时钟保持在逻辑高电平或过逻辑低电平的时间超过 MCD 超时, 单稳态将超时并产生复位。MCD 复位后, MCDRSF 标志 (RSTSRC.2) 的读数为 1, 表示 MCD 为复位源; 否则, 该位读数为 0。向 MCDRSF 位写入 1 启动缺失时钟探测器; 写入 0 将禁用缺失时钟探测器。RST 针脚状态不受该复位影响。

## 21.5. 比较器 0 复位

向 CORSEF 标志 (RSTSRC.5) 写入 1, 可将比较器 0 配置为复位源。应在写入 CORSEF 前启用比较器 0 并等待输出稳定, 以防止通电瞬间在输出端产生抖动, 从而导致不必要的复位。比较器 0 复位为低态有效: 如果同相端输入电压 (CP0+) 小于反相端输入电压 (CP0-), 则设备被置于复位状态。比较器 0 复位后, CORSEF 标志 (RSTSRC.5) 的读数为 1, 表示比较器 0 为复位源; 否则该位读数为 0。RST 针脚状态不受该复位影响。

## 21.6. PCA 看门狗复位

可编程计数器阵列 (PCA) 的可编程看门狗 (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。可以通过软件启用或禁用 PCA 的 WDT 功能 (见第 236 页的"29.4. 看门狗模式"一条); 在每次复位后, WDT 被启用并使用 SYSCLK / 12 作为计数时钟。如系统故障阻止用户软件更新 WDT, 将产生一次复位且 WDTRSF 位 (RSTSRC.5) 设置为'1'。RST 针脚状态不受该复位影响。

## 21.7. 闪存故障复位

如闪存读取 / 写入 / 擦除或程序读取的目标为非法地址, 将产生系统复位。这可能是因以下任何一种情况导致:

- 闪存写入或擦除地址超出了用户代码空间。这种情况发生在 PSWE 设置为 1, 且 MOVX 写入操作的目标地址大于地址 0x3DFF。
- 闪存读取地址超出了用户代码空间。这种情况发生在 MOVC 操作目标地址大于地址 0x3DFF。
- 程序读取地址超出了用户代码空间。这种情况发生在用户代码试图转移到大于 0x3DFF 的地址时。
- 闪存的读取、写入或擦除被闪存安全设置限制 (见第 114 页"19.3. 安全选项"一条)。

闪存故障复位后, FERROR 位 (RSTSRC.6) 被设置。RST 针脚状态不受该复位影响。

## 21.8. 软件复位

软件可通过向 SWRSF 位 (RSTSRC.4) 写入 1 进行强制复位。在软件强制复位后, SWRSF 位的读数为 1。RST 针脚状态不受该复位影响。

# C8051F80x-83x

## SFR 定义 21.2: RSTSRC: 复位源

位	7	6	5	4	3	2	1	0
名称		FERROR	CORSEF	SWRSF	WDTRSF	MCDRSF	PORSF	PINRSF
类型	R	R	R/W	R/W	R	R/W	R/W	R
复位	0	变化	变化	变化	变化	变化	变化	变化

SFR 地址 = 0xEF

位	名称	说明	写入	读取
7	未使用	未使用。	无需在意。	0
6	FERROR	闪存故障复位标志。	不可用	如闪存读取 / 写入 / 擦除故障导致上次复位, 设置为 1。
5	CORSEF	比较器 0 复位启用及标志。	写入 1 启用作为复位源 (低态有效) 的比较器 0。	如比较器 0 导致上次复位, 设置为 1。
4	SWRSF	软件复位力及标志。	写入 1 强制系统复位。	如写入 SWRSF 导致上次复位, 设置为 1。
3	WDTRSF	看门狗复位标志。	不可用	如看门狗溢出导致上次复位, 设置为 1。
2	MCDRSF	缺失时钟探测器启用及标志。	写入 1 启用缺失时钟探测器。如检测到缺失时钟条件, MCD 触发复位。	如缺失时钟探测器超时导致上次复位, 设置为 1。
1	PORSF	启动 / V <sub>DD</sub> 监视器复位标志, 且 V <sub>DD</sub> 监视器复位启用。	写入 1 启用作为复位源的 V <sub>DD</sub> 监视器。 在 V <sub>DD</sub> 监视器启用并稳定前在有关位数中写入 1 会导致系统复位。	开机或 V <sub>DD</sub> 监视器复位时, 设置为 1。 设置为 1 时, 所有其他 RSTSRC 标志是不确定的。
0	PINRSF	HW 针脚复位标志。	不可用	如 RST 针脚导致上次复位, 设置为 1。

附注: 请勿在该寄存器上使用读取 - 调节 - 写入操作。

## 22. 振荡器和时钟选择寄存器

C8051F80x-83x 设备包括可编程的内部高频振荡器以及外部振荡器驱动电路。可用 OSCICN 及 OSCICL 寄存器启用 / 禁用及校准内部高频振荡器（如图 22.1）。可通过外部振荡器电路或内部振荡器（默认）获取系统时钟。内部振荡器提供可选的后分频功能，其初始设置为时钟除以 8。

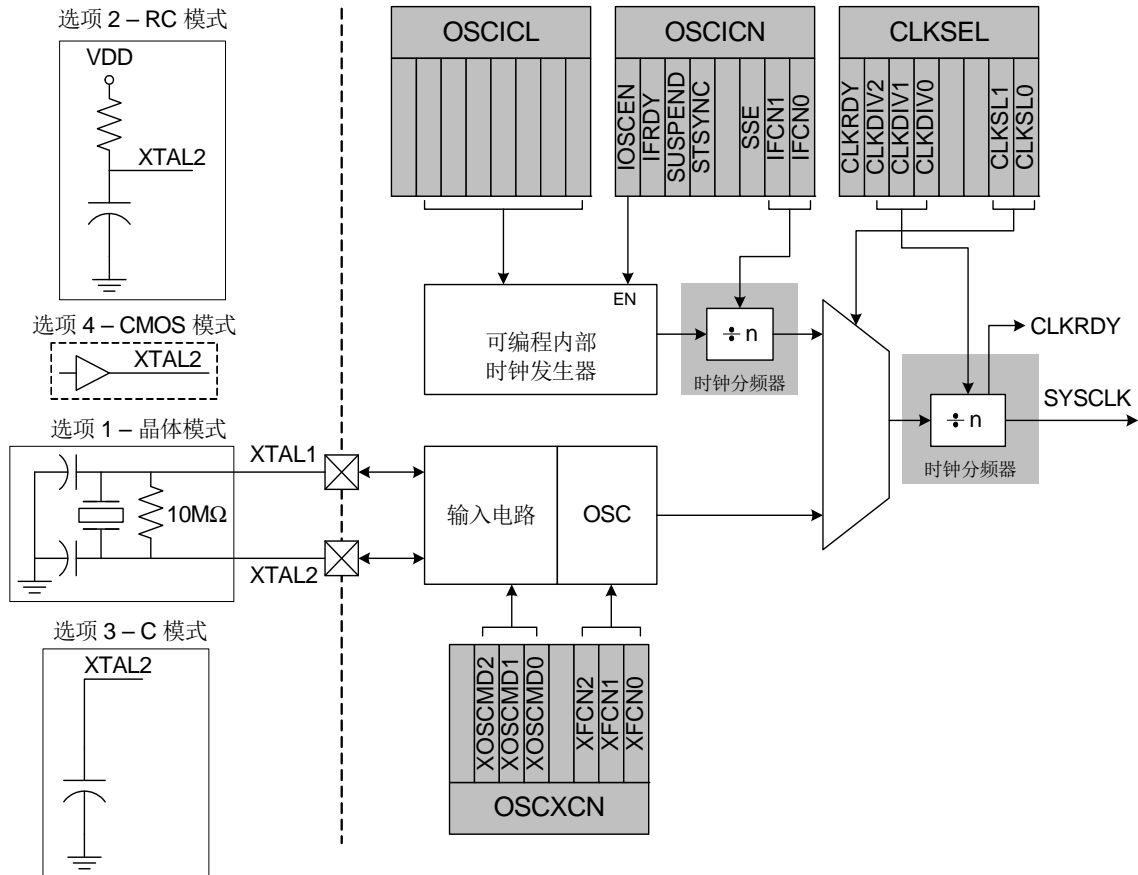


图 22.1. 振荡器选项

### 22.1. 系统时钟选择

可使用 CLKSEL 寄存器选择 MCU 系统时钟源。时钟被选为系统时钟后，可除以 1、2、4、8、16、32、64 或 128。当在两个时钟除数值间转换时，跳变可占用未分频时钟源的 128 次循环。查询 CLKRDY 标志，确定何时可用新时钟除数。进入暂停模式后，时钟分频器必须设置为“除以 1”。亦可接通正在运行的系统时钟源。在较慢的振荡器运行一个时钟周期后，转换生效。

# C8051F80x-83x

## SFR 定义 22.1: CLKSEL: 时钟选择

位	7	6	5	4	3	2	1	0
名称	CLKRDY	CLKDIV[2:0]				CLKSEL[2:0]		
类型	R	R/W	R/W	R/W	R	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xA9

位	名称	功能
7	CLKRDY	<b>系统时钟分频器时钟就绪标志。</b> 0: 选定的时钟分频设置尚未应用于系统时钟。 1: 选定的时钟分频设置已应用于系统时钟。
6:4	CLKDIV	<b>系统时钟分频器位。</b> 选择将应用于选定源（内部或外部）的时钟分频。 000: 选定时钟除以 1。 001: 选定时钟除以 2。 010: 选定时钟除以 4。 011: 选定时钟除以 8。 100: 选定时钟除以 16。 101: 选定时钟除以 32。 110: 选定时钟除以 64。 111: 选定时钟除以 128。
3	未使用	读 = 0b。必须写入 0b。
2:0	CLKSEL[2:0]	<b>系统时钟选择。</b> 选择将用作未分频系统时钟源的振荡器。 000: 内部振荡器 001: 外部振荡器  保留所有其他值。

## 22.2. 可编程内部高频 (H-F) 振荡器

所有 C8051F80x-83x 设备均包括一个可编程内部高频振荡器，其在系统复位后被默认为系统时钟。可通过 OSCICL 寄存器（定义见 SFR 定义 22.2）调整内部振荡器周期。

在 C8051F80x-83x 设备上，OSCICL 为工厂校准，以获取 24.5 兆赫基本频率。

内部振荡器输出频率可除以 1、2、4 或 8（定义见寄存器 OSCICN 中的 IFCN 位）。复位后，除数默认为 8。

为减少系统产生的电磁干扰（EMI），精确振荡器支持可调制输出频率的扩频模式。启用时（SSE = 1），通过梯形三角波调制振荡器输出频率，三角波频率 = 振荡器频率 / 384（工厂校准为 63.8 千赫）。中心频率的最大偏差值为 ±0.75%。输出频率每 32 次循环更新一次，且步长一般为中心频率的 0.25%。

### SFR 定义 22.2: OSCICL: 内部 H-F 振荡器校准

位	7	6	5	4	3	2	1	0
名称	OSCICL[6:0]							
类型	R/W							
复位	变化	变化	变化	变化	变化	变化	变化	变化

SFR 地址 = 0xB3

位	名称	功能
6:0	OSCICL[7:0]	<b>内部振荡器校准位</b> 这些位确定了内部振荡器周期。当设置为 00000000b 时，H-F 振荡器以其设置的最快速率运转。当设置为 11111111b 时，H-F 振荡器在以其设置的最慢速率运转。复位值为工厂校准，以产生 24.5 兆赫的内部振荡器频率。

# C8051F80x-83x

## SFR 定义 22.3: OSCICN: 内部 H-F 振荡器控制

位	7	6	5	4	3	2	1	0
名称	IOSCEN	IFRDY	SUSPEND	STSYNC	SSE		IFCN[1:0]	
类型	R/W	R	R/W	R	R/W	R	R/W	
复位	1	1	0	0	0	0	0	0

SFR 地址 = 0xB2

位	名称	功能
7	IOSCEN	<b>内部 H-F 振荡器启用位。</b> 0: 内部 H-F 振荡器禁用。 1: 内部 H-F 振荡器启用。
6	IFRDY	<b>内部 H-F 振荡器频率就绪标志</b> 0: 内部 H-F 振荡器尚未以编程频率运行。 1: 内部 H-F 振荡器正在以编程频率运行。
5	SUSPEND	<b>内部振荡器暂停启用位。</b> 设置此位至逻辑 1，将内部振荡器设为暂停模式。一旦发生暂停模式唤醒事件，内部振荡器将恢复操作。
4	STSYNC	<b>暂停定时器同步位。</b> 此位用于表示何时可安全读取及写入与暂停唤醒定时器有关的寄存器。如暂停唤醒源（除定时器 2）使振荡器脱离暂停模式，在定时器可读取或写入前，该源会占用三个定时器时钟。 0: 定时器 2 寄存器可安全读取。 1: 不应读取或写入定时器 2 寄存器。
3	SSE	<b>扩频启用。</b> 扩频启用位。 0: 扩频时钟抖动禁用。 1: 扩频时钟抖动启用。
2	未使用	读 = 0b; 写 = 无需在意
1:0	IFCN[1:0]	<b>内部 H-F 振荡器分频器控制位。</b> 00: 源自内部 H-F 振荡器的 SYSCLK 除以 8。 01: 源自内部 H-F 振荡器的 SYSCLK 除以 4。 10: 源自内部 H-F 振荡器的 SYSCLK 除以 2。 11: 源自内部 H-F 振荡器的 SYSCLK 除以 1。



## 22.3. 外部振荡器驱动电路

外部振荡器电路能驱动外部晶体、陶瓷谐振器、电容器或 RC 网络。CMOS 时钟也能提供时钟输入。就晶体或陶瓷谐振器配置而言，晶体 / 谐振器必须接线穿过 XTAL1 及 XTAL2 针脚（如图 22.1 选项 1 所示）。须将 10M 电阻器接线穿过 XTAL1 及 XTAL2 针脚，以便进行晶体/谐振器配置。在 RC、谐振器或 CMOS 时钟配置中，时钟源应连接至 XTAL2 针脚（如图 22.1 选项 2、3 或 4 所示）。须在 OSCXCN 寄存器中选择外部振荡器类型，且须选择适当的频率控制位 (XFCN)（请参阅 SFR 定义 22.4）。

**使用外部振荡器的重要事项：**当使用外部振荡器电路时，须配置端口针脚。当外部振荡器驱动电路启用晶体 / 谐振器模式时，端口针脚 P0.2 及 P0.3 分别用作 XTAL1 及 XTAL2。当外部振荡器驱动电路启用电容器、RC 或 CMOS 时钟模式时，端口针脚 P0.3 用作 XTAL2。端口 I/O 交叉开关须配置为可跳过振荡器电路使用的端口针脚（欲知交叉开关配置，请参阅 143 页“23.3 优先级交叉开关编码”一节）。此外当外部振荡器电路启用晶体 / 谐振器、电容器或 RC 模式中时，相关端口针脚应配置为**模拟输入**。CMOS 时钟模式中，相关针脚应配置为**数字输入**（欲知端口输入模式选择详情，请参阅 147 页“23.4. 端口 I/O 初始化”一节）。

# C8051F80x-83x

## SFR 定义 22.4: OSCXCN: 外部振荡器控制寄存器

位	7	6	5	4	3	2	1	0
名称	XTLVLD	XOSCMD[2:0]				XFCN[2:0]		
类型	R	R/W			R	R/W		
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xB1

位	名称	功能																																				
7	XTLVLD	<b>晶体振荡器有效标志</b> (仅当 XOSCMD = 11x 时读取) 0: 晶体振荡器未使用或不稳定。 1: 晶体振荡器稳定运行。																																				
6:4	XOSCMD[2:0]	<b>外部振荡器模式选择。</b> 000x: 外部振荡器电路关闭。 010: 外部 CMOS 时钟模式。 011: 外部 CMOS 时钟模式分为两个阶段。 100: RC 振荡器模式。 101: 电容器振荡器模式。 110: 晶体振荡器模式。 111: 晶体振荡器模式分为两个阶段。																																				
3	未使用	读 = 0; 写 = 无需在意																																				
2:0	XFCN[2:0]	<b>外部振荡器频率控制位。</b> 根据晶体或 RC 模式的适宜频率进行设置。 根据 C 模式的适宜 K 系数进行设置。																																				
		<table border="1"> <thead> <tr> <th>XFCN</th> <th>晶体模式</th> <th>RC 模式</th> <th>C 模式</th> </tr> </thead> <tbody> <tr> <td>000</td> <td><math>f \leq 32</math> 千赫</td> <td><math>f \leq 25</math> 千赫</td> <td>K 系数 = 0.87</td> </tr> <tr> <td>001</td> <td><math>32 \text{ 千赫} &lt; f \leq 84 \text{ 千赫}</math></td> <td><math>25 \text{ 千赫} &lt; f \leq 50 \text{ 千赫}</math></td> <td>K 系数 = 2.6</td> </tr> <tr> <td>010</td> <td><math>84 \text{ 千赫} &lt; f \leq 225 \text{ 千赫}</math></td> <td><math>50 \text{ 千赫} &lt; f \leq 100 \text{ 千赫}</math></td> <td>K 系数 = 7.7</td> </tr> <tr> <td>011</td> <td><math>225 \text{ 千赫} &lt; f \leq 590 \text{ 千赫}</math></td> <td><math>100 \text{ 千赫} &lt; f \leq 200 \text{ 千赫}</math></td> <td>K 系数 = 22</td> </tr> <tr> <td>100</td> <td><math>590 \text{ 千赫} &lt; f \leq 1.5 \text{ 兆赫}</math></td> <td><math>200 \text{ 千赫} &lt; f \leq 400 \text{ 千赫}</math></td> <td>K 系数 = 65</td> </tr> <tr> <td>101</td> <td><math>1.5 \text{ 兆赫} &lt; f \leq 4 \text{ 兆赫}</math></td> <td><math>400 \text{ 千赫} &lt; f \leq 800 \text{ 千赫}</math></td> <td>K 系数 = 180</td> </tr> <tr> <td>110</td> <td><math>4 \text{ 兆赫} &lt; f \leq 10 \text{ 兆赫}</math></td> <td><math>800 \text{ 千赫} &lt; f \leq 1.6 \text{ 兆赫}</math></td> <td>K 系数 = 664</td> </tr> <tr> <td>111</td> <td><math>10 \text{ 兆赫} &lt; f \leq 30 \text{ 兆赫}</math></td> <td><math>1.6 \text{ 兆赫} &lt; f \leq 3.2 \text{ 兆赫}</math></td> <td>K 系数 = 1590</td> </tr> </tbody> </table>	XFCN	晶体模式	RC 模式	C 模式	000	$f \leq 32$ 千赫	$f \leq 25$ 千赫	K 系数 = 0.87	001	$32 \text{ 千赫} < f \leq 84 \text{ 千赫}$	$25 \text{ 千赫} < f \leq 50 \text{ 千赫}$	K 系数 = 2.6	010	$84 \text{ 千赫} < f \leq 225 \text{ 千赫}$	$50 \text{ 千赫} < f \leq 100 \text{ 千赫}$	K 系数 = 7.7	011	$225 \text{ 千赫} < f \leq 590 \text{ 千赫}$	$100 \text{ 千赫} < f \leq 200 \text{ 千赫}$	K 系数 = 22	100	$590 \text{ 千赫} < f \leq 1.5 \text{ 兆赫}$	$200 \text{ 千赫} < f \leq 400 \text{ 千赫}$	K 系数 = 65	101	$1.5 \text{ 兆赫} < f \leq 4 \text{ 兆赫}$	$400 \text{ 千赫} < f \leq 800 \text{ 千赫}$	K 系数 = 180	110	$4 \text{ 兆赫} < f \leq 10 \text{ 兆赫}$	$800 \text{ 千赫} < f \leq 1.6 \text{ 兆赫}$	K 系数 = 664	111	$10 \text{ 兆赫} < f \leq 30 \text{ 兆赫}$	$1.6 \text{ 兆赫} < f \leq 3.2 \text{ 兆赫}$	K 系数 = 1590
XFCN	晶体模式	RC 模式	C 模式																																			
000	$f \leq 32$ 千赫	$f \leq 25$ 千赫	K 系数 = 0.87																																			
001	$32 \text{ 千赫} < f \leq 84 \text{ 千赫}$	$25 \text{ 千赫} < f \leq 50 \text{ 千赫}$	K 系数 = 2.6																																			
010	$84 \text{ 千赫} < f \leq 225 \text{ 千赫}$	$50 \text{ 千赫} < f \leq 100 \text{ 千赫}$	K 系数 = 7.7																																			
011	$225 \text{ 千赫} < f \leq 590 \text{ 千赫}$	$100 \text{ 千赫} < f \leq 200 \text{ 千赫}$	K 系数 = 22																																			
100	$590 \text{ 千赫} < f \leq 1.5 \text{ 兆赫}$	$200 \text{ 千赫} < f \leq 400 \text{ 千赫}$	K 系数 = 65																																			
101	$1.5 \text{ 兆赫} < f \leq 4 \text{ 兆赫}$	$400 \text{ 千赫} < f \leq 800 \text{ 千赫}$	K 系数 = 180																																			
110	$4 \text{ 兆赫} < f \leq 10 \text{ 兆赫}$	$800 \text{ 千赫} < f \leq 1.6 \text{ 兆赫}$	K 系数 = 664																																			
111	$10 \text{ 兆赫} < f \leq 30 \text{ 兆赫}$	$1.6 \text{ 兆赫} < f \leq 3.2 \text{ 兆赫}$	K 系数 = 1590																																			

## 22.3.1. 外部晶体范例

如晶体或陶瓷谐振器用作 MCU 的外部振荡器源，电路应按图 22.1 选项 1 所示进行配置。外部振荡器频率控制值 (XFCN) 应从 SFR 定义 22.4 (OSCXCN 寄存器) 所载表格的晶体一栏中选择。例如，11.0592 兆赫的晶体要求 111b 的 XFCN 设置，而 32.768 千赫表面晶体要求 001b 的 XFCN 设置。待外部 32.768 千赫的振荡器稳定后，XFCN 设置可转换为 000 以节省电量。建议在系统时钟转换为任何外部振荡器源前，启用缺失时钟探测器。

首次启用晶体振荡器时，振荡器振幅探测电路需要一定的设置时间，以达到合适偏压。若在启用振荡器与检查 XTLVLD 位之间介入 1 毫秒延时时间，将会阻止系统时钟过早转换外部振荡器。在晶体振荡器稳定前转换外部振荡器会导致不可预测的反应。建议的程序如下：

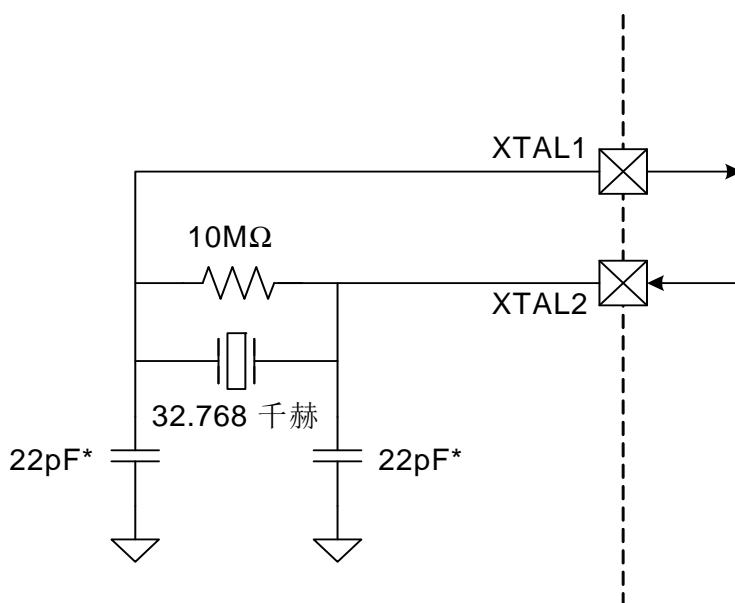
1. 强制 XTAL1 及 XTAL2 处于低状态。这包括启用交叉开关，并在 XTAL1 及 XTAL2 的关联端口引脚写入 0
2. 将 XTAL1 及 XTAL2 配置为模拟输入。
3. 启用外部振荡器。
4. 至少等待 1 毫秒。
5. 查询 XTLVLD = 1。
6. 如需要，启用缺失时钟探测器。
7. 将系统时钟转换至外部振荡器。

**有关外部晶体的重要事项：**晶体振荡器电路对 PCB 布局非常敏感。晶体放置的位置应尽可能靠近设备的 XTAL 引脚处。迹线应尽可能短，并受到屏蔽面的保护，远离可能产生噪音或干扰的任何其他迹线。

外部晶体配置所示电容器提供了晶体进行适当震荡所需的负载电容。上述电容器对晶体为"串联"设置，并与 XTAL1 及 XTAL2 引脚的寄生电容"并联"。

**附注：**所需负载电容取决于晶体及制造商。完成上述计算后，请参考晶体数据表。

例如，建议负载电容为 12.5 pF 的 32.768 千赫音叉晶体的应使用图 22.1 选项 1 所示的配置。电容器与 XTAL 引脚寄生电容的总值应等于 25 pF。寄生电容为每引脚 3 pF 的负载电容 22 pF 电容器通过晶体获得 12.5 pF 的等效电容（如图 22.2）。



\* 电容器值取决于晶体规格。

图 22.2. 外部 32.768 千赫石英晶体振荡器接线图

### 22.3.2. 外部 RC 范例

如 RC 网络用于 MCU 的外部振荡器源，电路应按图 22.1 选项 2 所示进行配置。电容器不应超过 100 pF；但对微型电容器而言，总电容可能以 PBC 配置图中的寄生电容为主。为确定 OSCXCN 寄存器所需的外部振荡器频率控制值 (XFCN)，根据方程式 22.1（其中  $f$  = 振荡器频率（兆赫）， $C$  = 电容值（pF），及  $R$  = 上拉寄存器值(k)），选择 RC 网络值，产生振荡器所需频率。

#### 方程 22.1: RC 模式振荡器频率

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

例如：如所需频率为 100 千赫，使  $R = 246 \text{ k}$  且  $C = 50 \text{ pF}$ ：

$$f = 1.23 (10^3) / RC = 1.23 (10^3) / [246 \times 50] = 0.1 \text{ 兆赫} = 100 \text{ 千赫}$$

参考 SFR 定义 22.4 中的表格，所需 XFCN 设置为 010b。

### 22.3.3. 外部电容器范例

如电容器用于 MCU 的外部振荡器，电路应按图 22.1 选项 3 所示进行配置。电容器不应超过 100 pF；但对微型电容器而言，总电容可能以 PCB 配置图中的寄生电容为主。为确定 OSCXCN 寄存器所需的外部振荡器频率控制值 (XFCN)，首先根据方程式 22.2（其中  $f$  = 振荡器频率（兆赫）， $C$  = 电容值（pF），及  $V_{DD}$  = MCU 供电（伏）），选择将使用的电容器，并找出振荡器频率。

#### 方程 22.2: C 模式振荡器频率

$$f = (KF)/(R \times V_{DD})$$

例如：假设  $V_{DD} = 3.0$  伏 且  $f = 150$  千赫：

$$f = KF / (C \times V_{DD})$$

$$0.150 \text{ 兆赫} = KF / (C \times 3.0)$$

由于所需频率大约为 150 千赫，从 SFR 定义 22.4 的表格中选择 K 系数。  
(OSCXCN) 当  $KF = 22$  时：

$$0.150 \text{ 兆赫} = 22 / (C \times 3.0)$$

$$C \times 3.0 = 22 / 0.150 \text{ 兆赫}$$

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

因此，本范例中所用的 XFCN 值为 011b，且  $C = 50$  pF。

# C8051F80x-83x

---

## 23. 端口输入/输出

数字及模拟源在 17 个 I/O 针脚（24 - 针脚及 20 - 针脚封装）或 13 个 I/O 针脚（16 - 针封装）中可用。端口针脚 P0.0-P1.7 可定义为通用 I/O (GPIO)，或分配至内部数字源之一（如图 23.4）。端口针脚 P2.0 可用作 GPIO，与 C2 接口数据信号 (C2D) 共用。设计者可完全控制功能分配，仅受限于实体 I/O 针脚数量。通过使用优先级交叉开关编码器，可实现源的灵活分配。请注意端口 I/O 针脚状态可在相应的端口锁存器中随时读取，无需理会交叉开关设置。

交叉开关根据优先级编码器，将所选内部数字源分配至 I/O 针脚（图 23.5）。可使用寄存器 XBR0 与 XBR1（定义见 SFR 定义 23.1 及 SFR 定义 23.2）选择内部数字功能。

所有端口 I/O 可承受 5 伏电量（欲知端口单元电路，请参照图 23.2）。在端口输出模式寄存器 (PnMDOUT，其中 n=0,1) 中，端口 I/O 单元的配置为推挽式或漏极开路。端口 I/O 的完整电气规格载于第 39 页的“7. 电气特性”一节。

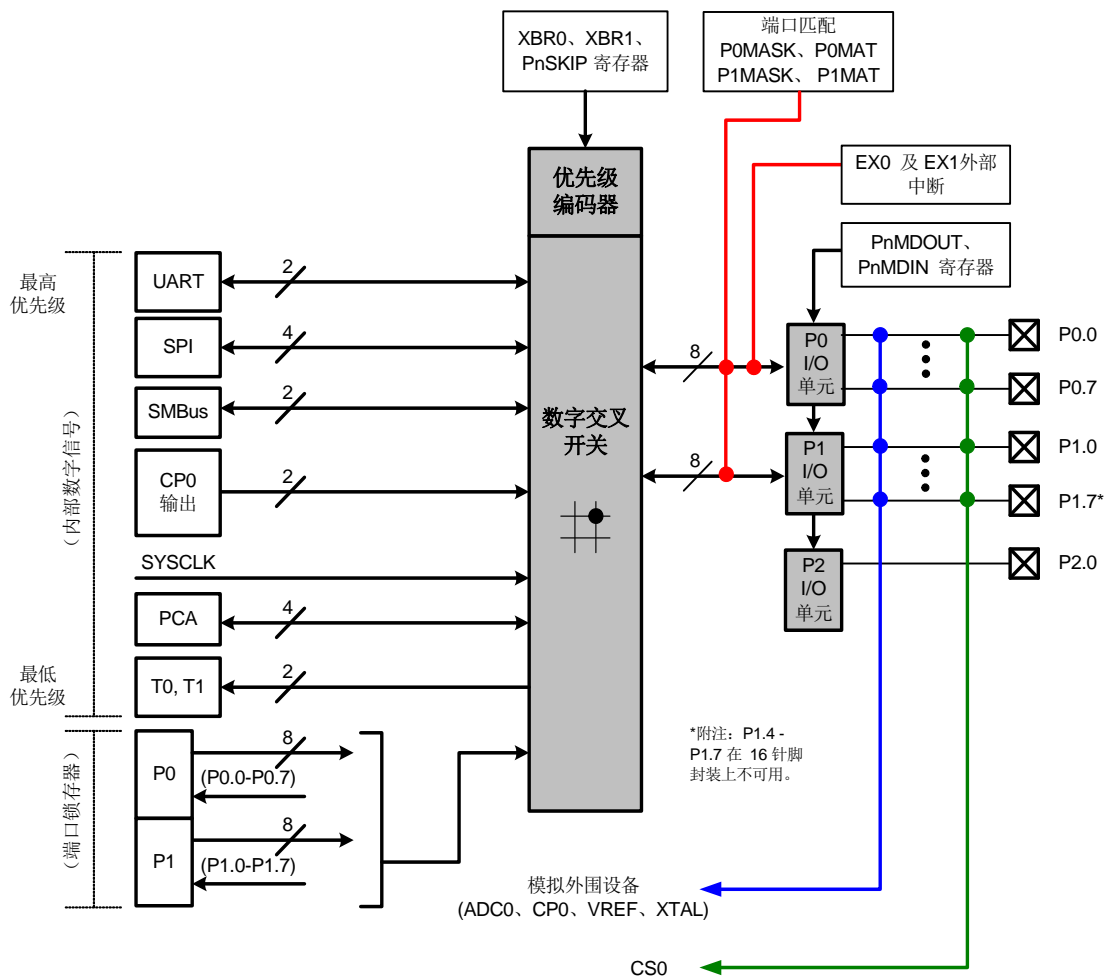


图 23.1. 端口 I/O 功能结构图

## 23.1. 端口 I/O 模式操作

端口引脚 P0.0-P1.7 使用图 23.2 所示的端口 I/O 单元。可使用 PnMDIN 与 PnMDOUT 寄存器通过软件为模拟 I/O 或数字 I/O 配置每个端口 I/O 单元。可使用 P2MDOUT 寄存器通过软件为数字 I/O 配置端口引脚 P2.0。复位时，所有端口 I/O 单元默认为高阻态，弱上拉模式启用。在交叉开关启用前 (XBARE = 1)，在所有交叉开关引脚上均明确禁用高低端口 I/O 驱动电路。

### 23.1.1. 为模拟 I/O 配置端口引脚

应为模拟 I/O 配置任何用作比较器或 ADC 输入、电容传感输入、外部振荡器输入 / 输出、VREF 输出或者 AGND 连接的任何引脚 (PnMDIN.n = 0、Pn.n = 1)。当为模拟 I/O 配置引脚时，其弱上拉模式、数字驱动和数字接收器均被禁用。为防止低端口 I/O 驱动电路将引脚拉低，应在相应的端口锁存器 (Pn.n = 1) 写入 '1'。为模拟 I/O 配置的端口引脚将随时读回 0 值，不论引脚上的实际电压为多少。

将引脚设置为模拟 I/O 可节省电量，并使端口引脚免受数字干扰。仍可通过模拟外围设备使用配置为数字 I/O 的端口引脚；然而由于此操作可能导致测量失误，因此并不建议如此行事。

### 23.1.2. 为数字 I/O 配置端口引脚

数字外围设备 (UART、SPI、SMBus 等)、外部数字时间捕捉功能所使用的任何引脚，或者用作 GPIO 的任何引脚，都应被配置为数字 I/O (PnMDIN.n = 1)。对于数字 I/O 引脚，须使用 PnMDOUT 寄存器选择两种输出模式 (推挽或漏极开路) 的其中之一。

推挽输出 (PnMDOUT.n = 1) 根据端口引脚的输出逻辑值，将端口板驱动至 VDD 或 GND 供电轨。漏极开路输出使高压侧驱动器禁用；因此相关输出仅在输出逻辑值为 0 时将端口板驱动至 GND，且当输出逻辑值为 1 时成为高阻抗输入 (高低驱动器均关闭)。

当数字 I/O 单元为高阻态时，弱上拉晶体管将端口板拉至 VDD 供电电压，以确保数字输入处于已定义逻辑状态。为降低功耗，当将 I/O 单元驱动至 GND 时弱上拉模式被禁用，且若 WEAKPUD 被设置为 1，弱上拉将被全部禁用。用户应确保数字 I/O 始终在内部或外部被上拉或驱动至有效逻辑状态，以降低功耗。为数字 I/O 配置的端口引脚会随时读回端口板的逻辑状态，不论端口引脚的输出逻辑值为多少。

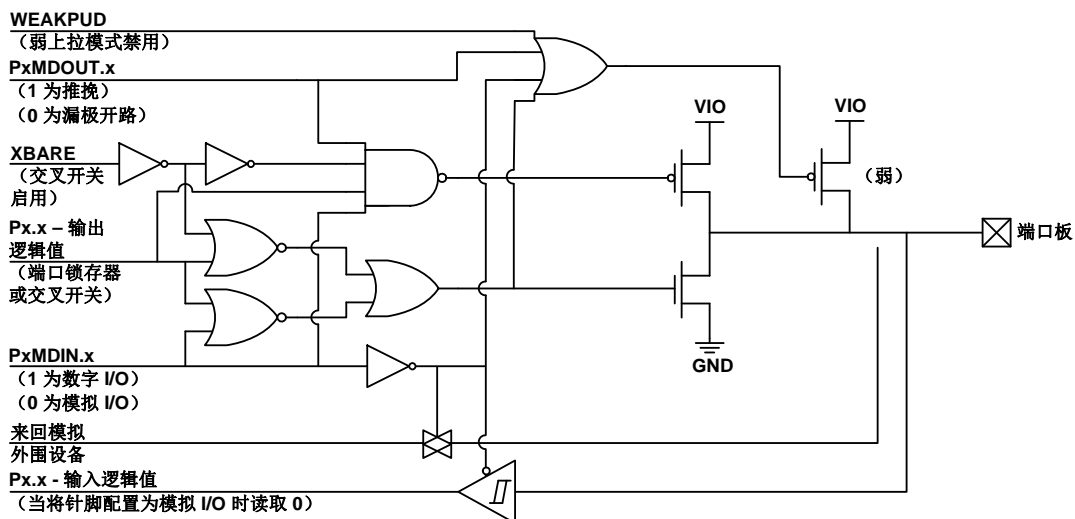


图 23.2. 端口 I/O 单元结构图



### 23.1.3. 连接端口 I/O 至 5 伏逻辑

为数字漏极开路操作配置的所有端口 I/O 能够以高于 2 伏（高于 VDD）低于 5.25 伏的供电电压，连接至数字逻辑操作。大部分系统通常需要能达到较高供电电压的外部上拉电阻器。

**重要事项：**在多极电压接口中，当供电电压介于（VDD + 0.6 伏）及（VDD + 1.0 伏）之间时，应调整外部上拉电阻器，让至少 150 微安的电流流入端口针脚。一旦端口针脚电压超出此范围，流入端口针脚的电流将非常微弱。图 23.3 显示了超出 VDD 的端口针脚的输入电流特性。当端口针脚电压达到约（VDD + 0.7 伏）时，其需要 150 微安峰值超速电流。

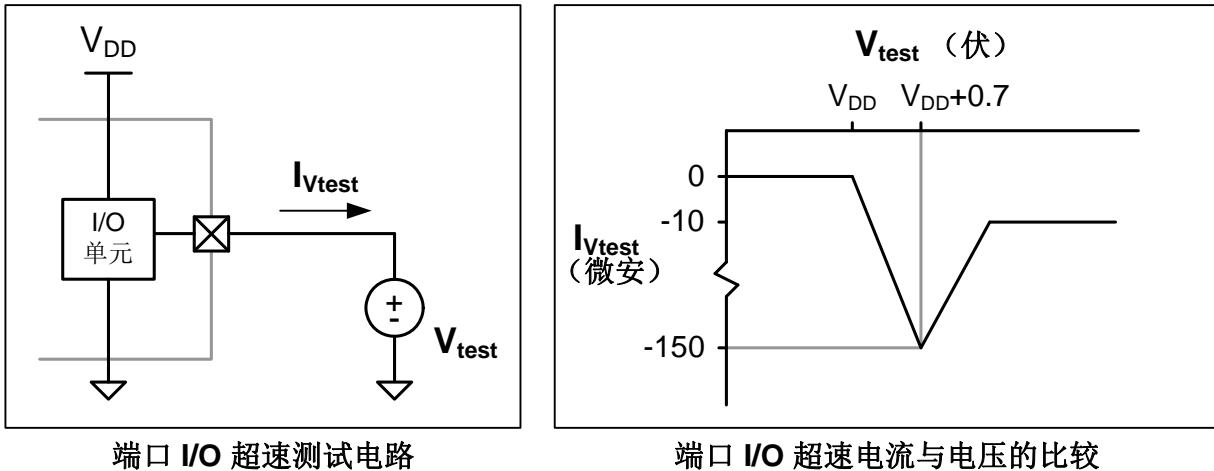


图 23.3. 端口 I/O 超速电流

## 23.2. 分配端口 I/O 针脚至模拟和数字功能

可将端口 I/O 针脚 P0.0-P1.7 分配至各种模拟、数字及外部中断功能。应将分配至模拟功能的端口针脚配置给模拟 I/O，并且将分配至数字或外部中断功能的端口针脚配置给数字 I/O。

### 23.2.1. 分配端口 I/O 针脚至模拟功能

表 23.1 显示了需要端口 I/O 分配的所有可用模拟功能。为上述模拟功能选择的端口针脚应在设置为 1 的 PnSKIP 中拥有相应位。这样可保留模拟功能使用的针脚，并防止其被交叉开关占用。任何选定针脚亦应在设置为 1 (Pn.n = 1) 的端口锁存器中拥有相应位。这可防止低端口 I/O 驱动电路将针脚拉低。表 23.1 显示了端口 I/O 与各个模拟功能的潜在映射。

表 23.1: 端口 I/O 模拟功能分配

模拟功能	潜在可分配的端口引脚	用于分配的 SFR
ADC 输入	P0.0–P1.7	ADC0MX、PnSKIP、PnMDIN
比较器 0 输入	P0.0–P1.7	CPT0MX、PnSKIP、PnMDIN
CS0 输入	P0.0–P1.7	CS0MX、CS0SS、CS0SE、PnMDIN
电压基准 (VREF0)	P0.0	REF0CN、P0SKIP、PnMDIN
接地基准 (AGND)	P0.1	REF0CN、P0SKIP
晶体模式中的外部振荡器 (XTAL1)	P0.2	OSCXCN、P0SKIP、P0MDIN
RC、C 或晶体模式 (XTAL2)中的外部振荡器	P0.3	OSCXCN、P0SKIP、P0MDIN

### 23.2.2. 分配端口 I/O 引脚至数字功能

可将未分配至模拟功能的任何端口引脚分配至数字功能，或用作 GPIO。多数数字功能依赖交叉开关进行引脚分配；然而，某些数字功能会以类似上述模拟功能的方式绕过交叉开关。**这些数字功能使用的端口引脚，以及被选用为 GPIO 的任何端口引脚，都应在设置为 1 的 PnSKIP 中拥有相应位。**表 23.2 显示了所有可用数字功能，及端口 I/O 与各个数字功能的潜在映射。

表 23.2: 端口 I/O 数字功能分配

数字功能	潜在可分配的端口针脚	用于分配的 SFR
UART0、SPI0、SMBus、SYSCLK、PCA0 (CEX0-2 及 ECI)、T0 或 T1。	任何可由交叉开关分配的端口针脚。其中包括 P0.0 - P1.7 <sup>2</sup> 针脚，其 PnSKIP 位设置为 0。 <sup>1</sup>	XBR0、XBR1
用作 GPIO 的任何针脚	P0.0–P2.0 <sup>2</sup>	PnSKIP
附注： 1. 交叉开关将随时把 UART0 针脚分配至 P0.4 及 P0.5。 2. 端口针脚 P1.4 - P1.7 在 16 针封装上不可用。		

### 23.2.3. 分配端口 I/O 针脚至外部数字事件捕捉功能

当数字 I/O 针脚发生转变时，外部数字事件捕捉功能可用来触发中断，或将设备从低电能模式唤醒。数字事件捕捉功能不需要专用针脚，且在 GPIO 针脚 (PnSKIP = 1) 及交叉开关所用针脚 (PnSKIP = 0) 上都可运行。外部数字事件捕捉功能不能在为模拟 I/O 配置的针脚上使用。表 23.3 显示了所有可用的外部数字事件捕捉功能。

表 23.3: 端口 I/O 外部数字事件捕捉功能分配

数字功能	潜在可分配的端口针脚	用于分配的 SFR
外部中断 0	P0.0–P0.7	IT01CF
外部中断 1	P0.0–P0.7	IT01CF
端口匹配	P0.0–P1.7 <sup>*</sup>	P0MASK、P0MAT P1MASK、P1MAT
附注：端口针脚 P1.4 - P1.7 在 16 针封装上不可用。		

## 23.3. 优先级交叉开关编码器

优先级交叉开关编码器将优先级分配至各个 I/O 功能，从 UART0 最高处开始。当数字源选定后，将最低有效未分配端口针脚分配至该源（UART0 除外，其随时处于针脚 4 与 5）。如端口已分配，当分配下一个选定源时，交叉开关会跳过该针脚。此外，交叉开关将跳过在 PnSKIP 寄存器中已设置相关位的端口针脚。PnSKIP 寄存器允许软件跳过用于模拟输入、专用功能或 GPIO 的端口针脚。

因为优先级交叉开关编码器的性质，不是所有外围设备都可在所有端口针脚上定位。图 23.4 将外围设备和可能出现外围设备 I/O 的潜在端口针脚相对应。

**交叉开关配置重要事项：**如外围设备占用了端口针脚，而不使用交叉开关，应设置其相应的 PnSKIP 位。这适用于 P0.0（如使用 VREF）、P0.1（如使用 AGND）、P0.3 及/或 P0.2（如启用外部振荡器电路）、P0.6（如将 ADC 配置为使用外部转换启动信号 (CNVSTR)）以及任何已选 ADC、比较器或电容传感输入。交叉开关跳过已选针脚，犹如该等针脚已分配，并移至下一未分配针脚。

使用寄存器 XBR0、XBR1 及 XBR2 将数字 I/O 源分配至物理 I/O 端口针脚。请注意，当选择 SMBus 时，交叉开关会分配与 SMBus（SDA 及 SCL）有关联的针脚；当选择 UART 时，交叉开关会分配与 UART（TX 及 RX）有关联的针脚。UART0 针脚的分配是为加载而设定：UART TX0 将始终分配至 P0.4；UART RX0 将始终分配至 P0.5。优先功能分配完后，标准端口 I/O 将连续出现。

**重要事项：**可在 3 线或 4 线模式中操作 SPI，这取决于寄存器 SPI0CN 中 NSSMD1-NSSMD0 位的状态。根据 SPI 模式决定是否可将 NSS 信号改为端口针脚。

端口	P0							P1							P2		
引脚编号	0	1	2	3	4	5	6	7	0	1	2	3	4 <sup>1</sup>	5 <sup>1</sup>	6 <sup>1</sup>	7 <sup>1</sup>	0
特殊功能信号	VREF	AGND	XTAL1	XTAL2			CNVSTR										
TX0																	
RX0																	
SCK																	
MISO																	
MOSI																	
NSS <sup>2</sup>																	
SDA																	
SCL																	
CP0																	
CP0A																	
SYSCLK																	
CEX0																	
CEX1																	
CEX2																	
ECI																	
T0																	
T1																	
引脚跳过设置	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	P0SKIP							P1SKIP									

信号不可用于交叉开关

可将引脚 P0.0-P1.7<sup>1</sup> 分配至交叉开关外围设备。

交叉开关外围设备根据本时间图按优先级别从最高到最低进行分配。

- 以上方框代表可被潜在地分配至外围设备的端口引脚。
- 特殊功能信号不由交叉开关分配。当这些信号启用时，应手动配置交叉开关，跳过相应端口引脚。
- 通过在 PnSKIP 中将相应位设置为 '1'，可“跳过”引脚。

附注：  
 1. P1.4 - P1.7 在 16 针封装上不可用。  
 2. 当 SPI 处于 4 线模式时，NSS 仅为引脚输出。

图 23.4. 优先级交叉开关编码器潜在引脚分配

# C8051F80x-83x

端口	P0							P1							P2		
引脚编号	0	1	2	3	4	5	6	7	0	1	2	3	4 <sup>1</sup>	5 <sup>1</sup>	6 <sup>1</sup>	7 <sup>1</sup>	0
特殊功能信号	VREF	AGND	XTAL1	XTAL2			CNVSTR										
TX0					■												
RX0						■											
SCK	■																
MISO		■															
MOSI			■														
NSS <sup>2</sup>				■													
SDA																	
SCL																	
CP0																	
CP0A																	
SYSCLK																	
CEX0							■										
CEX1								■									
CEX2									■								
ECI																	
T0																	
T1																	
引脚跳过设置	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	P0SKIP							P1SKIP									

信号不可用于交叉开关

此例中，交叉开关被配置为分配 UART TX0 及 RX0 信号、SPI 信号以及 PCA 信号。请注意，SPI 信号被分配为多重信号，且使用 P0SKIP 或 P1SKIP 寄存器时未跳过引脚。

■ 以上方框代表本配置中外围设备所用的端口引脚。

第一 TX0 被分配至 P0.4  
 第二 RX0 被分配至 P0.5  
 第三 SCK、MISO、MOSI 及 NSS 分别被分配至 P0.0、P0.1、P0.2 及 P0.3。  
 第四 CEX0、CEX1 及 CEX2 分别被分配至 P0.6、P0.7 及 P1.0。

所有未分配引脚均可用作 GPIO，或用于其他非交叉开关功能。

附注：  
 1. P1.4 - P1.7 在 16 针封装上不可用。  
 2. 当 SPI 处于 4 线模式时，NSS 仅为引脚输出。

图 23.5. 优先级交叉开关编码器范例 1—无跳过引脚

端口	P0							P1							P2		
引脚编号	0	1	2	3	4	5	6	7	0	1	2	3	4 <sup>1</sup>	5 <sup>1</sup>	6 <sup>1</sup>	7 <sup>1</sup>	0
特殊功能信号	VREF	AGND	XTAL1	XTAL2			CNVSTR										
TX0					■												
RX0						■											
SCK	■																
MISO							■										
MOSI								■									
NSS <sup>2</sup>									■								
SDA																	
SCL																	
CP0																	
CP0A																	
SYSCLK																	
CEX0										■							
CEX1											■						
CEX2												■					
ECI																	
T0																	
T1																	
引脚跳过设置	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	P0SKIP							P1SKIP									

信号不可用P0.0跳过于交叉开关

此例中，交叉开关被配置为分配 UART TX0 及 RX0 信号、SPI 信号以及 PCA 信号。请注意，SPI 信号被分配为多重信号，且使用 P0SKIP 或 P1SKIP 寄存器时未跳过引脚。此外，使用 P0SKIP 寄存器时，引脚 P0.0、P0.2 及 P0.3 被配置为跳过。

■ 以上方框代表本配置中外围设备所用的端口引脚。

第一 TX0 被分配至 P0.4  
 第二 RX0 被分配至 P0.5  
 第三 SCK、MISO、MOSI 及 NSS 分别被分配至 P0.1、P0.6、P0.7 及 P1.0。  
 第四 CEX0、CEX1 及 CEX2 分别被分配至 P1.1、P1.2 及 P1.3。

所有未分配引脚（包括 XBR0 跳过的引脚）均可用作 GPIO，或用于其他非交叉开关功能。

附注：  
 1. P1.4 - P1.7 在 16 针封装上不可用。  
 2. 当 SPI 处于 4 线模式时，NSS 仅为引脚输出。

图 23.6. 优先级交叉开关编码器范例 2—跳过引脚

## 23.4. 端口 I/O 初始化

端口 I/O 初始化包括以下步骤：

1. 使用端口输入模式寄存器 (PnMDIN)，为所有端口针脚选择输入模式（模拟或数字）。如针脚为模拟模式，还须在相应的端口锁存器 (Pn) 上写入 '1'。
2. 使用端口输出模式寄存器 (PnMDOUT)，为所有端口针脚选择输出模式（漏极开路或推挽）。
3. 使用端口跳过寄存器 (PnSKIP)，选择 I/O 交叉开关将跳过的任何针脚。
4. 为拟定的外围设备指定端口针脚（XBR0、XBR1）。
5. 启用交叉开关 (XBARE = 1)。

所有端口针脚均须配置为模拟或数字输入。针脚配置为模拟输入后，其弱上拉模式、数字驱动和数字接收器均被禁用。此过程节省能源并降低模拟输入的噪音。模拟外围设备仍可使用配置为数字输入的针脚；但是并不建议进行此项操作。

此外，所有模拟输入针脚应配置为可被交叉开关跳过（通过设置 PnSKIP 中的相关位完成）。PnMDIN 寄存器中设置为端口输入模式，其中 1 表示数字输入，0 表示模拟输入。所有模拟模式的端口针脚必须在相应的端口锁存寄存器上设置为 '1'。复位后所有针脚默认为数字输入。欲知 PnMDIN 寄存器详情，请参阅 SFR 定义 23.8 和 SFR 定义 23.12。

I/O 针脚的输出驱动特性定义为使用端口输出模式寄存器 (PnMDOUT)。各端口输出驱动可配置为漏极开路或推挽。由于该选项并非自动选项，因此必须选择此选项，即便是 XBRn 寄存器中所选的数字源亦然。仅 SMBus (SDA、SCL) 针脚除外，无论 PnMDOUT 如何设置，该针脚均配置为漏极开路。XBR1 中的 WEAKPUD 位为 0 时，所有配置为漏极开路的端口 I/O 均启用弱上拉模式。WEAKPUD 并不影响推挽端口 I/O。此外，配置为 0 的输出上的弱上拉模式被关闭，避免不必要的功率损耗。

寄存器 XBR0 和 XBR1 必须负载适当数值，以选择设计所需的数字 I/O 功能。将 XBR1 中的 XBARE 位设置为 1，启用交叉开关。交叉开关启动前，外部针脚始终维持为标准端口 I/O（输入模式），而不论 XBRn 寄存器如何设置。对于给定的 XBRn 寄存器设置，可使用优先解码表确定 I/O 针脚输出；作为替代方法，配置向导 (Configuration Wizard) 实用程序将根据 XBRn 寄存器设置确定端口 I/O 针脚分配。

必须启用交叉开关，以将端口针脚用作输出模式中的标准端口 I/O。禁用交叉开关后，端口输出驱动被禁用。



## SFR 定义 23.1: XBR0: 端口 I/O 交叉开关寄存器 0

位	7	6	5	4	3	2	1	0
名称			CP0AE	CP0E	SYSCKE	SMB0E	SPI0E	URT0E
类型	R	R	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xE1

位	名称	功能
7:6	未使用	读 = 00b。写 = 无需在意。
5	CP0AE	<b>比较器 0 异步输出启用。</b> 0: 异步 CP0 在端口引脚上不可用。 1: 异步 CP0 改为端口引脚。
4	CP0E	<b>比较器 0 输出启用。</b> 0: CP0 在端口引脚上不可用。 1: CP0 改为端口引脚。
3	SYSCKE	<b>SYSCCLK 输出启用。</b> 0: $\overline{\text{SYSCCLK}}$ 在端口引脚上不可用。 1: SYSCCLK 输出改为端口引脚。
2	SMB0E	<b>SMBus I/O 启用。</b> 0: SMBus I/O 在端口引脚上不可用。 1: SMBus I/O 改为端口引脚。
1	SPI0E	<b>SPI I/O 启用。</b> 0: SPI I/O 在端口引脚上不可用。 1: SPI I/O 改为端口引脚。请注意，SPI 可配置为 3 或 4 GPIO 引脚。
0	URT0E	<b>UART I/O 输出启用。</b> 0: UART I/O 在端口引脚上不可用。 1: UART TX0、RX0 改为端口引脚 P0.4 和 P0.5。

# C8051F80x-83x

## SFR 定义 23.2: XBR1: 端口 I/O 交叉开关寄存器 1

位	7	6	5	4	3	2	1	0
名称	WEAKPUD	XBARE	T1E	T0E	ECIE		PCA0ME[1:0]	
类型	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xE2

位	名称	功能
7	WEAKPUD	<b>Port I/O 弱上拉模式禁用。</b> 0: 弱上拉模式启用 (I/O 配置为模拟模式的端口除外)。 1: 弱上拉模式禁用。
6	XBARE	<b>交叉开关启用。</b> 0: 交叉开关禁用。 1: 交叉开关启用。
5	T1E	<b>T1 启用。</b> 0: T1 在端口针脚上不可用。 1: T1 改为端口针脚。
4	T0E	<b>T0 启用。</b> 0: T0 在端口针脚上不可用。 1: T0 改为端口针脚。
3	ECIE	<b>PCA0 外部计数器输入启用。</b> 0: ECI 在端口针脚上不可用。 1: ECI 改为端口针脚。
2	未使用	读 = 0b; 写 = 无需在意。
1:0	PCA0ME[1:0]	<b>PCA 模块 I/O 启用位。</b> 00: 所有 PCA I/O 在端口针脚上不可用。 01: CEX0 改为端口针脚。 10: CEX0、CEX1 改为端口针脚。 11: CEX0、CEX1、CEX2 改为端口针脚。

---

## 23.5. 端口匹配

端口匹配功能允许 P0 或 P1 上的逻辑值改变触发系统事件。存储在 PnMATCH 寄存器上的软件控制值指明 P0 和 P1 的预期或正常逻辑值。如端口输入针脚的逻辑电平与软件控制值不匹配，则会发生端口不匹配事件。这可在 P0 或 P1 输入针脚发生某种改变或模式时，使软件获得通知，而不论 XBRn 如何设置。

PnMASK 寄存器可用于个别的选择使用何种 P0 和 P1 针脚来和 PnMATCH 寄存器相比较。如  $(P0 \& P0MASK)$  并不等于  $(P0MATCH \& P0MASK)$  或如  $(P1 \& P1MASK)$  不等于  $(P1MATCH \& P1MASK)$ ，则会发生端口不匹配事件。

端口不匹配事件可用于在低电能模式中中断或唤醒设备，如空闲或暂停模式。如需更多有关中断和唤醒源，请参阅"中断和电能选项"章节。

# C8051F80x-83x

## SFR 定义 23.3: P0MASK: 端口 0 屏蔽寄存器

位	7	6	5	4	3	2	1	0
名称	P0MASK[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xFE

位	名称	功能
7:0	P0MASK[7:0]	<b>端口 0 屏蔽值。</b> 选择 P0 引脚与 P0MAT 中相应位作比较。 0: P0.n 引脚逻辑值被忽略, 不会引致端口不匹配事件。 1: P0.n 引脚逻辑值与 P0MAT.n 作比较。

## SFR 定义 23.4: P0MAT: 端口 0 匹配寄存器

位	7	6	5	4	3	2	1	0
名称	P0MAT[7:0]							
类型	R/W							
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0xFD

位	名称	功能
7:0	P0MAT[7:0]	<b>端口 0 匹配值。</b> 将用在 P0MASK (设置为 1) 中的位上的端口 0 的比较值相匹配。 0: P0.n 引脚逻辑值与逻辑低电平作比较。 1: P0.n 引脚逻辑值与逻辑高电平作比较。

**SFR 定义 23.5: P1MASK: 端口 1 屏蔽寄存器**

位	7	6	5	4	3	2	1	0
名称	P1MASK[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xEE

位	名称	功能
7:0	P1MASK[7:0]	<p><b>端口 1 屏蔽值。</b></p> <p>选择 P1 针脚与 P1MAT 中相应位作比较。</p> <p>0: P1.n 针脚逻辑值被忽略, 不会引致端口不匹配事件。</p> <p>1: P1.n 针脚逻辑值与 P1MAT.n 作比较。</p> <p>附注: P1.4 - P1.7 在 16 针封装上不可用。</p>

**SFR 定义 23.6: P1MAT: 端口 1 匹配寄存器**

位	7	6	5	4	3	2	1	0
名称	P1MAT[7:0]							
类型	R/W							
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0xED

位	名称	功能
7:0	P1MAT[7:0]	<p><b>端口 1 匹配值。</b></p> <p>将用在 P1MASK (设置为 1) 中的位上的端口 1 的比较值相匹配。</p> <p>0: P1.n 针脚逻辑值与逻辑低电平作比较。</p> <p>1: P1.n 针脚逻辑值与逻辑高电平作比较。</p> <p>附注: P1.4 - P1.7 在 16 针封装上不可用。</p>

**23.6. 访问和配置端口 I/O 的特殊功能寄存器**

所有端口 I/O 均可通过相应的特殊功能寄存器 (SFR) 进行访问, 特殊功能寄存器可按字节和位寻址。写入端口时, 写入 SFR 的值被锁定, 以维持各针脚的输出数据值。读取时, 无论 XBRn 如何设置, 端口输入针脚的逻辑电平均会返回 (即, 即使交叉开关将针脚分配至另一信号, 端口寄存器依然可读取相应的端口 I/O 针脚)。但执行读取 - 调节 - 写入指示是例外, 该指示将端口锁存寄存器列为目标。在端口 SFR 上运行时, 读取 - 调节 - 写入指示如下: ANL、ORL、XRL、JBC、CPL、INC、DEC、DJNZ 和 MOV、CLR 或 SETB (当目标为端口 SFR 上的单个位)。对上述指示而言, 读取、调节锁存寄存器 (不是针脚) 的值并写回 SFR。

# C8051F80x-83x

各个端口拥有相应的 PnSKIP 寄存器，该寄存器可允许其单个端口引脚分配至数字功能或被交叉开关跳过。所有用于模拟功能或 GPIO 的端口引脚应拥有设为 1 的 PnSKIP 位。

I/O 引脚的端口输入模式通过使用端口输入模式存取器 (PnMDIN) 界定。各个端口单元可为模拟或数字 I/O 进行配置。由于该选项并非自动选项，因此必须选择此选项，即便是 XBRn 寄存器中所选的数字源亦然。但 P2.0 是例外，P2.0 仅可用于数字 I/O。

I/O 引脚的输出驱动特性定义为使用端口输出模式寄存器 (PnMDOUT)。各端口输出驱动可配置为漏极开路或推挽。由于该选项并非自动选项，因此必须选择此选项，即便是 XBRn 寄存器中所选的数字源亦然。仅 SMBus (SDA、SCL) 引脚除外，无论 PnMDOUT 如何设置，该引脚均配置为漏极开路。

## SFR 定义 23.7: P0: 端口 0

位	7	6	5	4	3	2	1	0
名称	P0[7:0]							
类型	R/W							
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0x80; 可位寻址

位	名称	说明	写入	读取
7:0	P0[7:0]	<b>端口 0 数据。</b> 设置端口锁存逻辑值或读取为数字 I/O 配置的端口单元内的端口引脚逻辑状态。	<b>0:</b> 将输出锁存设置为逻辑低电平。 <b>1:</b> 将输出锁存设置为逻辑高电平。	<b>0:</b> P0.n 端口引脚为逻辑低电平。 <b>1:</b> P0.n 端口引脚为逻辑高电平。

**SFR 定义 23.8: P0MDIN: 端口 0 输入模式**

位	7	6	5	4	3	2	1	0
名称	P0MDIN[7:0]							
类型	R/W							
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0xF1

位	名称	功能
7:0	P0MDIN[7:0]	<p><b>P0.7-P0.0 的模拟配置位（分别）。</b></p> <p>为模拟模式配置的端口引脚禁用弱上拉模式、数字驱动和数字接收器。为将 P0.n 引脚设为模拟模式，与该引脚相对应的端口锁存寄存器上须设置为'1'。</p> <p>0: 为模拟模式配置有相应的 P0.n 引脚。</p> <p>1: 未为模拟模式配置相应的 P0.n 引脚。</p>

**SFR 定义 23.9: P0MDOUT: 端口 0 输出模式**

位	7	6	5	4	3	2	1	0
名称	P0MDOUT[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xA4

位	名称	功能
7:0	P0MDOUT[7:0]	<p><b>P0.7-P0.0 的输出配置位（分别）。</b></p> <p>如寄存器 P0MDIN 中相应位为逻辑 0，上述位则被忽略。</p> <p>0: 相应的 P0.n 输出为漏极开路。</p> <p>1: 相应的 P0.n 输出为推挽。</p>

# C8051F80x-83x

## SFR 定义 23.10: POSKIP: 端口 0 跳过寄存器

位	7	6	5	4	3	2	1	0
名称	POSKIP[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xD4

位	名称	功能
7:0	POSKIP[7:0]	<p><b>端口 0 交叉开关跳过启用位。</b></p> <p>上述位选择端口 0 被交叉开关编码器跳过。模拟、特殊功能或 GPIO 的端口引脚应被交叉开关跳过。</p> <p>0: 相应 P0.n 引脚未被交叉开关跳过。</p> <p>1: 相应 P0.n 引脚被交叉开关跳过。</p>

## SFR 定义 23.11: P1: 端口 1

位	7	6	5	4	3	2	1	0
名称	P1[7:0]							
类型	R/W							
复位	1	1	1	1	1	1	1	1

SFR 地址 = 0x90; 可位寻址

位	名称	说明	写入	读取
7:0	P1[7:0]	<p><b>端口 1 数据。</b></p> <p>设置端口锁存逻辑值或读取为数字 I/O 配置的端口单元内的端口引脚逻辑状态。</p> <p><b>附注:</b> P1.4 - P1.7 在 16 针封装上不可用。</p>	<p>0: 将输出锁存设置为逻辑低电平。</p> <p>1: 将输出锁存设置为逻辑高电平。</p>	<p>0: P1.n 端口引脚为逻辑低电平。</p> <p>1: P1.n 端口引脚为逻辑高电平。</p>



**SFR 定义 23.12: P1MDIN: 端口 1 输入模式**

位	7	6	5	4	3	2	1	0
名称	P1MDIN[7:0]							
类型	R/W							
复位	1*	1*	1*	1*	1	1	1	1

SFR 地址 = 0xF2

位	名称	功能
7:0	P1MDIN[7:0]	<p><b>P1.7-P1.0 的模拟配置位（分别）。</b></p> <p>为模拟模式配置的端口引脚禁用弱上拉模式、数字驱动和数字接收器。为将 P1.n 引脚设为模拟模式，与该引脚相应的端口锁存寄存器上须设置为 1。</p> <p>0: 相应 P1.n 引脚配置为模拟模式。</p> <p>1: 相应 P1.n 引脚未配置为模拟模式。</p> <p>附注: P1.4 - P1.7 在 16 针封装上不可用，P1MDIN[7:4] 的复位值为 0000b。</p>

**SFR 定义 23.13: P1MDOUT: 端口 1 输出模式**

位	7	6	5	4	3	2	1	0
名称	P1MDOUT[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xA5

位	名称	功能
7:0	P1MDOUT[7:0]	<p><b>P1.7-P1.0 的输出配置位（分别）。</b></p> <p>如寄存器 P1MDIN 中相应位为逻辑 0，上述位则被忽略。</p> <p>0: 相应的 P1.n 输出为漏极开路。</p> <p>1: 相应的 P1.n 输出为推挽。</p> <p>附注: P1.4 - P1.7 在 16 针封装上不可用。</p>

# C8051F80x-83x

## SFR 定义 23.14: P1SKIP: 端口 1 跳过寄存器

位	7	6	5	4	3	2	1	0
名称	P1SKIP[7:0]							
类型	R/W							
复位	0*	0*	0*	0*	0	0	0	0

SFR 地址 = 0xD5

位	名称	功能
7:0	P1SKIP[7:0]	<p><b>端口 1 交叉开关跳过启用位。</b></p> <p>上述位选择端口 1 针脚被交叉开关编码器跳过。模拟、特殊功能或 GPIO 的端口针脚应被交叉开关跳过。</p> <p>0: 相应 P1.n 针脚未被交叉开关跳过。</p> <p>1: 相应 P1.n 针脚被交叉开关跳过。</p> <p>附注: P1.4 - P1.7 在 16 针封装上不可用, P1SKIP[7:4] 的复位值为 1111b。</p>

## SFR 定义 23.15: P2: 端口 2

位	7	6	5	4	3	2	1	0
名称								P2[0]
类型	R	R	R	R	R	R	R	R/W
复位	0	0	0	0	0	0	0	1

SFR 地址 = 0xA0; 可位寻址

位	名称	说明	写入	读取
7:1	未使用	未使用。	无需在意	0000000b
0	P2[0]	<p><b>端口 2 数据。</b></p> <p>设置端口锁存逻辑值或读取为数字 I/O 配置的端口单元内的端口针脚逻辑状态。</p>	<p>0: 将输出锁存设置为逻辑低电平。</p> <p>1: 将输出锁存设置为逻辑高电平。</p>	<p>0: P2.0 端口针脚为逻辑低电平。</p> <p>1: P2.0 端口针脚为逻辑高电平。</p>

## SFR 定义 23.16: P2MDOUT: 端口 2 输出模式

位	7	6	5	4	3	2	1	0
名称								P2MDOUT[0]
类型	R	R	R	R	R	R	R	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xA6

位	名称	功能
7:1	未使用	读 = 0000000b; 写 = 无需在意。
0	P2MDOUT[0]	<b>P2.0 的输出配置位。</b> 0: P2.0 输出为漏极开路。 1: P2.0 输出为推挽。

# C8051F80x-83x

---

## 24. 循环冗余检验单位 (CRC0)

C8051F80x-83x 设备包括一个可使用 16 位或 32 位多项式执行 CRC 的循环冗余检验单位 (CRC0)。CRC0 接受写入 CRC0IN 寄存器的 8 位数据。CRC0 将 16 位或 32 位结果记入内部寄存器。可使用 CRC0PNT 位和 CRC0DAT 寄存器（如图 24.1 所示），间接访问内部结果寄存器。CRC0 还包括进行快速数据操作的位元反转寄存器。

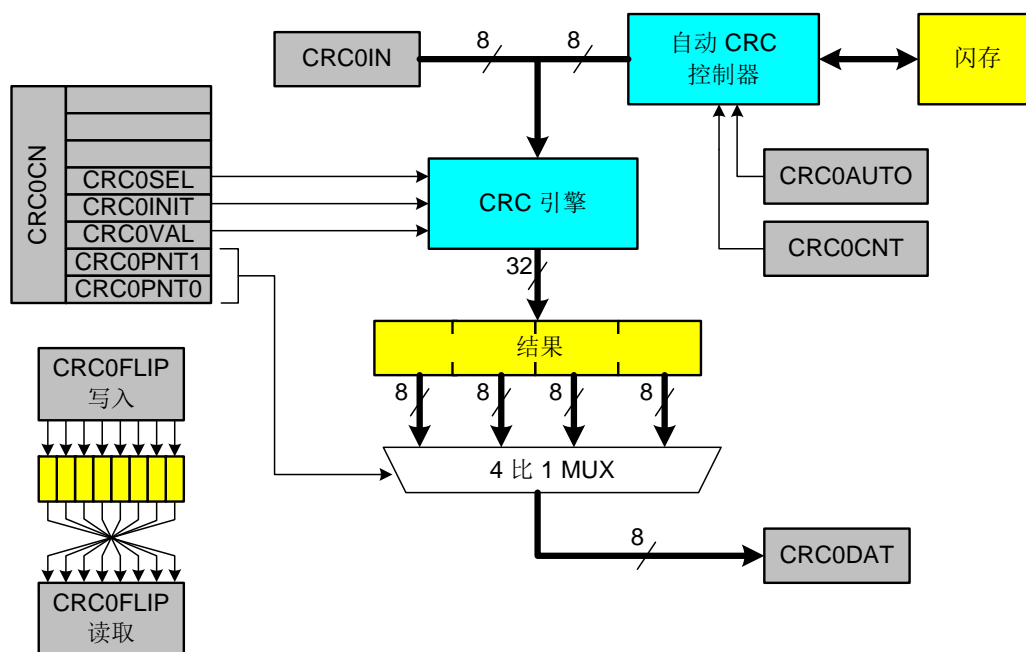


图 24.1. CRC0 结构图

# C8051F80x-83x

## 24.1. 16 位 CRC 算法

C8051F80x-83x CRC 单位使用多项式 0x1021 优先计算 16 位 CRC MSB。硬件执行的 16 位 CRC 算法如下所述：

1. XOR 目前 CRC 结果的最高有效字节和输入字节。如这是 CRC 单位的第一次迭代，目前的 CRC 结果将为设定的初始值（0x0000 或 0xFFFF）。
2. 如 CRC 结果的 MSB 已被设定，左移 CRC 结果，然后 XOR CRC 结果和多项式 (0x1021)。
3. 如果 CRC 结果的 MSB 未设定，左移 CRC 结果。
4. 重复步骤 2，重复次数为所输入位数 (8)。

例如，16 位 C8051F80x-83x CRC 算法可通过以下代码加以说明：

```
unsigned short UpdateCRC (unsigned short CRC_acc, unsigned char CRC_input){
    unsigned char i;                // 循环计数器
    #define POLY 0x1021
    // 为多项式计算创建 CRC"被除数"（二进制运算无进位）
    CRC_acc = CRC_acc ^ (CRC_input << 8);
    // 使用 CRC XOR 减法将多项式"分为"被除数
    // CRC_acc 保持每次划分的"余数"
    // 输入为 1 字节 (i = 0; i < 8; i++) 后，仅完成 8 位的除法
    for (i = 0; i < 8; i++)
    {
        // 核实 MSB 是否设定（如 MSB 为 1，多项式可"分为"
        // "除数"）
        if ((CRC_acc & 0x8000) == 0x8000)
        {
            // 如是，移位 CRC 值，XOR"减去"多项式
            CRC_acc = CRC_acc << 1;
            CRC_acc ^= POLY;
        }
        else
        {
            // 如不是，仅移位 CRC 值
            CRC_acc = CRC_acc << 1;
        }
    }
    return CRC_acc; // 返回最终余数（CRC 值）
}
```

表 24.1 使用 16 位 C8051F80x-83x CRC 算法列出输入值范例和相关输出（已使用初始值 0xFFFF）：

表 24.1: 16 位 CRC 输出范例

输入	输出
0x63	0xBD35
0xAA、0xBB、0xCC	0x6CF6
0x00、0x00、0xAA、0xBB、0xCC	0xB166

## 24.2. 32 位 CRC 算法

C8051F80x-83x CRC 单位使用多项式 0x04C11DB7 计算 32 位 CRC。CRC-32 算法被"反射",意思是所有输入字节和最终 32 位输出在处理引擎中均为位反转。以下为对简化的 CRC 算法的描述,该算法生成的结果和硬件相同:

1. XOR 目前 CRC 结果的最低有效字节和输入字节。如这是 CRC 单位的第一次迭代,目前的 CRC 结果将为设定的初始值 (0x00000000 或 0xFFFFFFFF)。
2. 右移 CRC 结果。
3. 如 CRC 结果的 LSB 已设定, XOR CRC 结果和反射多项式 (0xEDB88320)。
4. 重复步骤 2, 重复次数为所输入位数 (8)。

例如, 32 位 C8051F80x-83x CRC 算法可通过以下代码加以说明:

```
unsigned long UpdateCRC (unsigned long CRC_acc, unsigned char CRC_input){
    unsigned char i; // 循环计数器
    #define POLY 0xEDB88320 // 多项式 0x04C11DB7 位的反转版本
    // 为多项式计算创建 CRC"被除数" (二进制运算无进位)
    CRC_acc = CRC_acc ^ CRC_input;
    // 使用 CRC XOR 减法将多项式"分为"被除数
    // CRC_acc 保持每次划分的"余数"
    // 由于输入为 1 字节 (i = 0; i < 8; i++), 仅完成 8 位的除法
    for (i = 0; i < 8; i++)
    {
        // 核实 MSB 是否设定 (如 MSB 为 1, 多项式可"分为"
        // "除数")
        if ((CRC_acc & 0x00000001) == 0x00000001)
        {
            // 如是, 移位 CRC 值, XOR"减去"多项式
            CRC_acc = CRC_acc >> 1;
            CRC_acc ^= POLY;
        }
        else
        {
            // 如不是, 仅移位 CRC 值
            CRC_acc = CRC_acc >> 1;
        }
    }
    return CRC_acc; // 返回最终余数 (CRC 值)
}
```

表 24.2 使用 32 位 C8051F80x-83x CRC 算法列出输入值范例和相关输出 (已使用初始值 0xFFFFFFFF):

**表 24.2: 32 位 CRC 输出范例**

输入	输出
0x63	0xF9462090
0xAA、0xBB、0xCC	0x41B207B3
0x00、0x00、0xAA、0xBB、0xCC	0x78D129BC

# C8051F80x-83x

---

## 24.3. 准备 CRC 计算

准备 CRC0 进行 CRC 计算，软件应当选择拟定的多项式并设定结果的初始值。有两个多项式可用：0x1021（16 位）和 0x04C11DB7（32 位）。CRC0 结果可初始化为两个值中的一个：0x00000000 或 0xFFFFFFFF。以下步骤可用于初始化 CRC0。

1. 选择多项式（设定 CRC0SEL 的 0 为 32 位，或 1 为 16 位）。
2. 选择初始结果值（设定 CRC0VAL 的 0 为 0x00000000，或 1 为 0xFFFFFFFF）。
3. 设定结果为其初始值（将 1 写入 CRC0INIT）。

## 24.4. 执行 CRC 计算

CRC0 初始化后，随后输入数据流写入 CRC0IN，一次一个字节。每一字节写入后，CRC0 结果自动更新。CRC 引擎也可配置为自动在一个或多个闪存扇区执行 CRC。可使用以下步骤自动在闪存上执行 CRC。

1. 为上文所述的 CRC 计算准备 CRC0。
2. 将起始页的变址写入 CRC0AUTO。
3. 设定 CRC0AUTO 中的 AUTOEN 位。
4. 写入闪存扇区的数字，执行 CRC 计算中的 CRC0CNT。

**附注：** 各个闪存扇区为 512 字节。

5. 将任意值写入 CRC0CN（或用 0x00 OR 其内容），以启动 CRC 计算。在 CRC 操作完成前，CPU 不会执行任何其他代码。
6. 清除 CRC0AUTO 中的 AUTOEN 位。
7. 使用以下程序读取 CRC 结果。

## 24.5. 存取 CRC0 结果

内部 CRC0 结果为 32 位 (CRC0SEL = 0b) 或 16 位 (CRC0SEL = 1b)。CRC0PNT 位选择 CRC0DAT 上读写操作的目标字节并在每次读或写后增量。计算结果仍为内部的 CR0 结果寄存器，直至该结果被设定、改写或将其他数据写入 CRC0IN。



## SFR 定义 24.1: CRC0CN: CRC0 控制寄存器

位	7	6	5	4	3	2	1	0
名称				CRC0SEL	CRC0INIT	CRC0VAL	CRC0PNT[1:0]	
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xCE

位	名称	功能
7:5	未使用	读 = 000b; 写 = 无需在意。
4	CRC0SEL	<b>CRC0 多项式选择位。</b> 该位选择 CRC0 多项式和结果位数 (32 位或 16 位)。 0: CRC0 使用 32 位多项式 0x04C11DB7 计算 CRC 结果。 1: CRC0 使用 16 位多项式 0x1021 计算 CRC 结果。
3	CRC0INIT	<b>CRC0 结果初始化位。</b> 将 1 写入该位, 可在 CRC0VAL 的基础上将整个 CRC 结果初始化。
2	CRC0VAL	<b>CRC0 设置值初始化位。</b> 该位选择 CRC 结果的设置值。 0: 将 1 写入 CRC0INIT 后, CRC 结果设置为 0x00000000。 1: 将 1 写入 CRC0INIT 后, CRC 结果设置为 0xFFFFFFFF。
1:0	CRC0PNT[1:0]	<b>CRC0 结果指针。</b> 指明将在下次访问 CRC0DAT 时读写的 CRC 结果字节。上述字节的值在每次读取或写入后将自动增量。 如 CRC0SEL = 0: 00: CRC0DAT 访问 32 位 CRC 结果位数 7-0。 01: CRC0DAT 访问 32 位 CRC 结果位数 15-8。 10: CRC0DAT 访问 32 位 CRC 结果位数 23-16。 11: CRC0DAT 访问 32 位 CRC 结果位数 31-24。 如 CRC0SEL = 1: 00: CRC0DAT 访问 16 位 CRC 结果位数 7-0。 01: CRC0DAT 访问 16 位 CRC 结果位数 15-8。 10: CRC0DAT 访问 16 位 CRC 结果位数 7-0。 11: CRC0DAT 访问 16 位 CRC 结果位数 15-8。

# C8051F80x-83x

## SFR 定义 24.2: CRC0IN: CRC 数据输入寄存器

位	7	6	5	4	3	2	1	0
名称	CRC0IN[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xDD

位	名称	功能
7:0	CRC0IN[7:0]	<b>CRC0 数据输入。</b> 根据第 24.1 节所述的 CRC 算法，每次在写入数据中写入 CRC0IN 结果，写入数据被算入现有的 CRC 结果中。

## SFR 定义 24.3: CRC0DATA: CRC 数据输出寄存器

位	7	6	5	4	3	2	1	0
名称	CRC0DAT[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xDE

位	名称	功能
7:0	CRC0DAT[7:0]	<b>CRC0 数据输出寄存器。</b> 每次在 CRC0DAT 上进行的读取或写入以 CRC 结果位为目标，CRC0 结果指针（CRC0CN 中的 CRC0PNT 位）指向该位。

**SFR 定义 24.4: CRC0AUTO: CRC 自动控制寄存器**

位	7	6	5	4	3	2	1	0
名称	AUTOEN	CRCCPT	Reserved	CRC0ST[4:0]				
类型	R/W							
复位	0	1	0	0	0	0	0	0

SFR 地址 = 0xD2

位	名称	功能
7	AUTOEN	自动 CRC 计算启用。 AUTOEN 设为 1 时, 任何对 CRC0CN 的写入均会将自动 CRC 初始化, 自动 CRC 启用闪存扇区 CRC0ST, 并继续使用 CRC0CNT 扇区。
6	CRCCPT	自动 CRC 计算完成。 当 CRC 计算正在进行时, 设为 0。CRC 计算期间, 代码停止执行, 因此固件读取将始终返回 1。
5	备用	必须写入 0。
4:0	CRC0ST[4:0]	自动 CRC 计算启用闪存扇区。 上述位指明闪存扇区进行自动 CRC 计算。纳入自动 CRC 计算的首个闪存扇区起始地址为 CRC0ST x 512。

**SFR 定义 24.5: CRC0CNT: CRC 自动闪存扇区总数寄存器**

位	7	6	5	4	3	2	1	0
名称	CRC0CNT[5:0]							
类型	R	R	R/W					
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xD3

位	名称	功能
7:6	未使用	读 = 00b; 写 = 无需在意。
5:0	CRC0CNT[5:0]	自动 CRC 计算闪存扇区总数寄存器。 上述位指明执行自动 CRC 计算时将纳入的闪存扇区的数目。纳入自动 CRC 计算的最终闪存扇区基本地址等同于 (CRC0ST + CRC0CNT) x 512。

# C8051F80x-83x

## 24.6. CRC0 位元反转特征

CRC0 包括图 24.1 所示的字节中反转各个位数位元顺序的硬件。写入 CRC0FLIP 的各个数据字节为回读位元反转。例如，如 0xC0 写入 CRC0FLIP，数据回读为 0x03。位元反转是算法中有用的数字计算功能，如 FFT。

### SFR 定义 24.6: CRC0FLIP: CRC 位翻转寄存器

位	7	6	5	4	3	2	1	0
名称	CRC0FLIP[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xCF

位	名称	功能
7:0	CRC0FLIP[7:0]	<b>CRC0 位翻转寄存器。</b> 任何写入 CRC0FLIP 的字节均按位翻转次序回读，即录入的 LSB 成为 MSB。 例如： 如 0xC0 写入CRC0FLIP，数据回读将为 0x03。 如 0x05 写入 CRC0FLIP，数据回读将为 0xA0。

## 25. 增强型串行外围接口 (SPI0)

增强型串行外围接口 (SPI0) 提供访问灵活的全双工同步串行总线的路径。SPI0 可作为主设备或子设备在 3-线或4-线模式中运作, 并支持单个 SPI 总线上的多种主设备或子设备。从设备选择 (NSS) 信号可被配置为输入, 以选择从模式中的 SPI0, 或禁用多个主设备环境中的主模式操作, 避免在多个主设备试图同步进行数据转移时, 争用 SPI 总线。NSS 亦可配置为主模式的组件选择输入, 或禁用 3-线操作。其他通用端口 I/O 针脚可用于选择主模式中的多个从设备。

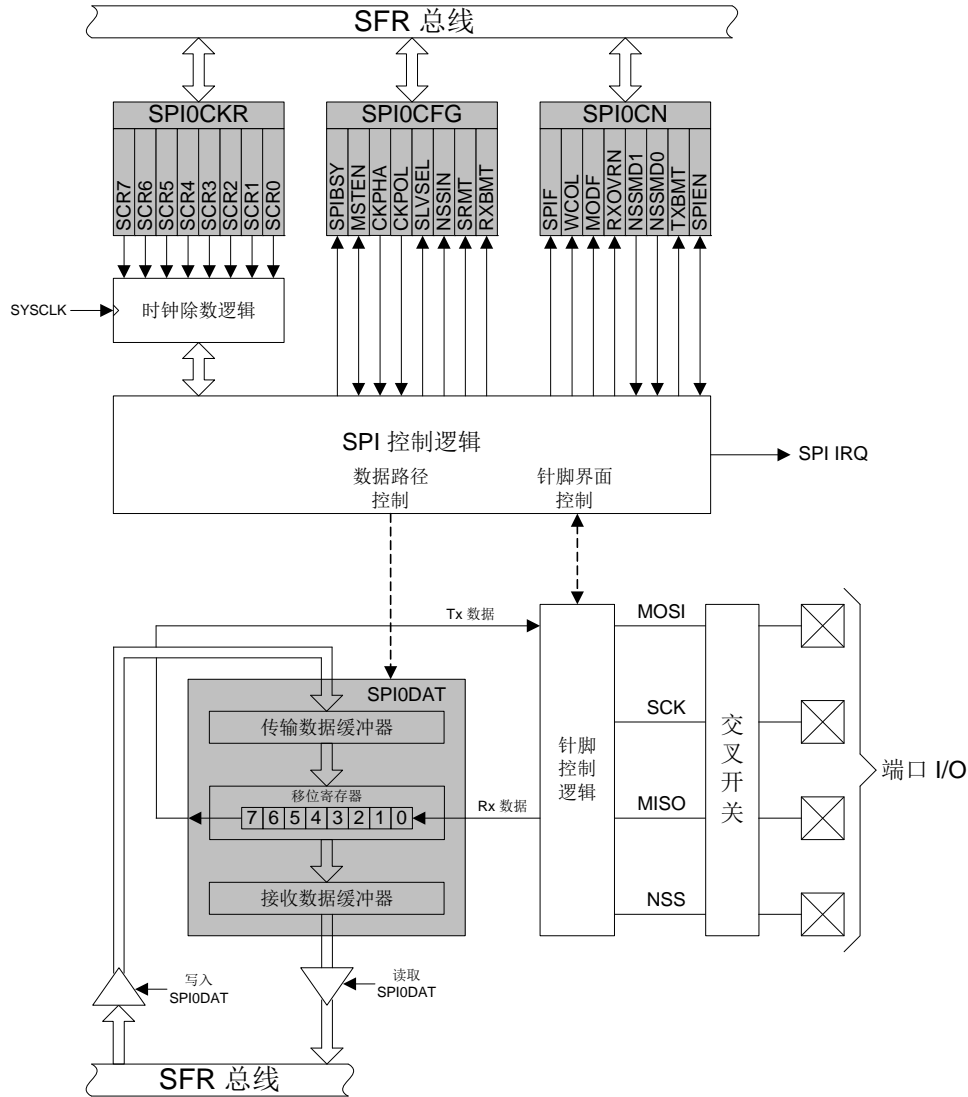


图 25.1. SPI 结构图

## 25.1. 信号说明

下面介绍 SPI0 所使用的四种信号（MOSI、MISO、SCK、NSS）。

### 25.1.1. 主输出、从输入 (MOSI)

主出从入 (MOSI) 信号是主设备的输出和从设备的输入。用于从主设备到从设备的串行数据传输。当 SPI0 作为主设备操作时，此信号为输出信号；当 SPI0 作为从设备操作时，此信号为输入信号。数据传输时最高位在先。当被配置为主设备在 3 线和 4 线模式下操作时，MOSI 由移位寄存器的 MSB 驱动。

### 25.1.2. 主输入、从输出 (MISO)

主入从出 (MISO) 信号是从设备的输出和主设备的输入，用于从从设备到主设备的串行数据传输。当 SPI0 作为主设备操作时，此信号为输入信号；当 SPI0 作为从设备操作时，此信号为输出信号。数据传输时最高位在先。当 SPI 模块被禁用或 SPI 作为从装置在 4 线模式下操作而未被选中时，MISO 引脚被置于高阻态。当作为从设备在 3 线模式下操作时，MISO 一直由移位寄存器的 MSB 驱动。

### 25.1.3. 串行时钟 (SCK)

串行时钟 (SCK) 信号是主设备的输出和从设备的输入，用于同步主设备和从设备之间在 MOSI 和 MISO 线上的数据传输。当 SPI0 作为主设备时产生此信号。在 4 线从模式下，当 SPI 从设备未被选中时 (NSS = 1)，SCK 信号被忽略。

### 25.1.4. 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。有三种设有上述两种位的可能被选中的模式：

1. NSSMD[1:0] = 00: 3 线主模式或从模式：SPI0 在 3 线模式下操作，NSS 被禁用。当作为从设备在 3 线模式下操作时，SPI0 总是被选择。由于无选择信号，SPI0 在 3 线模式下操作时必须为总线唯一的从设备。这种情况用于一个主设备和一个从设备之间的点对点通信。
2. NSSMD[1:0] = 01: 4 线从模式或多主模式：SPI0 在 4 线模式下操作，NSS 作为输入被启用。当作为从设备操作时，NSS 选择 SPI0 设备。当作为主设备操作时，NSS 信号的负跳变禁用 SPI0 的主设备功能，因此可以在同一个 SPI 总线上使用多个主设备。
3. NSSMD[1:0] = 1x: 4 线主模式：SPI0 在 4 线模式下操作，NSS 作为输出被启用。NSSMD0 的设置值决定 NSS 针脚的输出逻辑电平。此配置只能在 SPI0 作为主设备操作时使用。

请参见图 25.2、25.3 和 25.4，了解不同操作模式下的典型连接图。**请注意，NSSMD 位的设置影响设备的引脚分布。**当在 3 线主模式或从模式下操作时，NSS 将不被交叉开关映射引脚。在所有其它模式下，NSS 信号将被映射到设备引脚。请参见 138 页的"23. 端口输入/输出"一条，了解通用端口 I/O 和交叉开关的信息。

## 25.2. SPI0 主模式操作

SPI 主设备可启动 SPI 总线上的所有数据传输。通过设置主启用标志 (MSTEN、SPI0CN.6) 将 SPI0 置于主模式。当处于主模式时，向 SPI0 数据寄存器 (SPI0DAT) 写入一个字节时是写入发送缓冲器。如果 SPI 移位寄存器为空，发送缓冲器中的数据字节则被传送到移位寄存器，然后开始数据传输。SPI0 主设备立即在 MOSI 线上串行移出数据，同时在 SCK 上提供串行时钟。在传输结束时，SPIF (SPI0CN.7) 标志设置为逻辑 1。如中断被允许，设置 SPIF 标志时会产生一个中断请求。在全双工操作中，当 SPI0 主设备在 MOSI 线上向从设备传送数据时，被寻址的 SPI 从设备可同时在 MISO 线上向 SPI 主设备传送其移位寄存器中的内容。因此，SPIF 标志既作为传输完成标志，又作为待接收数据标志。从从设备接收数据字节以 MSB 为先的形式传送到主设备的移位寄存器。当一个数据字节被完全移入移位寄存器时，即被传送到接收缓冲器，在此处理器可通过读 SPI0DAT 来读取数据字节。

当被配置为主设备时，SPI0 可在三种不同模式中的任一种操作：多主模式、3 线单主模式和 4 线单主模式。当  $NSSMD1$  (SPI0CN.3) = 0 且  $NSSMD0$  (SPI0CN.2) = 1 时，默认的多主模式被激活。在这种模式下，NSS 是设备的输入，并在另一主设备访问总线时用于禁用主 SPI0。NSS 在该模式被拉为低电平时，MSTEN (SPI0CN.6) 和 SPIEN (SPI0CN.0) 设置为 0，以禁用 SPI 主设备，且产生模式错误 (MODF, SPI0CN.5 = 1)。如模式错误被启用，将产生中断。在这种情况下，必须用软件手动重启 SPI0。在多主系统中，当设备不作为系统主设备使用时，一般默认为从设备。在多主模式下，可使用通用 I/O 引脚对从设备单独寻址（如需）。图 25.2 展示了多主模式下两个主设备之间的连接图。

当  $NSSMD1$  (SPI0CN.3) = 0 且  $NSSMD0$  (SPI0CN.2) = 0 时，3 线单主模式被激活。在这种模式下，NSS 未被使用，且不会通过交叉开关映射到外部端口引脚。在此模式下，须寻址的任何从设备应使用通用 I/O 引脚进行选择。图 25.3 展示了 3 线主模式下的一个主设备和一个从设备之间的连接图。

当  $NSSMD1$  (SPI0CN.3) = 1 时，4 线单主模式被激活。在这种模式下，NSS 配置为输出引脚，可被用作为单个 SPI 设备选择信号的从设备。在这种模式下，NSS 的输出值由位  $NSSMD0$  (SPI0CN.2) 控制（用软件）。额外的从设备可使用通用 I/O 引脚寻址。图 25.4 展示了 4 线主模式下的一个主设备和两个从设备之间的连接图。

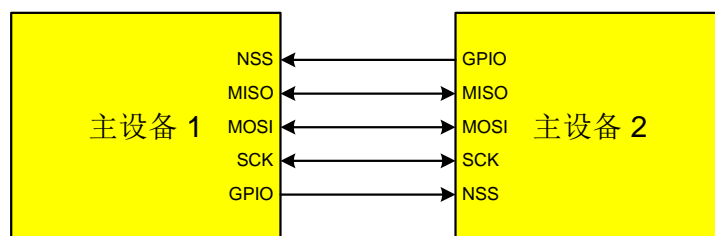


图 25.2. 多主模式接线图

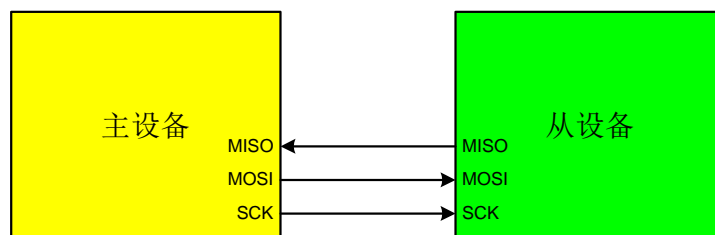


图 25.3. 3 线单主模式和 3 线单从模式接线图

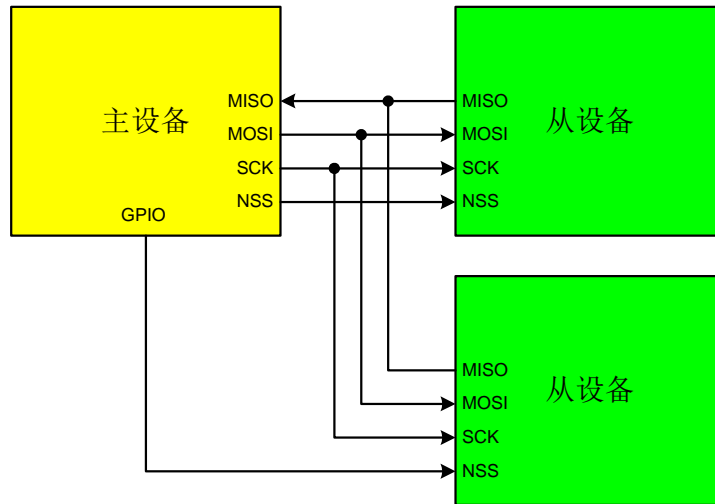


图 25.4. 4 线单主模式和 4 线从模式接线图

### 25.3. SPI0 从模式操作

SPI0 被启用且未被配置为主设备时，将作为 SPI 从设备操作。作为从设备，由控制串行时钟 (SCK) 信号的主设备从 MOSI 针脚移入数据字节，从 MISO 针脚移出数据字节。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转移之后，SPIF 标志设置为逻辑 1，接收到的字节被传送到接收缓冲器。通过读 SPI0DAT 来读取接收缓冲器中的数据。从设备不能启动数据传送。待传送至主设备的数据通过写入 SPI0DAT 预先载入移位寄存器。写入 SPI0DAT 的数据为双缓冲，首先被置于发送缓冲器。如移位寄存器为空，发送缓冲器中的内容将立即被传送至移位寄存器。当移位寄存器中已有数据时，SPI 将在下次（或本次）SPI 传送最后一个 SCK 边沿之后，再将发送缓冲器中的内容载入移位寄存器。

当被配置为从设备时，SPI0 可配置为在 4 线或 3 线模式下操作。当 NSSMD1 (SPI0CN.3) = 0 且 NSSMD0 (SPI0CN.2) = 1 时，默认的 4 线从模式被激活。在 4 线模式下，NSS 信号被发送到端口针脚且被配置为数字输入。当 NSS 为逻辑 0 时，SPI0 被启用；当 NSS 为逻辑 1 时，SPI0 被禁用。在 NSS 的下降沿，位计数器被复位。请注意，对于每次字节传输，NSS 信号必须在第一个有效的 SCK 边沿到来之前，驱动到低电平至少 2 个系统时钟周期。图 25.4 展示了 4 线从模式下的两个从设备和一个主设备之间的连接图。

当 NSSMD1 (SPI0CN.3) = 0 且 NSSMD0 (SPI0CN.2) = 0 时，3 线从模式被激活。在这种模式下，NSS 未被使用，且不会通过交叉开关映射到外部端口针脚。由于在 3 线从模式下无法唯一地为该设备寻址，因此，SPI0 须为总线上唯一的从设备。须注意，在 3 线从模式下，无外部方式可对决定收到一个完整字节的时间的位计数器复位。仅可通过使用 SPIEN 位禁用并重启 SPI0，方可复位位计数器。图 25.3 展示了 3 线从模式下的一个从设备和一个主设备之间的连接图。



## 25.4. SPI0 中断源

当 SPI0 中断被允许，下列四种标志设置为逻辑 1 时将产生中断：

下列所有位都必须用软件清除。

- 在每次字节传输结束时，SPI 中断标志 SPIF (SPI0CN.7) 设置为逻辑 1。此标记可在 SPI0 所有模式下产生。
- 当发送缓冲器中的数据尚未传送至 SPI 移位寄存器时，如欲写入 SPI0DAT，写入冲突标志 WCOL (SPI0CN.6) 设置为逻辑 1。这种情况发生时，写入 SPI0DAT 的操作将被忽略，也将不会写入发送缓冲器。此标志可在 SPI0 所有模式下产生。
- 当 SPI0 配置为多主模式的主设备且 NSS 引脚被拉为低电平时，模式错误标志 MODF (SPI0CN.5) 设置为逻辑 1。模式错误产生时，SPI0CN 中的 MSTEN 和 SPIEN 位设置为逻辑 0，以禁用 SPI0 并使另一个主设备访问总线。
- 当被配置为从设备且一次传输完成，而接收缓冲器仍保持着上次传输的未被读取的数据时，接收溢出标志 RXOVRN (SPI0CN.4) 设置为逻辑 1。新接收的字节不会传送至接收缓冲器，允许先前接收的数据字节被读取。引起溢出的数据字节丢失。

## 25.5. 串行时钟相位和极性

使用 SPI0 配置寄存器 (SPI0CFG) 中的时钟控制位可在串行时钟相位和极性的四种组合中选择其一。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) 在高电平有效或低电平有效的时钟之间选择。主设备和从设备都必须被配置为使用相同的时钟相位和极性。在改变时钟相位或极性时，SPI0 应被禁用（通过清除 SPIEN 位，SPI0CN.0）。主模式下的时钟和数据线的时序关系如图 25.5 所示。从模式的时钟和数据线的时序关系如图 25.6 和图 25.7 所示。请注意，在 Silicon Labs C8051 两种设备之间通信时，SPI 主设备和从设备上的 CKPHA 都应设置为 0。

SFR 定义 25.3 所示的 SPI0 时钟速率寄存器 (SPI0CKR) 控制主模式的串行时钟频率。在从模式下操作时，寄存器被忽略。SPI 被配置为主设备时，最大数据传输率（位/秒）是系统时钟频率的二分之一或 12.5 MHz（以较低的频率为准）。SPI 被配置为从设备时，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，前提是主设备与从设备的系统时钟同步发出 SCK、NSS（在 4 线从模式下）和串行输入数据。如果主设备发出的 SCK、NSS 和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。在主设备仅传送数据到从设备而无需接收从设备发出的数据（即半双工操作）这一特殊情况下，SPI 从设备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。这一情况发生的前提是主设备发出 SCK、NSS 和串行输入数据与从设备系统时钟同步。

# C8051F80x-83x

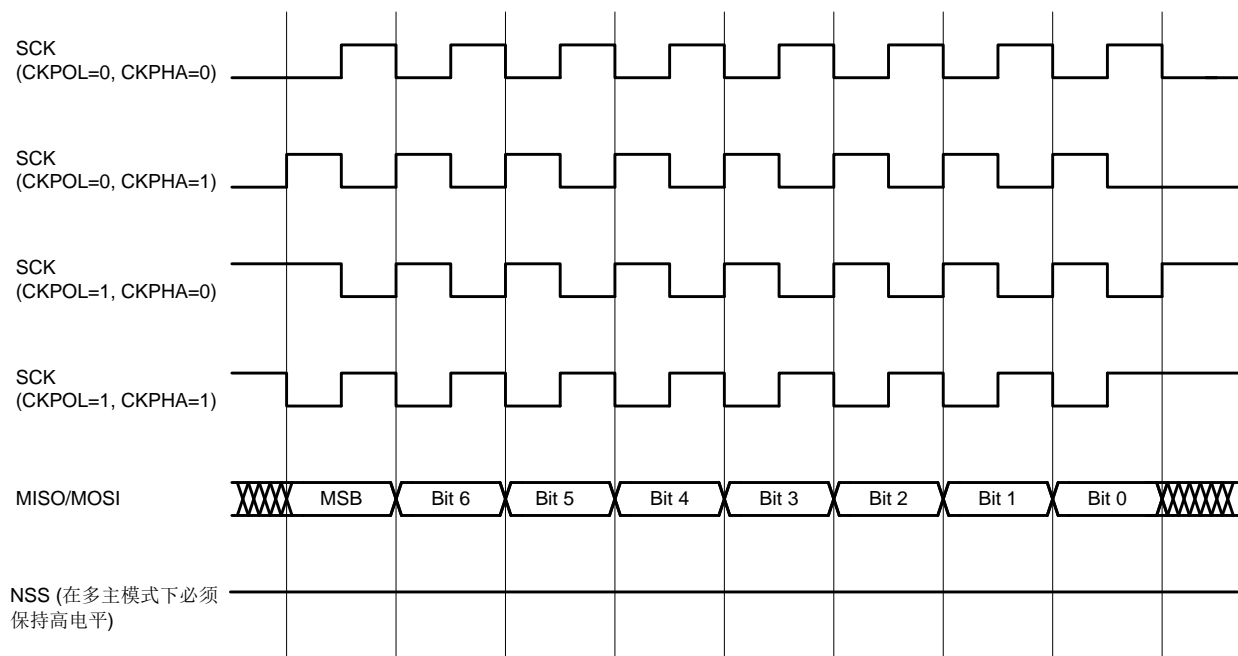


图 25.5. 主模式数据/时钟定时

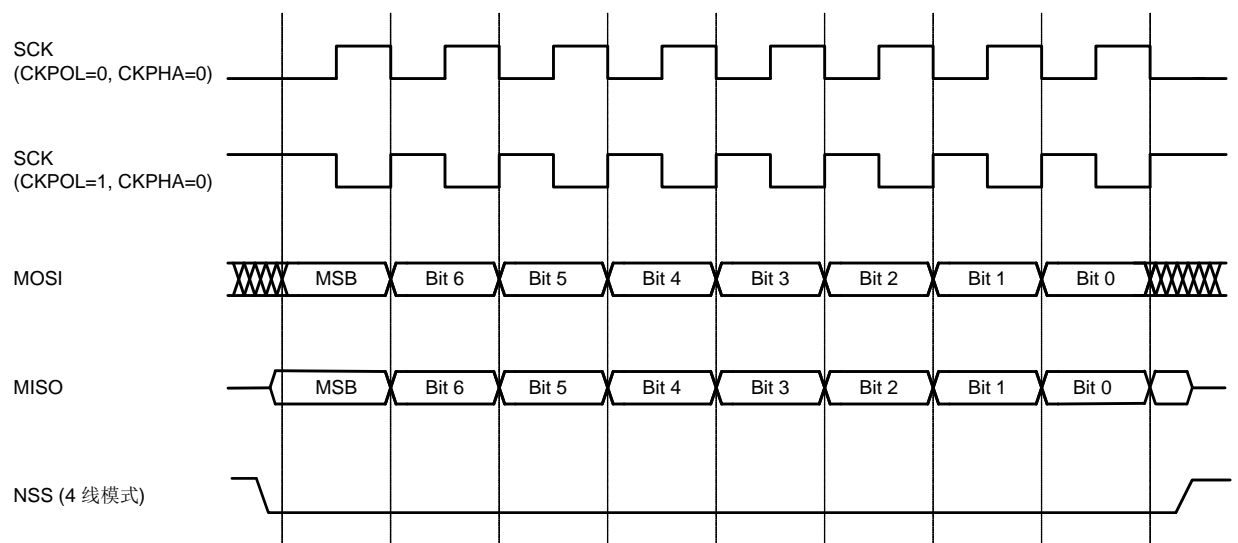


图 25.6. 从模式数据/时钟定时 (CKPHA = 0)

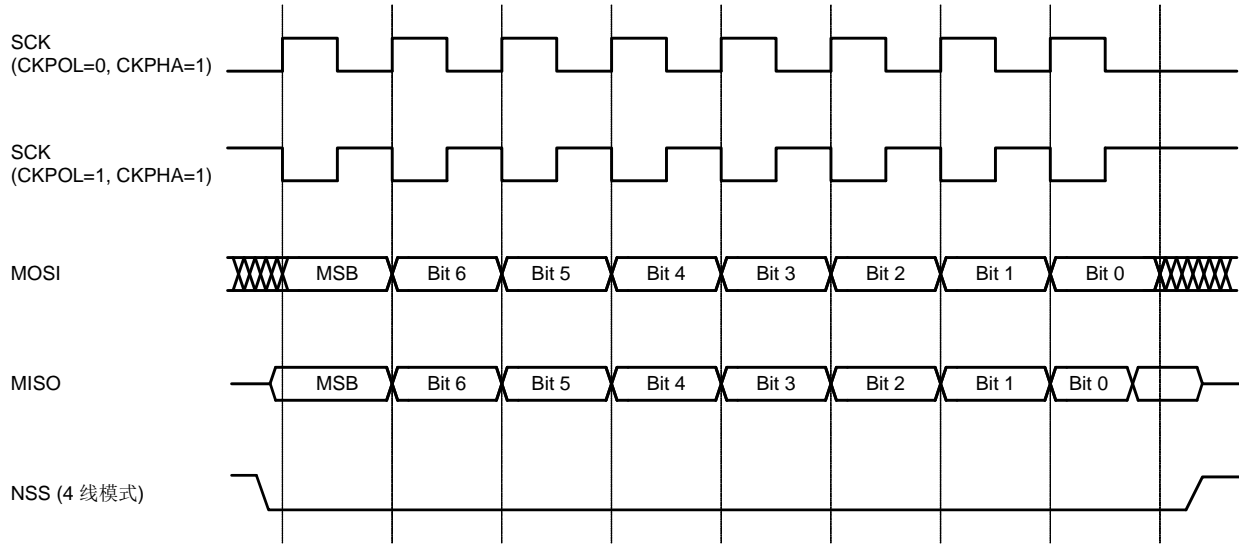


图 25.7. 从模式数据/时钟定时 (CKPHA = 1)

## 25.6. 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四种特殊功能寄存器实现：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。下图将介绍这四种有关 SPI0 总线操作的特殊功能寄存器。

# C8051F80x-83x

## SFR 定义 25.1: SPI0CFG: SPI0 配置寄存器

位	7	6	5	4	3	2	1	0
名称	SPIBSY	MSTEN	CKPHA	CKPOL	SLVSEL	NSSIN	SRMT	RXBMT
类型	R	R/W	R/W	R/W	R	R	R	R
复位	0	0	0	0	0	1	1	1

SFR 地址 = 0xA1

位	名称	功能
7	SPIBSY	<b>SPI 忙标志。</b> 当一次 SPI 传输正在进行时（主或从模式），此位设置为逻辑 1。
6	MSTEN	<b>主模式启用。</b> 0: 禁用主模式。在从模式下操作。 1: 启用主模式。以主设备操作。
5	CKPHA	<b>SPI0 时钟相位。</b> 0: 在 SCK 周期的第一个边沿采样数据。 <sup>*</sup> 1: 在 SCK 周期的第二个边沿采样数据。 <sup>*</sup>
4	CKPOL	<b>SPI0 时钟极性。</b> 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。
3	SLVSEL	<b>从选择标志。</b> 只要 NSS 针脚为低电平，则此位设置为逻辑 1，表示 SPI0 为被选中的从设备。当 NSS 针脚为高电平时（未被选中的从设备），此位被清为逻辑 0。此位并非指 NSS 针脚的即时值，而是指该针脚输入的去噪信号。
2	NSSIN	<b>NSS 针脚输入的即时值。</b> 此位模拟寄存器被读时 NSS 端口针脚上所示的即时值。此输入未被去噪。
1	SRMT	<b>移位寄存器空（仅在从模式下有效）。</b> 当所有数据都已被传入/传出移位寄存器且无新的数据可从发送缓冲器读取或写入接收缓冲器时，此位将设置为逻辑 1。当数据字节从发送缓冲器传入移位寄存器或 SCK 发生变化时，此位还原为逻辑 0。在主模式下 SRMT = 1。
0	RXBMT	<b>接收缓冲器空（仅在从模式下有效）。</b> 当接收缓冲器已被读且内无新数据时，此位将设置为逻辑 1。如尚未被读的接收缓冲器内有可用的新数据，此位将被还原为逻辑 0。在主模式下 RXBMT = 1。

**附注:** 在从模式下，MOSI 上的数据在每个数据位的中间被采样。在主模式下，MISO 上的数据在每个数据位结束前的一个 SYSCLK 被采样，为从设备提供最大的设置时间。请参见表 25.1，了解定时参数。

## SFR 定义 25.2: SPI0CN: SPI0 控制寄存器

位	7	6	5	4	3	2	1	0
名称	SPIF	WCOL	MODF	RXOVRN	NSSMD[1:0]		TXBMT	SPIEN
类型	R/W	R/W	R/W	R/W	R/W		R	R/W
复位	0	0	0	0	0	1	1	0

SFR 地址 = 0xF8; 可位寻址

位	名称	功能
7	SPIF	<b>SPI0 中断标志。</b> 在数据传输结束时，此位被硬件设置为逻辑 1。如 SPI 中断被允许，则将产生中断。此位不能由硬件自动清除，必须用软件清除。
6	WCOL	<b>写入冲突标志。</b> 当 TXBMT 为 0 时，如欲写入 SPI0DAT，则此位设置为逻辑 1。这种情况发生时，写入 SPI0DAT 的操作将被忽略，也将不会写入发送缓冲器。如 SPI 中断被允许，则将产生中断。此位不能由硬件自动清除，必须用软件清除。
5	MODF	<b>模式错误标志。</b> 当检测到主模式冲突（NSS 为低电平，MSTEN = 1 且 NSSMD[1:0] = 01）时，此位被硬件设置为逻辑 1。如 SPI 中断被允许，则将产生中断。此位不能由硬件自动清除，必须用软件清除。
4	RXOVRN	<b>接收溢出标志（仅在从模式下有效）。</b> 当接收缓冲器仍存有上次传输的未读数据且本次传输的最后一位已移入 SPI0 移位寄存器时，此位被硬件设置为逻辑 1。如 SPI 中断被允许，则将产生中断。此位不能由硬件自动清除，必须用软件清除。
3:2	NSSMD[1:0]	<b>从选择模式。</b> 从下列 NSS 操作模式中选择： （请参见第 25.2 和 25.3 条）。 00: 3 线从模式或主模式。NSS 信号不被发送到端口引脚。 01: 4 线从模式或多主模式（默认）。NSS 为设备的输入。 1x: 4 线单主模式。NSS 信号作为设备的输出被映射，并将假定 NSSMD0 的值。
1	TXBMT	<b>发送缓冲器空。</b> 当新数据写入发送缓冲器时，此位将设置为逻辑 0。当发送缓冲器中的数据被传送到 SPI 移位寄存器时，此位将设置为逻辑 1，表示可以将新的数据字节写入发送缓冲器。
0	SPIEN	<b>SPI0 启用。</b> 0: SPI 禁用。 1: SPI 启用。

# C8051F80x-83x

## SFR 定义 25.3: SPI0CKR: SPI0 时钟频率寄存器

位	7	6	5	4	3	2	1	0
名称	SCR[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xA2

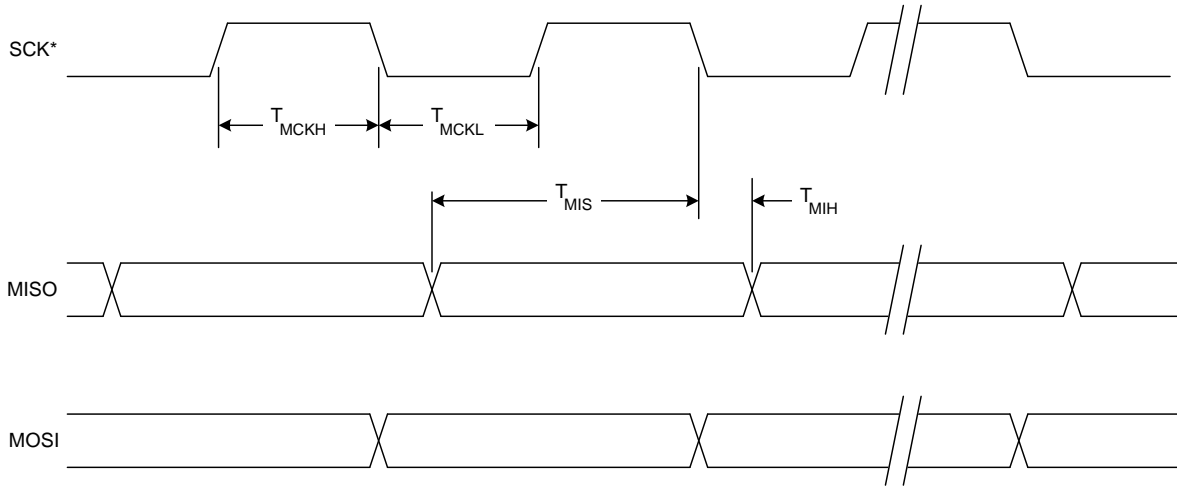
位	名称	功能
7:0	SCR[7:0]	<p><b>SPI0 时钟频率。</b></p> <p>当 SPI0 模块被配置为主模式操作时，这些位决定 SCK 输出的频率。SCK 时钟频率是从系统时钟分频得到的，由下面的方程计算而得，其中：SYSCLK 为系统时钟频率，SPI0CKR 为 SPI0CKR 寄存器中的 8 位值。</p> $f_{SCK} = \frac{SYSCLK}{2 \times (SPI0CKR[7:0] + 1)}$ <p>for 0 &lt;= SPI0CKR &lt;= 255</p> <p>例如：如 SYSCLK = 2 MHz，SPI0CKR = 0x04，则</p> $f_{SCK} = \frac{2000000}{2 \times (4 + 1)}$ $f_{SCK} = 200kHz$

## SFR 定义 25.4: SPI0DAT: SPI0 数据寄存器

位	7	6	5	4	3	2	1	0
名称	SPI0DAT[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

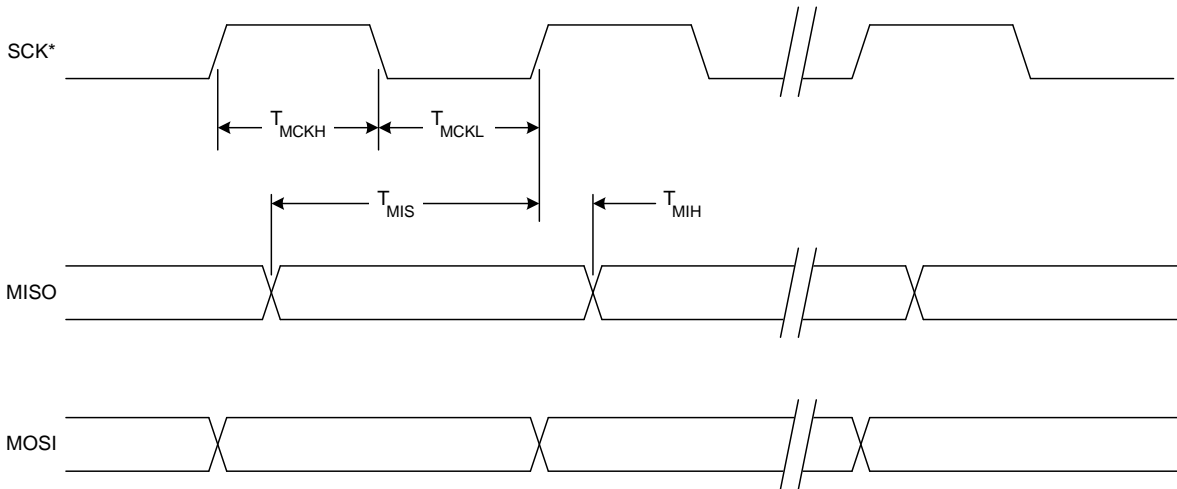
SFR 地址 = 0xA3

位	名称	功能
7:0	SPI0DAT[7:0]	<p><b>SPI0 发送和接收数据。</b></p> <p>SPI0DAT 寄存器用于发送和接收 SPI0 数据。在主模式下，向 SPI0DAT 写入数据，数据被置于发送缓冲器中并启动发送。读 SPI0DAT 即还原接收缓冲器中的内容。</p>



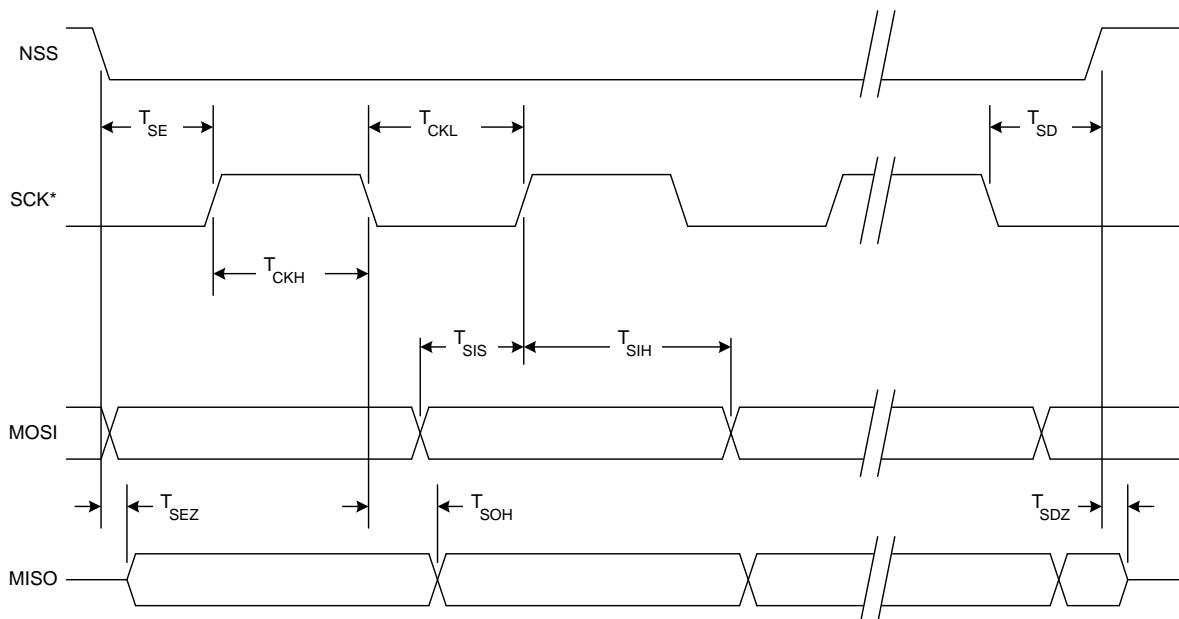
\* 这是对应 CKPOL = 0 时的 SCK 波形。对于 CKPOL = 1，SCK 波形的极性反向。

**图 25.8. SPI 主定时 (CKPHA = 0)**



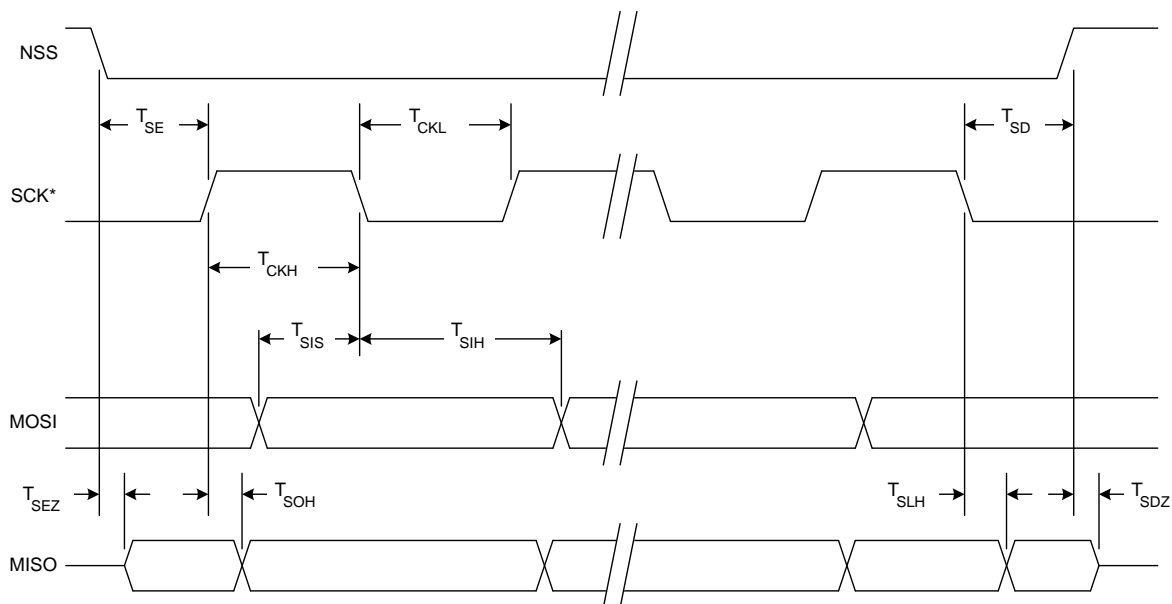
\* 这是对应 CKPOL = 0 时的 SCK 波形。对于 CKPOL = 1，SCK 波形的极性反向。

**图 25.9. SPI 主定时 (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 25.10. SPI 从定时 (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 25.11. SPI 从定时 (CKPHA = 1)



表 25.1: SPI 从定时参数

参数	说明	最小值	最大值	单位
主模式定时 (见图 25.8 和 25.9)				
$T_{MCKH}$	SCK 高电平时间	$1 \times T_{SYSCLK}$	—	ns
$T_{MCKL}$	SCK 低电平时间	$1 \times T_{SYSCLK}$	—	ns
$T_{MIS}$	MISO 有效到 SCK 移位边沿	$1 \times T_{SYSCLK} + 20$	—	ns
$T_{MIH}$	SCK 移位边沿到 MISO 发生改变	0	—	ns
从模式定时 (见图 25.10 和 25.11)				
$T_{SE}$	NSS 下降沿到第一个 SCK 边沿	$2 \times T_{SYSCLK}$	—	ns
$T_{SD}$	最后一个 SCK 边沿到 NSS 上升沿	$2 \times T_{SYSCLK}$	—	ns
$T_{SEZ}$	NSS 下降沿到 MISO 有效	—	$4 \times T_{SYSCLK}$	ns
$T_{SDZ}$	NSS 上升沿到 MISO 变为高阻态	—	$4 \times T_{SYSCLK}$	ns
$T_{CKH}$	SCK 高电平时间	$5 \times T_{SYSCLK}$	—	ns
$T_{CKL}$	SCK 低电平时间	$5 \times T_{SYSCLK}$	—	ns
$T_{SIS}$	MOSI 有效到 SCK 采样边沿	$2 \times T_{SYSCLK}$	—	ns
$T_{SIH}$	SCK 采样边沿到 MOSI 发生改变	$2 \times T_{SYSCLK}$	—	ns
$T_{SOH}$	SCK 移位边沿到 MISO 发生改变	—	$4 \times T_{SYSCLK}$	ns
$T_{SLH}$	最后一个 SCK 边沿到 MISO 发生改变 (仅在 CKPHA = 1 时)	$6 \times T_{SYSCLK}$	$8 \times T_{SYSCLK}$	ns
附注: $T_{SYSCLK}$ 为设备系统时钟 (SYSCLK) 的一个周期。				

# C8051F80x-83x

---

## 26. SMBus

SMBus I/O 接口是一个双线的双向串行总线。SMBus 符合系统管理总线规格 1.1 版，与 I<sup>2</sup>C 串行总线兼容。系统控制器对接口的读写操作都是以字节为单位，由 SMBus 接口自主控制数据的串行传输。在作为主设备或从设备时，数据传输的最大速率可达系统时钟频率的二十分之一（这可能比 SMBus 的规定速度快，取决于所使用的系统时钟）。可以采用延长低电平时间的方法协调同一总线上不同速度的设备。

SMBus 接口可作为主设备和/或从设备操作，且可在一个总线上操作多个主设备。SMBus 提供 SDA（串行数据）控制、SCL（串行时钟）产生和同步、仲裁逻辑以及起始/停止控制和产生电路。SMBus 外围可被软件完全驱动（即软件接受/拒绝从地址并产生 ACK），或硬件从地址识别和 ACK 自动产生可被启动，以降低软件开销。SMBus 外围的结构图和相关的 SFR 如图 26.1 所示。

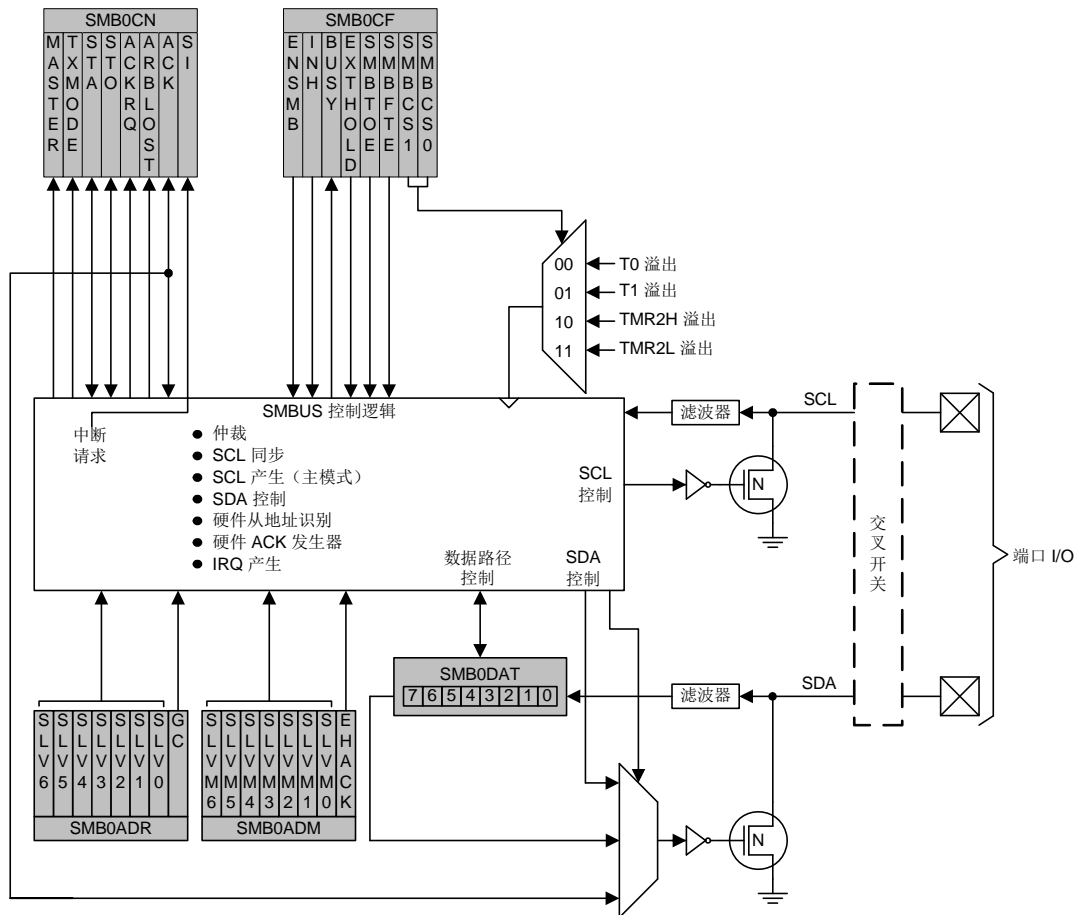


图 26.1. SMBus 结构图

# C8051F80x-83x

## 26.1. 辅助文件

假设读者熟悉或有条件读取下列辅助文件：

1. I<sup>2</sup>C 总线和如何使用（包括规格），飞利浦半导体。
2. I<sup>2</sup>C 总线规格-2.0 版，飞利浦半导体。
3. 系统管理总线规格-1.1 版，SBS 实施者论坛。

## 26.2. SMBus 配置寄存器

图 26.2 展示了 SMBus 的典型配置。SMBus 规定的电压为 3.0 V 至 5.0 V，总线上不同设备的工作电压可以不同。SCL（串行时钟）和 SDA（串行数据）线为双向，必须通过一个上拉电阻或等效电路连接到正电源电压。连接到总线上的每个设备的 SCL 和 SDA 线须为漏极开路或集电极开路，因此，当总线空闲时，这两条线都被拉到高电平（隐状态）。总线上的最大设备数目仅受规定的总线上上升和下降时间的限制，上升和下降时间分别不得超过 300 ns 和 1000 ns。

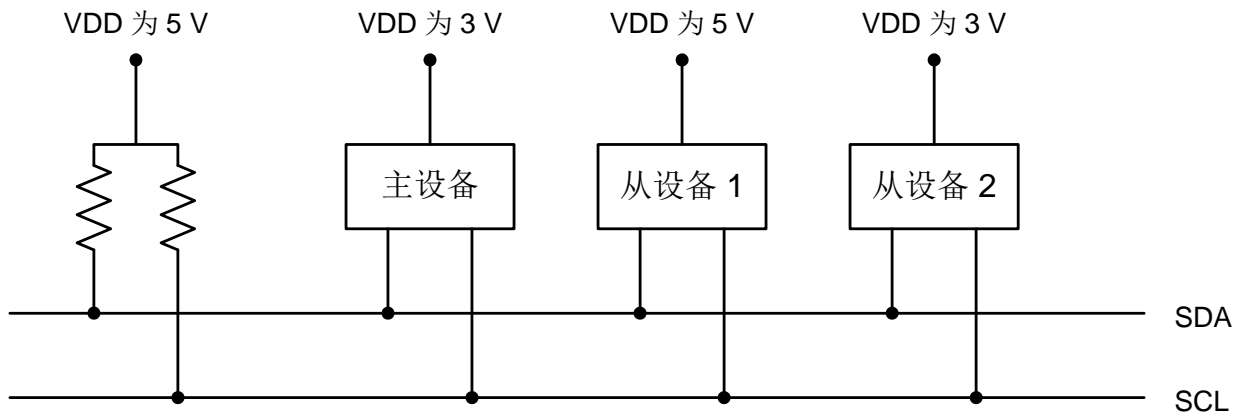


图 26.2. 典型 SMBus 配置

## 26.3. SMBus 操作

有两种可能的数据传输类型：从主发送器到寻址的从接收器（写）的数据传输和从寻址的从发送器到主接收器（读）。这两种数据传输都由主设备启动，主设备还在 SCL 上提供串行时钟脉冲。SMBus 接口可作为主设备或从设备操作，同一总线上可有多个主设备。如果两个或更多个主设备试图同时启动数据传输，仲裁机制将保证总会有一个主设备赢得总线。请注意，无必要在系统中指定一个主设备；任何发送起始和从地址的设备都可成为数据传输期间的主设备。

一次典型的 SMBus 传输包含一个起始条件、一个地址字节（位 7-1：7 位从地址；位 0：R/W 方向位）、一个或多个数据字节和一个停止条件。接收到的字节（由主设备或从设备发送）都必须用 SCL 高电平期间的 SDA 低电平（见图 26.3）来确认 (ACK)。如果接收设备不确认，传送设备将读到一个非确认 (NACK)，这用 SCL 高电平期间的 SDA 高电平表示。

方向位 (R/W) 占据地址字节的最低位位置。方向位设置为逻辑 1，表示一个“读”的操作；被清为逻辑 0，表示一个“写”的操作。

所有传输都是由主设备启动，以一个或多个寻址从设备为目标。主设备产生一个起始条件，然后传送从地址和方向位。如传输为主设备到从设备的写的操作，则主设备每传送一个数据字节后等待来自从设备的确认。对于读的操作，从设备传送每个数据字节后等待来自主设备的确认。数据传输结束时，主设备产生一个停止条件，终止传输并释放总线。图 26.3 列示了典型的 SMBus 传输。

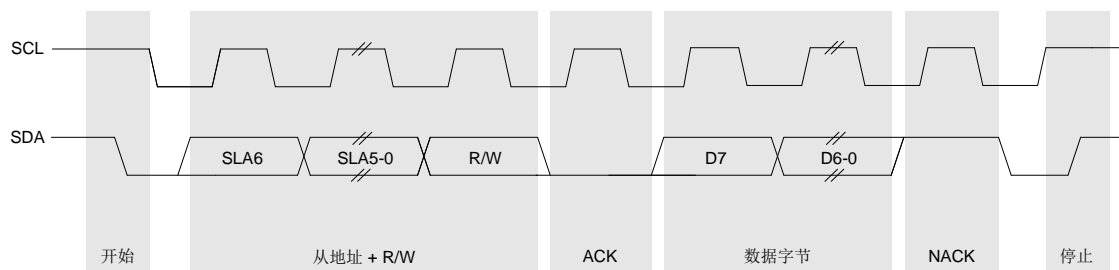


图 26.3. SMBus 传输

### 26.3.1. 发送器与接收器

在 SMBus 通信接口上，发送地址或数据字节至总线上另一设备时，设备为“发送器”。当地址或数据字节被总线上另一设备发送至此，设备为“接收器”。地址或数据字节传输时，发送器控制 SDA 线。发送器传送地址或数据信息的字节完成之后，接收器在传输 ACK 相位时发送 ACK 或 NACK 位，此时接收器控制 SDA 线。

### 26.3.2. 仲裁

一个主设备只有在总线空闲时才能启动一次传输。停止条件之后或 SCL 和 SDA 线保持高电平已经超过了指定时间（参见第 183 页“26.3.5. SCL 高（SMBus 除外）超时”一条），则总线空闲。如两个或多个设备试图同时开始数据传输，使用仲裁机制迫使一个主设备放弃总线。主设备继续传输，直至其中一个主设备发送高电平而其它主设备发送低电平。由于总线为漏极开路，因此总线将被拉为低电平。试图发送高电平的主设备将探测到 SDA 上的低电平而在竞争中失败。赢得总线的主设备继续无中断传输，而未赢得总线的主设备成为从设备并接收传输的其余数据（如被寻址）。此仲裁机制为非破坏性：总会有一个设备赢得总线，而无数据丢失。

### 26.3.3. 时钟低电平扩展

SMBus 提供一种类似 I2C 的时钟同步机制，允许不同速度的设备共存于同一个总线上。为了使低速从设备能与高速主设备通信，在传输时使用时钟低电平扩展。从设备可临时保持 SCL 线为低电平以扩展时钟低电平时间，有效地降低了串行时钟频率。

### 26.3.4. SCL 低电平超时

如 SCL 线被总线上的从设备保持为低电平，则不能再进行通信。并且主设备不能强制 SCL 线为高电平来纠正这种错误情况。为解决这一问题，SMBus 协议规定，参加数据传输的设备须探测任何时钟的低电平时间，若超过 25 ms 则为“超时”。探测到超时情况的设备须在探测到此情况后 10 ms 内复位通信电路。

当 SMB0CF 中的 SMBTOE 位被设置时，定时器 3 被用于探测 SCL 低电平超时。定时器 3 在 SCL 为高电平时被强制重载，在 SCL 为低电平时开始计数。如果定时器 3 被启用且溢出周期被配置为 25 ms（且 SMBTOE 被设置），则在发生 SCL 低电平超时事件时用定时器 3 中断服务程序，对 SMBus 复位（禁用后重新启用）。

## 26.3.5. SCL 高电平（SMBus 空闲）超时

SMBus 规格规定，如 SCL 和 SDA 线保持高电平的时间超过 50  $\mu$ s，则认为总线处于空闲状态。当 SMB0CF 中的 SMBFTE 位被设置时，如 SCL 和 SDA 保持高电平的时间超过 10 个 SMBus 时钟源周期（在为 SMBus 时钟源配置的定时器中定义），则总线将被视为空闲。如 SMBus 正待产生一个主起始条件，则该起始条件将在超时后立即产生。总线空闲超时探测需要一个时钟源，即使对从设备也不例外。

## 26.4. 使用 SMBus

SMBus 可在主模式和从模式下操作。接口电路为串行传输提供定时和移位控制；更高层的协议由用户软件实现。SMBus 接口提供下述无关应用的特征：

- 以字节为单位的串行数据传输
- SCL 时钟信号产生（仅限主模式）和 SDA 数据同步
- 超时/总线错误识别（在 SMB0CF 配置寄存器中定义）
- START/STOP 定时、探测和产生
- 总线仲裁
- 中断产生
- 状态信息
- 从地址的可选硬件识别和地址/数据自动确认

每次数据字节或从地址传输都产生 SMBus 中断。当硬件确认被禁用，此时中断产生取决于硬件是作为数据发送器还是接收器。作为发送器时（即发送地址/数据、接收 ACK），中断于 ACK 周期后产生，从而使软件可读取接收到的 ACK 值；接收数据时（即接收地址/数据、发送 ACK），中断于 ACK 周期之前产生，从而使软件可确定要发出的 ACK 值。如启用硬件确认，则这些中断总在 ACK 周期之后产生。请参见第 26.5 条，了解有关传输顺序的更多详情。

当主设备产生起始条件或从设备探测到停止条件时也会产生中断，表示数据传输开始或结束。软件应通过读 SMB0CN（SMBus 控制寄存器）来确定 SMBus 中断的原因。SMB0CN 寄存器如第 26.4.2 条所述；表 26.5 载列了 SMB0CN 快速解码参考。

### 26.4.1. SMBus 配置寄存器

SMBus 配置寄存器 (SMB0CF) 用于启用 SMBus 主和/或从模式，选择 SMBus 时钟源以及选择 SMBus 定时和超时选项。当 ENSMB 位被设置时，SMBus 的所有主和从事件都被允许。可通过设置 INH 位来禁止从事件。在从事件被禁止的情况下，SMBus 接口将仍然监视 SCL 和 SDA 引脚；然而，接口将在接收到所有地址时发出 NACK 信号，且将不会产生任何从中断。当 INH 位被设置时，在下一个起始条件后，所有的从事件都将被禁止（当前传输过程中的中断将继续）。

表 26.1: SMBus 时钟源选择

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

SMBCS1-0 位选择 SMBus 时钟源, 时钟源仅在作为主设备或空闲超时探测被启用时使用。作为主设备操作时, 所选时钟源的溢出周期决定 SCL 低电平和高电平的绝对值最小时间 (如方程 26.1 定义)。请注意, 只要定时器一直保持运行状态, 所选时钟源就可与其它外围设备共享。例如, 定时器 1 溢出可同时产生 SMBus 和 UART 波特率。定时器配置见第 209 页"28. 定时器"一条。

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

方程 26.1: 最小 SCL 高和低电平时间

所选时钟源应被配置为能产生方程 26.1 所定义的最小 SCL 高和低电平时间。当接口作为主设备操作 (且 SCL 不被总线上任何其它设备驱动或扩展) 时, 典型的 SMBus 位速率可由下面的方程 26.2 估算:

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

方程 26.2: 典型 SMBus 位速率

图 26.4 列示了方程 26.2 所述的典型 SCL 波形。请注意,  $T_{HIGH}$  通常为  $T_{LOW}$  的两倍。实际的 SCL 输出波形可能会因总线上有其它设备而发生改变 (SCL 可能被低速从设备扩展为低电平, 或被其它参与竞争的主设备驱动为低电平)。作为主设备操作时, 位速率将不能超过由方程 26.1 定义的极限值。

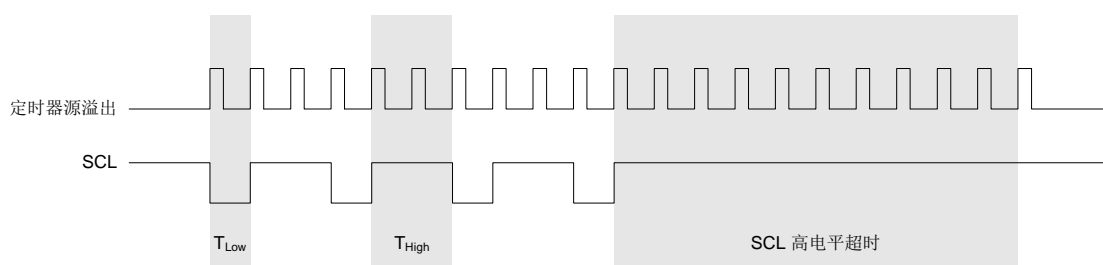


图 26.4. 典型 SMBus SCL 发生器

设置 EXTHOLD 位将扩展 SDA 线的最小建立时间和保持时间。最小 SDA 建立时间定义了 SCL 上升沿到来之前 SDA 稳定的绝对值最小时间。最小 SDA 保持时间定义了 SCL 下降沿过去之后 SDA 值保持稳定的绝对值最小时间。应设置 EXTHOLD, 从而使最小建立和保持时间分别符合 SMBus 规格规定的 250 ns 和 300 ns。表 26.2 列示了两种 EXTHOLD 设置情况的最小建立和保持时间。当 SYSCLK 大于 10 MHz 时, 通常需扩展建立和保持时间。

表 26.2: 最小 SDA 建立和保持时间

EXTHOLD	最小 SDA 建立时间	最小 SDA 保持时间
0	$T_{low} - 4$ 个系统时钟 或 1 个系统时钟 + 软件延时*	3 个系统时钟
1	11 个系统时钟	12 个系统时钟
<b>附注:</b> 传输 ACK 位和所有数据传输中 MSB 的建立时间。使用软件确认时, 软件延时发生在写入 SMB0DAT 或 ACK 到 SI 被清除之间。请注意, 如果 SI 被清除与确定要发出的 ACK 值发生在同一个写的操作, 则软件延时为零。		

在 SMBTOE 位被设置的情况下, 定时器 3 应被配置为以 25 ms 为周期溢出, 以探测 SCL 低电平超时 (请参见第 182 页"26.3.4. SCL 低电平超时"一条)。SMBus 接口将在 SCL 为高电平时强制重载定时器 3, 在 SCL 为低电平时允许定时器 3 计数。应通过先禁用后重启 SMBus, 使用定时器 3 中断服务程序对 SMBus 通信复位。

SMBus 空闲超时探测可通过设置 SMBFTE 位启用。当此位被设置, 如 SDA 和 SCL 保持高电平的时间超过 10 个 SMBus 时钟源周期 (见图 26.4), 则总线将被视为空闲。



## SFR 定义 26.1: SMB0CF: SMBus 时钟/配置寄存器

位	7	6	5	4	3	2	1	0
名称	ENSMB	INH	BUSY	EXTHOLD	SMBTOE	SMBFTE	SMBCS[1:0]	
类型	R/W	R/W	R	R/W	R/W	R/W	R/W	
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xC1

位	名称	功能
7	ENSMB	<b>SMBus 启用。</b> 当设置为 1 时，此位启用 SMBus 接口。启用时，接口一直监视 SDA 和 SCL 引脚。
6	INH	<b>SMBus 从禁止。</b> 此位设置为逻辑 1 时，SMBus 在从事件发生时不产生中断。这样有效地将 SMBus 从设备移出总线。主模式中断不受影响。
5	BUSY	<b>SMBus 忙状态标志。</b> 正在进行传输时，此位被硬件设置为逻辑 1。当检测到停止条件或空闲超时时，此位被清为逻辑 0。
4	EXTHOLD	<b>SMBus 建立和保持时间扩展启用。</b> 此位根据表 26.2 控制 SDA 建立和保持时间。 0: 禁用 SDA 建立和保持时间扩展。 1: 启用 SDA 建立和保持时间扩展。
3	SMBTOE	<b>SMBus SCL 超时探测启用。</b> 此位启用 SCL 低电平超时探测。如设置为逻辑 1，则 SMBus 接口将在 SCL 为高电平时强制重载定时器 3，在 SCL 为低电平时允许定时器 3 计数。如定时器 3 被配置为分割模式，则当 SCL 为高电平时，仅有定时器的高字节被重载。应将定时器 3 编程为每 25 ms 产生一次中断，并使用定时器 3 中断服务程序对 SMBus 通信复位。
2	SMBFTE	<b>SMBus 空闲超时探测启用。</b> 当此位设置为逻辑 1，如 SCL 和 SDA 保持高电平的时间超过 10 个 SMBus 时钟源周期，则总线将被视为空闲。
1:0	SMBCS[1:0]	<b>SMBus 时钟源选择。</b> 这两个位选择用于产生 SMBus 位速率的 SMBus 时钟源。所选设备应根据方程 26.1 配置。 00: 定时器 0 溢出 01: 定时器 1 溢出 10: 定时器 2 高字节溢出 11: 定时器 2 低字节溢出

# C8051F80x-83x

## 26.4.2. SMB0CN 控制寄存器

SMB0CN 用于控制接口和提供状态信息（见 SFR 定义 26.2）。SMB0CN 中的高四位（MASTER、TXMODE、STA 和 STO）组成一个状态向量，可用于转移到服务程序。MASTER 指当前传输期间设备为主设备或从设备。TXMODE 指设备当前正发送或接收数据字节。

STA 和 STO 指自上次 SMBus 中断以来，探测或产生的起始和/或停止条件。STA 和 STO 也用于在作为主设备操作时，产生起始和停止条件。在总线为空闲时，向 STA 写 1 将使 SMBus 接口进入主模式并产生一个起始条件（STA 在起始条件产生之后并未被硬件清除）。在主模式下操作时，向 STO 写 1 将使接口产生一个停止条件并在下一个 ACK 周期之后结束当前的传输。如 STO 和 STA 都被设置（在主模式下），起始条件产生之后将产生一个停止条件。

ARBLOST 位指此接口已在总线竞争中失败。当接口正在传输（主或从）时，可能会出现此情况。在作为从设备操作时，出现此情况表示总线错误。在每次 SI 被清除后，ARBLOST 被硬件清除。

在每次传输开始和结束、每个字节帧之后或竞争失败时，SI 位（SMBus 中断标志）被设置。请参见表 26.3，了解更多详情。

**有关 SI 位的重要事项：**当 SI 被设置时，SMBus 接口暂停工作；而 SCL 保持低电平，总线状态被冻结，直到软件将 SI 清除。

### 26.4.2.1. 软件 ACK 发生器

当寄存器 SMB0ADM 中的 EHACK 位被清 0 时，设备上的固件须探测传入的从地址和 ACK 或 NACK 从地址和传入的数据字节。作为接收器，写 ACK 位确定要发出的 ACK 值；作为发送器，读 ACK 位表示在最后的 ACK 周期中接收到的值。每次接收到字节时设置 ACKRQ，表示需要一个要发出的 ACK 值。当 ACKRQ 被设置，软件应在清除 SI 前向 ACK 位写拟发出的值。如软件未在清除 SI 前写 ACK 位，则将产生一个 NACK。写入 ACK 位之后，SDA 随即反射出已定义的 ACK 值，而 SCL 将保持低电平直至 SI 被清除。如接收到的从地址未被确认，其它的从事件将被忽略，直至探测到下一个起始条件。

### 26.4.2.2. 硬件 ACK 发生器

当寄存器 SMB0ADM 中的 EHACK 位设置为 1 时，将启用从地址自动识别和 ACK 发生器。有关从地址自动识别的更多详情，请参见第 26.4.3 条。作为接收器，ACK 位当前指定的值将在传入数据字节的 ACK 周期中自动发送到总线上。作为发送器，读 ACK 位表示最后的 ACK 周期中接收到的值。启用硬件 ACK 发生器时，不可使用 ACKRQ 位。如接收到的从地址未被硬件确认，其它的从事件将被忽略直至探测到下一个起始条件，且将不会产生任何中断。

表 26.3 列示了影响 SMB0CN 位的所有硬件源。有关使用 SMB0CN 寄存器进行 SMBus 状态解码，请参见表 26.5。

## SFR 定义 26.2: SMB0CN: SMBus 控制寄存器

位	7	6	5	4	3	2	1	0
名称	MASTER	TXMODE	STA	STO	ACKRQ	ARBLOST	ACK	SI
类型	R	R	R/W	R/W	R	R	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xC0; 可位寻址

位	名称	说明	读	写
7	MASTER	<b>SMBus 主/从标志。</b> 此只读位表示 SMBus 是否在主模式下操作。	0: SMBus 在从模式下操作。 1: SMBus 在主模式下操作。	不可用
6	TXMODE	<b>SMBus 发送模式标志。</b> 此只读位表示 SMBus 是否作为发送器操作。	0: SMBus 在接收器模式下。 1: SMBus 在发送器模式下。	不可用
5	STA	<b>SMBus 起始标志。</b>	0: 未探测到起始或重复起始条件。 1: 探测到起始或重复起始条件。	0: 未产生起始条件。 1: 配置为主模式时, 启动起始或重复起始条件。
4	STO	<b>SMBus 停止标志。</b>	0: 未探测到停止条件。 1: 探测到停止条件 (在从模式下) 或挂起 (在主模式下)。	0: 未发送停止条件。 1: 配置为主模式时, 产生停止条件, 以在下一次 ACK 周期后发送。被硬件清除。
3	ACKRQ	<b>SMBus 确认请求。</b>	0: 请求未确认 1: 请求确认	不可用
2	ARBLOST	<b>SMBus 竞争失败标志。</b>	0: 无仲裁错误。 1: 在竞争中失败	不可用
1	ACK	<b>SMBus 确认。</b>	0: 收到 NACK。 1: 收到 ACK。	0: 发送 NACK 1: 发送 ACK
0	SI	<b>SMBus 中断标志。</b> 在表 15.3 所列情况下, 此位被硬件设置。SI 必须被软件清除。SI 被设置时, SCL 保持低电平且 SMBus 暂停工作。	0: 无中断挂起 1: <b>中断挂起</b>	0: 清除中断, 并启动下一状态的机器事件。 1: 强制中断。

**表 26.3: SMB0CN 硬件更换源**

位	在下列情况被硬件设置:	在下列情况被硬件清除:
MASTER	<ul style="list-style-type: none"> <li>产生一个起始条件。</li> </ul>	<ul style="list-style-type: none"> <li>产生一个停止条件。</li> <li>在总线竞争中失败。</li> </ul>
TXMODE	<ul style="list-style-type: none"> <li>产生一个起始条件。</li> <li>在 SMBus 帧开始之前写 SMB0DAT。</li> </ul>	<ul style="list-style-type: none"> <li>探测到一个起始条件。</li> <li>在总线竞争中失败。</li> <li>在 SMBus 帧开始之前未写 SMB0DAT。</li> </ul>
STA	<ul style="list-style-type: none"> <li>起始条件后接收到一个地址字节。</li> </ul>	<ul style="list-style-type: none"> <li>必须被软件清除。</li> </ul>
STO	<ul style="list-style-type: none"> <li>作为从设备被寻址时探测到一个停止条件。</li> <li>由于探测到停止条件, 因此在竞争中失败。</li> </ul>	<ul style="list-style-type: none"> <li>产生一个挂起的停止条件。</li> </ul>
ACKRQ	<ul style="list-style-type: none"> <li>接收到一个字节并需要一个 ACK 响应值 (仅限硬件 ACK 未启用时)。</li> </ul>	<ul style="list-style-type: none"> <li>每个 ACK 周期之后。</li> </ul>
ARBLOST	<ul style="list-style-type: none"> <li>当 STA 为低电平时, 主设备探测到一个重复的起始条件 (不希望的重复起始条件)。</li> <li>在试图产生一个停止条件或重复起始条件时, 探测到 SCL 为低电平。</li> <li>在发送 1 时, 探测到 SDA 为低电平 (ACK 位除外)。</li> </ul>	<ul style="list-style-type: none"> <li>每次 SI 被清除时。</li> </ul>
ACK	<ul style="list-style-type: none"> <li>传入的 ACK 值为低 (确认)。</li> </ul>	<ul style="list-style-type: none"> <li>传入的 ACK 值为高 (非确认)。</li> </ul>
SI	<ul style="list-style-type: none"> <li>已产生一个起始条件。</li> <li>竞争失败。</li> <li>已发送一个字节且接收到 ACK/NACK。</li> <li>已接收到一个字节。</li> <li>在起始或重复起始条件之后接收到一个从地址 + R/W。</li> <li>已接收到一个停止条件。</li> </ul>	<ul style="list-style-type: none"> <li>必须被软件清除。</li> </ul>

### 26.4.3. 硬件从地址识别

SMBus 硬件能自动地识别传入的从地址并在无软件干预的情况下发送 ACK。从地址自动识别通过将寄存器 SMB0ADM 中的 EHACK 位设为 1 启用。这将对接收到的字节启用从地址自动识别和自动硬件 ACK 发生器 (作为主或从设备)。有关自动硬件 ACK 发生器的更多详情, 请参见第 26.4.2.2 条。

用于定义硬件识别地址的寄存器为 SMBus 从地址寄存器 (SFR 定义 26.3) 和 SMBus 从地址屏蔽寄存器 (SFR 定义 26.4)。单个或多个地址 (包括广播定址 0x00) 可使用这两种寄存器指定。两种寄存器最高的七位用于定义被确认的地址。从地址屏蔽寄存器 SLVM[6:0] 的位位置中的 1 就这些位对接收到的从地址和硬件的从地址 SLV[6:0] 进行比较。从地址寄存器的位中的 0 表示此位将不进行比较。在这种情况下, 在传入从地址过程中, 1 或 0 值都可被接受。此外, 如寄存器 SMB0ADR 中的 GC 位设置为 1, 则硬件将识别广播定址 (0x00)。表 26.4 列示了部分示例参数设置以及在该等情况下将被硬件识别的从地址。

表 26.4: 硬件地址识别范例 (EHACK = 1)

硬件从地址 SLV[6:0]	从地址屏蔽寄存器 SLVM[6:0]	GC 位	被硬件识别的从地址
0x34	0x7F	0	0x34
0x34	0x7F	1	0x34、0x00 (广播定址)
0x34	0x7E	0	0x34、0x35
0x34	0x7E	1	0x34、0x35、0x00 (广播定址)
0x70	0x73	0	0x70、0x74、0x78、0x7C

# C8051F80x-83x

## SFR 定义 26.3: SMB0ADR: SMBus 从地址寄存器

位	7	6	5	4	3	2	1	0
名称	SLV[6:0]							GC
类型	R/W							R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xD7

位	名称	功能
7:1	SLV[6:0]	<b>SMBus 硬件从地址。</b> 定义自动硬件确认的 SMBus 从地址。仅 SLVM[6:0] 中相应的位位置中含有 1 的地址位会被检查传入的地址。允许识别多个地址。
0	GC	<b>启用广播定址。</b> 当硬件地址识别启用 (EHACK = 1) 时, 此位将决定广播定址 (0x00) 是否还被硬件识别。 0: 忽略广播定址。 1: 识别广播定址。

## SFR 定义 26.4: SMB0ADM: SMBus 从地址屏蔽寄存器

位	7	6	5	4	3	2	1	0
名称	SLVM[6:0]							EHACK
类型	R/W							R/W
复位	1	1	1	1	1	1	1	0

SFR 地址 = 0xD6

位	名称	功能
7:1	SLVM[6:0]	<b>SMBus 从地址屏蔽。</b> 定义寄存器 SMB0ADR 被用作与传入的地址字节比较的位以及被忽略的位。SLVM[6:0] 中任何设置为 1 的位与 SLV[6:0] 中相应的位比较。设置为 0 的位被忽略 (在传入的地址中可为 0 或 1)。
0	EHACK	<b>启用硬件确认。</b> 启用从地址和接收到的数据字节的硬件确认。 0: 固件须手动确认所有传入的地址和数据字节。 1: 从地址自动确认和硬件确认已启用。

#### 26.4.4. 数据寄存器

SMBus 数据寄存器 SMB0DAT 保存要发送的或刚接收到的串行数据字节。SI 标志被设置时，软件可安全地读或写数据寄存器。当 SMBus 被启用而 SI 标志被清为逻辑 0 时，软件不应访问 SMB0DAT 寄存器，因为接口可能正在将数据字节移入或移出寄存器。

SMB0DAT 中的数据总是先移出 MSB。在接收到一个字节后，接收数据的第一个位位于 SMB0DAT 的 MSB。在数据被移出的同时，总线上的数据被移入。SMB0DAT 总是含有最后出现在总线上的数据字节。一旦竞争失败，从主发送器变为从接收器时 SMB0DAT 中的正确数据或地址保持不变。

#### SFR 定义 26.5: SMB0DAT: SMBus 数据寄存器

位	7	6	5	4	3	2	1	0
名称	SMB0DAT[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xC2

位	名称	功能
7:0	SMB0DAT[7:0]	<p><b>SMBus 数据。</b></p> <p>SMB0DAT 寄存器含有要发送到 SMBus 串行接口上的一个数据字节，或刚从 SMBus 串行接口接收到的一个字节。一旦 SI 串行中断标志 (SMB0CN.0) 设置为逻辑 1，CPU 即可读或写该寄存器。只要 SI 标志被设置，寄存器中的串行数据就保持稳定。当 SI 标志未被设置，系统可能正在移入/移出数据，此时 CPU 不应访问该寄存器。</p>

## 26.5. SMBus 传输模式

SMBus 接口可被配置为在主和/或从模式下操作。在任一时刻，它将在下列四个模式中任意一个下操作：主发送器、主接收器、从发送器或从接收器。SMBus 在产生起始条件时进入主模式，并保持在主模式直至其竞争失败或产生一个停止条件。SMBus 在每个 SMBus 字节帧结束后都产生一个中断。请注意，作为接收器时 ACK 中断的位置取决于硬件 ACK 发生器的启用状况。作为接收器，ACK 中断在硬件 ACK 发生器禁用的情况下于 ACK 之前产生，硬件 ACK 发生器启用时则于 ACK 之后产生。作为发送器，不论硬件 ACK 发生器启用与否，中断都于 ACK 之后产生。

### 26.5.1. 写入顺序（主）

就写入顺序而言，SMBus 主设备向从设备写入数据。在地址字节和所有数据字节传输中，主设备为发送器。SMBus 接口产生起始条件并发送含有目标从设备的地址的第一个字节和数据方向位。此时数据方向位 (R/W) 将为逻辑 0（写）。然后主设备发送一个或多个串行数据字节。每个字节发送之后，从设备将产生确认位。当 STO 位被设置且停止条件产生时，传输结束。请注意，如 SMB0DAT 在主发送器中断后未被写入数据，则接口将切换到主接收器模式。图 26.5 列示了典型的主发送器写入顺序。尽管可传输任意数目的字节，但仅列示了两个传输的数据字节。须注意，不论硬件 ACK 发生器启用与否，所有“数据字节传输结束”中断都于此模式下 ACK 周期之后产生。

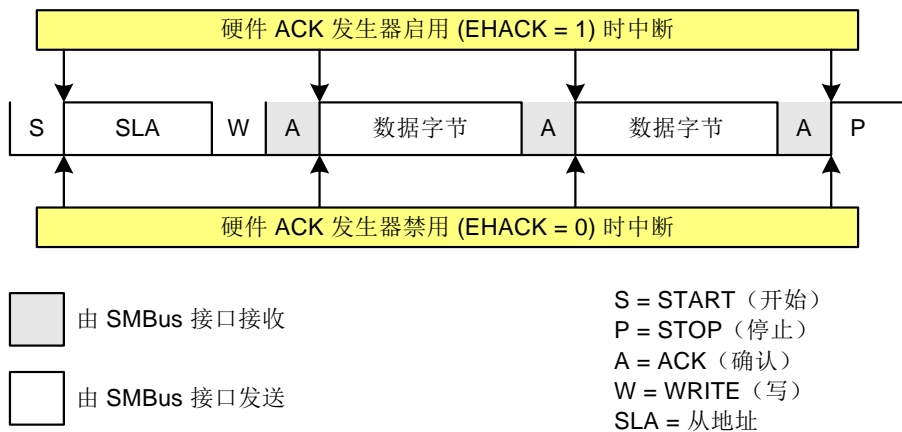


图 26.5. 典型主发送器写入顺序



### 26.5.2. 读取顺序（主）

就读取顺序而言，SMBus 主设备自从设备读取数据。在地址字节和所有数据字节传输中，主设备为接收器。SMBus 接口产生起始条件并发送含有目标从设备的地址的第一个字节和数据方向位。此时数据方向位 (R/W) 将为逻辑 1（读）。然后，SMBus 输出串行时钟的同时，串行数据自 SDA 上的从设备接收。从设备发送一个或多个串行数据字节。

如禁用硬件 ACK 发生器，则 ACKRQ 设置为 1 且在每收到一个字节后产生一个中断。那时软件必须写 ACK 位，以确认或不确认接收到的字节。

在启用硬件 ACK 发生器的情况下，SMBus 硬件将自动产生 ACK/NACK，然后发出中断。须注意，启用硬件 ACK 发生器时，在接收字节之前，软件应设置适当的 ACK 或 NACK 值。

向 ACK 位写 1 产生一个 ACK，写 0 则产生一个 NACK。软件应在接收到最后的数据后向 ACK 写 0，以发送 NACK。接口将在 STO 位被设置且产生停止条件后，退出主接收器模式。如在主接收器模式下执行 SMBODAT 写操作，接口将切换到主发送器模式。图 26.6 列示了典型的主接收器读取顺序。尽管可接收任意数目的字节，但仅列示了两个接收的数据字节。须注意，‘数据字节传输结束’中断发生于顺序中的不同位置，视乎硬件 ACK 发生器启用情况而定。中断在硬件 ACK 发生器禁用的情况下于 ACK 之前产生，硬件 ACK 发生器启用时则于 ACK 之后产生。

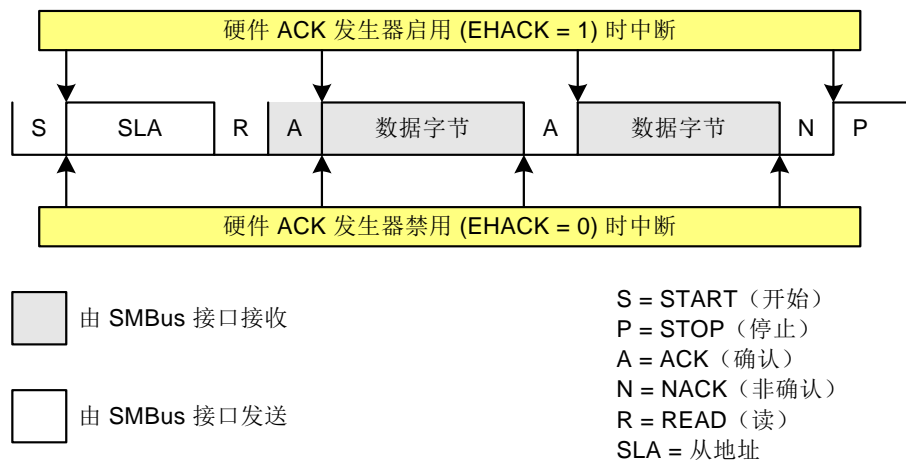


图 26.6. 典型主接收器读取顺序

## 26.5.3. 写入顺序（从）

就写入顺序而言，SMBus 主设备向从设备写入数据。在地址字节和所有数据字节传输中，从设备为接收器。在从事件被允许的情况下 (INH = 0)，当接收到一个起始条件和一个含有从地址和方向位（此处应为写）的字节时，接口进入从接收器模式。如硬件 ACK 发生器被禁用，则进入从接收器模式后，将产生一个中断且 ACKRQ 位被设置。软件须用一个 ACK 对接收到的从地址确认，或用一个 NACK 忽略接收到的从地址。如硬件 ACK 发生器被启用，硬件将对与 SMB0ADR 和 SMB0ADM 设置的标准相符的从地址回复 ACK。ACK 周期之后将产生中断。

如接收到的从地址被（软件或硬件）忽略，从中断将被禁止直至探测到下一个起始条件。如接收到的从地址被确认，则将收到零或更多数据字节。

如禁用硬件 ACK 发生器，则 ACKRQ 设置为 1 且在每收到一个字节后产生一个中断。那时软件必须写 ACK 位，以确认或不确认接收到的字节。

在启用硬件 ACK 发生器的情况下，SMBus 硬件将自动产生 ACK/NACK，然后发出中断。**须注意，启用硬件 ACK 发生器时，在接收字节之前，软件应设置适当的 ACK 或 NACK 值。**

收到一个 STOP 之后，接口退出从接收器模式。请注意，如在从接收器模式下执行 SMB0DAT 写操作，接口将切换到从发送器模式。图 26.7 列示了典型的从接收器写入顺序。尽管可接收任意数目的字节，但仅列示了两个接收的数据字节。须注意，“数据字节传输结”中断发生于顺序中的不同位置，视乎硬件 ACK 发生器启用情况而定。中断在硬件 ACK 发生器禁用的情况下于 ACK 之前产生，硬件 ACK 发生器启用时则于 ACK 之后产生。

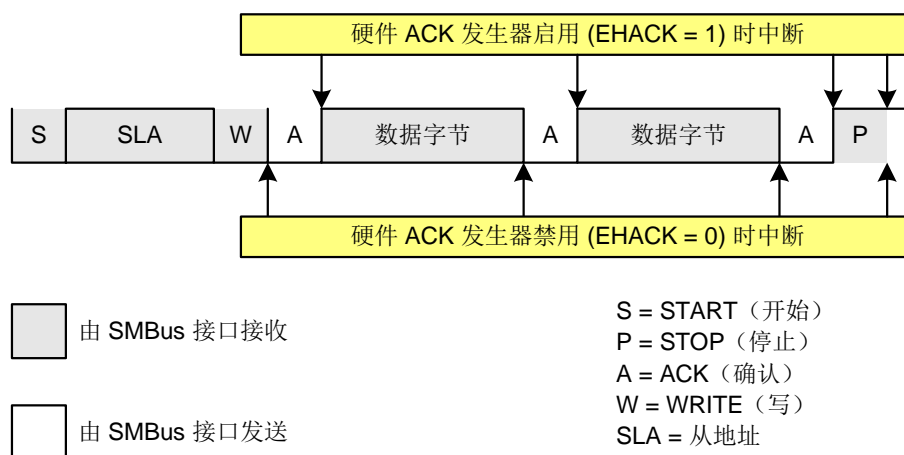


图 26.7. 典型从接收器写入顺序

#### 26.5.4. 读取顺序（从）

就读取顺序而言，SMBus 主设备自从设备读取数据。在地址字节传输中，从设备为接收器；在所有数据字节传输中，从设备为发送器。在从事件被允许的情况下 ( $INH = 0$ )，当接收到一个起始条件和一个含有从地址和方向位（此处应为读）的字节时，接口进入从接收器模式（以接收从地址）。如硬件 ACK 发生器被禁用，则进入从接收器模式后，将产生一个中断且 ACKRQ 位被设置。软件须用一个 ACK 对接收到的从地址确认，或用一个 NACK 忽略接收到的从地址。如硬件 ACK 发生器被启用，硬件将对与 SMB0ADR 和 SMB0ADM 设置的标准相符的从地址回复 ACK。ACK 周期之后将产生中断。

如接收到的从地址被（软件或硬件）忽略，从中断将被禁止直至探测到下一个起始条件。如接收到的从地址被确认，则将发送零或更多数据字节。如接收到的从地址被确认，则数据应被写入 SMB0DAT 以待发送。接口进入从发送器模式，且会发送一个或多个数据字节。发送每个字节之后，主设备发出确认位，如确认位为 ACK，则应向 SMB0DAT 写入下一个数据字节。如确认位为 NACK，则在 SI 被清除之前不应再写 SMB0DAT（在从发送器模式下，如在收到 NACK 后写 SMB0DAT，可能会产生一个错误条件）。收到一个 STOP 之后，接口退出从发送器模式。请注意，如 SMB0DAT 在从发送器中断后未被写入数据，则接口将切换到从接收器模式。图 26.8 列示了典型的从发送器读取顺序。尽管可发送任意数目的字节，但仅列示了两个发送的数据字节。须注意，不论硬件 ACK 发生器启用与否，所有“数据字节传输结束”中断都于此模式下 ACK 周期之后产生。

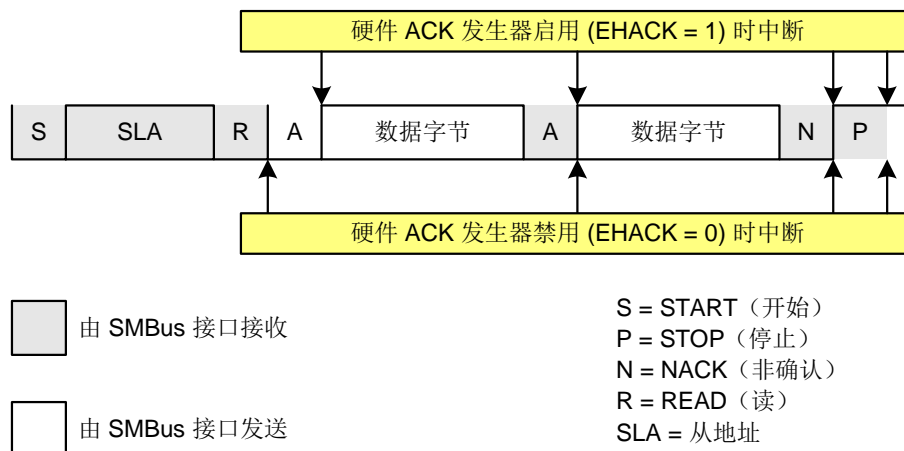


图 26.8. 典型从发送器读取顺序

## 26.6. SMBus 状态解码

使用 SMB0CN 寄存器可对 SMBus 当前的状态轻松解码。为响应 SMBus 事件而进行的适当操作视乎硬件从地址识别和 ACK 发生器的启用状况而定。表 26.5 列出了硬件从地址识别和 ACK 发生器禁用时典型的操作。表 26.6 列出了硬件从地址识别和 ACK 发生器启用时典型的操作。在下列表中，状态向量指 SMB0CN 的高四位：MASTER、TXMODE、STA 和 STO。表中仅列出了典型的响应选项，只要符合 SMBus 规格，特定应用程序就会被允许。突出显示的响应选项是被硬件允许的，但不符合 SMBus 规格。

# C8051F80x-83x

表 26.5: SMBus 状态解码 (禁用硬件 ACK 发生器) (EHACK = 0)

模式	读取值			SMBus 当前状态	典型响应选项	写入值			下一个状态向量	
	状态向量	ACKRQ	ARBLOST			ACK	STA	STO		ACK
主发送器	1110	0	0	X	主起始条件已产生。	将从地址 + R/W 装入 SMB0DAT。	0	0	X	1100
	1100	0	0	0	已发送主数据或地址字节; 收到 NACK。	设置 STA, 重新启动数据传输。	1	0	X	1110
						放弃发送。	0	1	X	—
		0	0	1	已发送主数据或地址字节; 收到 ACK。	将下一个数据字节装入 SMB0DAT。	0	0	X	1100
						并用停止条件结束数据传输。	0	1	X	—
						用停止条件结束数据传输并开始另一次传输。	1	1	X	—
						发送重复起始条件。	1	0	X	1110
				切换至主接收器模式 (清除 SI, 不向 SMB0DAT 写新的数据)。	0	0	X	1000		
主接收器	1000	1	0	X	收到主数据字节; 请求确认。	确认接收字节; 读 SMB0DAT。	0	0	1	1000
						发送 NACK 表示最后一个字节, 并发送停止条件。	0	1	0	—
						发送 NACK 表示最后一个字节, 并发送停止条件, 然后发送起始条件。	1	1	0	1110
						发送 ACK 后发送重复起始条件。	1	0	1	1110
						发送 NACK 表示最后一个字节, 并发送重复起始条件。	1	0	0	1110
						发送 ACK 并切换至主发送器模式 (清除 SI 前写入 SMB0DAT)。	0	0	1	1100
						发送 NACK 并切换至主发送器模式 (清除 SI 前写入 SMB0DAT)。	0	0	0	1100

表 26.5: SMBus 状态解码 (禁用硬件 ACK 发生器) (EHACK = 0) (Continued)

模式	读取值			SMBus 当前状态	典型响应选项	写入值			下一个状态向量	
	状态向量	ACKRQ	ARBLOST			ACK	STA	STO		ACK
从发送器	0100	0	0	0	从字节已发送; 收到 NACK。	0	0	X	0001	
		0	0	1	从字节已发送; 收到 ACK。	0	0	X	0100	
		0	1	X	从字节已发送; 探测到错误。	0	0	X	0001	
	0101	0	X	X	从传输过程中探测到非法的停止条件或总线错误。	0	0	X	—	
从接收器	0010	1	0	X	收到从地址 + R/W; 请求确认。	如写, 确认接收到的地址	0	0	1	0000
					如读, 将数据字节装入 SMBODAT; 确认接收到的地址	0	0	1	0100	
					不确认接收到的地址。	0	0	0	—	
	0010	1	1	X	作为主设备竞争失败; 收到从地址 + R/W; 请求确认。	如写, 确认接收到的地址	0	0	1	0000
					如读, 将数据字节装入 SMBODAT; 确认接收到的地址	0	0	1	0100	
					不确认接收到的地址。	0	0	0	—	
	0001	0	0	X	作为从发送器或从接收器被寻址时探测到停止条件。	清除 STO。	0	0	X	—
					试图发送停止条件时竞争失败。	无需任何操作 (传输完成/放弃)。	0	0	0	—
					收到从数据字节; 请求确认。	确认接收字节; 读 SMBODAT。	0	0	1	0000
	总线错误条件	0010	0	1	X	试图发送重复起始条件时竞争失败。	放弃失败的传输。	0	0	X
重新启动失败的传输。						1	0	X	1110	
0001		0	1	X	由于探测到停止条件, 因此在竞争中失败。	放弃失败的传输。	0	0	X	—
					重新启动失败的传输。	1	0	X	1110	
0000	1	1	X	作为主设备发送数据字节时竞争失败。	放弃失败的传输。	0	0	0	—	
				重新启动失败的传输。	1	0	0	1110		

# C8051F80x-83x

表 26.6: SMBus 状态解码 (启用硬件 ACK 发生器) (EHACK = 1)

模式	读取值			SMBus 当前状态	典型响应选项	写入值			下一个状态向量	
	状态向量	ACKRQ	ARBLOST			ACK	STA	STO		ACK
主发送器	1110	0	0	X	主起始条件已产生。	将从地址 + R/W 装入 SMBODAT。	0	0	X	1100
	1100	0	0	0	已发送主数据或地址字节; 收到 NACK。	设置 STA, 重新启动数据传输。	1	0	X	1110
						放弃发送。	0	1	X	—
		0	0	1	已发送主数据或地址字节; 收到 ACK。	将下一个数据字节装入 SMBODAT。	0	0	X	1100
						并用停止条件结束数据传输。	0	1	X	—
						用停止条件结束数据传输并开始另一次传输。	1	1	X	—
						发送重复起始条件。	1	0	X	1110
切换至主接收器模式 (清除 SI, 不向 SMBODAT 写新的数据)。将首个数据字节设为 ACK。	0	0	1	1000						
主接收器	1000	0	0	1	收到主数据字节; 发送 ACK。	将下一个数据字节设为 ACK; 读 SMBODAT。	0	0	1	1000
						设为 NACK 表示下一个数据字节为最后的数据字节; 读 SMBODAT。	0	0	0	1000
						启动重复起始条件。	1	0	0	1110
						切换至主发送器模式 (清除 SI 前写入 SMBODAT)。	0	0	X	1100
	0	0	0	0	收到主数据字节; 发送 NACK (最后的字节)。	读 SMBODAT; 发送停止条件。	0	1	0	—
						读 SMBODAT; 发送停止条件, 然后发送起始条件。	1	1	0	1110
						启动重复起始条件。	1	0	0	1110
切换至主发送器模式 (清除 SI 前写入 SMBODAT)。	0	0	X	1100						

表 26.6: SMBus 状态解码 (启用硬件 ACK 发生器) (EHACK = 1) (Continued)

模式	读取值			SMBus 当前状态	典型响应选项	写入值			下一个状态向量	
	状态向量	ACKRQ	ARBLOST			ACK	STA	STO		ACK
从发送器	0100	0	0	0	从字节已发送; 收到 NACK。	无需用任何操作 (等待停止条件)。	0	0	X	0001
		0	0	1	从字节已发送; 收到 ACK。	将下一个要发送的字节装入 SMBODAT。	0	0	X	0100
		0	1	X	从字节已发送; 探测到错误。	无需用任何操作 (等待主设备结束传输)。	0	0	X	0001
	0101	0	X	X	从传输过程中探测到非法的停止条件或总线错误。	清除 STO。	0	0	X	—
从接收器	0010	0	0	X	收到从地址 + R/W; 发送 ACK。	如写, 将第一个数据字节设为 ACK。	0	0	1	0000
					如读, 将数据字节装入 SMBODAT	0	0	X	0100	
		0	1	X	作为主设备竞争失败; 收到从地址 + R/W; 发送 ACK。	如写, 将第一个数据字节设为 ACK。	0	0	1	0000
					如读, 将数据字节装入 SMBODAT	0	0	X	0100	
	0001	0	0	X	作为从发送器或从接收器被寻址时探测到停止条件。	清除 STO。	0	0	X	—
					试图发送停止条件时竞争失败。	无需用任何操作 (传输完成/放弃)。	0	0	0	—
	0000	0	0	X	收到从字节。	将下一个数据字节设为 ACK; 读 SMBODAT。	0	0	1	0000
					将下一个数据字节设为 NACK; 读 SMBODAT。	0	0	0	0000	
总线错误条件	0010	0	1	X	试图发送重复起始条件时竞争失败。	放弃失败的传输。	0	0	X	—
					重新启动失败的传输。	1	0	X	1110	
	0001	0	1	X	由于探测到停止条件, 因此在竞争中失败。	放弃失败的传输。	0	0	X	—
					重新启动失败的传输。	1	0	X	1110	
	0000	0	1	X	作为主设备发送数据字节时竞争失败。	放弃失败的传输。	0	0	X	—
					重新启动失败的传输。	1	0	X	1110	

# C8051F80x-83x

---



## 27. UART0

UART0 是一个异步、全双工串口，提供标准 8051 UART 的模式 1 和 3。UART0 具有增强型波特率发生器电路，有多个时钟源可用于产生标准波特率。（详情请参见本页“27.1. 增强型波特率发生器”一条）。接收数据缓冲机制允许 UART0 在软件完成读取前一个数据字节之前开始接收第二个传入的数据字节。

UART0 有两个相关的 SFR：串行控制寄存器 0 (SCON) 和串行数据缓冲器 0 (SBUF0)。用同一个 SBUF0 地址可访问发送寄存器和接收寄存器。**写 SBUF0 时通常访问发送寄存器。读 SBUF0 时通常访问缓冲的接收寄存器；不可能从发送寄存器读取数据。**

如 UART0 中断被允许，则每次发送完成（SCON0 中的 TI0 被设置）或接收到数据字节（SCON0 中的 RI0 被设置）时会产生一个中断。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手动清除，这就允许软件查询 UART0 中断的原因（发送完成或接收完成）。

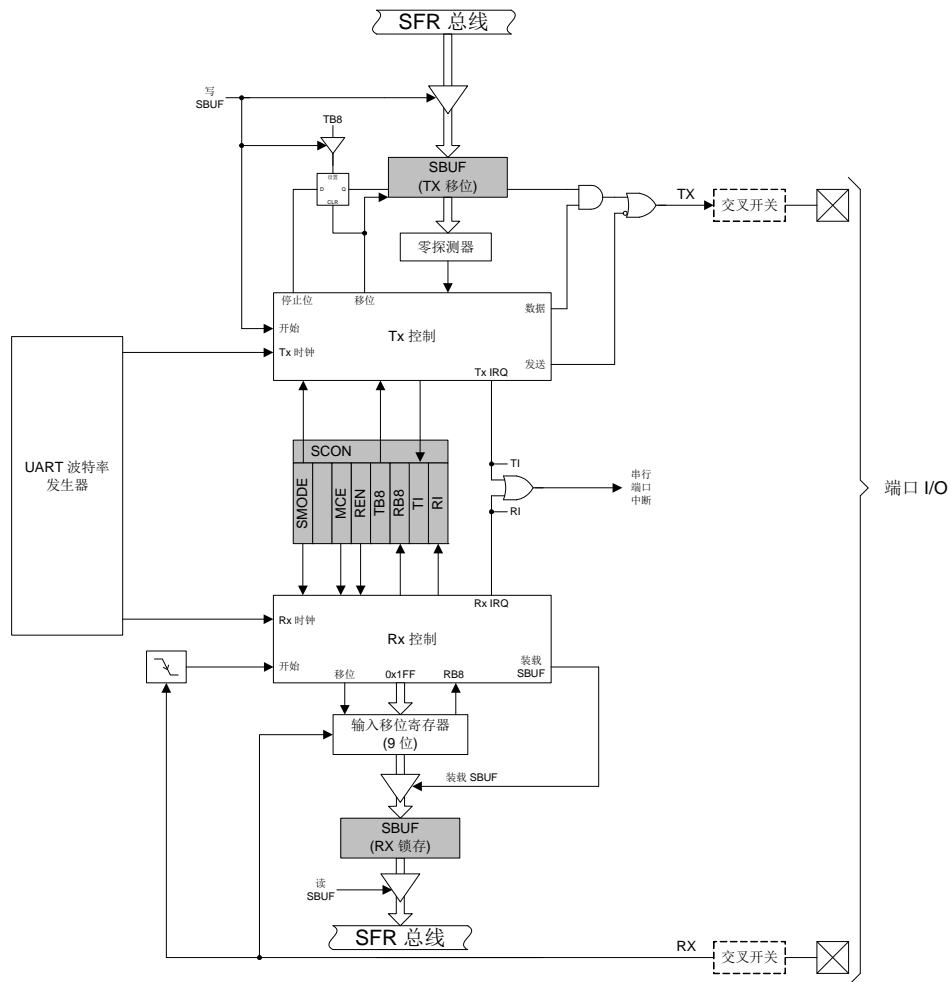


图 27.1. UART0 结构图

## 27.1. 增强型波特率发生器

UART0 波特率由定时器 1 在 8 位自动重载的模式下产生。TX 时钟由 TL1 产生；RX 时钟由 TL1 的拷贝寄存器（图 27.2 所示的 RX 定时器）产生，该寄存器不被用户访问。TX 和 RX 定时器溢出信号经过二分频后用于产生 TX 和 RX 波特率。定时器 1 被启用时，RX 定时器运行并使用相同的重载值 (TH1)。而当 RX 引脚上探测到起始条件时，RX 定时器被强制重载。这允许在探测到起始条件时开始接收过程，而与 TX 定时器的状态无关。

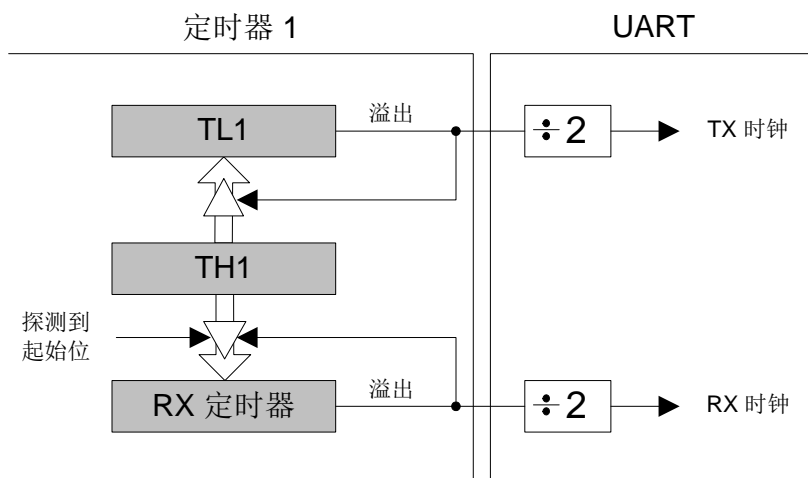


图 27.2. UART0 波特率逻辑

定时器 1 应被配置为模式 2，即 8 位自动重载模式（请参见第 210 页“28.1.3. 模式 2：8 位自动重载计数器/定时器”一条）。定时器 1 的重载值应被设置，以使溢出频率为所期望的 UART 波特率频率的两倍。请注意，定时器 1 的时钟可在下列六个时钟源中选择：SYSCLK、SYSCLK/4、SYSCLK/12、SYSCLK/48、外部振荡器时钟/8 或外部输入 T1。就任何给定的定时器 1 时钟源而言，UART0 波特率由方程 27.1-A 和方程 27.1-B 决定。

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

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

### 方程 27.1: UART0 波特率

其中  $T1_{CLK}$  是定时器 1 的时钟频率，而 TH1 是定时器 1 的高字节（重载值）。定时器 1 时钟频率的选择方法见第 207 页“28. 定时器”一条。表 27.1 至 27.2 载列了典型波特率和系统时钟频率的快速对照表。在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。

## 27.2. 操作模式

UART0 提供标准的异步、全双工通信。UART 模式（8 位或 9 位）通过 S0MODE 位 (SCON0.7) 来选择。图 27.3 展示了典型的 UART 连接选项。

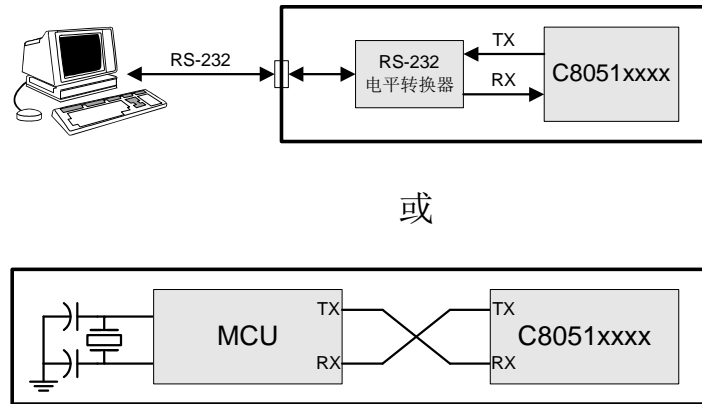


图 27.3. UART 互连图

### 27.2.1. 8 位 UART

在 8 位 UART 模式下，每个数据字节共使用 10 位：一个起始位、八个数据位（LSB 在先）和一个停止位。数据（LSB 在先）从 TX0 针脚发送，在 RX0 针脚接收。接收时，八个数据位存入 SBUF0，停止位进入 RB80 (SCON0.2)。

当软件向 SBUF0 寄存器写入一个数据字节时，数据传输开始。在数据传输结束时（停止位开始），TI0 发送中断标志 (SCON0.1) 被设置。REN0 接收允许位 (SCON0.4) 设置为逻辑 1 之后，数据接收便可开始。收到停止位后，如满足下列条件，则数据字节将载入 SBUF0 接收寄存器：RI0 须为逻辑 0，如 MCE0 为逻辑 1，则停止位须为逻辑 1。在接收数据溢出的情况下，先接收到的 8 位数据将被锁存到 SBUF0 接收寄存器，而后面溢出的数据位则被丢弃。

如满足这些条件，则八位数据将被存入 SBUF0，停止位被存入 RB80，RI0 标志被设置。如不满足这些条件，则不装入 SBUF0 和 RB80，RI0 标志也不会被设置。如中断被允许，则在 TI0 或 RI0 被设置时将产生一个中断。

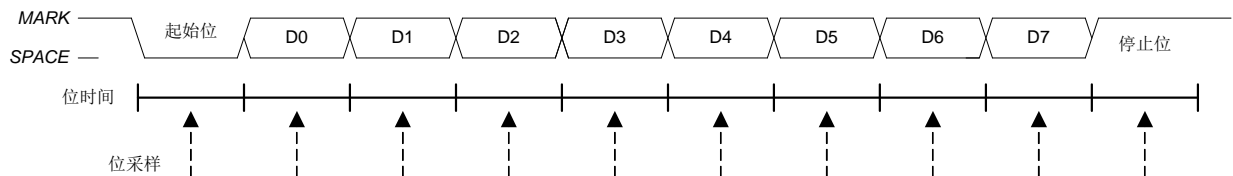


图 27.4. 8 位 UART 定时图

# C8051F80x-83x

## 27.2.2. 9 位 UART

在 9 位 UART 模式下，每个数据字节共使用十一位：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。可被赋值为寄存器 PSW 中的奇偶位 P（用于错误探测），或用于多处理器通信。接收时，第九位数据进入 RB80 (SCON0.2)，停止位被忽略。

当软件向 SBUF0 寄存器写入一个数据字节的指令时，数据传输开始。在数据传输结束时（停止位开始），TI0 发送中断标志 (SCON0.1) 被设置。REN0 接收允许位 (SCON0.4) 设置为 1 之后，数据接收便可开始。收到停止位后，如满足下列条件，则数据字节将载入 SBUF0 接收寄存器：(1) RI0 须为逻辑 0，(2) 如 MCE0 为逻辑 1，则第九位须为逻辑 1（MCE0 为逻辑 0 时，第九位数据的状态并不重要）。如满足这些条件，则八位数据将被存入 SBUF0，第九位被存入 RB80，RI0 标志设置为 1。如不满足这些条件，则不装入 SBUF0 和 RB80，RI0 标志也不会设置为 1。如 UART0 中断被允许，则在 TI0 或 RI0 设置为 1 时将产生一个中断。

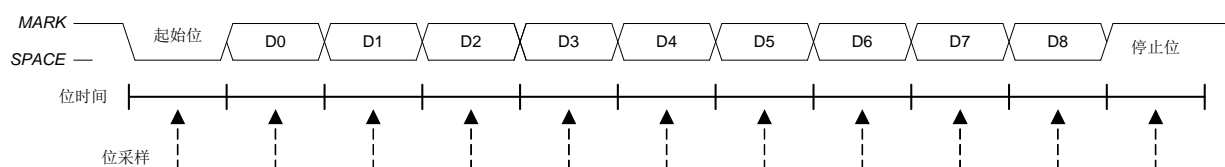


图 27.5. 9 位 UART 定时图

### 27.3. 多处理器通信

9 位 UART 模式通过第九位数据的特殊使用，支持一个主处理器和一个或多个从处理器之间的多处理器通信。当主处理器要发送数据至一个或多个从处理器时，首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：地址字节的第九位为逻辑 1；而数据字节的第九位总是设置为逻辑 0。

将从处理器的 MCE0 位 (SCON0.5) 设置为其 UART，以便在收到停止位且仅在该第九位为逻辑 1 ( $RB80 = 1$ ) (标志着收到地址字节) 时，UART 产生一个中断。在 UART 中断处理程序中，软件将接收到的地址与从处理器自身配备的 8 位地址进行比较。如地址匹配，则从处理器将清除其 MCE0 位，以允许后面接收数据字节时产生中断。未被寻址的从处理器仍保持其 MCE0 位的设置，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。一旦接收完整消息，被寻址的从处理器复位其 MCE0 位，以忽略所有数据传输，直至收到下一个地址字节。

可将多个地址分配给一个从处理器，和/或将一个地址分配给多个从处理器，从而允许同时向多个从处理器"广播"发送。主处理器可被配置为接收所有传输数据，或通过实现某种协议使主/从角色能临时变换，以允许原来的主处理器和从处理器之间进行半双工通信。

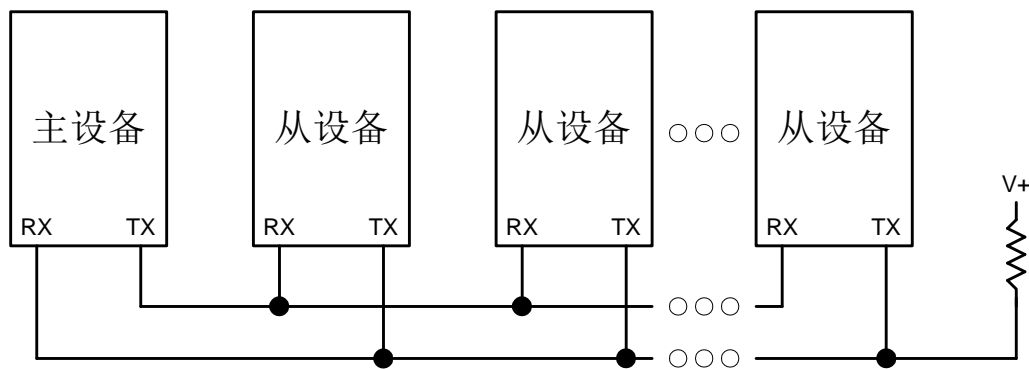


图 27.6. UART 多处理器模式互连图

# C8051F80x-83x

## SFR 定义 27.1: SCON0: 串行端口 0 控制寄存器

位	7	6	5	4	3	2	1	0
名称	S0MODE		MCE0	REN0	TB80	RB80	TI0	RI0
类型	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	1	0	0	0	0	0	0

SFR 地址 = 0x98; 可位寻址

位	名称	功能
7	S0MODE	<b>串行端口 0 操作模式。</b> 选择 UART0 操作模式。 0: 波特率可变的 8 位 UART。 1: 波特率可变的 9 位 UART。
6	未使用	读 = 1b, 写 = 无需在意。
5	MCE0	<b>多处理器通信允许。</b> 此位的功能视串行端口 0 操作模式而定: <b>模式 0: 检查有效停止位。</b> 0: 停止位的逻辑电平被忽略。 1: 仅在停止位为逻辑电平 1 时 RI0 方被激活。 <b>模式 1: 多处理器通信允许。</b> 0: 第 9 位的逻辑电平被忽略。 1: 仅在第 9 位为逻辑 1 时, RI0 才被设置并产生一个中断。
4	REN0	<b>接收允许。</b> 0: UART0 接收禁用。 1: UART0 接收允许。
3	TB80	<b>第 9 传输位。</b> 此位的逻辑电平将作为第 9 传输位并以 9 位 UART 模式 (模式 1) 发送。在 8 位模式中未使用 (模式 0)。
2	RB80	<b>第 9 接收位。</b> 在模式 0 下, RB80 被赋予停止位值; 在模式 1 下, 被赋予第 9 数据位值。
1	TI0	<b>传输中断标志。</b> 当 UART0 传输一个字节数据后, 此位被硬件设置 (在 8 位 UART 模式下, 是在传输第 8 位以后, 或在 9 位 UART 模式下, 是在停止位开始时)。当 UART0 中断被允许时, 设置此位会导致 CPU 转至 UART0 中断服务程序。此位必须用软件手动清除。
0	RI0	<b>接收中断标志。</b> 当 UART0 接收到一个字节数据时, 此位被硬件设置为 1 (设置于停止位采样时)。当 UART0 中断被允许时, 将此位设置为 1 会导致 CPU 转至 UART0 中断服务程序。此位必须用软件手动清除。

---

**SFR 定义 27.2: SBUF0: 串行 (UART0) 端口数据缓冲器**


---

位	7	6	5	4	3	2	1	0
名称	SBUF0[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x99

位	名称	功能
7:0	SBUF0[7:0]	<p><b>串行数据缓冲器位 7-0 (MSB-LSB)。</b></p> <p>SFR 访问两个寄存器：发送移位寄存器和接收锁存寄存器。当数据被写入 SBUF0 时，数据会进入发送移位寄存器，等待串行传输。向 SBUF0 写入一个字节即启动传输过程。读 SBUF0 时返回接收锁存器的内容。</p>

表 27.1: 使用内部 24.5 MHz 振荡器进行标准波特率的定时器设置

频率: 24.5 MHz							
	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 (预分频选择) <sup>1</sup>	T1M <sup>1</sup>	定时器 1 重载值 (十六进制)
SYSCLK 源自 内部振荡器	230400	-0.32%	106	SYSCLK	XX <sup>2</sup>	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

附注:

1. SCA1-SCA0 和 T1M 位的定义见第 28.1 条。
2. X = 无需在意。

表 27.2: 使用外部 22.1184 MHz 振荡器进行标准波特率的定时器设置

频率: 22.1184 MHz							
	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 (预分频选择) <sup>1</sup>	T1M <sup>1</sup>	定时器 1 重载值 (十六进制)
SYSCLK 源自 外部振荡器	230400	0.00%	96	SYSCLK	XX <sup>2</sup>	1	0xD0
	115200	0.00%	192	SYSCLK	XX	1	0xA0
	57600	0.00%	384	SYSCLK	XX	1	0x40
	28800	0.00%	768	SYSCLK / 12	00	0	0xE0
	14400	0.00%	1536	SYSCLK / 12	00	0	0xC0
	9600	0.00%	2304	SYSCLK / 12	00	0	0xA0
	2400	0.00%	9216	SYSCLK / 48	10	0	0xA0
	1200	0.00%	18432	SYSCLK / 48	10	0	0x40
SYSCLK 源自 内部振荡器	230400	0.00%	96	EXTCLK / 8	11	0	0xFA
	115200	0.00%	192	EXTCLK / 8	11	0	0xF4
	57600	0.00%	384	EXTCLK / 8	11	0	0xE8
	28800	0.00%	768	EXTCLK / 8	11	0	0xD0
	14400	0.00%	1536	EXTCLK / 8	11	0	0xA0
	9600	0.00%	2304	EXTCLK / 8	11	0	0x70

附注:

1. SCA1-SCA0 和 T1M 位的定义见第 28.1 条。
2. X = 无需在意。



## 28. 定时器

每个 MCU 内部都有三个计数器/定时器：其中两个与标准 8051 中的 16 位计数器/定时器兼容，一个是 16 位自动重载定时器，用于 ADC、SMBus 或作为通用定时器使用。这些定时器可用于测量时间间隔、对外部事件计数及生成周期性的中断请求。定时器 0 和定时器 1 几乎完全相同，有四种主要操作模式。定时器 2 具有 16 位和两个 8 位定时器的功能，并可自动重载。此外，定时器 2 还具有在装置暂停时由外部振荡器定时的功能，并可用作唤醒源。可借此实现超低能耗系统，包括 RTC 功能。

定时器 0 和定时器 1 模式	定时器 2 模式
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（仅限定定时器 0）	

定时器 0 和 1 有五个可选的时钟源，由定时器模式选择位 (T1M- T0M) 和时钟分频位 (SCA1-SCA0) 确定。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（请参见 SFR 定义 28.1，了解预分频时钟选择）。

定时器 0/1 可经配置使用此预分频时钟信号或系统时钟。定时器 2 可使用系统时钟、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 还可用作计数器。如用作计数器，则每当所选输入引脚 (T0 或 T1) 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。输入信号无需具周期性，但在给定电平上保持的时间至少应为两个完整系统时钟周期，以确保电平能够被正确采样。

# C8051F80x-83x

## SFR 定义 28.1: CKCON: 时钟控制寄存器

位	7	6	5	4	3	2	1	0
名称			T2MH	T2ML	T1M	T0M	SCA[1:0]	
类型	R	R	R/W	R/W	R/W	R/W	R/W	
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x8E

位	名称	功能
7:6	未使用	读 = 0b; 写 = 无需在意
5	T2MH	<b>定时器 2 高字节时钟选择。</b> 选择供给定时器 2 高字节的时钟（仅在两个 8 位定时器模式下）。 0: 定时器 2 高字节使用 TMR2CN 中 T2XCLK 位定义的时钟。 1: 定时器 2 高字节使用系统时钟。
4	T2ML	<b>定时器 2 低字节时钟选择。</b> 选择供给定时器 2 的时钟。如果定时器 2 被配置为两个 8 位定时器模式，则此位选择供给低 8 位定时器的时钟。 0: 定时器 2 低字节使用 TMR2CN 中 T2XCLK 位定义的时钟。 1: 定时器 2 低字节使用系统时钟。
3	T1	<b>定时器 1 时钟选择。</b> 选择供给定时器 1 的时钟源。当 C/T1 被设置为 1 时忽略。 0: 定时器 1 使用预分频位 SCA[1:0] 定义的时钟。 1: 定时器 1 使用系统时钟。
2	T0	<b>定时器 0 时钟选择。</b> 选择供给定时器 0 的时钟源。当 C/T0 被设置为 1 时忽略。 0: 计数器/定时器 0 使用预分频位 SCA[1:0] 定义的时钟。 1: 计数器/定时器 0 使用系统时钟。
1:0	SCA[1:0]	<b>定时器 0/1 预分频位。</b> 这些位控制定时器 0/1 时钟预分频数： 00: 系统时钟/12 01: 系统时钟/4 10: 系统时钟/48 11: 外部时钟/8（与系统时钟同步）

## 28.1. 定时器 0 和定时器 1

每个定时器都用作一个 16 位的寄存器，在被访问时以两个单独字节的形式出现：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) 用于允许定时器 0 和定时器 1 以及指示状态。可通过设置 IE 寄存器中的 ET0 位允许定时器 0 中断（第 104 页“18.2 中断寄存器说明”）；通过设置 IE 寄存器中的 ET1 位允许定时器 1 中断（第 104 页“18.2 中断寄存器说明”）。这两个计数器/定时器都有四种主要模式，通过设置计数器/定时器模式寄存器 (TMOD) 中的模式选位 T1M1-T0M0 来选择。每个定时器均可独立配置。下面对每种操作模式进行说明。

### 28.1.1. 模式 0：13 位计数器/定时器

在模式 0 下，定时器 0 和定时器 1 用作 13 位计数器/定时器。下面介绍了定时器 0 的配置和操作。但由于这两个定时器在操作上完全相同，定时器 1 的配置方法与定时器 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 位位置保持五个 LSB。TL0 的高 3 位 (TL0.7-TL0.5) 是不确定的，在读取时应屏蔽掉或忽略这 3 位。当 13 位定时器寄存器增加并从 0x1FFF（全 1）溢出到 0x0000 时，会在 TCON 设置定时器溢出标志 TF0，并且如果定时器 0 中断允许，会产生一个中断。

TMOD 寄存器中的 C/T0 位选择计数器/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（请参阅第 142 页“23.3. 优先级交叉开关编码器”，了解关于选择和配置外部 I/O 引脚的信息）。清除 C/T 将选择由 CKCON 寄存器中 T0M 位定义的时钟。当 T0M 被设置时，定时器 0 的时钟源为系统时钟；当 T0M 位被清除时，定时器 0 的时钟源由 CKCON（请参见 SFR 定义 28.1）中的时钟分频位定义。

当 TMOD 寄存器中的 GATE0 为逻辑 0 或输入信号  $\overline{\text{INT0}}$  有效时（有效电平由 IT01CF 寄存器中的 IN0PL 位定义，请参见 SFR 定义 18.7），设置 TR0 位 (TCON.4) 将启动定时器。将 GATE0 设置为 1 将允许定时器受外部输入信号  $\overline{\text{INT0}}$  的控制（请参见第 104 页“18.2. 中断寄存器说明”），便于脉冲宽度测量。

TR0	GATE0	$\overline{\text{INT0}}$	计数器/定时器
0	X	X	禁用
1	0	X	允许
1	1	0	禁用
1	1	1	允许
附注：X = 无需在意			

设置 TR0 并不强制定时器复位。应在定时器被允许前将所需要的初值载入定时器寄存器。

TL1 和 TH1 构成定时器 1 的 13 位寄存器的方式与上述的 TL0 和 TH0 相同。定时器 1 的配置和控制方法与定时器 0 一样，使用对应的 TCON 和 TMOD 位。输入信号  $\overline{\text{INT1}}$  为定时器 1 所用， $\overline{\text{INT1}}$  的极性由 IT01CF 寄存器中的 IN1PL 位定义（请参见 SFR 定义 18.7）。

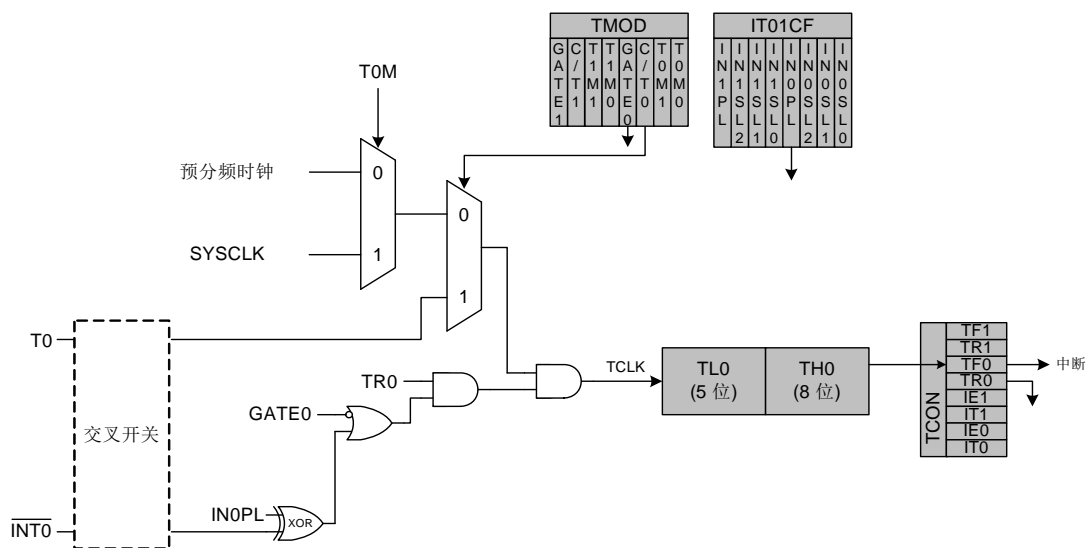


图 28.1. T0 模式 0 结构图

### 28.1.2. 模式 1：16 位计数器/定时器

模式 1 的操作与模式 0 相同，只是计数器/定时器寄存器使用全部 16 位。在模式 1 下允许和控制计数器/定时器的方法与模式 0 相同。

### 28.1.3. 模式 2：8 位自动重载计数器/定时器

模式 2 将定时器 0 和定时器 1 配置为具有自动重载初值功能的 8 位计数器/定时器。TL0 保持计数值，而 TH0 保持重载值。当 TL0 中的计数值从全 1 溢出到 0x00 时，TCON 寄存器中的定时器溢出标志 TF0 被设置，TH0 中的计数值被重新载入到 TL0。如果定时器 0 中断被允许，在 TF0 标识被设置时将产生一个中断。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 初始化为所需初值。当在模式 2 下时，定时器 1 的操作与定时器 0 完全相同。

在模式 2 下，计数器/定时器的允许和配置方法均与模式 0 一样。当 TMOD 寄存器中的 GATE0 为逻辑 0 或输入信号  $\overline{INT0}$  有效时（有效电平由 IT01CF 寄存器中的 IN0PL 位定义，请参见第 111 页“18.3. INT0 和 INT1 外部中断”，了解外部输入信号 INT0 和 INT1 的详情），设置 TR0 位 (TCON.4) 将允许定时器。

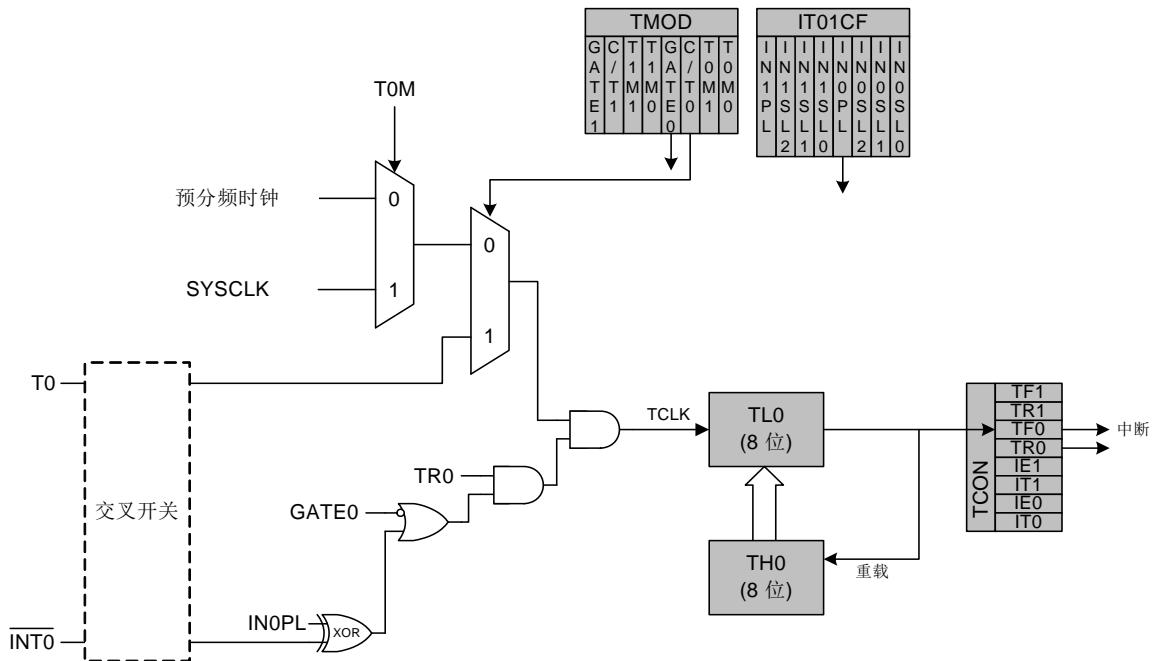


图 28.2. T0 模式 2 结构图

### 28.1.4. 模式 3：两个 8 位计数器/定时器（仅限定时器 0）

在模式 3 下，定时器 0 被配置两个独立的 8 位定时器/计数器，计数值分别在 TL0 和 TH0 中。TL0 中的计数器/定时器通过 TCON 和 TMOD 中定时器 0 的控制/状态位受控：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，以系统时钟或预分频时钟为时钟源。TH0 通过定时器 1 的运行控制位 TR1 被允许。TH0 在发生溢出时会设置定时器 1 的溢出标志 TF1，从而控制定时器 1 的中断。

定时器 1 在模式 3 下停止运行。当定时器 0 在模式 3 下操作时，定时器 1 可以在模式 0、1 或 2 下操作，但不能用外部信号作为时钟源，也不能设置 TF1 标志和产生中断。但是，定时器 1 溢出可以用于为其他外围设备产生波特率或溢出条件。当定时器 0 在模式 3 下操作时，定时器 1 的运行控制通过其模式设置处理。要在定时器 0 为模式 3 时运行定时器 1，应将定时器 1 的模式设为 0、1 或 2。可以通过将定时器 1 的模式配置为 3，禁用定时器 1。

# C8051F80x-83x

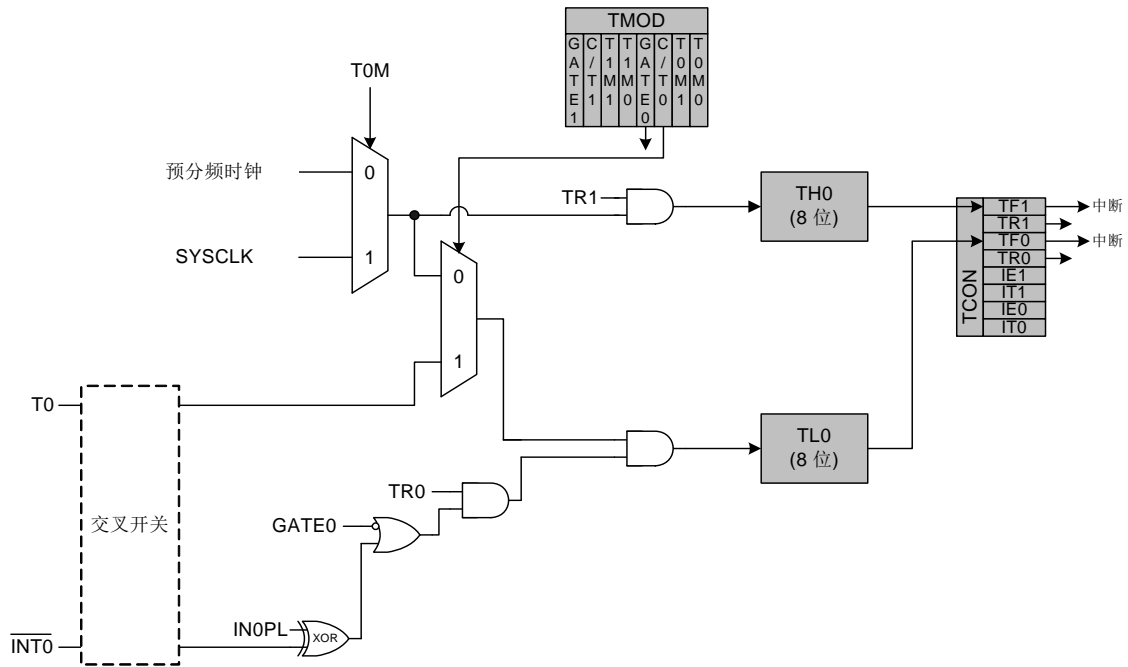


图 28.3. T0 模式 3 结构图

## SFR 定义 28.2: TCON: 定时器控制寄存器

位	7	6	5	4	3	2	1	0
名称	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x88; 可位寻址

位	名称	功能
7	TF1	<b>定时器 1 溢出标志。</b> 当定时器 1 溢出时由硬件设置为 1。该标志可通过软件清除，但当 CPU 转至定时器 1 中断服务程序时，该标志被自动清除。
6	TR1	<b>定时器 1 运行控制。</b> 通过将此位设置为 1 允许定时器 1。
5	TF0	<b>定时器 0 溢出标志。</b> 当定时器 0 溢出时由硬件设置为 1。该标志可通过软件清除，但当 CPU 转至定时器 0 中断服务程序时，该标志被自动清除。
4	TR0	<b>定时器 0 运行控制。</b> 通过将此位设置为 1 允许定时器 0。
3	IE1	<b>外部中断 1。</b> 当检测到一个由 IT1 定义的类型边沿/电平时，该标志由硬件设置。该标志可通过软件清除，但当边缘触发模式下 CPU 转至外部中断 1 服务程序时，该标志被自动清除。
2	IT1	<b>中断 1 类型选择。</b> 此位选择配置 /INT1 中断是边缘触发还是电平触发。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（请参见 SFR 定义 18.7）。 0: /INT1 为电平触发。 1: /INT1 为边缘触发。
1	IE0	<b>外部中断 0。</b> 当检测到一个由 IT1 定义的类型边沿/电平时，该标志由硬件设置。该标志可通过软件清除，但当边缘触发模式下 CPU 转至外部中断 0 服务程序时，该标志被自动清除。
0	IT0	<b>中断 0 类型选择。</b> 此位选择配置 INT0 中断是边缘触发还是电平触发。 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（请参见 SFR 定义 18.7）。 0: INT0 为电平触发。 1: INT0 为边缘触发。

# C8051F80x-83x

## SFR 定义 28.3: TMOD: 定时器模式

位	7	6	5	4	3	2	1	0
名称	GATE1	C/T1	T1M[1:0]		GATE0	C/T0	T0M[1:0]	
类型	R/W	R/W	R/W		R/W	R/W	R/W	
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x89

位	名称	功能
7	GATE1	<p><b>定时器 1 门控制。</b></p> <p>0: 当 TR1 = 1 时定时器 1 允许, 与 <math>\overline{INT1}</math> 的逻辑电平无关。 1: 仅当 TR1 = 1 并且 <math>\overline{INT1}</math> 有效 (有效电平由 IT01CF 寄存器中的 IN1PL 位定义, 请参见 SFR 定义 18.7) 时, 定时器 1 才被允许。</p>
6	C/T1	<p><b>计数器/定时器 1 选择。</b></p> <p>0: 定时器: 定时器 1 由 CKCON 寄存器中的 T1M 位定义的时钟加 1。 1: 计数器: 定时器 1 由外部引脚 (T1) 的负变跳加 1。</p>
5:4	T1M[1:0]	<p><b>定时器 1 模式选择。</b></p> <p>这些位选择定时器 1 的操作模式。 00: 模式 0, 13 位计数器/定时器 01: 模式 1, 16 位计数器/定时器 10: 模式 2, 8 位自动重载计数器/定时器 11: 模式 3, 定时器 1 停止运行</p>
3	GATE0	<p><b>定时器 0 门控制。</b></p> <p>0: 当 TR0 = 1 时定时器 0 被允许, 与 <math>\overline{INT0}</math> 的逻辑电平无关。 1: 仅当 TR0 = 1 并且 <math>\overline{INT0}</math> 有效 (有效电平由 IT01CF 寄存器中的 IN0PL 位定义, 请参见 SFR 定义 18.7) 时, 定时器 0 才被允许。</p>
2	C/T0	<p><b>计数器/定时器 0 选择。</b></p> <p>0: 定时器: 定时器 0 由 CKCON 寄存器中的 T0M 位定义的时钟加 1。 1: 计数器: 定时器 0 由外部引脚 (T0) 的负变跳加 1。</p>
1:0	T0M[1:0]	<p><b>定时器 0 模式选择。</b></p> <p>这些位选择定时器 0 的操作模式。 00: 模式 0, 13 位计数器/定时器 01: 模式 1, 16 位计数器/定时器 10: 模式 2, 8 位自动重载计数器/定时器 11: 模式 3, 两个 8 位计数器/定时器</p>



**SFR 定义 28.4: TL0: 定时器 0 低字节寄存器**

位	7	6	5	4	3	2	1	0
名称	TL0[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x8A

位	名称	功能
7:0	TL0[7:0]	定时器 0 低字节寄存器。 TL0 寄存器是 16 位定时器 0 的低字节寄存器。

**SFR 定义 28.5: TL1: 定时器 1 低字节寄存器**

位	7	6	5	4	3	2	1	0
名称	TL1[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x8B

位	名称	功能
7:0	TL1[7:0]	定时器 1 低字节寄存器。 TL1 寄存器是 16 位定时器 1 的低字节寄存器。

# C8051F80x-83x

## SFR 定义 28.6: TH0: 定时器 0 高字节寄存器

位	7	6	5	4	3	2	1	0
名称	TH0[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x8C

位	名称	功能
7:0	TH0[7:0]	定时器 0 高字节寄存器。 TL0 寄存器是 16 位定时器 0 的高字节寄存器。

## SFR 定义 28.7: TH1: 定时器 1 高字节寄存器

位	7	6	5	4	3	2	1	0
名称	TH1[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0x8D

位	名称	功能
7:0	TH1[7:0]	定时器 1 高字节寄存器。 TL1 寄存器是 16 位定时器 1 的高字节寄存器。

## 28.2. 定时器 2

定时器 2 是一个 16 位定时器，由两个 8 位 SFR 组成：TMR2L（低字节）和 TMR2H（高字节）。定时器 2 可以在 16 位自动重载模式下或（两个）8 位自动重载模式下操作。T2SPLIT 位 (TMR2CN.3) 定义定时器 2 的操作模式。定时器 2 还可被用于捕捉模式，捕捉比较器 0 输出的上升沿。

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/8。外部时钟模式非常适合实时时钟 (RTC) 功能，其中由内部振荡源驱动系统时钟，而定时器 2（和/或 PCA）以外部振荡源为时钟源。在所有操作模式下（暂停模式除外），外部振荡源/8 均与系统时钟同步。当内部振荡源被置为暂停模式时，外部时钟/8 的信号可以直接驱动定时器。可以使用外部时钟将装置从暂停模式下唤醒。定时器将继续在暂停模式下运行并计数。当定时器发生溢出时，装置将从暂停模式中醒来，并重新开始执行代码。可在输入暂停前设置定时器值，从而在所需时间段（时钟数）溢出以唤醒装置。如果定时器以外的唤醒源将装置从暂停模式中唤醒，则可能要用 3 个定时器方可读取或写入定时器寄存器。在此期间，OSCICN 寄存器中的 STSYNC 位将被设置为 1，表明读取或写入定时器寄存器不安全。

### 28.2.1. 16 位自动重载定时器

当 T2SPLIT (TMR2CN.3) 为 0 时，定时器 2 作为 16 位自动重载定时器操作。定时器 2 可以 SYSCLK、SYSCLK/12 或外部振荡器时钟源/8 为时钟源。当 16 位定时器寄存器增值并从 0xFFFF 溢出到 0x0000 时，定时器 2 重载寄存器（TMR2RLH 和 TMR2RLL）中的 16 位值被载入定时器 2 寄存器（如图 28.4 所示），并且定时器 2 高字节溢出标志 (TMR2CN.7) 被设置。如果定时器 2 中断被允许（如已设置 IE.5），则每次定时器 2 溢出时均将产生一个中断。此外，如果定时器 2 中断被允许，并且 TF2LEN 位被设置 (TMR2CN.5)，则每次低 8 位 (TMR2L) 从 0xFF 溢出到 0x00 时将产生一个中断。

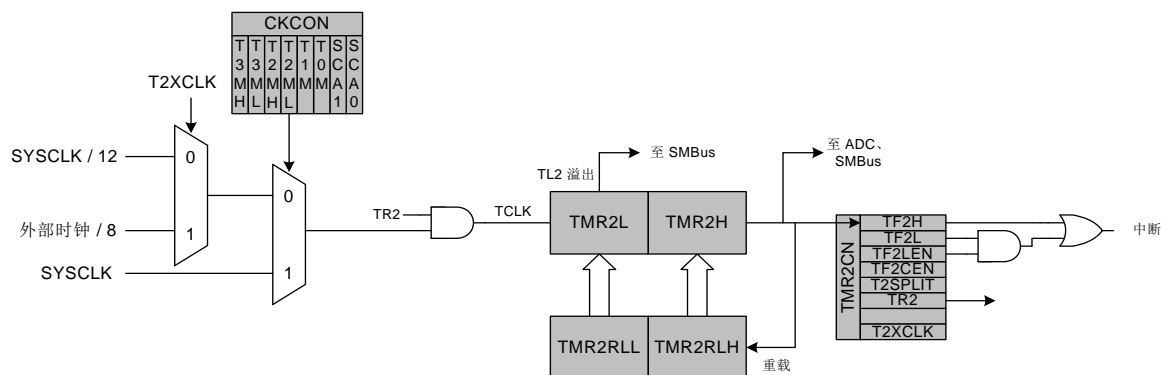


图 28.4. 定时器 2 16 位模式结构图

# C8051F80x-83x

## 28.2.2. 8 位自动重载定时器

当 T2SPLIT 被设置时，定时器 2 作为两个 8 位定时器（TMR2H 及 TMR2L）操作。这两个 8 位定时器都在自动重载模式下操作（如图 28.5 所示）。TMR2RLL 保持 TMR2L 的重载值，TMR2RLH 保持 TMR2H 的重载值。TMR2CN 中的 TR2 位处理 TMR2H 的运行控制。当配置为 8 位模式时，TMR2L 总是处于运行状态。定时器 2 还可被用于捕捉模式，捕捉比较器 0 输出的上升沿。

每个 8 位定时器都可以被配置为使用 SYSCLK、SYSCLK/12 或外部振荡器时钟源/8 作为其时钟源。定时器 2 时钟选择位（CKCON 中的 T2MH 和 T2ML）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，如下所示：

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

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

当 TMR2H 从 0xFF 溢出到 0x00 时，TF2H 位被设置；当 TMR2L 从 0xFF 溢出到 0x00 时，TF2L 位被设置。如果定时器 2 中断被允许 (IE.5)，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 (TMR2CN.5) 被设置，则每次 TMR2L 或 TMR2H 发生溢出时将产生一个中断。当 TF2LEN 位被允许时，软件应检查 TF2H 和 TF2L 标志，以确定定时器 2 中断的来源。TF2H 和 TF2L 中断标志不能由硬件自动清除，必须通过软件手动清除。

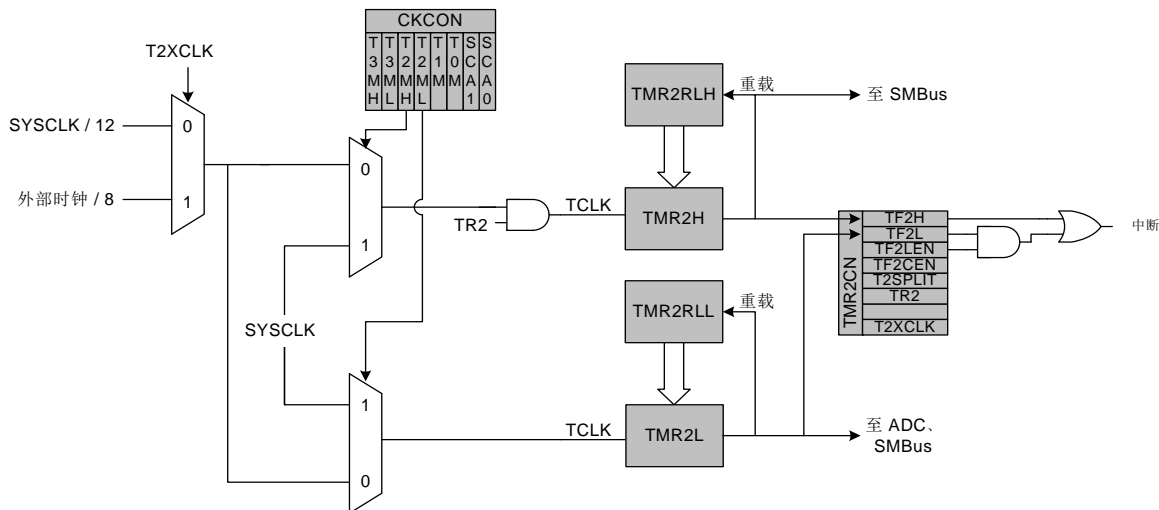


图 28.5. 定时器 2 8 位模式结构图

## 28.2.3. 比较器 0 捕捉模式

定时器 2 的捕捉模式允许使用以系统时钟或系统时钟/12 作为时钟源的定时器捕捉比较器 0 上升沿。通过将 TF2CEN 设置为 1 并将 T2SPLIT 设置为 0 允许定时器 2 捕捉模式。

允许捕捉模式后，每个比较器 0 上升沿均会产生一个捕捉事件。当捕捉事件发生时，定时器 2 的内容 (TMR2H:TMR2L) 会载入定时器 2 重载寄存器 (TMR2RLH:TMR2RLL)，并且 TF2H 标志将被设置（如果定时器 2 中断被允许，则会触发一个中断）。通过记录两个连续定时器捕捉值的差异，可以确定比较器 0 的周期（相对于定时器 2 时钟）。为获得精确的读数，定时器 2 时钟必须远快于捕捉时钟。

这种模式允许软件确定连续比较器 0 上升沿间的时间，上升沿可用于检查电容开关的容量变化，或测量低电平模拟信号的频率。

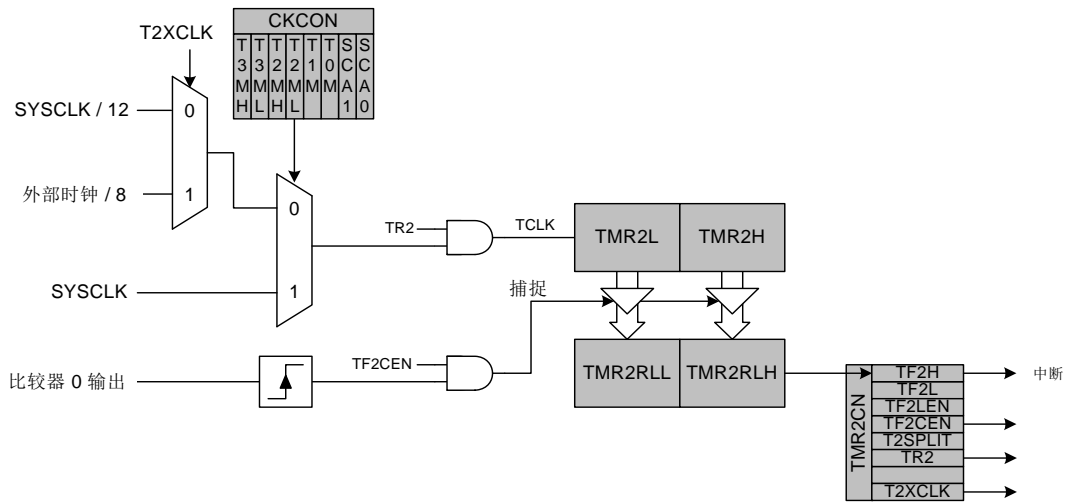


图 28.6. 定时器 2 捕捉模式结构图

# C8051F80x-83x

## SFR 定义 28.8: TMR2CN: 定时器 2 控制寄存器

位	7	6	5	4	3	2	1	0
名称	TF2H	TF2L	TF2LEN	TF2CEN	T2SPLIT	TR2		T2XCLK
类型	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xC8; 可位寻址

位	名称	功能
7	TF2H	<b>定时器 2 高字节溢出标志。</b> 当定时器 2 高字节从 0xFF 溢出到 0x00 时由硬件设置。在 16 位模式下，当定时器 2 从 0xFFFF 溢出到 0x0000 时发生上述情况。当启用定时器 2 中断时，设置此位会导致 CPU 转至定时器 2 中断服务程序。此位不能由硬件自动清除。
6	TF2L	<b>定时器 2 低字节溢出标志。</b> 当定时器 2 低字节从 0xFF 溢出到 0x00 时由硬件设置。TF2L 在低字节溢出时被设置，与定时器 2 的模式无关。此位不能由硬件自动清除。
5	TF2LEN	<b>定时器 2 低字节中断允许。</b> 当此位被设置为 1 时，允许定时器 2 低字节中断。如同时还允许定时器 2 中断，则当定时器 2 低字节发生溢出时产生一个中断。
4	TF2CEN	<b>定时器 2 比较器捕捉允许。</b> 当此位被设置为 1 时，允许定时器 2 比较器捕捉模式。如 TF2CEN 被设置于比较器 0 输出的上升沿，则 TMR2H:TMR2L 中的当前 16 位定时器值将被复制到 TMR2RLH:TMR2RLL 中。如定时器 2 中断也被允许，此事件发生时将产生一个中断。
3	T2SPLIT	<b>定时器 2 双 8 位模式允许。</b> 当此位被设置时，定时器 2 作为两个 8 位自动重载定时器操作。 0: 定时器 2 在 16 位自动重载模式下操作。 1: 定时器 2 作为两个 8 位自动重载定时器操作。
2	TR2	<b>定时器 2 运行控制。</b> 定时器 2 通过将此位设置为 1 被允许。在 8 位模式下，此位仅允许/禁用 TMR2H；在双模式下，TMR2L 始终被允许。
1	未使用	读 = 0b; 写 = 无需在意。
0	T2XCLK	<b>定时器 2 外部时钟选择。</b> 此位为定时器 2 选择外部时钟源。如果定时器 2 在 8 位模式下，此位为两个定时器字节选择外部振荡器时钟源。但仍可用定时器 2 时钟选择位（CKCON 寄存器中的 T2MH 和 T2ML）为任一定时器在外部时钟和系统时钟之间作出选择。 0: 系统时钟/12。 1: 外部时钟/8（未暂停时与 SYSCLK 同步）。

**SFR 定义 28.9: TMR2RLL: 定时器 2 重载寄存器低字节**

位	7	6	5	4	3	2	1	0
名称	TMR2RLL[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xCA

位	名称	功能
7:0	TMR2RLL[7:0]	定时器 2 重载寄存器低字节。 TMR2RLL 保持定时器 2 重载值的低字节。

**SFR 定义 28.10: TMR2RLH: 定时器 2 重载寄存器高字节**

位	7	6	5	4	3	2	1	0
名称	TMR2RLH[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xCB

位	名称	功能
7:0	TMR2RLH[7:0]	定时器 2 重载寄存器高字节。 TMR2RLH 保持定时器 2 重载值的高字节。

# C8051F80x-83x

## SFR 定义 28.11: TMR2L: 定时器 2 低字节寄存器

位	7	6	5	4	3	2	1	0
名称	TMR2L[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xCC

位	名称	功能
7:0	TMR2L[7:0]	定时器 2 低字节寄存器。 在 16 位模式下，TMR2L 寄存器保持 16 位定时器 2 的低字节。在 8 位模式下，TMR2L 中保持 8 位低字节定时器值。

## SFR 定义 28.12: TMR2H 定时器 2 高字节寄存器

位	7	6	5	4	3	2	1	0
名称	TMR2H[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xCD

位	名称	功能
7:0	TMR2H[7:0]	定时器 2 高字节寄存器。 在 16 位模式下，TMR2H 寄存器保持 16 位定时器 2 的高字节。在 8 位模式下，TMR2H 中保持 8 位高字节定时器值。



## 29. 可编程计数器阵列

可编程计数器阵列 (PCA0) 提供增强型定时器功能, 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/比较模块都有其自己相关的 I/O 线 (CEXn), 这些 I/O 线在被允许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驱动, 时基可以在六个时钟源中选择: 系统时钟、系统时钟/4、系统时钟/12、外部振荡器时钟源/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/比较模块均可配置为以六种模式中的一种独立操作: 边沿触发捕捉、软件定时器、高速输出、频率输出、8 至 15 位 PWM 或 16 位 PWM (请参见第 228 页"29.3. 捕捉/比较模块", 了解每种模式的详情)。外部振荡器时钟选项非常适合实时时钟 (RTC) 功能, 这一功能允许 PCA 以精确外部振荡器为时钟源, 而由内部振荡器驱动系统时钟。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 的结构图示于图 29.1

**重要事项:** PCA 模块 2 可被用作看门狗 (WDT), 在系统复位后即被允许为此模式。在 WDT 模式被允许时, 对某些 PCA 寄存器的访问受到限制。详见第 29.4 条。

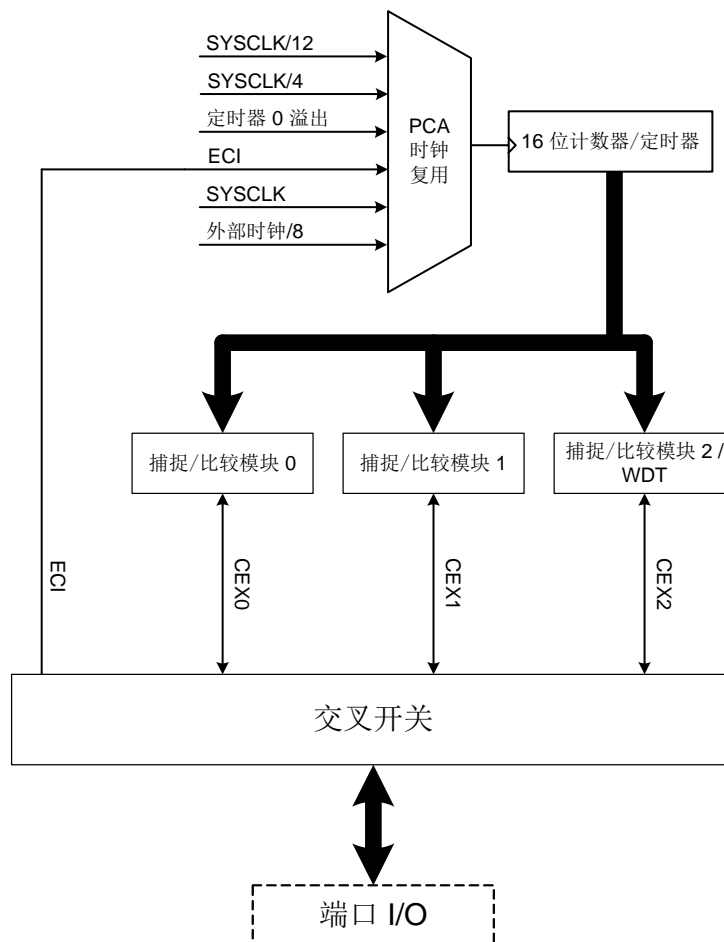


图 29.1. PCA 结构图

# C8051F80x-83x

## 29.1. 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。先读 PCA0L 寄存器可以保证准确读取整个 16 位 PCA0 计数器。读 PCA0H 或 PCA0L 不影响计数器操作。PCA0MD 寄存器中的 CPS2-CPS0 位用于选择计数器/定时器的时基，如表 29.1 所示。

当计数器/定时器从 0xFFFF 溢出到 0x0000 时，PCA0MD 中的计数器溢出标志 (CF) 被设置为逻辑 1 并产生一个中断请求（如果 CF 中断被允许）。将 PCA0MD 中的 ECF 位设置为逻辑 1 即可允许 CF 标志产生一个中断请求。当 CPU 转向中断服务程序时，CF 位不能由硬件自动清除，必须用软件清除。清除 PCA0MD 寄存器中的 CIDL 位将允许 PCA 在 CPU 处于空闲模式时继续正常操作。

表 29.1: PCA 时基输入选项

CPS2	CPS1	CPS0	时基
0	0	0	系统时钟/12
0	0	1	系统时钟/4
0	1	0	定时器 0 溢出
0	1	1	ECI 负变跳（最大速率 = 系统时钟/4）
1	0	0	系统时钟
1	0	1	外部振荡器源/8（附注）
1	1	x	备用

附注：外部振荡器源/8 与系统时钟同步。

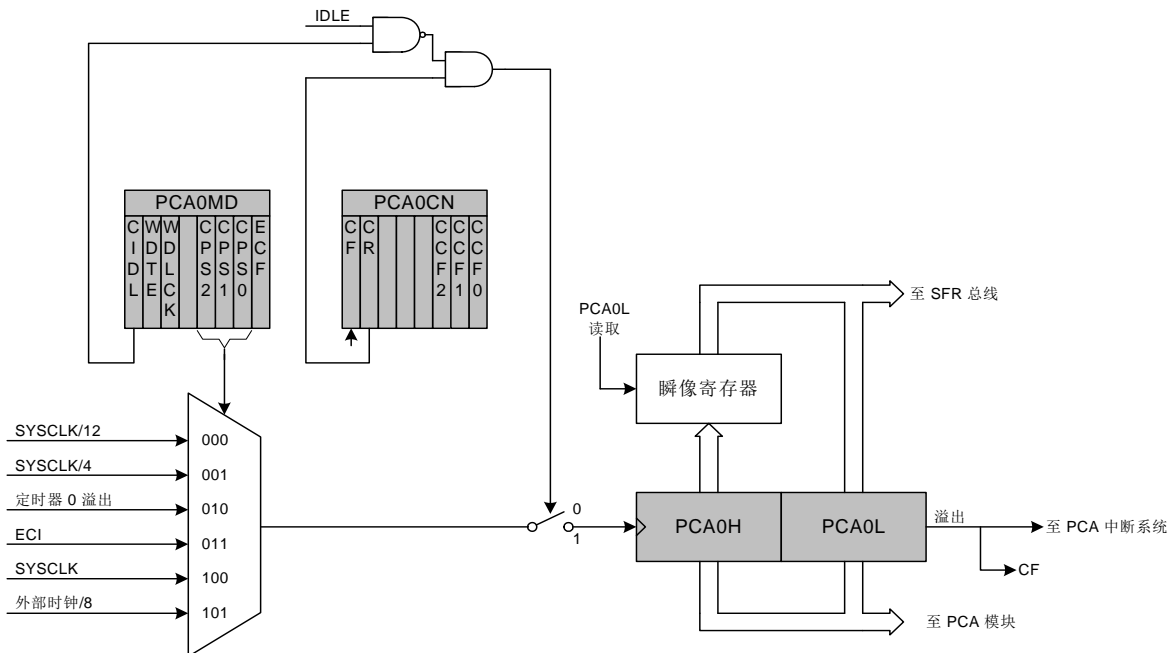


图 29.2. PCA 计数器/定时器结构图

## 29.2. PCA0 中断源

图 29.3 列示了 PCA 中断树的结构图。共有五个独立事件标志可用于产生 PCA0 中断。分别为：主 PCA 计数器溢出标志 (CF) (在 PCA0 计数器出现 16 位溢出时被设置)、中级溢出标志 (COVF) (在 PCA0 计数器从 8 位溢出到 15 位时被设置)，以及每个 PCA 通道的独立标志 (CCF0、CCF1 和 CCF2) (根据该模块的操作模式设置)。这些事件标志通常在触发条件发生时被设置。通过对应的中断允许标志 (CF 对应 ECF，COVF 对应 ECOV，CCFn 对应 ECCFn) 可独立选择每种标志用于产生 PCA0 中断。在处理器识别任何独立中断源前，必须全局允许 PCA0 中断。可通过将 IE 寄存器中的 EA 位和 EIE1 寄存器中的 EPCA0 位设置为逻辑 1 全局允许 PCA0 中断。

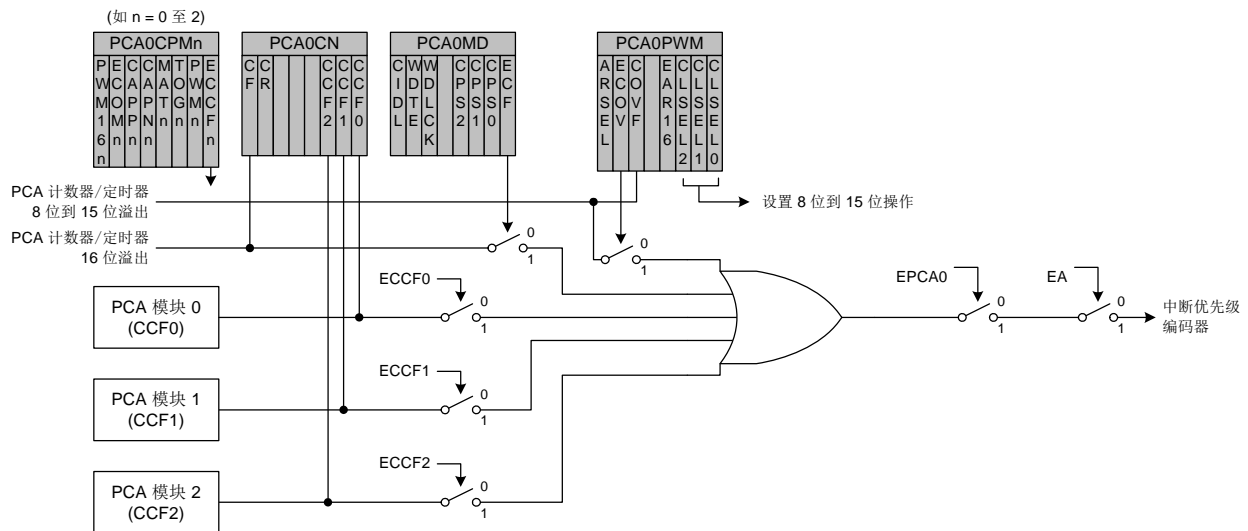


图 29.3. PCA 中断结构图

## 29.3. 捕捉/比较模块

每个模块都可被配置为独立操作，有六种操作模式：边沿触发捕捉、软件定时器、高速输出、频率输出、8位至15位脉冲宽度调制器或16位脉冲宽度调制器。每个模块在CIP-51系统控制器中都有属于自己的特殊功能寄存器(SFR)。这些寄存器用于与模块交换数据和配置模块的操作模式。表29.2概述了PCA0CPMn和PCA0PWM寄存器的位设置情况，这些寄存器用于选择PCA捕捉/比较模块的操作模式。请注意，设置为使用8位到15位PWM模式的所有模块均必须使用同一周期长度(8-15位)。设置PCA0CPMn寄存器中的ECCFn位会允许模块CCFn中断。

表 29.2: PCA 捕捉/比较模块的 PCA0CPM 和 PCA0PWM 位设置<sup>1,2,3,4,5,6</sup>

操作模式	PCA0CPMn								PCA0PWM					
	7	6	5	4	3	2	1	0	7	6	5	4	3	2-0
用 CEXn 的正沿触发捕捉	X	X	1	0	0	0	0	A	0	X	B	X	X	XXX
用 CEXn 的负沿触发捕捉	X	X	0	1	0	0	0	A	0	X	B	X	X	XXX
用 CEXn 的任一跳变触发捕捉	X	X	1	1	0	0	0	A	0	X	B	X	X	XXX
软件定时器	X	C	0	0	1	0	0	A	0	X	B	X	X	XXX
高速输出	X	C	0	0	1	1	0	A	0	X	B	X	X	XXX
频率输出	X	C	0	0	0	1	1	A	0	X	B	X	X	XXX
8 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	0	X	B	X	X	000
9 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	D	X	B	X	X	001
10 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	D	X	B	X	X	010
11 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	D	X	B	X	X	011
12 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	D	X	B	X	X	100
13 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	D	X	B	X	X	101
14 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	D	X	B	X	X	110
15 位脉冲宽度调制器 <sup>7</sup>	0	C	0	0	E	0	1	A	D	X	B	X	X	111
16 位脉冲宽度调制器	1	C	0	0	E	0	1	A	0	X	B	X	0	XXX
16 位自动重载脉冲宽度调制器	1	C	0	0	E	0	1	A	D	X	B	X	1	XXX

附注:

1. X = 无需在意 (无论等于 1 或 0 对于独立模块的功能而言并无差异)。
2. A = 允许此模块中断 (CCFn 被设置为 1 时, 触发 PCA 中断)。
3. B = 允许 8 到 15 位溢出中断 (取决于 CLSEL[2:0] 的设置)。
4. C = 当设置为 0 时, 数字比较器关闭。对于高速和频率输出模式, 相关引脚将不会转换。在任意 PWM 模式下, 这均会产生 0% 负载循环 (输出 = 0)。
5. D = 通过地址 PCA0CPHn 和 PCA0CPLn 评估为相关通道选择捕捉/比较寄存器 (0) 还是自动重载寄存器 (1)。
6. E = 设置时, 匹配事件将导致相关通道的 CCFn 标识被设置。
7. 设置为 8 位到 15 位 PWM 模式的所有模块使用同一周期长度设置。

## 29.3.1. 边缘触发捕捉模式

在该模式下，CEXn 针脚上出现的有效跳变导致 PCA 捕捉 PCA 计数器/定时器的值，并将其载入对应模块的 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跳变类型：正变跳（正沿）、负变跳（负沿）或任意变跳（正沿或负沿）。当捕捉发生时，PCA0CN 中的捕捉/比较标志 (CCFn) 被设置为逻辑 1，并且如果该模块的 CCFn 中断被允许，将产生一个中断请求。当 CPU 转向中断服务程序时，CCFn 位不能由硬件自动清除，必须用软件清除。如果 CAPPn 和 CAPNn 位都被设置为逻辑 1，则可以通过直接读 CEXn 对应端口针脚的状态，来确定捕捉是由上升沿触发还是由下降沿触发。

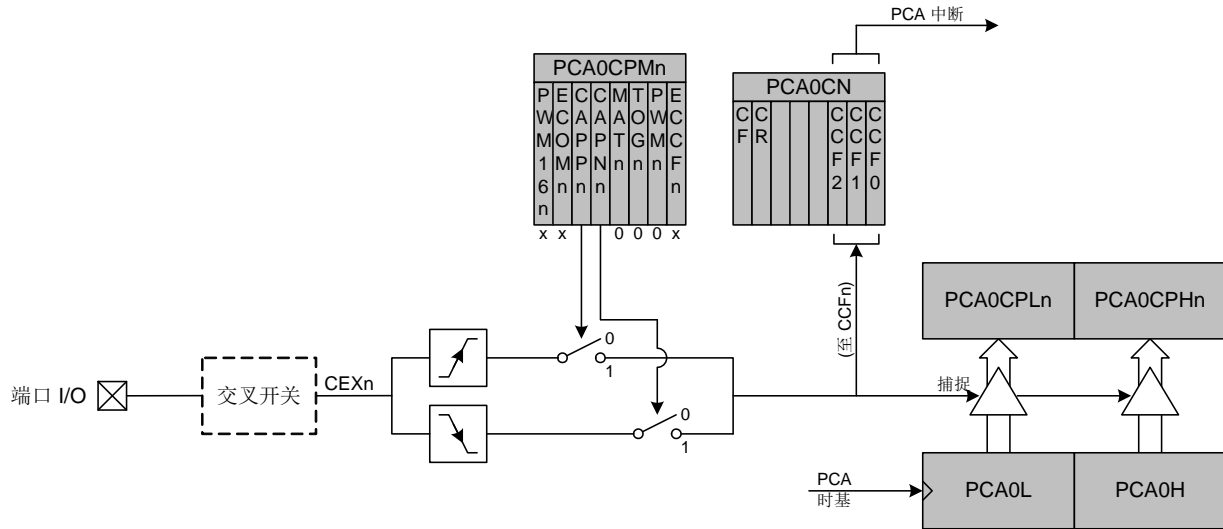


图 29.4. PCA 捕捉模式图

附注：CEXn 输入信号必须保持高电平或低电平至少两个系统时钟周期，以保证能够被硬件识别。

# C8051F80x-83x

## 29.3.2. 软件定时器（比较）模式

在软件定时器模式下，PCA 计数器/定时器的值与模块的 16 位捕捉/比较寄存器（PCA0CPLn 和 PCA0CPHn）进行比较。当发生匹配时，PCA0CN 中的捕捉/比较标志（CCFn）被设置为逻辑 1，并且如果该模块的 CCFn 中断被允许，将产生一个中断请求。当 CPU 转向中断服务程序时，CCFn 位不能由硬件自动清除，必须用软件清除。设置 PCA0CPMn 寄存器中的 ECOMn 和 MATn 位将允许软件定时器模式。

**关于计数器/定时器的重要事项：**当向 PCA0 的捕捉/比较寄存器写入一个 16 位数值时，应先写入低字节。向 PCA0CPLn 写入时会把 ECOMn 位清 0；向 PCA0CPHn 写入时会把 ECOMn 位设置为 1。

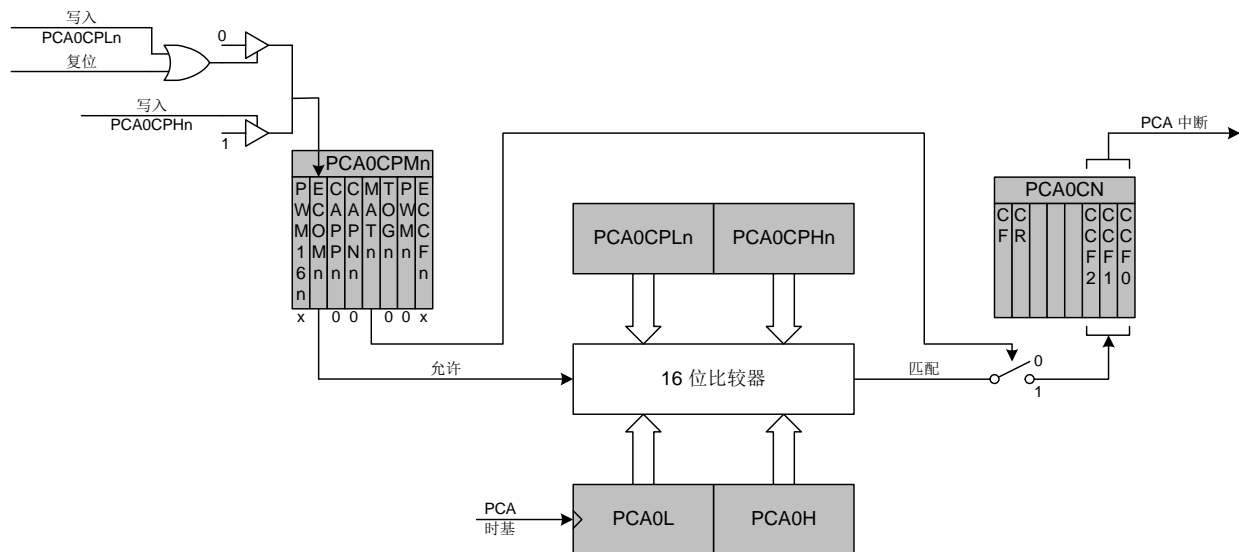


图 29.5. PCA 软件定时器模式图

### 29.3.3. 高速输出模式

在高速输出模式下，每当 PCA 计数器与模块的 16 位捕捉/比较寄存器（PCA0CPLn 和 PCA0CPHn）发生匹配时，模块的相关 CEXn 引脚将发生转换。当发生匹配时，PCA0CN 中的捕捉/比较标志 (CCFn) 被设置为逻辑 1，并且如果该模块的 CCFn 中断被允许，将产生一个中断请求。当 CPU 转向中断服务程序时，CCFn 位不能由硬件自动清除，必须用软件清除。设置 PCA0CPMn 寄存器中的 TOGn、MATn 和 ECOMn 位将允许高速输出模式。如果 ECOMn 被清除，相关引脚将保持其状态，而不会在下次匹配事件时发生转换。

**关于计数器/定时器的重要事项：**当向 PCA0 的捕捉/比较寄存器写入一个 16 位数值时，应先写入低字节。向 PCA0CPLn 写入时会将 ECOMn 位清 0；向 PCA0CPHn 写入时会将 ECOMn 位设置为 1。

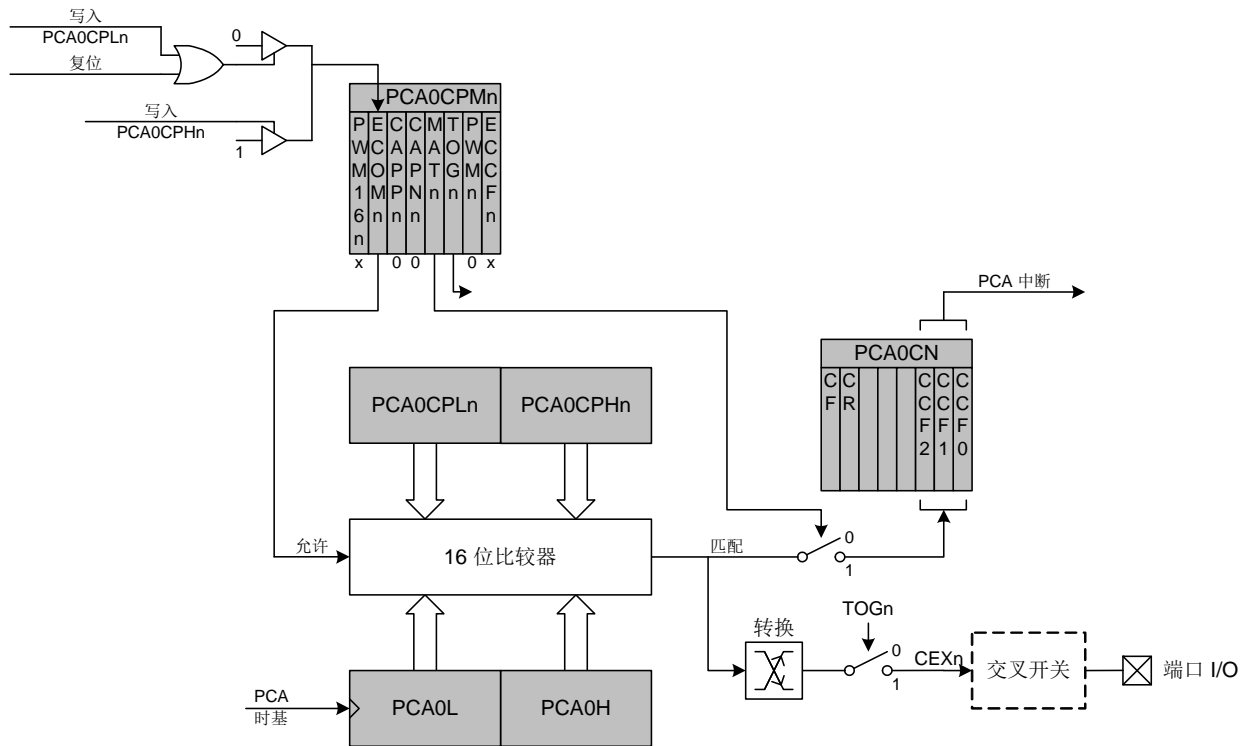


图 29.6. PCA 高速输出模式图

## 29.3.4. 频率输出模式

频率输出模式会在模块的相关 CEXn 针脚上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转换前要计的 PCA 时钟数。方波的频率由方程 29.1 定义。

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

附注：对于该方程，PCA0CPHn 寄存器中的值为 0x00 时，相当于 256。

### 方程 29.1：方波频率输出

其中， $F_{PCA}$  是由 PCA 模式寄存器 PCA0MD 中 CPS2-0 位选择的时钟的频率。捕捉/比较模块的低字节与 PCA 计数器的低字节比较，两者匹配时，CEXn 的电平发生转换，高字节中的偏移值被加到 PCA0CPLn 的匹配值中。通过设置 PCA0CPMn 寄存器中 ECOMn、TOGn 和 PWMn 位允许频率输出模式。在这种模式下，通常应将 MATn 位设置为 0。如果 MATn 位被设置为 1，则当通道的 16 位 PCA0 计数器与 16 位捕捉/比较寄存器相等时，通道的 CCFn 标志将被设置。

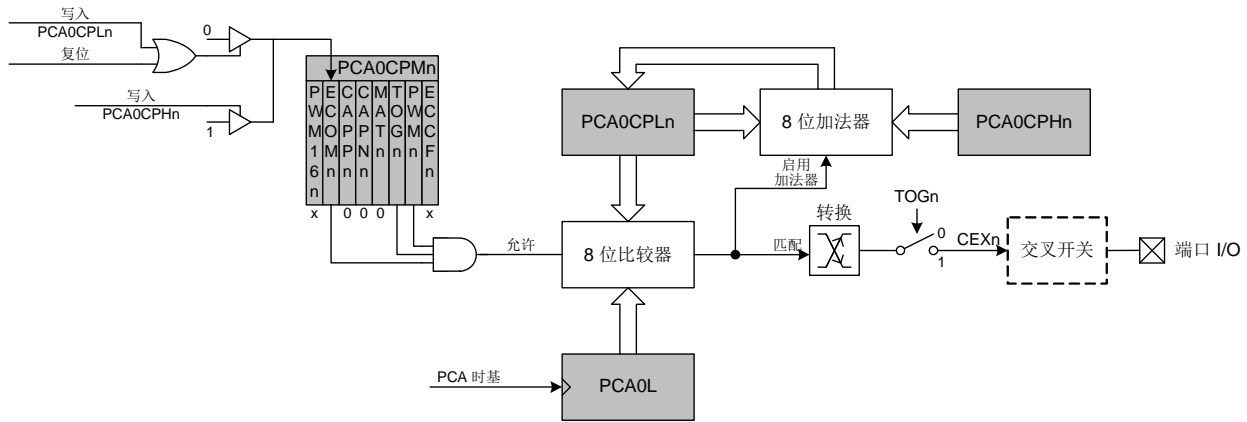


图 29.7. PCA 频率输出模式

## 29.3.5. 8 位至 15 位脉冲宽度调制器模式

每个模块都可以被独立地用于在对应的 CEXn 针脚产生脉冲宽度调制 (PWM) 输出。输出的频率取决于 PCA 计数器/定时器的时基以及 PWM 周期长度的设置（8、9、10、11、12、13、14 或 15 位）。就与其他装置的 8 位 PWM 模式向后兼容而言，8 位 PWM 模式的操作与 9、10、11、12、13、14 及 15 位 PWM 模式略有不同。**务请注意，为 8 位至 15 位 PWM 模式配置的所有通道均将使用同一周期长度。**例如，不能为 8 位 PWM 配置一条通道，而为 11 位模式配置另一条。但其他 PCA 通道可被独立配置用于针脚捕捉、高速输出、软件定时器、频率输出或 16 位 PWM 模式。



## 29.3.5.1. 8 位脉冲宽度调制器模式

使用模块的 PCA0CPLn 捕捉/比较寄存器改变 PWM 的输出信号在 8 位 PWM 模式下的负载循环。当 PCA 计数器/定时器的低字节 (PCA0L) 值与 PCA0CPLn 中的值相等时, CEXn 针脚上的输出将被设置。当 PCA0L 中的计数值溢出时, CEXn 输出将被复位 (见图 29.8)。而且, 当计数器/定时器的低字节 (PCA0L) 从 0xFF 溢出到 0x00 时, 保存在模块计数器/定时器高字节 (PCA0CPHn) 中的值被自动重载到 PCA0CPLn, 不需软件干预。软件可利用这一同步更新特征异步写入新的 PWM 高电平时间, 并在随后的 PWM 期间生效。

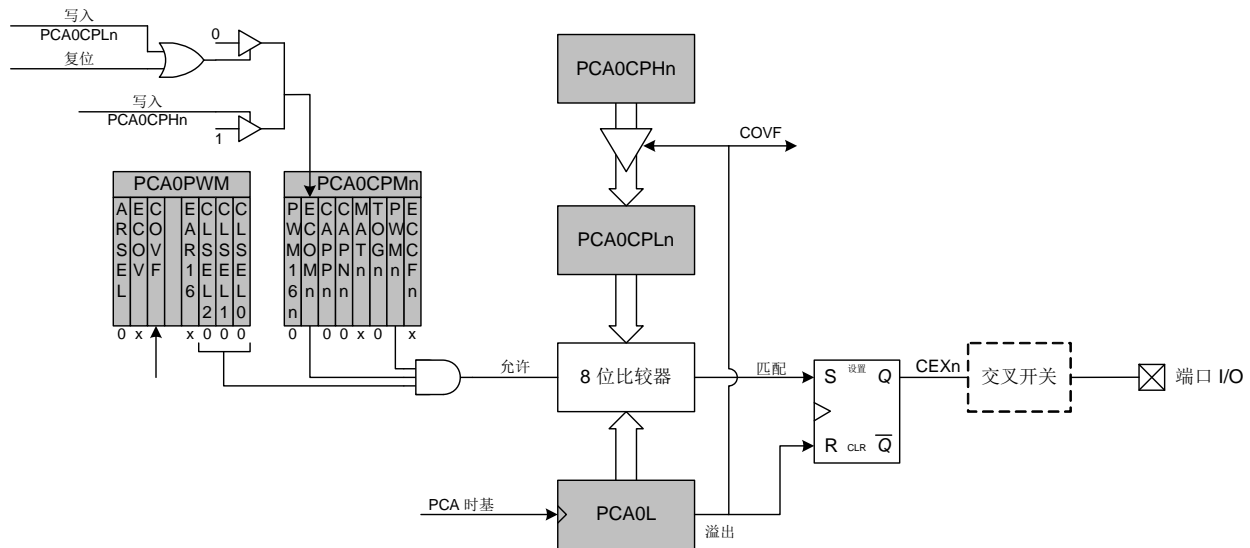
可通过设置 PCA0CPMn 寄存器中的 ECOMn 和 PWMn 位和将 PCA0PWM 寄存器中的 CLSEL 位设置为 000b 来允许 8 位脉冲宽度调制器模式。如果 MATn 位被设置为 1, 模块的 CCFn 标志将在每次 8 位比较器发生匹配 (上升沿) 时被设置。PCA0PWM 中的 COVF 标志可被用于探测溢出 (下降沿), 该溢出将每 256 个 PCA 时钟周期发生一次。8 位 PWM 模式的负载循环由方程 29.2 给出。

**关于计数器/定时器的重要事项:** 当向 PCA0 的捕捉/比较寄存器写入一个 16 位数值时, 应先写入低字节。向 PCA0CPLn 写入时会将 ECOMn 位清 0; 向 PCA0CPHn 写入时会将 ECOMn 位设置为 1。

$$\text{负载循环} = \frac{(256 - \text{PCA0CPHn})}{256}$$

**方程 29.2: 8 位 PWM 负载循环**

由方程 29.2 可知, 最大负载循环为 100% (PCA0CPHn = 0), 最小负载循环为 0.39% (PCA0CPHn = 0xFF)。可以通过将 ECOMn 位清 0 产生 0% 负载循环。



**图 29.8. PCA 8 位 PWM 模式图**

## 29.3.5.2. 9 位至 15 位脉冲宽度调制器模式

通过写入一个被双映射至 PCA0CPHn 和 PCA0CPLn 寄存器位置的"自动重载"寄存器改变 PWM 输出信号在 N 位 (N = 9 至 15) PWM 模式下的负载循环。定义负载循环的写入数据应在寄存器中右对齐。当 PCA0PWM 中的 ARSEL 位被设置为 1 时访问 (读或写) 自动重载寄存器。当 ARSEL 被设置为 0 时, 访问捕捉/比较寄存器。

当 PCA0 计数器的最低 N 位与相关模块的捕捉/比较寄存器 (PCA0CPn) 中的值匹配时, CEXn 的输出被置为高电平。当计数器从第 N 位溢出时, CEXn 被置为低电平 (见图 29.9)。当从第 N 位溢出时, COVF 标志被设置, 保存在模块自动重载寄存器中的值被载入捕捉/比较寄存器。N 的值由 PCA0PWM 寄存器中的 CLSEL 位确定。软件可利用这一同步更新特征异步写入新的 PWM 高电平时间, 并在随后的 PWM 期间生效。

可通过设置 PCA0CPMn 寄存器中的 ECOMn 和 PWMn 位选择 9、10、11、12、13、14 及 15 位 PWM 模式, 并将 PCA0PWM 寄存器中的 CLSEL 位设置为所需的周期长度 (8 位除外)。如果 MATn 位被设置为 1, 模块的 CCFn 标志将在每次比较器发生匹配 (上升沿) 时被设置。PCA0PWM 中的 COVF 标志可被用于探测溢出 (下降沿), 该溢出将每 512 (9 位)、1024 (10 位)、2048 (11 位)、4096 (12 位)、8192 (13 位)、16384 (14 位) 或 32768 (15 位) 个 PCA 时钟周期发生一次。N 位 (N = 9 至 15) PWM 模式的负载循环由方程 29.2 给出, 其中 N 为 PWM 循环的位数。可以通过将 ECOMn 位清 0 产生 0% 负载循环。

**关于 PCA0CPHn 和 PCA0CPLn 寄存器的重要事项:** 当向 PCA0CPn 寄存器写入一个 16 位数值时, 应先写入低字节。向 PCA0CPLn 写入时会将 ECOMn 位清 0; 向 PCA0CPHn 写入时会将 ECOMn 位设置为 1。

$$\text{负载循环} = \frac{(2^N - \text{PCA0CPn})}{2^N}$$

方程 29.3: N 位 PWM 负载循环 (N = 9 至 15)

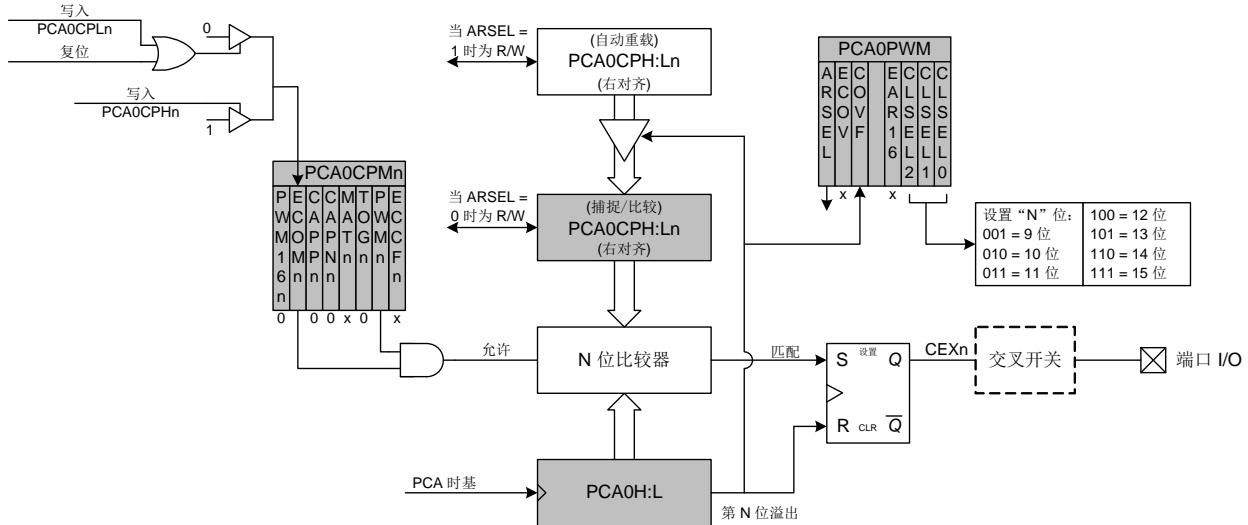


图 29.9. PCA 9 位至 15 位 PWM 模式图

## 29.3.6. 16 位脉冲宽度调制器模式

PCA 模块可在 16 位 PWM 模式下操作。16 位 PWM 模式独立于其他（8 位至 15 位）PWM 模式。在此模式下，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 计数器与模块内容匹配时，CEXn 的输出被置为高电平；当 16 位计数器溢出时，CEXn 被置为低电平。通过设置 PCA0CPMn 寄存器中 ECOMn、PWMn 和 PWM16n 位允许 16 位 PWM 模式。

通过写入一个被双映射至 PCA0CPHn 和 PCA0CPLn 寄存器位置的"自动重载"寄存器改变 PWM 输出信号的负载循环。当 PCA0PWM 中的 ARSEL 位被设置为 1 时，访问（读或写）自动重载寄存器。当 ARSEL 被设置为 0 时，访问捕捉/比较寄存器。软件可利用这一同步更新特征异步写入新的 PWM 高电平时间，并在随后的 PWM 期间生效。

就与其他装置的 16 位 PWM 模式向后兼容而言，可在不使用"自动重载"寄存器的情况下修改 PWM 负载循环。为了在不使用"自动重载"寄存器的情况下输出一个可变负载循环，新值的写入应与 PCA CCFn 匹配中断同步。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。如果 MATn 位被设置为 1，模块的 CCFn 标志将在每次 16 位比较器发生匹配（上升沿）时被设置。PCA0CN 中的 CF 标志可被用于探测溢出（下降沿）。16 位 PWM 模式的负载循环由方程 29.4 给出。

**关于计数器/定时器的重要事项：**当向 PCA0 的捕捉/比较寄存器写入一个 16 位数值时，应先写入低字节。向 PCA0CPLn 写入时会把 ECOMn 位清 0；向 PCA0CPHn 写入时会把 ECOMn 位设置为 1。

$$\text{负载循环} = \frac{(65536 - \text{PCA0CPn})}{65536}$$

方程 29.4: 16 位 PWM 负载循环

由方程 29.4 可知，最大负载循环为 100% (PCA0CPn = 0)，最小负载循环为 0.0015% (PCA0CPn = 0xFFFF)。可以通过将 ECOMn 位清 0 产生 0% 负载循环。

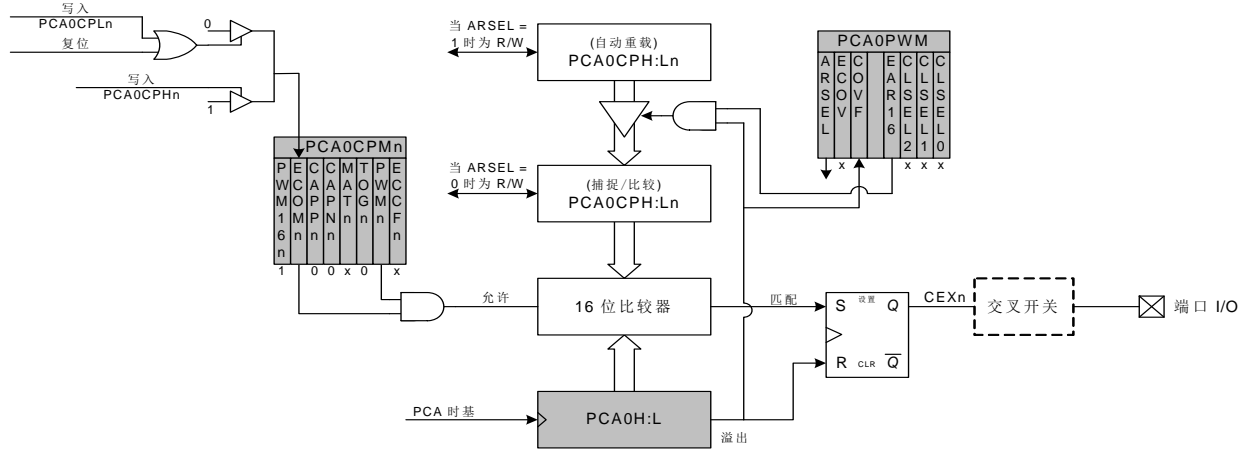


图 29.10. PCA 16 位 PWM 模式图

## 29.4. 看门狗模式

通过 PCA 的模块 2 可以实现可编程看门狗 (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允许/禁用 WDT。

当 PCA0MD 寄存器中的 WDTE 位被设置时，模块 2 作为看门狗 (WDT) 操作。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。**在复位后看门狗被启用。在看门狗被启用时，对某些 PCA 寄存器的写操作受到限制。**代码开始执行后 WDT 将很快产生复位。为了避免复位，必须明确禁用 WDT（如果在系统中使用 WDT，还可选择重新配置和重新允许）。

### 29.4.1. 看门狗操作

当 WDT 被启用时：

- PCA 计数器被强制运行。
- 不允许写 PCA0L 和 PCA0H。
- PCA 时钟源位 (CPS2-CPS0) 被冻结。
- PCA 等待控制位 (CIDL) 被冻结。
- 模块 2 被强制进入软件定时器模式。
- 对模块 2 模式寄存器 (PCA0CPM2) 的写操作被禁用。

当 WDT 被允许时，写入 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 被禁用。如果 WDT 被启用，但用户软件没有允许 PCA 计数器，则 PCA 计数器运行控制位 (CR) 将读 0。如果在 WDT 被允许时 PCA0CPH2 和 PCA0H 发生匹配，则系统将被复位。为了防止 WDT 复位，可通过向 PCA0CPH2 写入任意值来更新 WDT。在写入 PCA0CPH2 时，PCA0H 加上 PCA0CPL2 中保存的偏移值后被载入到 PCA0CPH2（见图 29.11）。

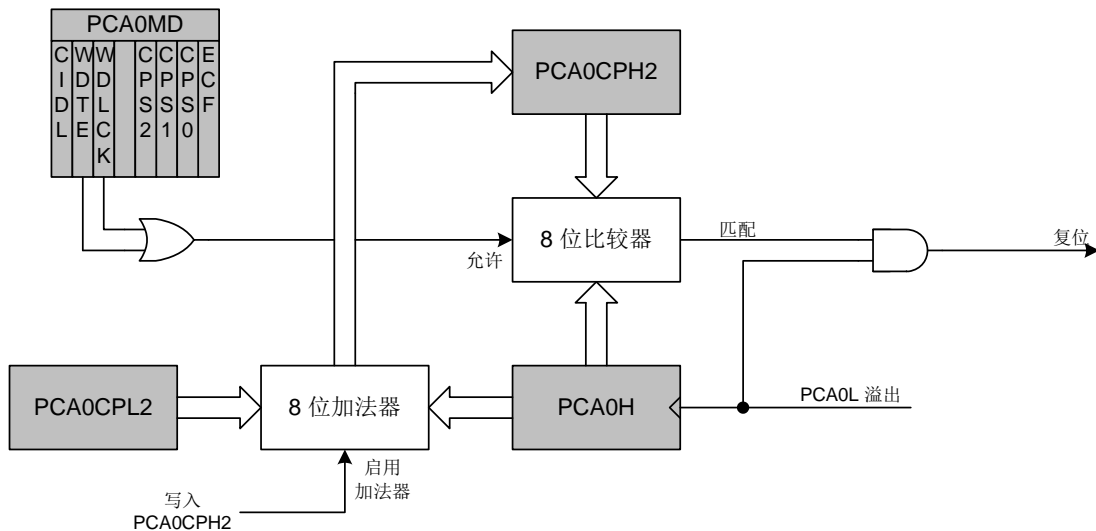


图 29.11. 启用看门狗的 PCA 模块 2

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 29.5 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 29.5: 看门狗偏移值（PCA 时钟数）

当 PCA0L 发生溢出并且 PCA0CPH2 和 PCA0H 匹配时，WDT 将产生复位。在启用 WDT 的情况下，软件可以通过向 CCF2 标志 (PCA0CN.2) 写入 1 来强制产生 WDT 复位。

#### 29.4.2. 看门狗用法

配置 WDT 的步骤如下：

1. 通过向 WDTE 位写 0 禁用 WDT。
2. 选择所需 PCA 时钟源（用 CPS2-CPS0 位）。
3. 向 PCA0CPL2 载入所需的 WDT 更新偏移值。
4. 配置 PCA 的空闲模式（如果希望在 CPU 处于空闲模式时 WDT 停止工作，则应设置 CIDL 位）。
5. 通过将 WDTE 位被设置为 1 来启用 WDT。
6. 通过写入 PCA0CPH2 复位 WDT 定时器。

在启用 WDT 时，不能改变所选的 PCA 时钟源和空闲模式。通过设置 PCA0MD 寄存器的 WDTE 或 WDLCK 位来启用看门狗。如 WDLCK 已被设置，下一次系统复位之前不得禁用 WDT。如果 WDCLK 未被设置，清除 WDTE 位将禁用 WDT。

WDT 在任何一次复位之后都被设置为允许状态。PCA0 计数器时钟默认为系统时钟/12，PCA0L 和 PCA0CPL2 的默认为 0x00。根据方程 29.5，WDT 的超时间隔为 256 个 PCA 时钟周期或 3072 个系统时钟周期。表 29.3 列出了典型系统时钟超时间隔的部分范例。

表 29.3: 看门狗超时间隔<sup>1</sup>

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

### 29.5. PCA0 寄存器说明

下面对与 PCA 操作有关的特殊功能寄存器进行详细说明。

# C8051F80x-83x

## SFR 定义 29.1: PCA0CN: PCA0 控制寄存器

位	7	6	5	4	3	2	1	0
名称	CF	CR				CCF2	CCF1	CCF0
类型	R/W	R/W	R	R	R	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xD8; 可位寻址

位	名称	功能
7	CF	<b>PCA 计数器/定时器溢出标志。</b> 当 PCA 计数器/定时器从 0xFFFF 溢出到 0x0000 时由硬件设置。当启用计数器/定时器溢出 (CF) 中断时, 设置此位会导致 CPU 转至 PCA 中断服务程序。此位不能由硬件自动清除, 必须用软件清除。
6	CR	<b>PCA 计数器/定时器运行控制。</b> 此位允许/禁用 PCA 计数器/定时器。 0: PCA 计数器/定时器禁用。 1: PCA 计数器/定时器启用。
5:3	未使用	读 = 000b; 写 = 无需在意。
2	CCF2	<b>PCA 模块 2 捕捉/比较标志。</b> 当发生匹配或捕捉时, 此位由硬件设置。当启用 CCF2 中断时, 设置此位会导致 CPU 转至 PCA 中断服务程序。此位不能由硬件自动清除, 必须用软件清除。
1	CCF1	<b>PCA 模块 1 捕捉/比较标志。</b> 当发生匹配或捕捉时, 此位由硬件设置。当启用 CCF1 中断时, 设置此位会导致 CPU 转至 PCA 中断服务程序。此位不能由硬件自动清除, 必须用软件清除。
0	CCF0	<b>PCA 模块 0 捕捉/比较标志。</b> 当发生匹配或捕捉时, 此位由硬件设置。当启用 CCF0 中断时, 设置此位会导致 CPU 转至 PCA 中断服务程序。此位不能由硬件自动清除, 必须用软件清除。

## SFR 定义 29.2: PCA0MD: PCA0 模式

位	7	6	5	4	3	2	1	0
名称	CIDL	WDTE	WDLCK		CPS2	CPS1	CPS0	ECF
类型	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
复位	0	1	0	0	0	0	0	0

SFR 地址 = 0xD9

位	名称	功能
7	CIDL	<b>PCA 计数器/定时器等待控制。</b> 设置 CPU 空闲模式下的 PCA 工作方式。 0: 当系统控制器处于空闲模式时, PCA 继续正常工作。 1: 当系统控制器处于空闲模式时, PCA 停止工作。
6	WDTE	<b>看门狗允许</b> 如果该位被设置, PCA 模块 2 被用作看门狗。 0: 看门狗被禁用。 1: PCA 模块 2 被用作看门狗。
5	WDLCK	<b>看门狗锁定。</b> 该位锁定/解锁看门狗允许。如 WDLCK 被设置, 下一次系统复位之前不得禁用 WDT。 0: 看门狗允许未被锁定。 1: 看门狗允许被锁定。
4	未使用	读 = 0b; 写 = 无需在意。
3:1	CPS[2:0]	<b>PCA 计数器/定时器脉冲选择</b> 这些位选择 PCA 计数器的时基 000: 系统时钟/12 001: 系统时钟/4 010: 定时器 0 溢出 011: ECI 负变跳 (最大速率 = 系统时钟/4) 100: 系统时钟 101: 外部时钟/8 (与系统时钟同步) 11x: 备用
0	ECF	<b>PCA 计数器/定时器溢出中断允许。</b> 该位是 PCA 计数器/定时器溢出 (CF) 中断的屏蔽位。 0: 禁用 CF 中断。 1: 当 CF (PCA0CN.7) 被设置时, 允许 PCA 计数器/定时器溢出中断请求。
<b>附注:</b> 当 WDTE 位设置为 1 时, 不能修改 PCA0MD 寄存器其他位的值。若要更改 PCA0MD 寄存器中的内容, 必须先禁用看门狗。		

# C8051F80x-83x

## SFR 定义 29.3: PCA0PWM: PCA0 PWM 配置寄存器

位	7	6	5	4	3	2	1	0
名称	ARSEL	ECOV	COVF		EAR16	CLSEL[1:0]		
类型	R/W	R/W	R/W	R	R/W	R/W		
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xF7

位	名称	功能									
7	ARSEL	<p><b>自动重载寄存器选择</b></p> <p>该位选择是否读写相同 SFR 地址上正常的 PCA 捕捉/比较寄存器 (PCA0CPn), 或自动重载寄存器。此功能用于定义 9 位至 15 位 PWM 模式和 16 位 PWM 模式的重载值。所有其他模式中, 自动重载寄存器都不具备此功能。</p> <p>0: 读/写捕捉/比较寄存器 (PCA0CPHn 和 PCA0CPLn)。</p> <p>1: 读/写自动重载寄存器 (PCA0CPHn 和 PCA0CPLn)。</p>									
6	ECOV	<p><b>周期溢出中断允许。</b></p> <p>该位设置周期溢出标志 (COVF) 中断的屏蔽。</p> <p>0: COVF 不会产生 PCA 中断。</p> <p>1: COVF 被设置时, 将产生 PCA 中断。</p>									
5	COVF	<p><b>周期溢出标志。</b></p> <p>该位表明主 PCA 计数器 (PCA0) 第 n (n 为 9 至 15) 位溢出。用于该标志的特定位置取决于 CLSEL 位的设置。该位可通过硬件或软件进行设置, 但必须通过软件清除。</p> <p>0: 从上次该位被清除后, 并无发生溢出。</p> <p>1: 从上次该位被清除后, 已发生溢出。</p>									
4	未使用	读 = 0b; 写 = 无需在意。									
3	EAR16	<p><b>16 位 PWM 自动重载允许。</b></p> <p>该位控制 16 位 PWM 模式的自动重载特征, 而该模式载入 PCA 计数器 (PCA0) 溢出上相同 SFR 地址的 PCA0CPn 捕捉/比较寄存器和自动重载寄存器值。该设置影响所有已经被配置使用 16 位 PWM 模式的 PCA 通道。</p> <p>0: 禁用 16 位 PWM 模式自动重载。该默认设置与其他设备的 16 位 PWM 模式向后兼容。</p> <p>1: 16 位 PWM 模式自动重载允许。</p>									
2:0	CLSEL[2:0]	<p><b>周期长度选择。</b></p> <p>16 位 PWM 模式未被选择时, 该等位会选择 PWM 周期的长度, 范围介于 8 到 15 位。这会影响到 PWM (未使用 16 位 PWM 模式) 配置的所有通道。就配置给 16 位 PWM 模式的个别通道而言, 忽略该位。</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%;">000: 8 位。</td> <td style="width: 33%;">011: 11 位。</td> <td style="width: 33%;">110: 14 位。</td> </tr> <tr> <td>001: 9 位。</td> <td>100: 12 位。</td> <td>111: 15 位。</td> </tr> <tr> <td>010: 10 位。</td> <td>101: 13 位。</td> <td></td> </tr> </table>	000: 8 位。	011: 11 位。	110: 14 位。	001: 9 位。	100: 12 位。	111: 15 位。	010: 10 位。	101: 13 位。	
000: 8 位。	011: 11 位。	110: 14 位。									
001: 9 位。	100: 12 位。	111: 15 位。									
010: 10 位。	101: 13 位。										



## SFR 定义 29.4: PCA0CPMn: PCA0 捕捉/比较模式寄存器

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

SFR 地址: PCA0CPM0 = 0xDA, PCA0CPM1 = 0xDB, PCA0CPM2 = 0xDC

位	名称	功能
7	PWM16n	<b>16 位脉冲宽度调制允许。</b> 当允许脉冲宽度调制模式时, 该位选择 16 位模式。 0: 选择 8 至 15 位 PWM。 1: 选择 16 位 PWM。
6	ECOMn	<b>比较器功能允许。</b> 该位设置为 1 时, 允许 PCA 模块 n 的比较器功能。
5	CAPPn	<b>正沿捕捉功能允许。</b> 该位设置为 1 时, 允许 PCA 模块 n 的正边沿捕捉。
4	CAPNn	<b>负沿捕捉功能允许。</b> 该位设置为 1 时, 允许 PCA 模块 n 的负边沿捕捉。
3	MATn	<b>匹配功能允许。</b> 该位设置为 1 时, 允许 PCA 模块 n 的匹配功能。如果允许, 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 位会设置为 1。
2	TOGn	<b>转换功能允许。</b> 该位设置为 1 时, 允许 PCA 模块 n 的转换功能。如果允许, 当 PCA 计数器与一个模块的捕捉/比较寄存器匹配时, CEXn 针脚的逻辑电平会进行转换。如果 PWMn 位也设置为 1, 模块将在频率输出模式下运行。
1	PWMn	<b>脉宽调制模式允许。</b> 该位设置为 1 时, 允许 PCA 模块 n 的 PWM 功能。当允许时, CEXn 针脚输出脉冲宽度调制信号。PWM16n 清除时, 使用 8 至 15 位 PWM 模式, PWM16n 设置为逻辑 1 时, 使用 16 位模式。如果 TOGn 位也设置为逻辑 1, 则模块在频率输出模式下运行。
0	ECCFn	<b>捕捉/比较标志中断允许。</b> 该位设置捕捉/比较标志 (CCFn) 的中断屏蔽。 0: 禁用 CCFn 中断。 1: 当 CCFn 位设置为 1 时, 允许捕捉/比较标志的中断请求。

**附注:** 当 WDTE 位设置为 1 时, 不得更改 PCA0CPM2 寄存器, 而模块 2 将用作看门狗。要更改 PCA0CPM2 寄存器中的内容或模块 2 的功能, 必须禁用看门狗。

# C8051F80x-83x

## SFR 定义 29.5: PCA0L: PCA0 计数器/定时器低字节寄存器

位	7	6	5	4	3	2	1	0
名称	PCA0[7:0]							
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xF9

位	名称	功能
7:0	PCA0[7:0]	<b>PCA 计数器/定时器低字节。</b> PCA0L 寄存器保存 16 位 PCA 计数器/定时器的低字节 (LSB)。

附注: WDTE 位设置为 1 时, PCA0L 寄存器不能通过软件修改。若要更改 PCA0L 寄存器中的内容, 必须先禁用看门狗。

## SFR 定义 29.6: PCA0H: PCA0 计数器/定时器高字节

位	7	6	5	4	3	2	1	0
名称	PCA0[15:8]							
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址 = 0xFA

位	名称	功能
7:0	PCA0[15:8]	<b>PCA 计数器/定时器高字节。</b> PCA0H 寄存器保存 16 位 PCA 计数器/定时器的高字节 (MSB)。读该寄存器将读取"瞬象"寄存器的内容, 其内容仅在读 PCA0L 的内容时更新 (参阅第 29.1 条)。

附注: WDTE 位设置为 1 时, PCA0H 寄存器不能通过软件修改。若要更改 PCA0H 寄存器中的内容, 必须先禁用看门狗。

**SFR 定义 29.7: PCA0CPLn: PCA0 捕捉模块低字节寄存器**

位	7	6	5	4	3	2	1	0
名称	PCA0CPn[7:0]							
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址: PCA0CPL0 = 0xFB, PCA0CPL1 = 0xE9, PCA0CPL2 = 0xEB

位	名称	功能
7:0	PCA0CPn[7:0]	<b>PCA 捕捉模块低字节寄存器</b> PCA0CPLn 寄存器保存 16 位捕捉模块 n 的低字节 (LSB)。该寄存器地址也会允许读取 9 位至 15 位 PWM 模式和 16 位 PWM 模式中相应 PCA 通道自动重载值的低字节。寄存器 PCA0PWM 中的 ARSEL 位决定会读取哪个寄存器。

附注: 写入该寄存器会将模块的 ECOMn 位清 0。

**SFR 定义 29.8: PCA0CPHn: PCA0 捕捉模块高字节寄存器**

位	7	6	5	4	3	2	1	0
名称	PCA0CPn[15:8]							
类型	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位	0	0	0	0	0	0	0	0

SFR 地址: PCA0CPH0 = 0xFC, PCA0CPH1 = 0xEA, PCA0CPH2 = 0xEC

位	名称	功能
7:0	PCA0CPn[15:8]	<b>PCA 捕捉模块高字节寄存器</b> PCA0CPHn 寄存器保存 16 位捕捉模块 n 的高字节 (MSB)。该寄存器地址也会允许读取 9 位至 15 位 PWM 模式和 16 位 PWM 模式中相应 PCA 通道自动重载值的高字节。寄存器 PCA0PWM 中的 ARSEL 位决定会读取哪个寄存器。

附注: 写入该寄存器会将模块的 ECOMn 位设置为 1。

# C8051F80x-83x

---

## 30. C2 接口

C8051F80x-83x 设备有一个 Silicon Labs 2 线 (C2) 调试接口, 支持 闪存编程和使用安装在最终应用程序中的部件进行系统内调试。C2 接口使用两个针脚进行运作: 一个双向的 C2 数据信号 (C2D) 和一个时钟输入 (C2CK)。有关 C2 协议的详细信息, 请参阅 C2 接口规格。

### 30.1. C2 接口寄存器

下面对通过 C2 接口执行 闪存编程功能所需的 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 接口规格)。

#### C2 寄存器定义 30.1: C2ADD: C2 地址寄存器

位	7	6	5	4	3	2	1	0
名称	C2ADD[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

位	名称	功能																																	
7:0	C2ADD[7:0]	<p><b>C2 地址寄存器。</b></p> <p>C2ADD 寄存器通过 C2 接口访问, 以选择 C2 数据读取和数据写入命令的目标数据寄存器。</p> <table border="1"> <thead> <tr> <th>地址</th> <th>名称</th> <th>说明</th> </tr> </thead> <tbody> <tr> <td>0x00</td> <td>DEVICEID</td> <td>选择设备 ID 寄存器 (只读)</td> </tr> <tr> <td>0x01</td> <td>REVID</td> <td>选择版本 ID 寄存器 (只读)</td> </tr> <tr> <td>0x02</td> <td>FPCTL</td> <td>选择 C2 闪存编程控制寄存器</td> </tr> <tr> <td>0xBF</td> <td>FPDAT</td> <td>选择 C2 闪存数据寄存器</td> </tr> <tr> <td>0xD2</td> <td>CRC0AUTO*</td> <td>选择 CRC0AUTO 寄存器</td> </tr> <tr> <td>0xD3</td> <td>CRC0CNT*</td> <td>选择 CRC0CNT 寄存器</td> </tr> <tr> <td>0xCE</td> <td>CRC0CN*</td> <td>选择 CRC0CN 寄存器</td> </tr> <tr> <td>0xDE</td> <td>CRC0DATA*</td> <td>选择 CRC0DATA 寄存器</td> </tr> <tr> <td>0xCF</td> <td>CRC0FLIP*</td> <td>选择 CRC0FLIP 寄存器</td> </tr> <tr> <td>0xDD</td> <td>CRC0IN*</td> <td>选择 CRC0IN 寄存器</td> </tr> </tbody> </table>	地址	名称	说明	0x00	DEVICEID	选择设备 ID 寄存器 (只读)	0x01	REVID	选择版本 ID 寄存器 (只读)	0x02	FPCTL	选择 C2 闪存编程控制寄存器	0xBF	FPDAT	选择 C2 闪存数据寄存器	0xD2	CRC0AUTO*	选择 CRC0AUTO 寄存器	0xD3	CRC0CNT*	选择 CRC0CNT 寄存器	0xCE	CRC0CN*	选择 CRC0CN 寄存器	0xDE	CRC0DATA*	选择 CRC0DATA 寄存器	0xCF	CRC0FLIP*	选择 CRC0FLIP 寄存器	0xDD	CRC0IN*	选择 CRC0IN 寄存器
地址	名称	说明																																	
0x00	DEVICEID	选择设备 ID 寄存器 (只读)																																	
0x01	REVID	选择版本 ID 寄存器 (只读)																																	
0x02	FPCTL	选择 C2 闪存编程控制寄存器																																	
0xBF	FPDAT	选择 C2 闪存数据寄存器																																	
0xD2	CRC0AUTO*	选择 CRC0AUTO 寄存器																																	
0xD3	CRC0CNT*	选择 CRC0CNT 寄存器																																	
0xCE	CRC0CN*	选择 CRC0CN 寄存器																																	
0xDE	CRC0DATA*	选择 CRC0DATA 寄存器																																	
0xCF	CRC0FLIP*	选择 CRC0FLIP 寄存器																																	
0xDD	CRC0IN*	选择 CRC0IN 寄存器																																	
*附注: CRC 寄存器和功能说明见第 158 页的第 24 条“循环冗余检验单位 (CRC0)”。																																			

#### C2 寄存器定义 30.2: DEVICEID: C2 设备 ID

位	7	6	5	4	3	2	1	0
名称	DEVICEID[7:0]							
类型	R/W							
复位	1	1	1	0	0	0	0	1

# C8051F80x-83x

C2 地址寄存器: 0x00

位	名称	功能
7:0	DEVICEID[7:0]	设备 ID。 该只读寄存器返回 8 位的设备 ID 号: 0x23 (C8051F80x-83x)。

## C2 寄存器定义 30.3: REVID: C2 版本 ID

位	7	6	5	4	3	2	1	0
名称	REVID[7:0]							
类型	R/W							
复位	不同	不同	不同	不同	不同	不同	不同	不同

C2 地址寄存器: 0x01

位	名称	功能
7:0	REVID[7:0]	版本 ID。 该只读寄存器返回 8 位的版本 ID。例如: 0x00 = 版本 A。

**C2 寄存器定义 30.4: FPCTL: C2 闪存编程控制寄存器**

位	7	6	5	4	3	2	1	0
名称	FPCTL[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

C2 地址寄存器: 0x02

位	名称	功能
7:0	FPCTL[7:0]	<p><b>C2 闪存编程控制寄存器。</b></p> <p>该寄存器用于通过 C2 接口启用闪存编程。欲启用 C2 闪存编程，下列代码须按顺序写入：0x02、0x01。一旦启用 C2 闪存编程，须发出系统复位以恢复至正常操作。</p>

**C2 寄存器定义 30.5: FPDAT: C2 闪存编程数据寄存器**

位	7	6	5	4	3	2	1	0
名称	FPDAT[7:0]							
类型	R/W							
复位	0	0	0	0	0	0	0	0

C2 地址寄存器: 0xBF

位	名称	功能										
7:0	FPDAT[7:0]	<p><b>C2 闪存编程数据寄存器。</b></p> <p>该寄存器用于在 C2 闪存访问期间传递闪存命令、地址和数据。下列命令为有效命令。</p> <table border="1"> <thead> <tr> <th>代码</th> <th>命令</th> </tr> </thead> <tbody> <tr> <td>0x06</td> <td>闪存区块读取</td> </tr> <tr> <td>0x07</td> <td>闪存区块写入</td> </tr> <tr> <td>0x08</td> <td>闪存页面擦除</td> </tr> <tr> <td>0x03</td> <td>设备擦除</td> </tr> </tbody> </table>	代码	命令	0x06	闪存区块读取	0x07	闪存区块写入	0x08	闪存页面擦除	0x03	设备擦除
代码	命令											
0x06	闪存区块读取											
0x07	闪存区块写入											
0x08	闪存页面擦除											
0x03	设备擦除											

# C8051F80x-83x

## 30.2. C2CK 针脚共享

C2 协议允许 C2 针脚与用户功能共享，因此，系统调试和闪存编程可执行。此种情况可能发生，因为 C2 通信通常在设备处于停止模式下时执行，该模式下片上所有外围设备和用户软件都暂停工作。在此停止模式下，C2 接口可安全地“借用”C2CK (RST) 和 C2D 针脚。在大多数应用中，外部电阻器需将 C2 接口流量从用户应用中分离出来。典型的分离配置如图 30.1 所示。

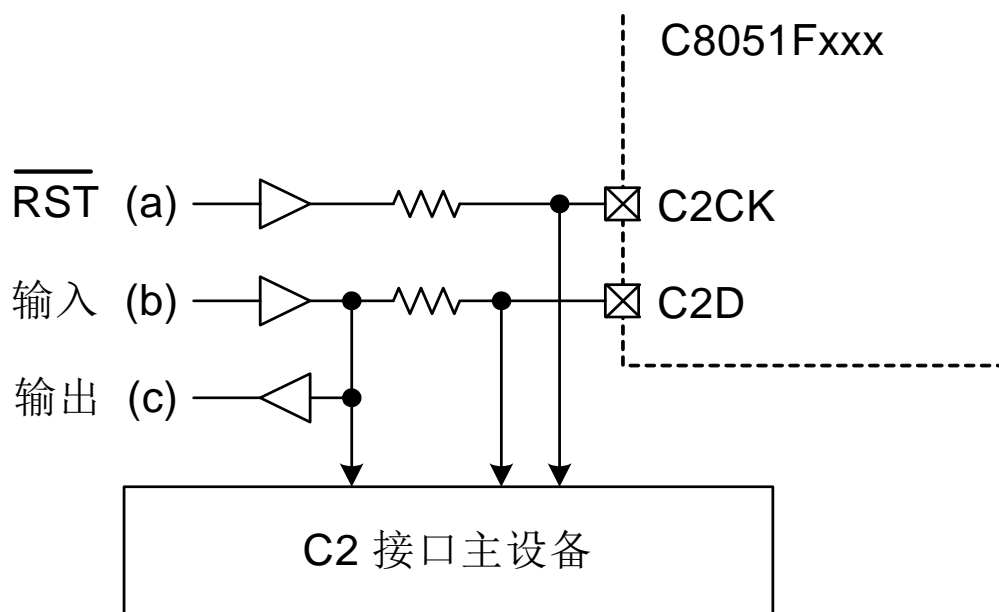


图 30.1: 典型 C2 针脚共享

图 30.1 中的配置以下列各项为条件:

1. 当目标设备处于停止模式时，用户输入 (b) 不得更改模式。
2. 目标设备上的  $\overline{\text{RST}}$  针脚仅可用作输入。

视乎具体应用的情况，可能需要额外的电阻器。



## 文件更新一览表

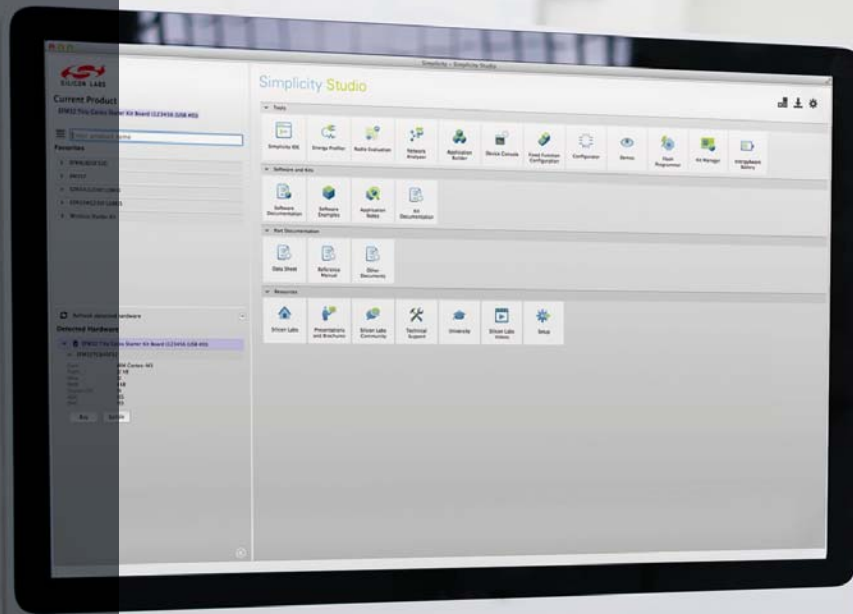
### 0.2 版更新为 1.0 版

- 更新后的电气特性表可反映生产特性数据。
- 增加了写入或擦除闪存所需的最低 SYSCLK 要求。
- 增加了在唤醒源活跃状态下进入暂停模式的注意事项（第 20.3 节）
- 将 VDMOCN 复位值更正为“变化”。
- 删除了输出引脚表中的 IDAC 部分。

# C8051F80x-83x

---

附注:



## Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



**IoT Portfolio**  
[www.silabs.com/IoT](http://www.silabs.com/IoT)



**SW/HW**  
[www.silabs.com/simplicity](http://www.silabs.com/simplicity)



**Quality**  
[www.silabs.com/quality](http://www.silabs.com/quality)



**Support and Community**  
[community.silabs.com](http://community.silabs.com)

### Disclaimer

Silicon Laboratories intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Laboratories products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Laboratories reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Laboratories shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products must not be used within any Life Support System without the specific written consent of Silicon Laboratories. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Laboratories products are generally not intended for military applications. Silicon Laboratories products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

### Trademark Information

Silicon Laboratories Inc., Silicon Laboratories, Silicon Labs, SiLabs and the Silicon Labs logo, CMEMS®, EFM, EFM32, EFR, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZMac®, EZRadio®, EZRadioPRO®, DSPLL®, ISOModem®, Precision32®, ProSLIC®, SiPHY®, USBXpress® and others are trademarks or registered trademarks of Silicon Laboratories Inc. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.



**SILICON LABS**

Silicon Laboratories Inc.  
 400 West Cesar Chavez  
 Austin, TX 78701  
 USA

<http://www.silabs.com>