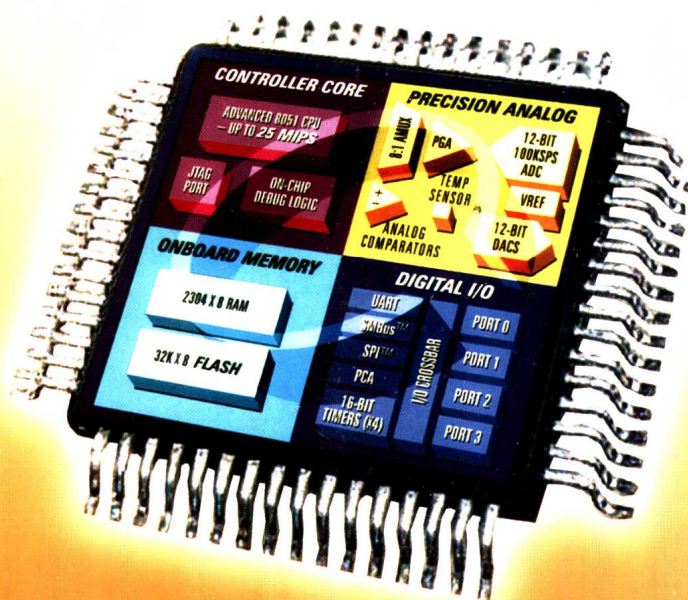


Cygnal 单片机



C8051F 单片机应用解析



[美] Cygnal Integrated Products, Inc. 著

潘琢金 孙德龙 夏秀峰 译



北京航空航天大学出版社

<http://www.buaapress.com.cn>

C8051F 单片机应用解析

[美] Cygnal Integrated Products, Inc. 著

潘琢金 孙德龙 夏秀峰 译

北京航空航天大学出版社

<http://www.buaapress.com.cn>

内 容 简 介

本书汇集了 Cygnal 集成产品公司迄今为止发布的有关 C8051F 高速 SoC 单片机的全部 29 篇应用笔记,内容涉及到 C8051F 单片机内部各种模拟和数字资源的原理介绍、操作和编程,并给出了大量完整的应用实例。这些应用实例对于快速学习和更好地应用 C8051F 单片机将起到事半功倍的作用。

本书可作为使用 C8051F 单片机进行产品开发的工程技术人员的硬件和软件设计参考书,其中有相当一部分内容对基于其他类型单片机的产品设计也具有很重要的参考价值。

图书在版编目(CIP)数据

C8051F 单片机应用解析/潘琢金等译. —北京:北京
航空航天大学出版社,2002.10
ISBN 7-81077-224-4

I. C… II. 潘… III. 单片微型计算机,C8051F
IV. TP368.1

中国版本图书馆 CIP 数据核字(2002)第 059730 号

该书出版得到原版权拥有者美国 Cygnal Integrated Products, Inc. 的授权。所有
权保留。

C8051F 单片机应用解析

[美] Cygnal Integrated Products, Inc. 著

潘琢金 孙德龙 夏秀峰 译

责任编辑 刘晓明

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100083) 发行部电话:010-82317024 传真:010-82328026

<http://www.buaapress.com.cn>

E-mail: pressell@publica.bj.cninfo.net

河北省涿州市新华印刷厂印装 各地书店经销

*

开本:787×1092 1/16 印张:26.75 字数:685 千字

2002 年 10 月第 1 版 2002 年 10 月第 1 次印刷 印数:5 000 册

ISBN 7-81077-224-4 定价:40.00 元

北京市版权局著作权登记号 图字:01-2002-3053 号

前 言

Cygnal 集成产品公司的 C8051F 系列单片机是完全集成的混合信号系统级芯片(SoC),具有与 MCS-51 指令集完全兼容的高速 CIP-51 内核;峰值速度可达 25 MIPS;在一个芯片内集成了构成一个单片机数据采集或控制系统所需要的几乎所有模拟和数字外设及其他功能部件(包括 PGA、ADC、DAC、电压比较器、电压基准、温度传感器、SMBus/I²C、UART、SPI、定时器、可编程计数器/定时器阵列、内部振荡器、看门狗定时器及电源监视器等);具有大容量的可在系统(ISP)和在应用(IAP)编程的 FLASH 存储器。由于 C8051F 单片机与其他 8 位单片机相比具有更为优异的性能,所以一经面世就得到了广大单片机系统设计工程师的青睐,成为很多测控系统设计的首选机型。

尽管 C8051F 单片机与 MCS-51 指令集完全兼容,但与标准 8051 相比扩展了很多功能部件,并使用了一些在单片机中前所未见的技术,在配置和编程方面比标准 8051 要复杂一些,因此很多用户迫切希望能得到一些应用和编程示例。Cygnal 集成产品公司为使用户能够快速学习和更好地应用 C8051F 单片机进行系统设计,陆续发布了 29 篇应用笔记。这些应用笔记内容广泛,涉及到如何提高 ADC 分辨率、用 DAC 作为函数发生器、用内部温度传感器测量环境温度、SMBus 通信、软件 UART、软件 SPI、端口 I/O 交叉开关和振荡器配置、FLASH 的在系统和在应用编程、FLASH 安全、电源管理及功耗计算、混合电源系统等很多单片机系统设计者所关心的技术。大部分应用笔记中都有完整的应用实例,并提供带有详细注释的 C51 或汇编语言源代码。这些源代码只须稍加修改即可在用户程序中使用,为 C8051F 单片机的软件开发提供了极大的方便。一些应用笔记中所介绍的技术不但适用于 C8051F 单片机,而且对使用其他单片机进行产品设计也具有重要的参考价值。

自 2001 年 10 月起,译者在沈阳新华龙电子有限公司的网站上陆

续发布了前 20 篇应用笔记的中文版,得到了读者的欢迎和好评。应广大读者的要求,译者对已经发布在网上的 20 篇应用笔记重新整理并根据原文的最新版本更新了内容,又将 Cygnal 公司后来发布的 9 篇应用笔记也翻译出来,汇成此书。希望此书能对广大单片机系统设计人员提供帮助。

在翻译过程中发现了原文中的一些错误并做了更正。有关 Cygnal 集成产品公司的最新单片机资料和应用信息,请访问 Cygnal 公司的网站 <http://www.cygnal.com> 或沈阳新华龙电子有限公司的网站 <http://www.xhl.com.cn>。

本书由潘琢金、孙德龙、夏秀峰翻译,全书由潘琢金统稿。北京航空航天大学何立民教授、马广云博士和沈阳新华龙电子有限公司对本书的出版给予了很多的关心和支持,在此表示衷心的感谢。

由于时间仓促,译者水平有限,对书中存在的错误和不准确之处,恳请读者不吝赐教。译者联系方式: panzhj@syiae.edu.cn 或 panzhuojin@sina.com。

译 者

2002 年 6 月

于沈阳航空工业学院

目 录

AN001——配置端口 I/O 交叉开关译码器	1
AN002——配置内部和外部振荡器	10
AN003——使用片内温度传感器	23
AN004——在 Cygnal IDE 中集成 Keil 8051 工具	37
AN005——通过 JTAG 接口对 FLASH 编程	42
AN006——外部 SRAM 与 C8051F000 接口	88
AN007——用 PCA 实现 16 位 PWM	97
AN008——实现一个实时时钟	114
AN009——从应用程序写 FLASH	124
AN010——用片内定时器实现 16 位 PWM	128
AN011——在 5 V 系统中使用 C8051Fxxx	138
AN012——C8051F0xx 引导装入程序设计	142
AN013——用 SMBus 实现串行通信	149
AN014——微细间距 QFP 器件手工焊接指南	189
AN015——软件 UART 示例	205
AN016——电源管理技术及功耗计算	243
AN017——使用 C8051Fxxx 片内 FLASH 编程接口 DLL	252
AN018——用过采样和求均值提高 ADC 分辨率	260
AN019——计算开关电容 ADC 的建立时间	278
AN020——FLASH 安全用户指南	285
AN021——从 Metalink ASM51 汇编器到 Keil A51 汇编器的转换	289
AN022——C8051F02x 系列 C 程序示例	290
AN023——使用 DAC 作为函数发生器	330
AN024——C2 接口的引脚共享技术	348
AN025——在 Cygnal IDE 中集成 Raisonance 8051 工具	352
AN026——在 Cygnal IDE 中集成 Tasking 8051 工具	357
AN027——通过 C2 接口对 FLASH 编程	362
AN028——C8051F30x 系列 SPI 程序示例	385
AN029——从 F30x 器件的应用程序写 FLASH	415

AN001——配置端口 I/O 交叉开关译码器

1 引言

本应用笔记的目的是说明如何配置和使用端口 I/O 交叉开关译码器。本文提供软件示例。

如图 1 所示,交叉开关是一个多路选择器,用于为器件内部的硬件外设分配 I/O 端口。例如,它可以决定 UART 的 RXD 和 TXD 连到哪一个端口引脚。

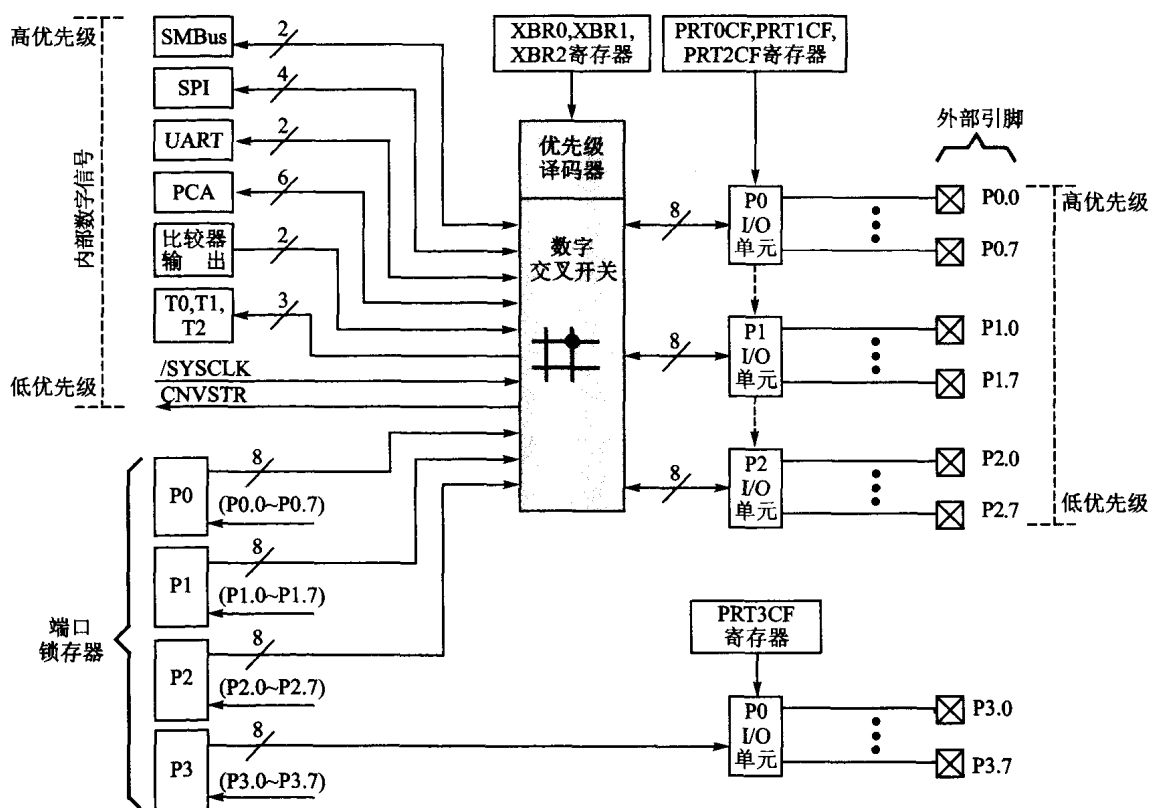


图 1 交叉开关译码器原理框图

交叉开关负责 SMBus、SPI、UART、定时器捕捉模块、外部 PCA 输入、比较器输出、定时器外部输入、/SYSCLK 以及 A/D 转换启动输入的引脚分配。必须在访问这些外设的 I/O 之前配置和允许交叉开关。未被指定的端口引脚作为通用 I/O 引脚。

交叉开关提供了两个关键的系统特性：

- 在端口 0、端口 1 和端口 2 上的所有未分配的通用 I/O 口引脚都被连续组合在一起。
- 对于引脚数量少的器件,它提供了外设选择的灵活性。外设选择只受限于可用的端口引脚数,而不受限于哪一个引脚是可用的。这就允许系统设计者在使用引脚数少的器件时,能选择将哪些数字外设分配到器件上的数字 I/O 引脚。

2 关键点

- 为了使用端口 0、端口 1 或端口 2 的任何一个引脚作为输出,交叉开关必须被允许。
- 交叉开关译码器必须在任何一个数字外设被允许前配置。
- 交叉开关通常在复位时被配置一次(在复位处理过程的最开始处),以后不再进行配置。
- 交叉开关的设置改变器件的引脚分配。
- 每一种交叉开关设置导致惟一的器件引脚分配,如果在交叉开关中允许或禁止外设,则引脚分配将发生变化。
- 对于输出端口引脚,其输出方式(漏极开路或推挽)必须被明确设置,即使那些由交叉开关分配的端口引脚也是如此。例外情况是 SMBus 上的 SDA 和 SCL 及 UART 的 RXD 引脚,这些引脚被自动配置为漏极开路。
- 交叉开关分配的输入引脚(例如 NSS 或/INT0)是漏极开路还是推挽并不重要。这些引脚被配置为输入,而与相应端口配置寄存器的设置无关。
- 为了将一个通用 I/O 引脚配置为输入,与这一引脚相关的端口配置寄存器位必须被清“0”,这样即可选择该引脚为漏极开路输出方式。另外,与该引脚相关的端口位必须被置 1,这样使该引脚处于高阻态,或在 XBR2 中的 WEAKPUD 被清“0”时弱上拉为高电平,这是端口引脚的复位配置。
- 在任意时刻可以通过读取相应端口 SFR 得到端口引脚的电平值,而不管交叉开关寄存器的设置如何或引脚被配置为输入还是输出。
- 交叉开关寄存器中的允许位是独立的,与数字外设本身的允许位是分开的。
 - 外设部件在使用时不需要在交叉开关中被允许(例如,一个 PCA 模块即使在输出没有被接到引脚的情况下也能产生中断)。
 - 那些在交叉开关中被允许但在它们自己的 SFR 中被禁止的外设,仍然控制端口引脚。这就是说,端口引脚可以在任何时刻读取,但是输出被占用它的外设所控制,不能作为通用输出口来访问。
- 端口 1 上的四个外部中断(P1.[4..7])是由引脚上的下降沿触发的,与下降沿触发电源、交叉开关设置或端口引脚的输出方式无关。
- 与标准 8051 不同的是,C8051 提供了真正的推挽输出。如果需要 8051 的上拉功能,则可以通过将相应的端口输出配置为“推挽”方式,然后再配置为“漏极开路”来进行软件仿真。

3 确定器件的引脚分配

本节介绍如何使用优先权交叉开关译码表即表 4 来决定器件的引脚分配,这种分配基于交叉开关寄存器中的外设选择,这些选择在图 2~图 4 中列出。

为了确定引脚分配,首先根据所需要的外设来配置交叉开关寄存器,然后从优先权交叉开关译码表的顶部开始向下扫描,直到遇到第一个被允许的外设部件。该设备将使用 P0.0,如果需要多个引脚,则可以按顺序从 P0.1 向后分配。例如,如果 SPI 是第一个被允许的外设,则 SCK, MISO, MOSI 和 NSS 将分别被分配到 P0.0, P0.1, P0.2, P0.3。下一个被允许的外

设将被分配到引脚 P0.4。所有未被分配的引脚作为通用 I/O。

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值:
CP0OEN	ECIE	PCA0ME			UARTEN	SPI0OEN	SMB0OEN	00000000
位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	SFR 地址: 0xE1

图 2 XBR0:端口 I/O 交叉开关寄存器 0

- 位 7 CP0OEN:比较器 0 输出允许位。
0:CP0 不连到端口引脚;
1:CP0 连到端口引脚。
- 位 6 ECIE:PCA0 计数器输入允许位。
0:ECI 不连到端口引脚;
1:ECI 连到端口引脚。
- 位 5~3 PCA0ME:PCA 模块 I/O 允许位。
000:所有的 PCA I/O 都不连到端口引脚;
001:CEX0 连到端口引脚;
010:CEX0,CEX1 连到 2 个端口引脚;
011:CEX0,CEX1,CEX2 连到 3 个端口引脚;
100:CEX0,CEX1,CEX2,CEX3 连到 4 个端口引脚;
101:CEX0,CEX1,CEX2,CEX3,CEX4 连到 5 个端口引脚;
110:保留;
111:保留。
- 位 2 UARTEN:UART I/O 允许位。
0:UART I/O 不连到端口引脚;
1:RX,TX 连到 2 个端口引脚。
- 位 1 SPI0OEN:SPI 总线 I/O 允许位。
0:SPI I/O 不连到端口引脚;
1:MISO,MOSI,SCK 和 NSS 连到 4 个端口引脚。
- 位 0 SMB0OEN:SMBus 总线 I/O 允许位。
0:SMBus I/O 不连到端口引脚;
1:SDA 连到 P0.0 ,SCL 连到 P0.1。

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值:
SYSCKE	T2EXE	T2E	INT1E	T1E	INT0E	T0E	CP1OEN	00000000
位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	SFR 地址: 0xE2

图 3 XBR1:端口 I/O 交叉开关寄存器 1

- 位 7 SYSCKE:SYSCLK 输出允许位。
0:SYSCLK 不连到端口引脚;
1:SYSCLK 连到端口引脚。

- 位 6 T2EXE: T2EX 允许位。
0: T2EX 不连到端口引脚;
1: T2EX 连到端口引脚。
- 位 5 T2E: T2 允许位。
0: T2 不连到端口引脚;
1: T2 连到端口引脚。
- 位 4 INT1E: /INT1 允许位。
0: /INT1 不连到端口引脚;
1: /INT1 连到端口引脚。
- 位 3 T1E: T1 允许位。
0: T1 不连到端口引脚;
1: T1 连到端口引脚。
- 位 2 INT0E: /INT0 允许位。
0: /INT0 不连到端口引脚;
1: /INT0 连到端口引脚。
- 位 1 T0E: T0 允许位。
0: T0 不连到端口引脚;
1: T0 连到端口引脚。
- 位 0 CP1OEN: 比较器 1 输出允许位。
0: CP1 不连到端口引脚;
1: CP1 连到端口引脚。

R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	复位值:
WEAKPUD	XBARE	—	—	—	—	—	CNVSTE	00000000
位 7	位 6	位 5	位 4	位 3	位 2	位 1	位 0	SFR 地址: 0xE3

图 4 XBR2: 端口 I/O 交叉开关寄存器 2

- 位 7 WEAKPUD: 端口 I/O 弱上拉禁止位。
0: 弱上拉允许(除了 I/O 被配置为推挽方式的端口);
1: 弱上拉禁止。
- 位 6 XBARE: 交叉开关允许位。
0: 交叉开关禁止;
1: 交叉开关允许。
- 位 5~1 未用。读=00000b, 写=忽略。
- 位 0 CNVSTE: ADC 转换启动输入允许位。
0: CNVSTR 不连到端口引脚;
1: CNVSTR 连到端口引脚。

例 1 假设一个应用需要：

- SPI；
- UART；
- 两个捕捉模块；
- /INT0；
- T2。

参考图 2~图 4 中的 I/O 端口交叉开关寄存器说明，将交叉开关寄存器配置如下：

XBR0=00010110b,允许 UART、两个捕捉模块和 SPI；

XBR1=00100100b,允许 T2 和 /INT0；

XBR2=01000000b,允许交叉开关。

从表 1 的顶部开始发现 SPI 引脚将占用 P0.[0..3](因为不使用 SMBus),UART 占用引脚 P0.[4..5],CEX0 占用引脚 P0.6,CEX1 占用引脚 P0.7,/INT0 占用引脚 P1.0,T2 占用引脚 P1.1。其他端口引脚 P1.[2..7]和引脚 P2.[0..7]作为通用端口引脚。引脚分配情况列于表 1。

表 1 例 1 的引脚分配

引 脚	功 能	引 脚	功 能
P0.0	SCK	P0.6	CEX0
P0.1	MISO	P0.7	CEX1
P0.2	MOSI	P1.0	/INT0
P0.3	NSS	P1.1	T2
P0.4	TX	P1.[2..7]	GPIO
P0.5	RX	P2.[0..7]	GPIO

例 2 假设一个应用需要：

- UART；
- /INT1；
- /SYSCLK；
- CNVSTR。

据此，交叉开关寄存器的配置如下：

XBR0=00000100b,允许 UART；

XBR1=10010000b,允许 /INT1 和 /SYSCLK；

XBR2=01000001b,允许 CNVSTR 和交叉开关。

引脚分配情况列于表 2。

表 2 例 2 的引脚分配

引 脚	功 能	引 脚	功 能
P0.0	TX	P0.4	CNVSTR
P0.1	RX	P0.[5..7]	GPIO
P0.2	/INT1	P1.[0..7]	GPIO
P0.3	/SYSCLK	P2.[0..7]	GPIO

续表 4

I/O 引脚	P0								P1								P2							
	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7	0	1	2	3	4	5	6	7
NSS				•		•																		
TX	•		•		•		•																	
RX		•		•		•		•																
CEX0	•		•		•		•		•															
CEX1		•		•		•		•		•														
CEX2			•		•		•		•		•													
CEX3				•		•		•		•		•												
CEX4					•		•		•		•		•											
EC1	•	•	•	•	•	•	•	•	•	•	•	•	•	•										
CP0	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•									
CP1	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•								
T0	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•							
/INT0	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•						
T1	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•					
/INT1	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•				
T2	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•			
T2EX	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•		
/SYSCLK	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	
CNVSTR	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

4 软件示例

下面的代码段说明如何配置交叉开关寄存器。

```

;-----
; CYGNAL INTEGRATED PRODUCTS, INC
;
;
; 文件名:      cross1. ASM
; 目标 MCU:    C8051F000
; 说明:        用于配置交叉开关的示例代码
;
; 实现笔记
;
;

```

```

;-----

;-----
; 等价定义
;-----

$MOD8F000

;-----
; 全局变量及定义
;-----
; 交叉开关端口引脚定义(可选)
SCK    EQU    P0.0
MISO   EQU    P0.1
MOSI   EQU    P0.2
NSS    EQU    P0.3
TX      EQU    P0.4
RX      EQU    P0.5
CEX0   EQU    P0.6
CEX1   EQU    P0.7
/INT0  EQU    P1.0
T2      EQU    P1.1
; 其他变量
;-----
; 中断向量表
;-----
; 复位向量
        org     00h
        ljmp    Reset
; 其他中断向量
;-----
; 中断向量代码
;-----

        org     0B3h
Reset:
        mov     XBR0, 00010110b    ; 允许 SPI, UART 和两个外部捕捉模块
        mov     XBR1, 00100100b    ; 允许 /INT0 和 T2
        mov     XBR2, 01000000b    ; 允许交叉开关和弱上拉
                                     ; 端口引脚分配如下
                                     ; P0.0 = SCK, 例: 主模式
                                     ; P0.1 = MISO
                                     ; P0.2 = MOSI

```

```

; P0.3 = NSS
; P0.4 = TX
; P0.5 = RX
; P0.6 = CEX0, 例: 输出
; P0.7 = CEX1, 例: 输入
; P1.0 = /INT0
; P1.1 = T2
; P1.[2..7], P2.[0..7] 为 GPIO
mov    PRT0CF, #01010101b ; CEX0, TX, MOSI 和 SCK 为推挽输出
mov    PRT1CF, #00000000b ; T2 和 /INT0 是由交叉开关分配的输入, 因此与它们的
; 端口配置值无关
mov    PRT2CF, #00000000b ; 配置端口 2 全部为输入
mov    P2, #11111111b ; 为了将未被分配的通用 I/O 脚配置为输入, 输出类型必
; 须为漏极开路, 且端口位必须被置 1

; 其他配置代码
ljmp   Main

; -----
; 主程序代码
; -----
Main:
    Sjmp    $ ; 原地跳转
    ; 此处为主程序代码

; -----
; 结束
; -----

END

```

AN002——配置内部和外部振荡器

1 引言

本应用笔记的目的是介绍如何配置和使用内部及外部振荡器。本文提供了配置说明、设置举例和示例代码。图 1 为内部和外部振荡器电路结构。

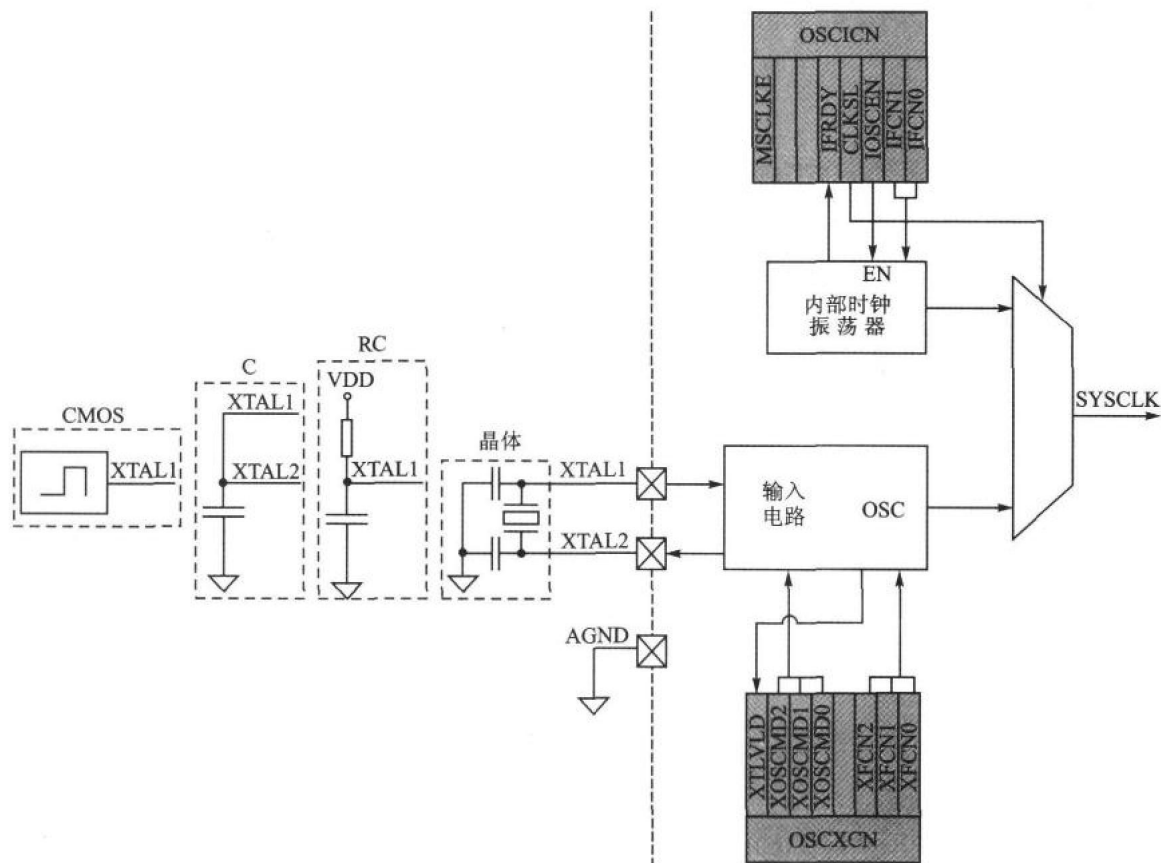


图 1 内部和外部振荡器电路结构

2 关键点

- 在器件复位时内部振荡器被自动使能并被选择为系统时钟，工作于 1.9 MHz。
- 系统时钟可以很容易地在内部振荡器和外部振荡器之间进行切换。
- 选择外部振荡器作为系统时钟和禁止内部振荡器可以在同一个写操作内完成。在外部振荡器运行时，可以用同一个写操作允许内部振荡器和选择内部振荡器作为系统时钟。
- 在使用内部振荡器作为系统时钟的情况下，允许改变内部振荡器的频率。
- 在所有的振荡器方式中，/SYSCLK 是系统时钟的缓冲输出，可以在交叉开关中允许其输出到输出端口引脚。

- 如果时钟丢失检测器被允许,将在系统时钟频率降到 10 kHz 左右时产生复位。
- 晶体振荡器有效标志可以用于在晶体振荡器稳定后产生中断,这样可以让中断处理程序将时钟切换到外部振荡器。

3 配置说明

除了具有高度可配置性外,C8051 的振荡器还具有灵活和易于使用的特点。

系统时钟可以自由地在内部振荡器和外部振荡器之间进行切换,也可以在选择内部振荡器时让外部振荡器保持在允许状态,这样可以避免在系统时钟被切换到外部振荡器时的启动延迟。

外部振荡器具有很高的可配置性,为系统设置者提供了多种选择。时基信号可以从外部 CMOS 电平时钟源、晶体或陶瓷谐振器、RC 组合电路或外部电容获得。

内部和外部振荡器的工作受两个 SFR 寄存器即 OSCICN(内部振荡器控制寄存器)和 OSCXCN(外部振荡器控制寄存器)控制。图 8 和图 9 分别给出了这两个寄存器的说明。

3.1 内部振荡器

在系统复位后,内部振荡器被选为系统时钟并工作在 1.9 MHz。内部振荡器可以被编程为表 1 中所列的四种频率之一。内部振荡器的频率可以在运行中切换,频率改变只需要几个系统周期。如果工作频率对于振荡器改变之后的指令很重要,则可以查询 IFRDY(内部振荡器频率准备好标志:OSCICN.4)。

内部振荡器本身的功耗与所选择的频率无关,但是整个器件的功耗与所选择的频率有关。内部振荡器的精度在 $\pm 20\%$,这个精度值已经考虑到工艺、供电电压及温度的不同。

3.2 外部 CMOS 时钟

系统时钟可以由一个接到 XTAL1 脚的外部 CMOS 电平时钟源提供,例如晶体振荡器模块或来自另一个微控制器的时钟。

3.3 外部晶体

通常在需要一个精确的时基信号时才会选择晶体,例如当 ADC 的绝对采样速率比较关键,或需要用于产生一个标准 UART 波特率时。一种解决方案是用一个低频率的音叉晶体(例如 32.768 kHz 钟表晶体)使器件工作在低功耗方式,当系统需要时再切换到高频率的内部振荡器。

3.4 外部 RC

时间基准还可以通过一个外部串连 RC 电路获得。在这种配置下,电容通过电阻充电,直到 XTAL1 的电压达到 $(1/3) \times AV_{+}$,此时 XTAL1 被驱动到地,使电容放电。这种工作方式在 XTAL1 脚产生一个锯齿波,其周期主要由电容两端电压的上升时间决定。对于一个 100 pF 的电容,放电时间小于 10 ns。该信号经过缓冲后被送到一个二分频器,分频器的输出作为系统时钟。在外部 RC 方式下,时基的精度主要由 R 和 C 元件的误差决定。