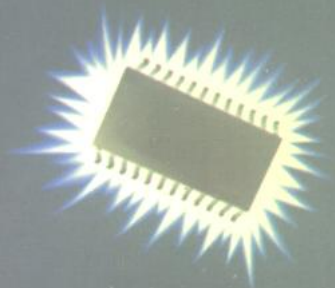
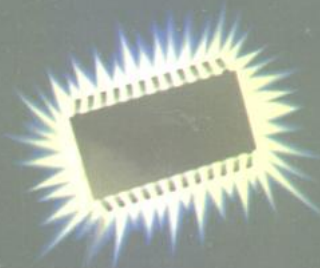
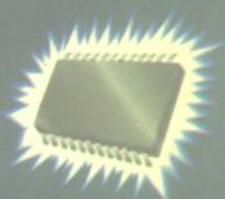
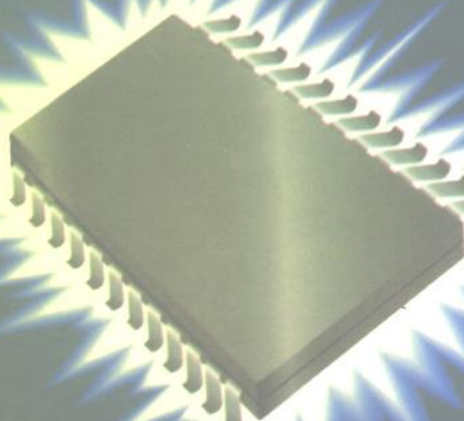
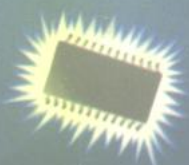
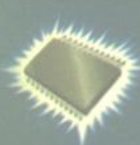




PIC16C5X 系列

单片机应用设计

·李东星·陈小牧·周立功·



电子工业出版社

PIC16C5X 系列单片机应用设计

李东星 陈小牧 周立功

电子工业出版社

(京)新登字 055 号

内 容 简 介

美国 Microchip 公司近年推出的 PIC 系列单片机,具有实用、低价、易学、省电、小巧和高速等优点。

Microchip 公司提供或推荐的 PIC 系列单片机仿真开发系统,都是基于 IBM PC 的,使用方便,功能全。

为此,目前国内单片机应用技术领域,PIC16C5X 系列单片机愈来愈受到用户的欢迎。

本书为广大 PIC16C5X 系列单片机用户,提供较全面的有关硬软件知识和设计应用实例。

PIC16C5X 系列

单片机应用设计

李东星 陈小牧 周立功

责任编辑:王惠民



电子工业出版社出版(北京 173 信箱)

电子工业出版社发行 各地新华书店经销

电子工业出版社广州科技公司排版

华南师范大学印刷厂印刷



开本:787×1092 毫米 1/16 印张:10 字数:220 千字

1996 年 1 月第一版 1996 年 1 月第一次印刷

印数:1~5000 册 定价:20.00 元

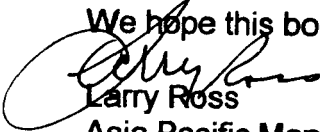
ISBN7-5053-3149-3/TP·1130

PREFACE

Microchip Technology Inc. is a world-wide leader of embedded control solutions to application problems. Our PIC series of 8-bit microcontrollers have already won thousands of design world-wide since first launched a few years ago. We are very pleased to see the first book on "PIC16C5X Design and Application" published and distributed in China. We are sure this book will help engineers in China design PIC microcontrollers in their applications.

Microchip is committed to develop high performance, low cost 8-bit microcontrollers for the industry. The characteristics of our PIC series microcontrollers, RISC architecture, field programmable, high speed, and low power consumption is quickly emerging to be the industry standard 8-bit microcontroller. You can find successful PIC designs in computers, automobiles, instrumentation, communications equipment and home appliances. We are confident the PIC will also be successful in China. There are already thousands of engineers in China designing their products using the PIC for embedded control. We believe there will be more and more successful designs using our PIC in the future.

We hope this book will have a great success in China.



Larry Ross

Asia Pacific Manager

Microchip Technology Inc.

序言

美国 MICROCHIP 公司的 PIC 系列单片机已经风行全球。现在我又非常高兴地看到《PIC16C5X 单片机应用设计》一书在中国首次发行,这必将帮助众多的中国工程师,科研人员以及大专院校的教师,学生开发应用 PIC 单片机。

总部位于美国 ARIZONA 的 MICROCHIP 公司一直致力于发展高性能,低价格的单片机产品。其 PIC 系列单片机正是体现了微控制器工业的新趋势:高性能、低价位、省电、高速以及小巧易用等。在全球范围内,从工业仪器仪表,汽车电子,通信设备到各种家用民生电器,PIC 都有优秀的表现。而更令我们鼓舞的是现在已有数以千计的中国用户正在开发应用 PIC,所以我们可以期望在中国不久将会有更多的 PIC 应用的成功范例。

MICROCHIP 注重其产品的推陈出新,实际上已有众多的 PIC 新产品问世。目前这本书无疑将会把读者引入 PIC 单片机的广阔天地。

让我预祝此书获得成功!

Microchip 公司亚太区经理

拉里·罗斯(签字)

前 言

随着单片机应用领域的不断开拓,单片机应用技术应根据不同的测控对象,不同的环境选择使用不同型号和功能的单片机。目前,单片机市场呈现着缤纷世界,新型单片机层出不穷,Mi—crochip 公司 PIC 系列 CMOS 八位单片机就是一种能够代表单片机发展方向的新型单片机,越来越多的用户选用它。

我们知道,廉价的单片机一般需要订制掩膜,周期长。面对瞬息万变的电子市场,这确实是一个很大的风险,针对这种情况,众多 IC 厂家均竞相推长出 OTP 型(One Time PROM 即一次烧写型)单片机,但其价格约为掩膜型的三倍左右。

美国 Microchip 公司是生产 RISC 单片机及 E²PROM 存储器为主的半导体厂家,近年来推出的 PIC 系列廉价 OTP 型单片机,其价格几乎与掩膜型接近。采用精简指令集(RISC),易学易用;CMOS 工艺制造,极为省电;其 I/O 端口有较大的输入输出负载能力,可以带 25mA 拉出电流。PIC 单片机单周期/单字节指令及哈佛总线和 RISC 结构,为设计带来了极高速度,比一般单片机快 4 倍。另外 PIC 单片机还具备诸如监视定时器 WDT(看门狗)、程序保密位及低价等优点,这些均是用户大量使用的根本原因。

近年来,Microchip 除现有的 PIC16C5X 系列之外,陆续推出 PIC16C71/84/61/64/65,增加中断功能,I/O 端口数目,堆栈嵌套级数,A/D,串行口及 E²PROM 结构。这些新型号在保持 PIC16C5X 系列优良性价比之外,依然是保密性极强而不可破译,其未来前景是必将成为热门单片机。

本书在编写过程中,得到电子工业出版社广州科技公司的大力的支持,美国 Microchip 公司亚太区经理拉里·罗斯先生为本书作序。香港分公司的 Larry Ross 先生和叶新先生对本书的出版均给予大力支持,在此谨表谢意。

因时间仓促,书中不妥之处敬请广大读者不吝赐教。

编 者

1995 年 10 月

有关 PIC 系列 CMOS 八位单片机应用技术疑难之处,
请与本书作者联系。

李东星 陈小牧
福州高奇电子科技有限公司
电 话:(0591)7844159

周立功
广州单片机应用实验室
电 话:(020)5510490

目 录

第一章 PIC16C5X 硬件结构	(1)
§ 1.1 PIC16C5X 主要功能特点	(1)
§ 1.2 PIC16C5X 型号介绍及引脚功能	(2)
§ 1.3 PIC16C5X 内部结构	(3)
§ 1.4 程序存储器及堆栈	(4)
一、程序存储器	(4)
二、堆栈	(5)
§ 1.5 数据存储器	(5)
一、操作寄存器	(5)
二、I/O 寄存器	(10)
三、通用寄存器	(11)
四、特殊功能寄存器	(11)
§ 1.6 预设倍数 Prescaler	(12)
§ 1.7 看门狗 WDT	(13)
§ 1.8 I/O 结构	(14)
§ 1.9 振荡电路	(16)
一、晶体/陶瓷振荡	(16)
二、RC 振荡	(16)
三、外部振荡	(17)
四、时钟/指令时序	(17)
§ 1.10 复位(RESET)	(18)
一、复位的条件和原因	(18)
二、复位时的 PIC 状态	(18)
三、振荡起振计时器(OST)	(19)
四、内部上电复位路	(19)
五、外部复位电路	(19)
§ 1.11 低功耗模式(SLEEP)	(20)
§ 1.12 定义 EPROM	(21)
一、程序保密位(Protect Fuse)	(21)
二、用户识别码(Customer ID Code)	(21)
第二章 PIC16C5X 指令集及程序设计技巧	(22)
§ 2.1 PIC16C5X 指令概述	(22)
§ 2.2 PIC16C5X 指令寻址方式	(23)
§ 2.3 面向字节操作类指令	(24)
§ 2.4 面向位操作指令	(30)
§ 2.5 常数和位操作指令	(31)

§ 2.6 特殊指令助记符	(34)
§ 2.7 PIC16C5X 程序设计技巧	(36)
一、程序的基本格式	(36)
二、程序设计基础	(37)
第三章 PIC16C5X 系统扩展方法	(47)
§ 3.1 I/O 的扩展	(47)
§ 3.2 数据存储器的扩展	(48)
第四章 PIC16C5X 设计实例	(50)
§ 4.1 开发步骤	(50)
§ 4.2 设计实例	(51)
一、数字显示(一)	(51)
二、数字显示(二)	(54)
三、按键显示	(57)
四、电子音价	(60)
五、交流电电源控制	(64)
六、电力线参数测量	(67)
七、阶梯波产生器(D/A 转换)	(68)
八、A/D 转换	(70)
九、异步串行通信	(76)
十、I ² C 串行总线的控制	(79)
十一、液晶 LCD 显示驱动	(86)
十二、PIC16C5X 模拟 EPLD、PLD 电路	(89)
§ 4.3 算术例程	(96)
一、无符号的 BCD 加法	(96)
二、无符号的 BCD 减法	(98)
三、二进制转换成 BCD 数	(99)
四、BCD 转换成二进制数	(101)
第五章 宏汇编	(103)
§ 5.1 汇编程序 PICASM	(103)
§ 5.2 汇编语言程序的语句格式	(103)
一、标号	(104)
二、助记符	(104)
三、操作数	(104)
四、伪指令	(105)
§ 5.3 宏定义	(106)
一、宏定义	(107)
二、宏调用	(107)
三、LOCAL	(108)
§ 5.4 错误信息	(108)

一、在列表文件中显示的错误	(109)
二、在屏幕中显示的信息表	(110)
第六章 仿真器 PICMATE - 5X	(112)
§ 6.1 PICMATE - 5X 功能简介	(112)
§ 6.2 硬件组成	(113)
§ 6.3 运行及准备	(113)
§ 6.4 功能热键	(115)
§ 6.5 调试命令	(118)
§ 6.6 主选菜单及其功能	(120)
§ 6.7 错误信息	(125)
§ 6.8 演示	(127)
第七章 烧写编程器 PICPROG 使用说明	(129)
§ 7.1 安装启动	(129)
§ 7.2 使用说明	(129)
一、程序区(ROM)的烧写	(130)
二、ID 码的烧写	(131)
三、Configuration 烧写	(132)
四、其它命令操作	(132)
第八章 如何选择开发工具	(135)
附 录 PIC16C5X 电特性参数(英文资料)	(136)

第一章 PIC16C5X 硬件结构

PIC16C5X 系列单片机是 8 位单片机,CMOS 工艺制造。本章将详细介绍其内部结构,寄存器组,I/O,时序,振荡形式等等。

§ 1.1 PIC16C5X 主要功能特点

- 采用精简指令集(RISC),仅 33 条指令。指令字长 12 位,全部指令都是单字节指令。除涉及 PC 值改变的指令外(如跳转指令等),其余指令都是单周期指令。

- 工作频率为 DC~20MHz。

- 系统为哈佛结构。数据总线和指令总线各自独立分开,数据总线宽度为 8 位,指令总线宽度为 12 位。

- 内部程序存储器(ROM)从 0.5K~2K 不等。内部寄存器组(RAM)有 25~72 个。

- 7 个特殊功能寄存器。

- 2 级子程序堆栈。

- 工作电源

- 商用级:2.5V~6.25V

- 工业级:2.5V~6.25V

- 军工级:2.5V~6.0V

- 内部自振式看门狗(WDT)

- 低功耗模式(Standby Mode),耗电小于 10 μ A。

- 内部复位电路

- 内带一个 8 位定时器/计数器(RTCC)

- 具备保密位。保密熔丝可在程序烧写时选择将其熔断,则程序不能被读出拷贝。

- 提供四种可选振荡方式

- 低成本的阻容(RC)振荡—RC

- 标准晶体/陶瓷振荡—XT

- 高速晶体/陶瓷振荡—HS

- 低功耗,低频晶体振荡—LP

- 12~20 根双向可独立编程 I/O 口。每根 I/O 口线都可由程序来编程决定其输入/输出方向。

- 低功耗

- <2mA @5V,4MHz

- <15 μ A @3V,32kHz

- <3 μ A 低功耗模式 @3V,0°C~70°C

§ 1.2 PIC16C5X 型号及引脚介绍

PIC16C5X 有四种型号,见下表:

表 1.1 PIC16C5X 型号表

型号	管脚	I/O	RAM	EPROM	振荡	最短指令周期
16C54	18	12	32×8	512×12	DC~20M	200 nS
16C55	28	20	32×8	512×12	DC~20M	200 nS
16C56	18	12	32×8	1024×12	DC~20M	200 nS
16C57	28	20	80×8	2048×12	DC~20M	200 nS
16C58	18	12	80×8	2048×12	DC~20M	200ns

PIC16C5X 管脚图如下:

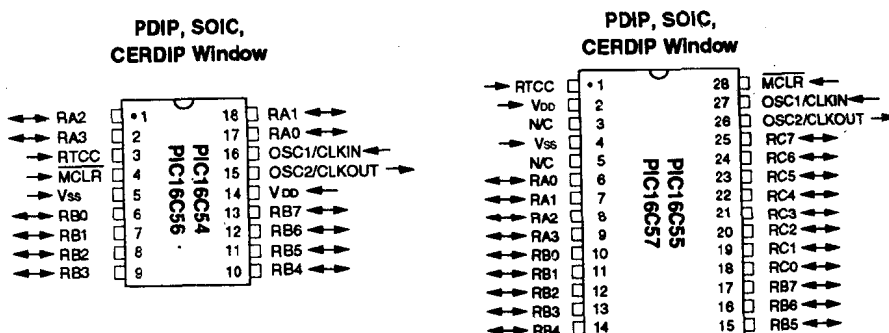


图 1.1 PIC16C5X 管脚图

表 1.2 描述了各引脚的功能。

表 1.2 PIC16C5X 引脚功能描述

引脚名	功能
RA0~RA3	I/O 口 双向可编程
RB0~RB7	I/O 口 双向可编程
RC0~RC7	I/O 口 双向可编程
RTCC	实时定时器/计数器
MCLR	复位脚,低电平有效
OSC1	振荡(输入)
OSC2	振荡(输出)
V _{DD}	电源
V _{SS}	地
N/C	未用

注:RTCC 设置成内部定时器时(由程序设定),这时应将 RTCC 端接 V_{SS}或 V_{DD},以避免

§ 1.3 PIC16C5X 内部结构

The diagram illustrates the internal architecture of the PIC16C55/C57 microcontroller. Key components and their interconnections include:

- Memory and Instruction Path:** EPROM (512 x 12 to 2048 x 12) feeds into the PC (Program Counter, 9-11 bits). The PC outputs a 9-bit address to the INSTRUCTION REGISTER (12 bits) and the INSTRUCTION DECODER (12 bits). The INSTRUCTION DECODER outputs an 8-bit DIRECT ADDRESS to the DATA BUS and a 9-bit address to the STACK 1 and STACK 2 (9-11 bits).
- Control and Timing:** The CONFIGURATION EPROM provides control signals: "DISABLE", "CODE PROTECT", "OSC SELECT" (to OSC1, OSC2, MCLR), and "SLEEP". The WATCHDOG TIMER and WDT/RTCC PRESCALER are connected to the RTCC PIN and provide a WDT TIME OUT signal. The OSCILLATOR/TIMING & CONTROL block generates OSC1, OSC2, MCLR, and CLKOUT signals.
- General Purpose Register File:** The GENERAL PURPOSE REGISTER FILE (16 registers) is accessed via the DATA BUS (8 bits) using an 8-bit address from the W register. It includes the STATUS register and the FSR (File Select Register, 5-7 bits).
- Arithmetic and Logic:** The ALU (Arithmetic Logic Unit) receives 8-bit literals from the W register and performs operations based on the 8-bit DIRECT ADDRESS. It is connected to the DATA BUS.
- Input/Output:** The RTCC (Real Time Clock) is connected to the DATA BUS (8 bits) and the W register. The FSR is connected to the DATA BUS (8 bits) and the GENERAL PURPOSE REGISTER FILE. The TRISA, PORTA, TRISB, PORTB, TRISC, and PORTC registers are connected to the DATA BUS (8 bits) and the W register. The RA0-RA3, RB0-RB7, and RC0-RC7 (PIC16C55/C57 ONLY) ports are connected to the DATA BUS (8 bits).

从图中可以看到, PIC16C5X 有个特点:就是把数据存储器 RAM 当作寄存器来寻址使用, 方便编程。寄存器组按功能分成二部分, 即特殊寄存器组和通用寄存器组。特殊寄存器组包括实时时钟计数器 RTCC, 程序计数器 PC, 状态寄存器 Status, I/O 口以及存储体选择寄存器 FSR。这些寄存器稍后我们还要详细论述。

PIC16C5X 提供二级堆栈(Stack),所以子程序调用只有二层。使用时一定要注意这点,否则程序运行将失去控制。

§ 1.4 程序存储器及堆栈

PIC16C5X 内部有 0.5K~2K 的只读程序存储器,下面论述其结构和堆栈。

一、程序存储器结构

PIC16C5X 程序存储器结构如图 1.3 所示:

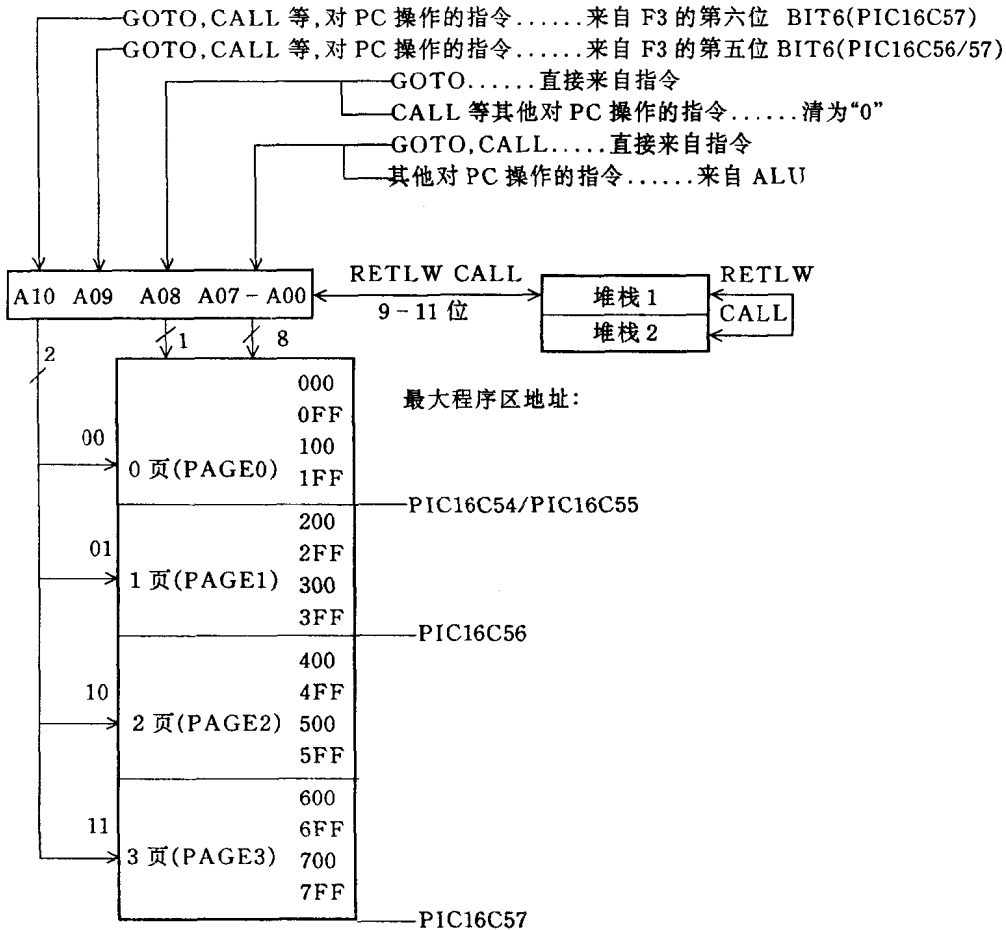


图 1.3 PIC16C5X 程序存储器结构

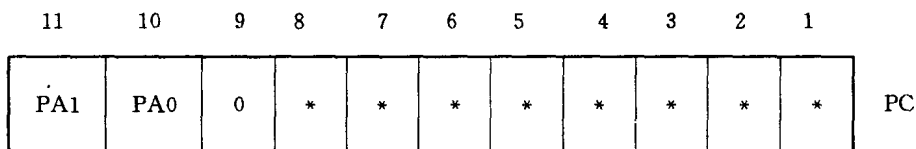
从上图可看出, PIC 程序存储器采用分页结构, 每页长 0.5K。因此对于 PIC16C54 和 PIC16C55 程序存储器容量为 1 页, 而 PIC16C56 和 PIC16C57 及 PIC16C58 的容量分别为 2 页和 4 页。页面地址由状态寄存器 f3 的第 5 位和第 6 位 (PA0, PA1) 确定。程序转移时, 在本页内可直接进行; 在需跨页跳转时 (GOTO, CALL 指令), 则必须根据将要跳转去的页面, 把 f3 中的 PA0, PA1 位置成相应的值。具体请参考 f3 寄存器描述及 § 2.7.2 程序设计基础的程序技巧例子。

二、堆栈

PIC16C5X 设有二层堆栈:堆栈 1 和堆栈 2 供子程序调用。涉及堆栈操作的指令有二条。

1. CALL——在主程序中第一次执行 CALL 指令时,将 PC 值加 1 后推入堆栈 1,堆栈 1 原有的内容则被推入堆栈 2 中。这时子程序中还可再做一次子程序嵌套,即再执行一次 CALL 指令。如果子程序调用多于二层时,堆栈中只存放最近的二个返回地址。

当执行一条 CALL 指令时,将状态寄存器 f3 中的页面寻址位 PA1,PA0 将被置入到 PC 的最高二位(第 11 位和第 10 位),而 PC 的第 9 位总是被置为 0,如图 1.3 所示。所以这时 PC 值将是



这意味着在 PIC16C5X 中,子程序起始地址只能放在每个程序存储页面的上半页,即低地址的那一半(000~0FF,200~2FF,400~4FF,600~6FF)。注意,这里指的是子程序的起始地址,即子程序头。而子程序体是可以延伸到下半页去的。

对于 PIC16C56 和 PIC16C57 及 PIC16C58,由于程序空间分别为 1K 和 2K,可能存在跨页面子程序调用。所以调用子程序前必须先把 f3 中的 PA0,PA1 设置成该子程序所在的页面地址。返回后再将其恢复成当前的页面值。当然如果这时子程序是在同一页面,则可省去这一过程。

2. RETLW——该指令把栈顶(堆栈 1)的值写入 PC,同时还把堆栈 2 的值拷贝到堆栈 1 去。子程序总是返回到调用它时所在的地方,不管它是处在什么页面,也不管 f3 寄存器中的 PA0,PA1 这时是指在什么页面。但是执行 RETLW(子程序返回)指令并不会改变 f3 中 PA0,PA1 的值,所以当从一次跨页面的子程序调用返回时,不要忘了恢复 f3 中的原先的 PA0 和 PA1 值。请参考上面关于 CALL 指令的叙述。

由于堆栈和 PC 的宽度是相同的,所以可以在程序的任何地方执行一条 CALL 指令来调用子程序。但是对于跨页面的调用,要小心处理 f3 中的页面地址位 PA0 和 PA1,请参考第二章指令详解中的 CALL 实例。

§ 1.5 数据存储器

PIC16C5X 把数据存储器 RAM 都当作寄存器来使用以使寻址简单明洁,它们功能上可分为操作寄存器,I/O 寄存器,通用寄存器和特殊功用寄存器。它们的组织结构如图 1.4 所示:这些寄存器用代号 F0~F79 来表示。F0~F4 是操作寄存器,F5~F7 是 I/O 寄存器,其余为通用寄存器。特殊功用寄存器地址对用户不透明。

一、操作寄存器

1. F0 间址寄存器

寻址 F0 实际上意味着间址寻址。实际地址为寄存器选择寄存器 F4 的内容。

例: MOVLW 10

```

MOVWF F4      ;10→F4
MOVLW 55
MOVWF F0      ;55→F10

```

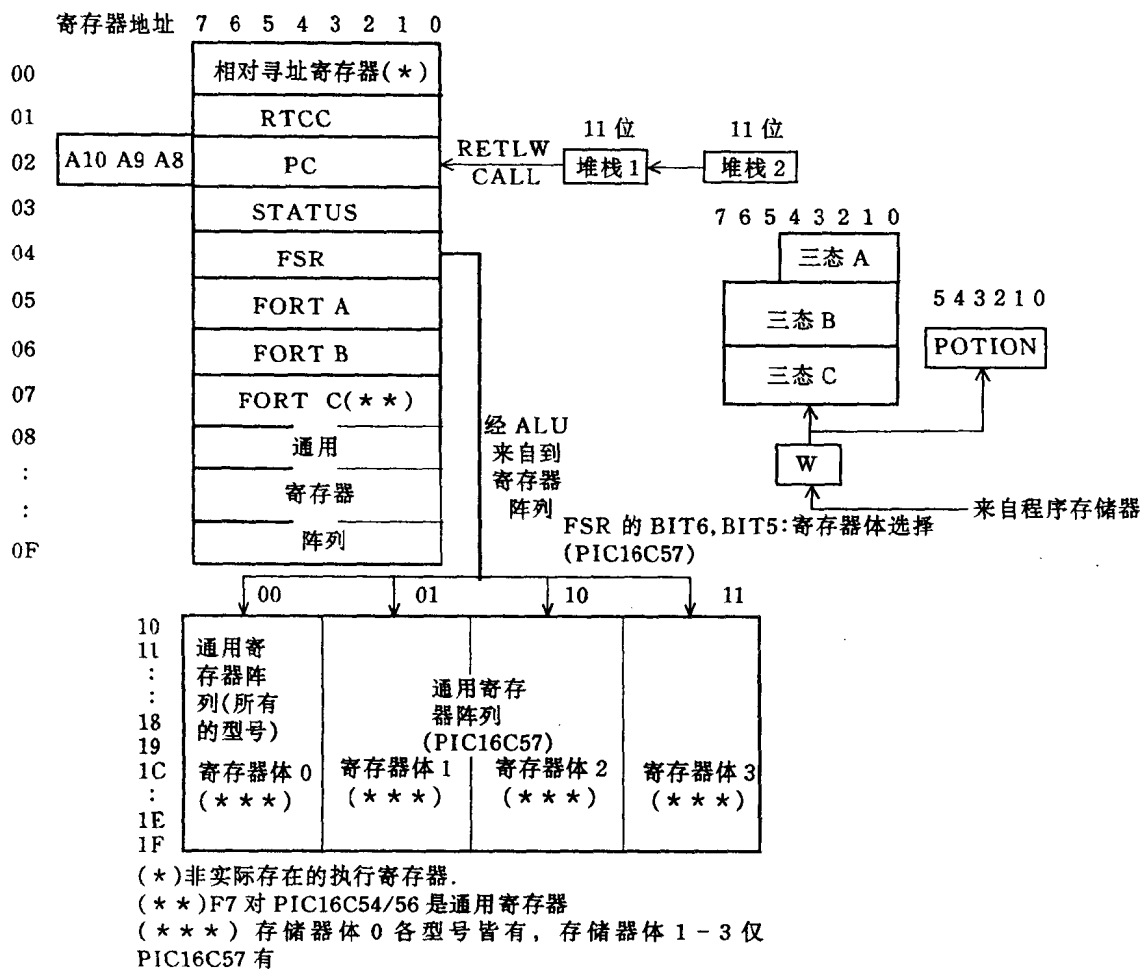


图 1.4 PIC16C5X 寄存器结构

2. F1 实时时钟/计数寄存器(RTCC)

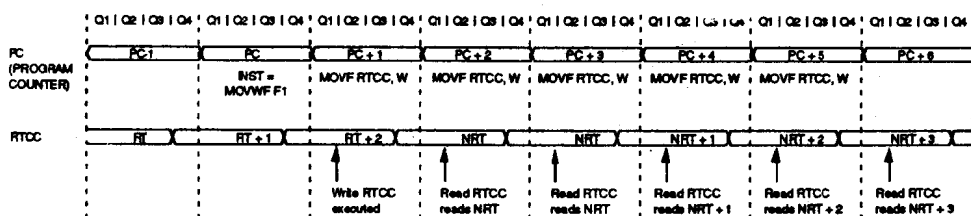
此寄存器是一个 8 位计数器,和其他寄存器一样可由程序进行读写操作。它用于对外加在 RTCC 引脚上的脉冲计数,或对内部时钟计数(起定时器作用)。

RTCC 及其相关电路如图 1.5 所示。

从图中可看出 RTCC 工作状态由 OPTION 寄存器控制(参见 § 1.5.4),其中 OPTION 寄存器的 RTS 位用来选择 RTCC 的计数信号源,当 RTS 为"1"时,信号源为内部时钟,RTS 为"0"时,信号源为来自 RTCC 引脚的外部信号。OPTION 寄存器的 PSA 位控制预分频器(Prescaler)分配对象,当 PSA 位为"1",分配给 RTCC,即外部或内部信号经过预分频器分频后再输出给 RTCC。预分频器的分频比率由 OPTION 内的 PS0~PS2 决定。这时涉及写 F1

- 图 1.5 RTCC 方块图

RTCC 对外部信号的采样周期为 2 个振荡周期。因此当不用预分频器时,外加在 RTCC 引脚上的脉冲宽度不得小于 2 个振荡周期即 $1/2$ 指令周期。同理,当使用预分频器时,预分频器的输出脉冲周期不得小于指令周期,因此预分频器最大输入频率可达 $N, f_{osc}/4, N$ 为预分频器的分频比,但不得大于 50MHz 。



Timing diagram for PC and RTCC registers. The PC register (Program Counter) is shown with values PC-1, PC, PC+1, PC+2, PC+3, PC+4, PC+5, and PC+6. The RTCC register (Real-Time Counter) is shown with values RT, RT+1, NRT, and NRT+1. The diagram illustrates the sequence of operations: PC is incremented by 1, then RTCC is read, then PC is incremented by 1, then RTCC is read, and so on. The RTCC register is shown to be updated from RT to RT+1, and then from NRT to NRT+1. The diagram also shows the RTCC register being read at various points, including when it is updated.

应注意的是尽管 PIC 对外部加于 RTCC 信号端上的信号宽度没有很严格的要求,但是如

果高电平或低电平的维持时间太短,也有可能使 RTCC 检测不到这个信号。一般要求信号宽度要大于是 10nS。

3. F2 程序计数器(PC)

程序计数器 PC 可寻址最多 2K 的程序存储器。表 1.3 列出了 PIC16C5X 各种型号的 PC 长度 和堆栈的长度。

表 1.3 PC 及堆栈的宽度

型 号	PC 宽度(位)	堆栈宽度(位)
PIC16C54/PIC16C55	9	9
PIC1656	10	10
PIC16C57/PIC16C58	11	11

单片机一复位(RESET),F2 的值全置为“1”。除非执行地址跳转指令,否则当执行一条 指令后,F2(PC)值会自动加 1 指向下一条指令。下面这些指令可能改变 PC 的值:

① “GOTO”指令。它可以直接写(改变)PC 的低 9 位。对于 PIC16C56/PIC16C57,状态寄存器 F3 的 PA1、PA0 两位将置入 PC 的最高二位。所示“GOTO”指令可以跳转到程序存储器的任何地方。

② “CALL”指令。它可以直接写 PC 低 8 位、同时将 PC 的第 9 位清零。对于 PIC16C56 / PIC16C57 及 PIC16C58,状态寄存器 F3 的 PA1、PA0 两位将置入 PC 的最高二位(第 10、11 位)。

③ “RETLW”指令。它把栈项(堆栈 1)的值写入 PC。

④ “MOVWF F2”指令。它把 W 寄存器的内容置入 PC。

⑤ “ADDWF F2”指令。它把 PC 值加 1 后再和 W 寄存器的值相加,结果写入 PC。

在以上②④和⑤中,PC 的第 9 位总是被清为零。所以用这三条指令来产生程序跳 转时,要把子程序或分支程序放在每页的上部地址(分别为 000~0FF,200~2FF,400~4FF, 600~6FF)。

4. F3 状态寄存器(STATUS)

如图 1.7 所示 F3 包含了 ALU 的算术状态,RESET 状态,程序存储器页面地址等。F3 中除 PD 和 TO 两位外,其他的位都可由指令来设置或清零。注意,当执行一条欲改变 F3 寄存器的 指令后,F3 中的情况可能出乎你的意料。

例:CLRF F3 ;清 F3 为零

得到的结果是 F3=000UU100(U 为未变)而不是想像中的全零。UU 两位是 PD 和 TO, 它们 维持不变,而 2 位由于清零操作被置成“1”。所以如果你要想改变 F3 的内容,建议使用 BCF,BSF 和 MOVWF 这三条指令,因为它们的执行不影响其他状态位。

例: MOVLW 0 ;0→W

MOVWF F3 ;把 F3 除 PD 和 TO 以外的位全部清零,则你可得到 F3=000UU000。

有关各条指令对状态位的影响请看第二章介绍。

在加法运算(ADDWF)时,C 是进位位。在减法运算(SUBWF)时,C 是借位的反 (Borrow)。

例: CLR F10 ;F10=0
 MOVLW 1 ;1→W
 SUBWF F10 ;F10-W=0-1=FFH→F10
 C=0:运算结果为负

例: MOVLW 1 ;1→W
 MOVWF F10 ;F10=1
 CLRW ;W=0
 SUBWF F10 ;F10-W=1-0=1→F10
 C=1:运算结果为正

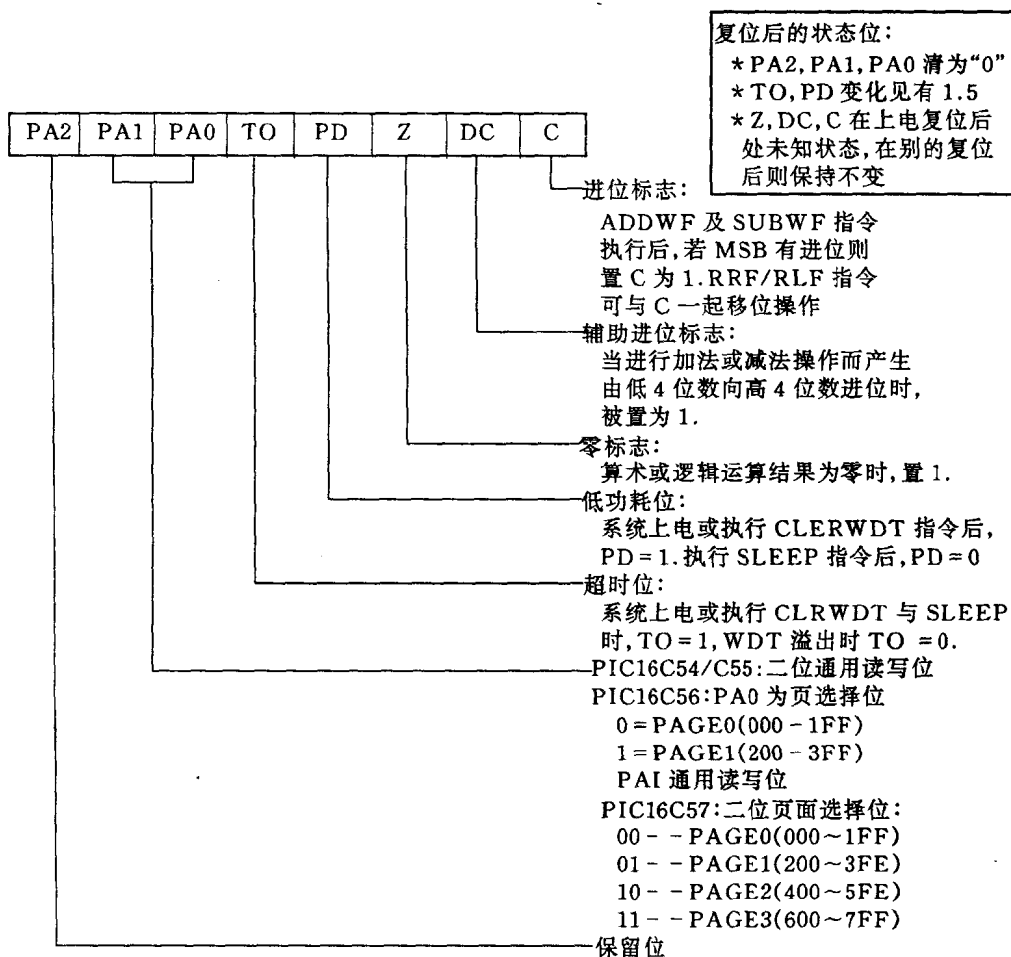


图 1.7 状态寄存器结构

PD 和 TO 两位可用来判断 RESET 的原因。例如判断 RESET 是由芯片上电引起的,或是由看门狗 WDT 计时溢出引起的,或是复位端加低电平引起的,或是由 WDT 唤醒 SLEEP 引起的。

表 1.4 列出了影响 TO,PD 位的事件。表 1.5 列出了在各种 RESET 后的 TO,PD 位状态。