



MICROCHIP

dsPIC30F3014/4013

数据手册

高性能 16 位
数字信号控制器

请注意以下有关 Microchip 器件代码保护功能的要点:

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信: 在正常使用的情况下, Microchip 系列产品是当今市场上同类产品中 safest 的产品之一。
- 目前, 仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知, 所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 (Digital Millennium Copyright Act)》。如果这种行为导致他人在未经授权的情况下, 能访问您的软件或其他受版权保护的成果, 您有权依据该法案提起诉讼, 从而制止这种行为。

提供本文档的中文版本仅为为了便于理解。请勿忽视文档中包含的英文部分, 因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为为您提供便利, 它们可能由更新之信息所替代。确保应用符合技术规范, 是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保, 包括但不限于针对其使用情况、质量、性能、适用性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用, 一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时, 会维护和保障 Microchip 免于承担法律责任, 并加以赔偿。在 Microchip 知识产权保护下, 不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、Microchip 徽标、dsPIC、KEELOQ、KEELOQ 徽标、MPLAB、PIC、PICmicro、PICSTART、PIC³² 徽标、rfPIC 和 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICtail、REAL ICE、rFLAB、Select Mode、Total Endurance、TSHARC、UniWinDriver、WiperLock 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

在此提及的所有其他商标均为各持有公司所有。

© 2011, Microchip Technology Inc. 版权所有。

ISBN: 978-1-60932-940-2

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC[®] MCU 与 dsPIC[®] DSC、KEELOQ[®] 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外, Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

高性能 16 位数字信号控制器

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

改进的高性能 RISC CPU:

- 改进的哈佛架构
- 优化的 C 编译器指令集架构
- 灵活的寻址模式
- 83 条基本指令
- 24 位宽指令，16 位宽数据路径
- 最高 48 KB 的片上闪存程序空间
- 2 KB 的片上数据 RAM
- 1 KB 的非易失性数据 EEPROM
- 16 x 16 位的工作寄存器阵列
- 运行速度最高为 30 MIPS:
 - DC 到 40 MHz 的外部时钟输入
 - 带有 PLL (4x、8x 和 16x) 的 4 MHz 至 10 MHz 振荡器输入
- 最多 33 个中断源:
 - 8 个可由用户选择的优先级
 - 3 个外部中断源
 - 4 个处理器陷阱

DSP 特性:

- 双数据取操作
- 模寻址和位反转寻址模式
- 两个带可选饱和逻辑的 40 位宽累加器
- 17 x 17 位单周期硬件整数 / 小数乘法器
- 所有 DSP 指令均为单周期指令
 - 乘 - 累加 (Multiply-Accumulate, MAC) 运算
- 单周期左移和右移 16 位

外设特性:

- I/O 引脚的高灌 / 拉电流: 25 mA/25 mA
- 最多 5 个 16 位定时器 / 计数器; 可选择将两个 16 位定时器配对成一个 32 位定时器模块
- 最多 4 个 16 位输入捕捉功能
- 最多 4 个 16 位输出比较 / PWM 功能
- 数据转换器接口 (Data Converter Interface, DCI) 支持常见音频编解码器协议, 包括 I²S 和 AC'97
- 3 线 SPI 模块 (支持 4 种帧模式)
- I²CTM 模块支持多主 / 从模式和 7 位 / 10 位寻址
- 最多 2 个带 FIFO 缓冲区的可寻址 UART 模块
- 兼容 CAN 2.0B 标准的 CAN 总线模块

模拟特性:

- 12 位模数转换器 (Analog-to-Digital Converter, ADC) 具有:
 - 200 ksps 的转换速率
 - 最多 13 个输入通道
 - 在休眠和空闲模式下进行转换
- 可编程低电压检测 (Programmable Low-Voltage Detection, PLVD)
- 可编程欠压复位

单片机特性:

- 增强型闪存程序存储器:
 - 工业级温度范围内至少可承受 10,000 次擦写, 容量的典型值为 100 KB
- 数据 EEPROM 存储器:
 - 在工业级温度范围内至少可承受 100,000 次擦写, 容量的典型值为 1 MB
- 可在软件控制下重新自编程
- 上电复位 (Power-on Reset, POR)、上电延时定时器 (Power-up Timer, PWRT) 和振荡器起振定时器 (Oscillator Start-up Timer, OST)
- 灵活的看门狗定时器 (Watchdog Timer, WDT) 带有片上低功耗 RC 振荡器, 能确保可靠运行
- 故障保护时钟监视器操作:
 - 检测时钟故障, 并在检测到故障时将时钟切换到片上低功耗 RC 振荡器
- 可编程代码保护
- 在线串行编程 (In-Circuit Serial ProgrammingTM, ICSPTM)
- 可选功耗管理模式:
 - 休眠、空闲和备用时钟模式

dsPIC30F3014/4013

CMOS 技术:

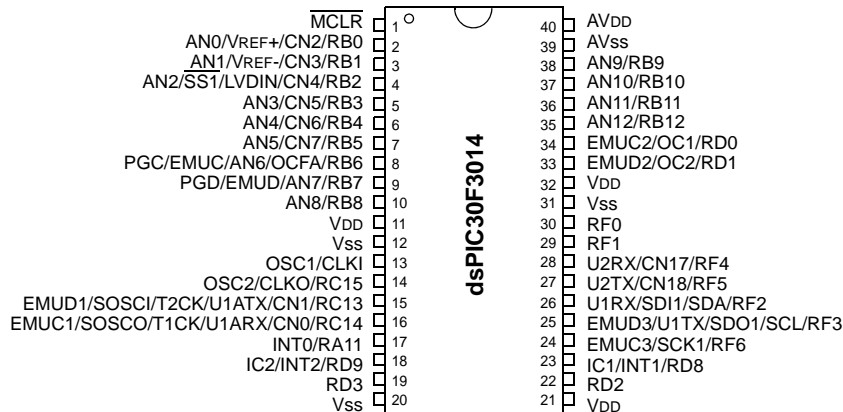
- 低功耗、高速闪存技术
- 宽工作电压范围 (2.5V 至 5.5V)
- 工业级和扩展级温度范围
- 低功耗

dsPIC30F3014/4013 控制器系列

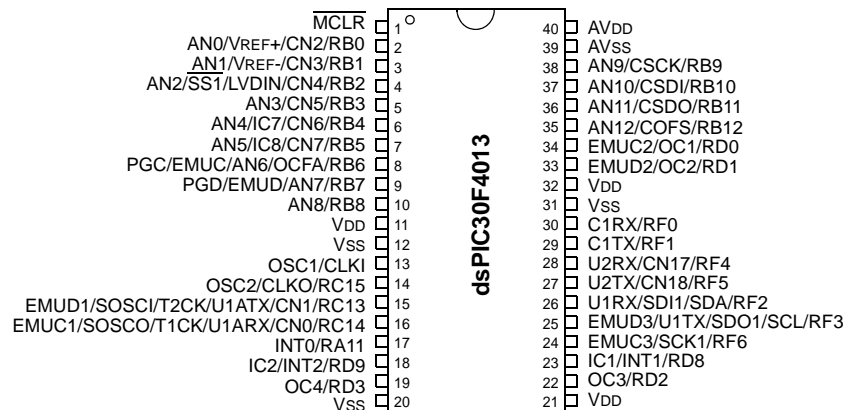
器件	引脚	程序存储器		SRAM 字节	EEPROM 字节	16 位 定时器	输入 捕捉	输出比较/ 标准 PWM	编解码器 接口	12 位 A/D 200 Ksps	UART	SPI	i ² C™	CAN
		字节	指令											
dsPIC30F3014	40/44	24K	8K	2048	1024	3	2	2	—	13 通道	2	1	1	0
dsPIC30F4013	40/44	48K	16K	2048	1024	5	4	4	AC'97, I ² S	13 通道	2	1	1	1

引脚图

40 引脚 PDIP

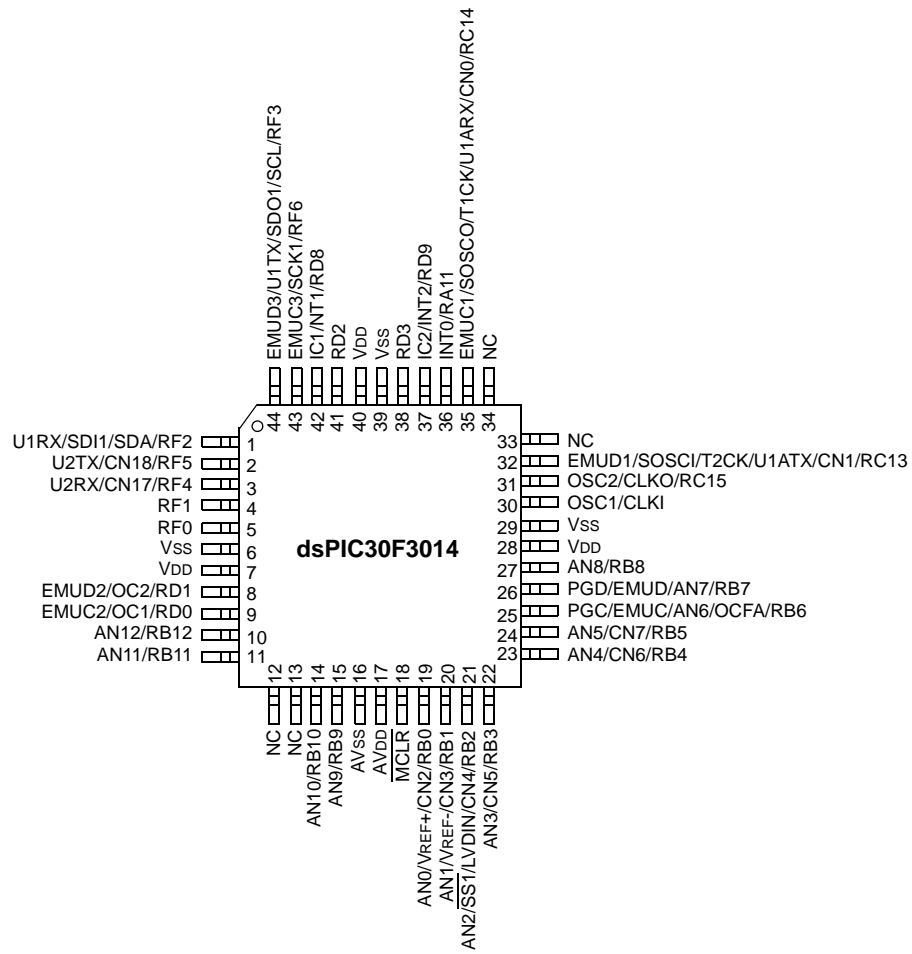


40 引脚 PDIP



引脚图 (续)

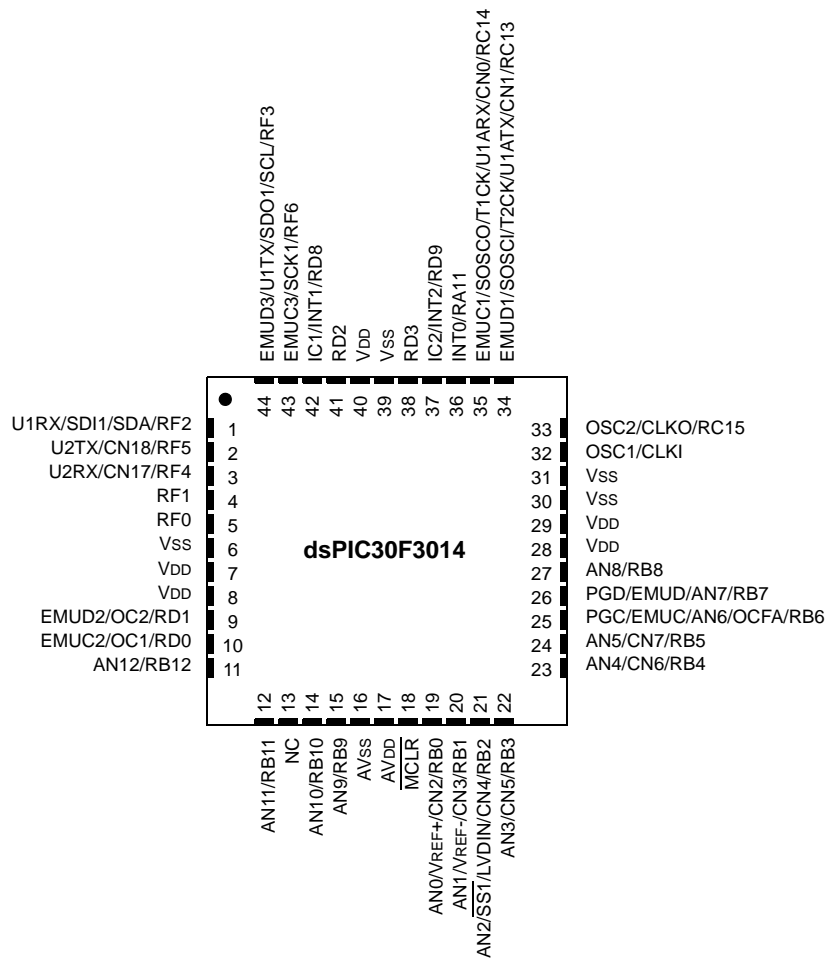
44 引脚 TQFP



dsPIC30F3014/4013

引脚图 (续)

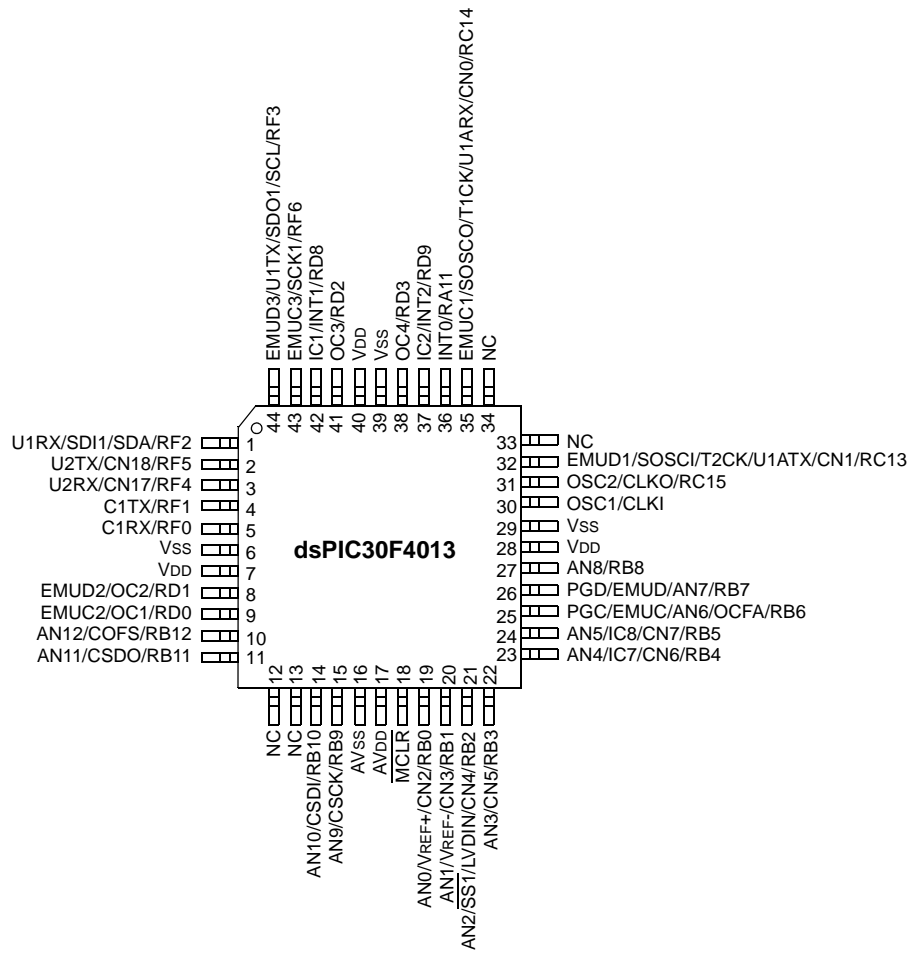
44 引脚 QFN⁽¹⁾



注 1: 器件底部的金属面未连接到任何引脚, 推荐从外部连接到 Vss。

引脚图 (续)

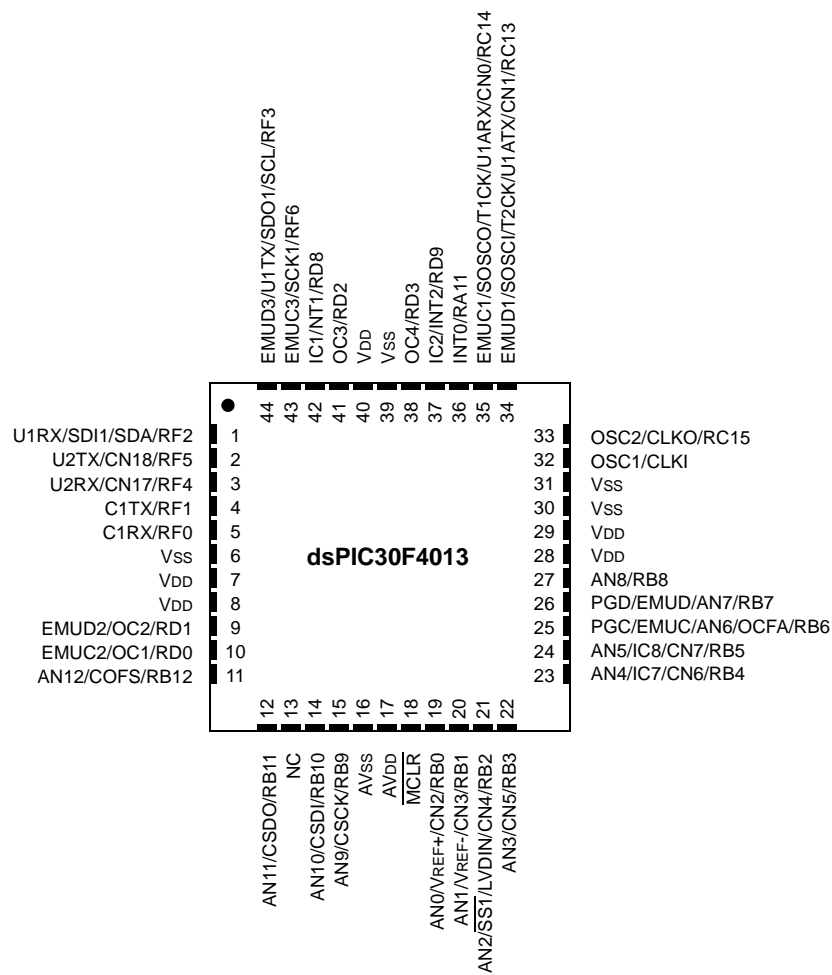
44 引脚 TQFP



dsPIC30F3014/4013

引脚图 (续)

44 引脚 QFN⁽¹⁾



注 1: 器件底部的金属面未连接到任何引脚, 推荐从外部连接到 Vss。

目录

1.0	器件概述	11
2.0	CPU 架构概述	15
3.0	存储器构成	25
4.0	地址发生器单元	37
5.0	闪存程序存储器	43
6.0	数据 EEPROM 存储器	49
7.0	I/O 端口	53
8.0	中断	59
9.0	Timer1 模块	67
10.0	Timer2/3 模块	71
11.0	Timer4/5 模块	77
12.0	输入捕捉模块	81
13.0	输出比较模块	85
14.0	I ² C™ 模块	91
15.0	SPI 模块	99
16.0	通用异步收发器 (UART) 模块	103
17.0	CAN 模块	111
18.0	数据转换器接口 (DCI) 模块	121
19.0	12 位模数转换器 (ADC) 模块	131
20.0	系统集成	141
21.0	指令集综述	159
22.0	开发支持	167
23.0	电气特性	171
24.0	封装信息	211
	索引	219
	Microchip 网站	225
	变更通知客户服务	225
	客户支持	225
	读者反馈表	226
	产品标识体系	227

致 客 户

我们旨在提供最佳文档供客户正确使用 Microchip 产品。为此，我们将不断改进出版物的内容和质量，使之更好地满足您的要求。出版物的质量将随新文档及更新版本的推出而得到提升。

如果您对本出版物有任何问题和建议，请通过电子邮件联系我公司 TRC 经理，电子邮件地址为 CTRC@microchip.com，或将本数据手册后附的《读者反馈表》传真到 86-21-5407 5066。我们期待您的反馈。

最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

<http://www.microchip.com>

查看数据手册中任意一页下边角处的文献编号即可确定其版本。文献编号中数字串后的字母是版本号，例如：DS30000A 是 DS30000 的 A 版本。

勘误表

现有器件可能带有一份勘误表，描述了实际运行与数据手册中记载内容之间存在的细微差异以及建议的变通方法。一旦我们了解到器件 / 文档存在某些差异时，就会发布勘误表。勘误表上将注明其所适用的硅片版本和文件版本。

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

- Microchip 网站：<http://www.microchip.com>
- 当地 Microchip 销售办事处（见最后一页）

在联络销售办事处时，请说明您所使用的器件型号、硅片版本和数据手册版本（包括文献编号）。

客户通知系统

欲及时获知 Microchip 产品的最新信息，请到我公司网站 www.microchip.com 上注册。

dsPIC30F3014/4013

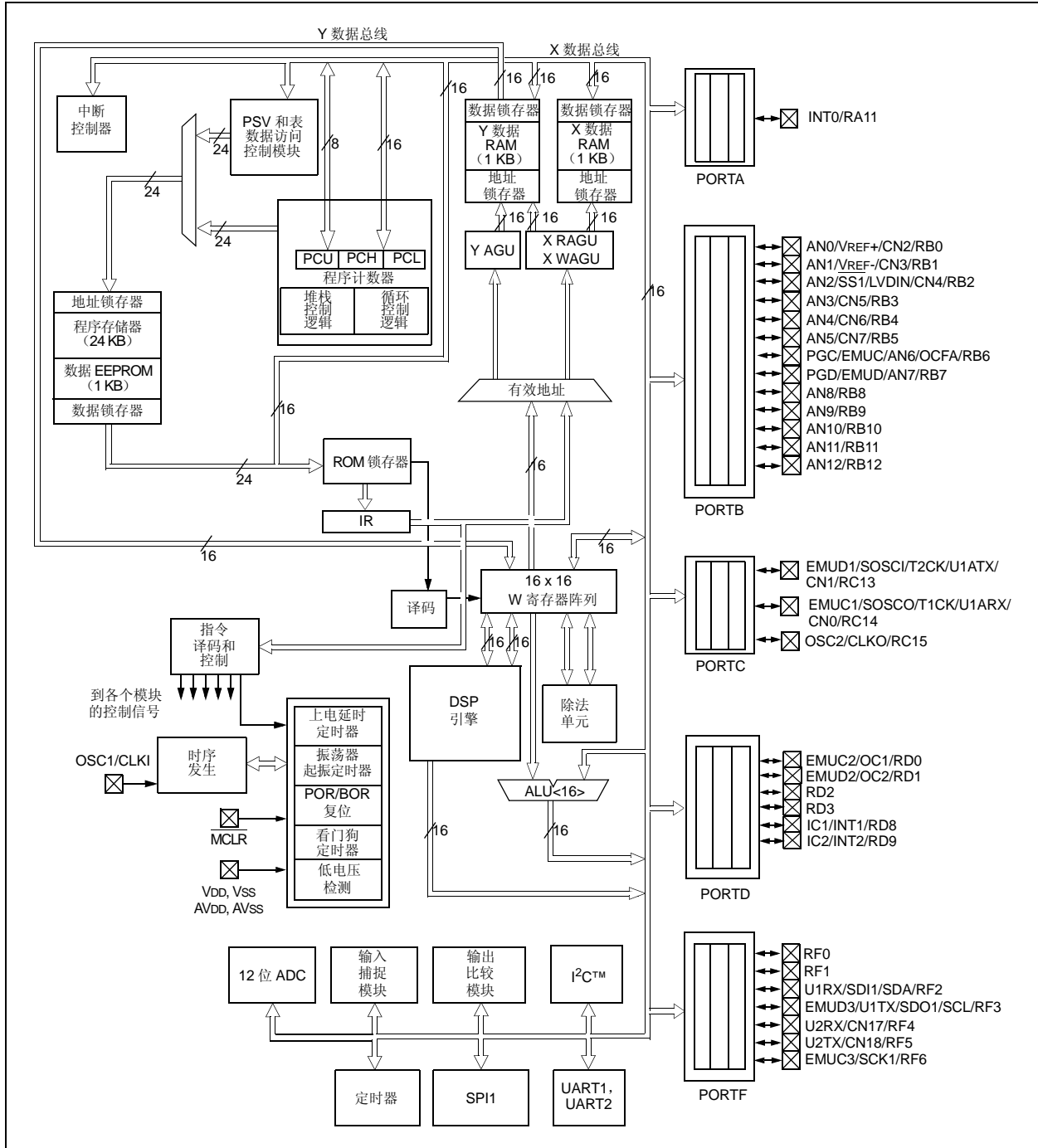
注:

1.0 器件概述

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

本文档包含有关 dsPIC30F3014/4013 数字信号控制器 (DSC) 器件的特定信息。dsPIC30F3014/4013 器件在高速 16 位单片机 (MCU) 架构下，提供丰富的数字信号处理 (DSP) 功能。图 1-1 和图 1-2 分别为 dsPIC30F3014 和 dsPIC30F4013 器件的框图。

图 1-1: dsPIC30F3014 的框图



dsPIC30F3014/4013

图 1-2: dsPIC30F4013 的框图

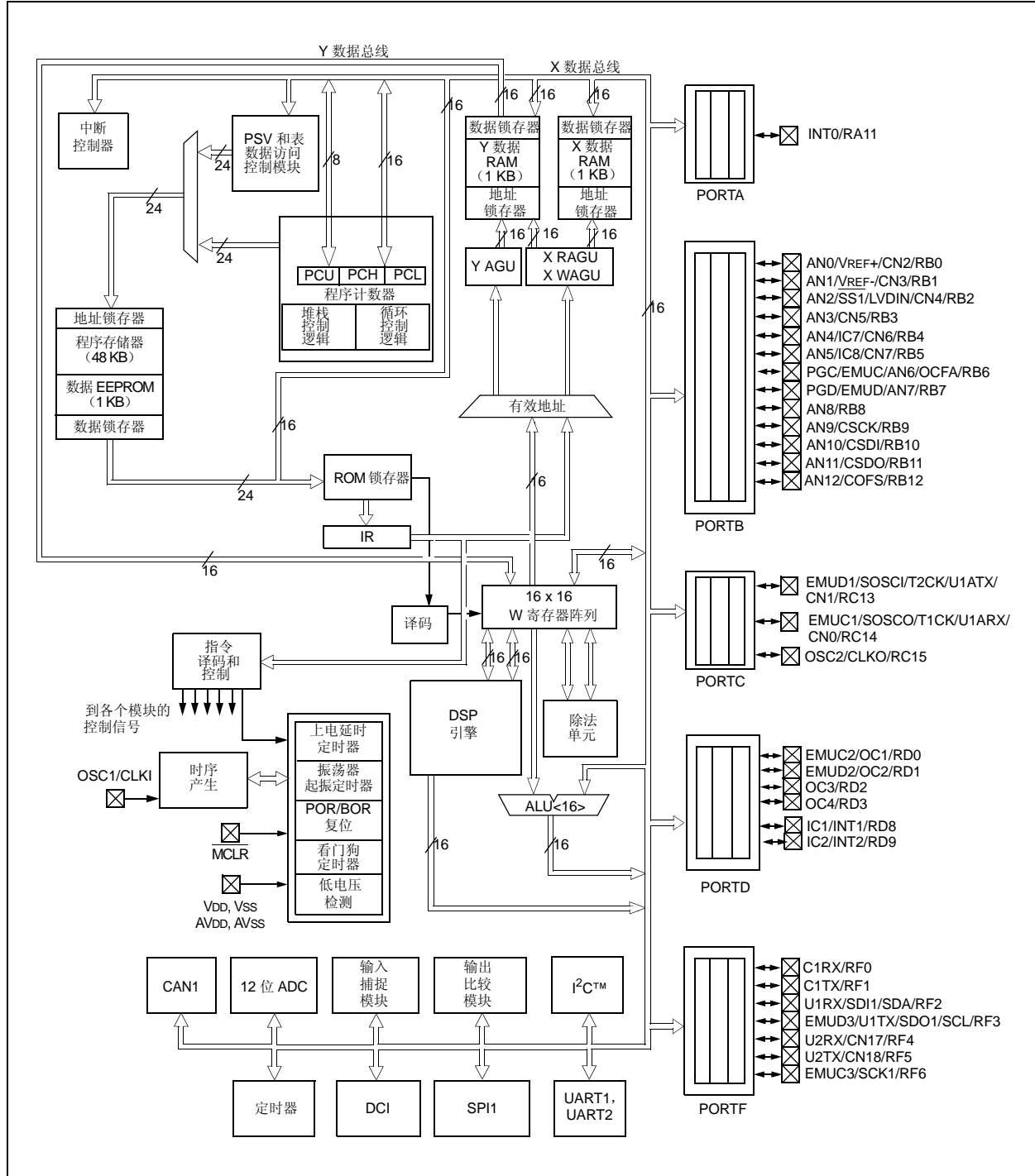


表 1-1 简单介绍了器件的 I/O 引脚配置及其可与端口引脚复用的功能。一个端口引脚可能具有多种功能。当发生复用时，外设模块的功能要求可能会强制改写端口引脚的数据方向配置。

表 1-1: I/O 引脚配置说明

引脚名称	引脚类型	缓冲器类型	说明
AN0-AN12	I	模拟	模拟输入通道。AN6 和 AN7 还分别用作器件的编程数据和时钟输入。
AVDD	P	P	模拟模块的正电源。此引脚必须始终连接。
AVSS	P	P	模拟模块的参考地。此引脚必须始终连接。
CLKI CLKO	I O	ST/CMOS —	外部时钟源输入。总是与 OSC1 引脚功能相关联。 晶振输出。在晶振模式下，与晶振或谐振器相连。在 RC 和 EC 模式下可选作 CLKO 引脚。 总是与 OSC2 引脚功能相关联。
CN0-CN7, CN17-CN18	I	ST	输入电平变化通知输入。 可通过软件编程设定所有输入为内部弱上拉。
COFS CCK CSDI CSDO	I/O I/O I O	ST ST ST —	数据转换器接口帧同步引脚。 数据转换器接口串行时钟输入 / 输出引脚。 数据转换器接口串行数据输入引脚。 数据转换器接口串行数据输出引脚。
C1RX C1TX	I O	ST —	CAN1 总线接收引脚。 CAN1 总线发送引脚。
EMUD EMUC EMUD1 EMUC1 EMUD2 EMUC2 EMUD3 EMUC3	I/O I/O I/O I/O I/O I/O I/O I/O	ST ST ST ST ST ST ST ST	ICD 第一通信通道数据输入 / 输出引脚。 ICD 第一通信通道时钟输入 / 输出引脚。 ICD 第二通信通道数据输入 / 输出引脚。 ICD 第二通信通道时钟输入 / 输出引脚。 ICD 第三通信通道数据输入 / 输出引脚。 ICD 第三通信通道时钟输入 / 输出引脚。 ICD 第四通信通道数据输入 / 输出引脚。 ICD 第四通信通道时钟输入 / 输出引脚。
IC1, IC2, IC7, IC8	I	ST	捕捉输入 1、2、7 和 8。
INT0 INT1 INT2	I I I	ST ST ST	外部中断 0。 外部中断 1。 外部中断 2。
LVDIN	I	模拟	低电压检测参考电压输入引脚。
MCLR	I/P	ST	主复位输入或编程电压输入。此引脚为低电平有效的器件复位引脚。
OCFA OC1-OC4	I O	ST —	比较故障 A 输入（用于比较通道 1、2、3 和 4）。 比较输出 1 到 4。
OSC1 OSC2	I I/O	ST/CMOS —	晶振输入。当配置为 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。 晶振输出。在晶振模式下，与晶振或谐振器相连。在 RC 和 EC 模式下，可选作 CLKO 引脚。
PGD PGC	I/O I	ST ST	在线串行编程数据输入 / 输出引脚。 在线串行编程时钟输入引脚。

图注: CMOS = CMOS 兼容的输入或输出
ST = CMOS 电平的施密特触发器输入
I = 输入

模拟 = 模拟输入
O = 输出
P = 电源

2.0 CPU 架构概述

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

2.1 内核概述

本节简单描述了 dsPIC30F 的 CPU 架构。

其内核具有 24 位指令字。程序计数器 (Program Counter, PC) 为 23 位宽，最低有效位 (Least Significant bit, LSb) 总是清零 (见第 3.1 节“程序地址空间”)，且在正常程序执行时忽略最高有效位 (Most Significant bit, MSb)，但某些特定的指令除外。鉴于此，PC 最多可寻址 4M 指令字的用户程序空间。dsPIC30F 使用指令预取机制来帮助维持吞吐量。使用 DO 和 REPEAT 指令可支持无需循环计数管理开销的程序循环结构，这两条指令在任何时候都可被中断。

工作寄存器阵列由 16 个 16 位的寄存器组成，每个工作寄存器都可以作为数据、地址或地址偏移量寄存器。工作寄存器 (W15) 用作中断和调用操作的软件堆栈指针。

数据空间为 64 KB (32 KW)，被分成两块，分别称为 X 数据存储区和 Y 数据存储区。每个存储区都有各自独立的地址发生器单元 (Address Generation Unit, AGU)。大部分指令只通过 X 存储器 AGU 进行操作，将整个数据空间看作一个统一的数据空间。如乘 - 累加 (MAC) 类双源操作数 DSP 指令通过 X 和 Y 的 AGU 进行操作，将数据地址空间分成两个部分 (见第 3.2 节“数据地址空间”)。X 和 Y 数据空间的边界视具体器件而定，不能由用户修改。每个数据字由 2 个字节组成，大部分指令可以按字或字节访问数据。

有两种方法可以访问存储在程序存储器中的数据：

- 可将数据存储空间的高 32 KB 映射到由 8 位程序空间可视性页 (Program Space Visibility Page, PSVPAG) 寄存器定义的任何 16K 程序字边界内的程序空间的低半部分。程序空间到数据空间的映射功能让任何指令都能像访问数据空间一样访问程序空间。但访问需要占用一个额外的指令周期。且使用此方法仅能访问每个指令字的低 16 位。
- 也可以通过表读 / 表写指令使用工作寄存器对程序空间内大小为 32K 字的页进行线性间接寻址。可使用表读和表写指令访问一个指令字的所有 24 位。

X 和 Y 地址空间都支持无开销的循环缓冲区 (模寻址)。模寻址主要用于减少 DSP 算法的循环开销。

X AGU 还支持对目标有效地址的位反转寻址，从而大幅简化了基 2 FFT 算法对输入或输出数据的重新排序。欲知有关模寻址和位反转寻址的详细信息，请参见第 4.0 节“地址发生器单元”。

内核支持固有 (无操作数) 寻址、相对寻址、立即数寻址、存储器直接寻址、寄存器直接和寄存器间接寻址，以及寄存器偏移量和立即数偏移量寻址模式。每条指令根据其功能要求，与一组预定义的寻址模式相关。

对于大多数指令，在每个指令周期，内核能执行一次数据 (或程序数据) 存储器读操作、一次工作寄存器 (数据) 读操作、一次数据存储器写操作和一次程序 (指令) 存储器读操作。因此，可以支持 3 个操作数的指令，使 $C = A+B$ 操作能在单周期内执行。

包含一个 DSP 引擎可显著增强内核的运算能力和吞吐量。DSP 引擎具备一个高速 17 位 \times 17 位乘法器、一个 40 位 ALU、两个 40 位饱和累加器和一个 40 位双向桶形移位寄存器。在单个周期内，至多可将累加器或工作寄存器中的数据右移 15 位或左移 16 位。DSP 指令可以无缝地与所有其他指令一起操作，其设计可实现最佳的实时性能。MAC 类指令可以同时从存储器中取出两个数据操作数并将两个 W 寄存器相乘。这要求数据空间对于这些指令拆分为两块，但对所有其他指令保持线性。这是通过为 MAC 类指令的每个地址空间指定某些工作寄存器，以透明和灵活的方式实现的。

dsPIC30F3014/4013

内核不支持多级指令流水线，其采用的是单级指令预取机制，该机制在执行指令的前一个周期访问要执行的指令并对其部分译码。从而使可用执行时间最长。除了某些特例外，大部分指令都在一个指令周期内完成。

内核具有用于处理陷阱和中断的向量异常处理结构，提供 62 个独立向量。异常由 8 个陷阱（其中 4 个保留）和 54 个中断组成。每个中断都具有由用户分配的从 1 到 7 的优先级（1 代表最低优先级，7 代表最高优先级）和预先确定的“自然优先级”。陷阱具有从 8 至 15 的固定优先级。

2.2 编程模型

图 2-1 为编程模型，它由 16X16 位工作寄存器（W0 至 W15）、2 个 40 位累加器（AccA 和 AccB）、状态寄存器（SR）、数据表页寄存器（TBLPAG）、程序空间可视性页寄存器（PSVPAG）、DO 和 REPEAT 寄存器（DOSTART、DOEND、DCOUNT 和 RCOUNT）以及程序计数器（PC）组成。工作寄存器可作为数据、地址或偏移量寄存器。所有寄存器均为存储器映射。W0 用作执行文件寄存器寻址的 W 寄存器。

其中的一些寄存器有与之关联的影子寄存器（见图 2-1）。影子寄存器被用作临时保存寄存器，它能够在事件发生时将其主寄存器中的内容送入其中或将自身内容送回主寄存器。影子寄存器均不可直接访问。如下指令适用于寄存器与影子寄存器间的送入和送出。

- PUSH.S 和 POP.S
传送 W0、W1、W2、W3、SR（仅 DC、N、OV、Z 和 C 位）。
- DO 指令
在循环开始时将 DOSTART、DOEND、DCOUNT 寄存器的内容压入其影子寄存器，在循环结束时将这些寄存器的内容弹出。

当对一个工作寄存器执行字节操作时，仅目标寄存器的低字节会受到影响。然而，工作寄存器是存储器映射的，其好处在于可通过对数据存储空间进行字节宽度的访问来对工作寄存器的低字节和高字节进行操作。

2.2.1 软件堆栈指针 / 帧指针

dsPIC® DSC 器件包含软件堆栈。W15 是专用的软件堆栈指针（Stack Pointer，SP）。它可被异常处理、子程序调用和返回自动修改，并且也能够被任何指令引用（方法与引用其他工作寄存器相同）。这样就简化了对堆栈指针的读、写和操作（例如，创建堆栈帧）。

注： 为了防止不对齐的堆栈访问，W15<0> 始终为零。

复位时 W15 被初始化为 0x0800。在初始化期间，用户可以将 SP 重新编程以指向数据空间的任何单元。

W14 是专用的堆栈帧指针，由 LNK 和 ULNK 指令定义。并且也能够被任何指令引用（方法与引用其他工作寄存器相同）。

2.2.2 状态寄存器

dsPIC DSC 内核具有一个 16 位状态寄存器（SR），它的低字节（Least Significant Byte，LSB）称为 SR 低字节（SRL），它的高字节（Most Significant Byte，MSB）称为 SR 高字节（SRH）。有关 SR 的布局请参见图 2-1。

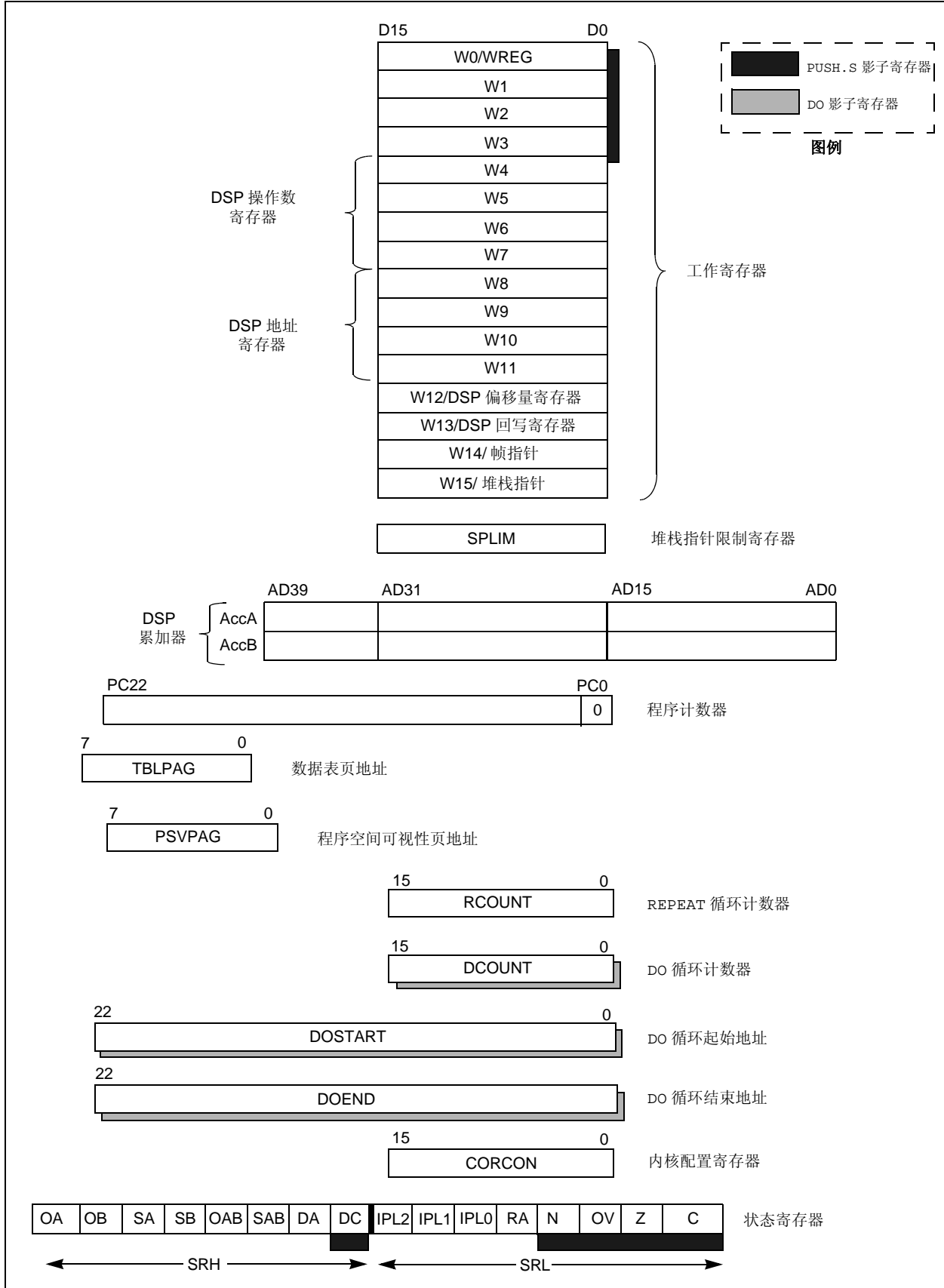
SRL 包含了所有 MCU ALU 运算的状态标志位（包括 Z 位）以及 CPU 中断优先级状态位 IPL<2:0> 和 Repeat 循环有效状态位 RA。在异常处理期间，SRL 与 PC 的 MSB 连在一起形成一个完整的字，然后将该字的值压入堆栈。

状态寄存器的高字节包含 DSP 加法 / 减法状态位、DO 循环有效位（DO Loop Active，DA）以及半进位（Digit Carry，DC）状态位。

2.2.3 程序计数器

程序计数器为 23 位宽；bit 0 总是为零。因此，PC 可寻址最多 4M 指令字。

图 2-1: 编程模型



2.4 DSP 引擎

DSP 引擎由一个高速 17 位 \times 17 位乘法器、桶形移位寄存器、40 位加法器/减法器（带有两个目标累加器以及舍入和饱和逻辑）组成。

DSP 引擎能够执行固有的“累加器 - 累加器”操作，而无需额外数据。这些指令为 ADD、SUB 和 NEG。

dsPIC30F 采用单周期指令流架构，因此 DSP 引擎操作和 MCU 指令流不能同时进行。但某些 MCU ALU 和 DSP 引擎资源可被同一条指令（如 ED 和 EDAC）同时使用。（有关 DSP 指令见表 2-2）。

通过 CPU 内核配置寄存器（CORCON）的不同位可以选择 DSP 引擎的不同功能，如下所示：

1. 小数或整数 DSP 乘法（IF）。
2. 有符号或无符号 DSP 乘法（US）。
3. 传统或收敛舍入（RND）。
4. AccA 的自动饱和和使能/禁止（SATA）。
5. AccB 的自动饱和和使能/禁止（SATB）。
6. 对于写数据存储器的自动饱和和使能/禁止（SATDW）。
7. 累加器饱和模式选择（ACCSAT）。

注： 请参见表 3-3 了解 CORCON 的布局。

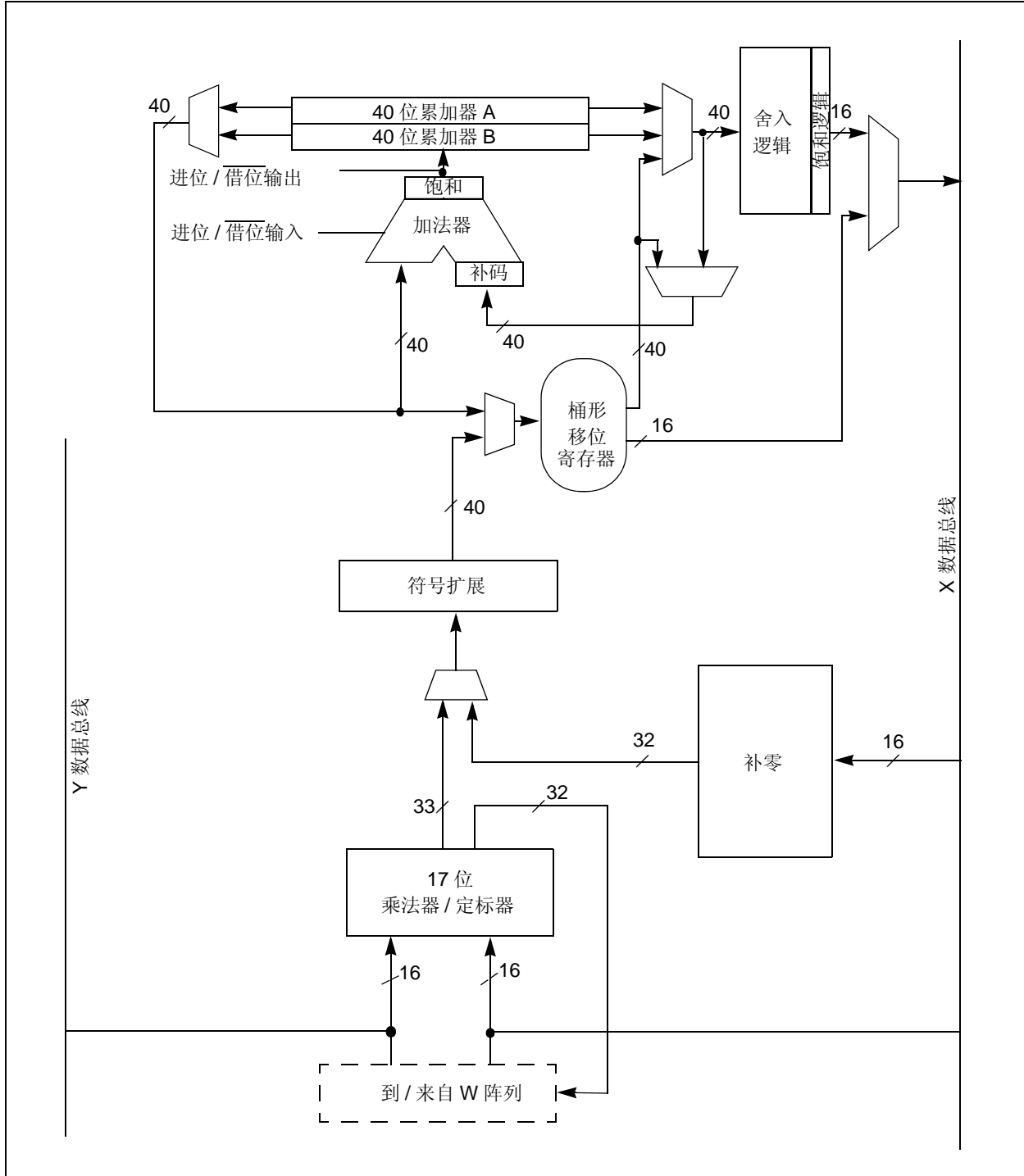
图 2-2 给出了 DSP 引擎的框图。

表 2-2: DSP 指令汇总

指令	代数运算	ACC 回写?
CLR	$A = 0$	是
ED	$A = (x - y)^2$	否
EDAC	$A = A + (x - y)^2$	否
MAC	$A = A + (x * y)$	是
MAC	$A = A + x^2$	否
MOVSAC	A 值不发生改变	是
MPY	$A = x * y$	否
MPY.N	$A = -x * y$	否
MSC	$A = A - x * y$	是

dsPIC30F3014/4013

图 2-2: DSP 引擎框图



2.4.1 乘法器

17 位 \times 17 位乘法器可以进行有符号或无符号运算，并通过使用定标器对输出进行换算，支持 1.31 小数 (Q31) 或 32 位整数结果。无符号操作数对乘法器输入值的第 17 位进行零扩展。有符号操作数对乘法器输入值的第 17 位进行符号扩展。17 位 \times 17 位乘法器 / 定标器的输出一个 33 位的值，并对其进行符号扩展使其长度为 40 位。整型数据本身为一个有符号的二进制补码值，其中 MSb 被定义为符号位。一般来说，N 位二进制补码整数的范围为 -2^{N-1} 到 $2^{N-1} - 1$ 。对于一个 16 位整数，该范围为 -32768 (0x8000) 到 32767 (0x7FFF)，包括 0 在内。对于一个 32 位整数，该范围为 -2,147,483,648 (0x8000 0000) 到 2,147,483,645 (0x7FFF FFFF)。

如果将乘法器配置为进行小数乘法运算，则以二进制补码小数形式表示数据。其中 MSb 被定义为符号位，且暗指小数点就在符号位之后 (QX 格式)。这种带暗指小数点的 N 位二进制补码小数的范围为 -1.0 到 $(1 - 2^{1-N})$ 。对于 16 位小数，Q15 的数据范围为 -1.0 (0x8000) 到 0.999969482 (0x7FFF)，包括 0 在内，且精度为 3.01518×10^{-5} 。在小数模式下，16 \times 16 乘法运算生成 1.31 格式的乘积，其精度为 4.65661×10^{-10} 。

相同的乘法器也用于支持 MCU 乘法指令，包含 16 位有符号、无符号和混合符号整数的乘法。

MUL 指令可以直接使用字节或字大小的操作数。字节操作数将 16 位结果保存到 W 阵列中的指定寄存器，而字操作数将 32 位结果保存到 W 阵列中的指定寄存器。

2.4.2 数据累加器和加法器 / 减法器

数据累加器具有一个带有自动符号扩展逻辑的 40 位加法器 / 减法器。它可以两个累加器 (A 或 B) 之一作为它累加前的源和累加后的目标。对于 ADD 和 LAC 指令，可选择通过桶形移位器在累加之前对将被累加或装入的数据进行调整。

2.4.2.1 加法器 / 减法器、溢出和饱和

加法器 / 减法器为一个 40 位的运算器，一端可选择输入零，另外一端为数据的原码或数据的补码。当运算器用作加法器时，进位 / 借位输入为高电平有效，另一端为数据原码 (未进行求补)。当运算器用作减法器时，进位 / 借位输入为低电平有效，另一端为数据的补码。加法器 / 减法器产生溢出状态位 SA/SB 和 OA/OB，它们被锁存到状态寄存器中并由状态寄存器反映出来：

- 从 bit 39 溢出。这是灾难性的溢出，因为它破坏了累加器的符号位。
- 溢出到警戒位 bit 32 至 bit 39：这是一种可恢复的溢出。这些警戒位不完全相同时，该位置 1。

加法器有一个额外的饱和模块，如果选用该模块将控制累加器的数据饱和。饱和模块使用加法器的结果、上一段所述的溢出状态位以及 SATA/B (CORCON<7:6>) 和 ACCSAT (CORCON<4>) 模式控制位来决定何时以及在何值达到饱和。

在状态寄存器中有 6 个支持饱和及溢出的位，它们是：

1. OA:
AccA 溢出至警戒位
2. OB:
AccB 溢出至警戒位
3. SA:
AccA 已饱和 (bit 31 溢出并饱和)
或
AccA 溢出至警戒位并饱和 (bit 39 溢出并饱和)
4. SB:
AccB 已饱和 (bit 31 溢出并饱和)
或
AccB 溢出至警戒位并饱和 (bit 39 溢出并饱和)
5. OAB:
OA 和 OB 的逻辑或 (OR)
6. SAB:
SA 和 SB 的逻辑或 (OR)

每次数据通过加法器 / 减法器，就会修改 OA 位和 OB 位。置 1 时，它们表明最近的运算已经溢出到累加器警戒位 (bit 32 到 bit 39)。如果 OA 和 OB 位置 1 而且 INTCON1 寄存器中相应的溢出陷阱允许位 (OVATE 和 OVBTE) 也置 1 的话，还可以选择用 OA 和 OB 位产生算术警告陷阱 (见第 8.0 节“中断”)。这使用户能立即采取措施，例如校正系统增益。

每次数据通过加法器/减法器，就会修改 SA 和 SB 位，但它们只能由用户清零。置 1 时，它们表明累加器已经溢出其最大范围（32 位饱和是 bit 31，而 40 位饱和是 bit 39），将发生饱和（如果饱和使能的话）。如果没有使能饱和，SA 和 SB 置 1 默认为 bit 39 溢出，即表明发生灾难性溢出。如果 INTCON1 寄存器中的 COVTE 位置 1，当饱和和被禁止时，将产生算术警告陷阱。

可选择将状态寄存器中的溢出和饱和状态位分别视作 OA 和 OB 的逻辑或（OAB 位）以及 SA 和 SB 的逻辑或（SAB 位）。这样，只需要检查状态寄存器中的一个位，编程人员就能够判断出是否有累加器溢出；检查状态寄存器中的另一个位，就可以判断出是否有累加器发生饱和。对于通常需要使用两个累加器的复杂数值运算而言，这很有用。

器件支持三种饱和及溢出模式：

- 1. Bit 39 溢出和饱和：**
当发生 bit 39 溢出和饱和时，饱和逻辑将最大的正数 9.31（0x7FFFFFFF）或最小的负数 -9.31 值（0x80000000）装入目标累加器。SA 或 SB 位被置 1 且保持置 1 状态，直至被用户清零。这称为“超饱和”，为错误数据或不可预期的算法问题（例如，增益计算）提供了保护机制。
- 2. Bit 31 溢出和饱和：**
当发生 bit 31 溢出和饱和时，饱和逻辑将最大的正数 1.31（0x007FFFFFFF）或最小的负数 -1.31（0x0080000000）装入目标累加器。SA 或 SB 位被置 1 且保持置 1 状态，直至被用户清零。当此饱和模式生效时，不使用警戒位，因此 OA、OB 或 OAB 位永远不会置 1。
- 3. Bit 39 灾难性溢出：**加法器的 bit 39 溢出会将 SA 或 SB 位置 1 并保持该状态直至被用户清零。不执行饱和操作，允许累加器溢出（破坏累加器的符号位）。如果 INTCON1 寄存器中的 COVTE 位置 1，则灾难性溢出可能引发陷阱异常。

2.4.2.2 累加器“回写”

MAC 类指令（MPY、MPY.N、ED 和 EDAC 除外）可以选择将累加器高位字（bit 16 至 bit 31）的舍入形式写入数据存储空间，前提是该累加器不是指令的目标。通过 X 总线将数据写入组合的 X 和 Y 地址空间。支持下列的寻址模式：

- 1. W13，寄存器直接寻址：**非目标累加器的舍入内容以 1.15 小数的形式写入 W13。
- 2. [W13]+2，带后递增的寄存器间接寻址：**非目标累加器的舍入内容以 1.15 小数的形式写入 W13 指向的地址。W13 的值增加 2（对于字写操作）。

2.4.2.3 舍入逻辑

舍入逻辑为一个组合的模块，在累加器写（存储）过程中可以执行传统的（有偏）或收敛的（无偏）数据舍入功能。由 CORCON 寄存器的 RND 位的状态决定舍入模式。它会产生一个 16 位的 1.15 数据值，该值被送入到数据空间写饱和和逻辑。如果指令没有明确指定舍入，那么将会存储一个截取的 1.15 数据值而直接丢弃低位字（least significant word, lsw）

传统舍入提取累加器 bit 15 的值，对其进行零扩展并将扩展值加到 ACCxH 字（累加器的 bit 16 至 bit 31）。如果 ACCxL 字（累加器的 bit 0 至 bit 15）的值在 0x8000 到 0xFFFF（含 0x8000）之间，ACCxH 的值递增 1。如果 ACCxL 字的值在 0x0000 到 0x7FFF 之间，ACCxH 的值不变。此算法的结果是经过一系列随机舍入操作之后，获得的值比真实值略微大一些。

收敛（或无偏）舍入操作与传统舍入操作的工作方式相同，但 ACCxL 等于 0x8000 时例外。在此种情况下，检查 ACCxH 的最低位（累加器的 bit 16）。如果该位为 1，ACCxH 的值递增 1。如果该位为 0，ACCxH 的值不变。假设 bit 16 本身就是一个有效随机数，那么此机制将消除任何可能累加的舍入偏差。

SAC 和 SAC.R 指令通过 X 总线将目标累加器内容的截取值（SAC）或舍入值（SAC.R）存储到数据存储（有关数据饱和的信息，见第 2.4.2.4 节“数据空间写饱和”）。请注意，对于 MAC 类指令，累加器回写操作以相同的方式工作，通过 X 总线访问组合的 MCU（X 和 Y）数据空间。对于此类指令，总是对数据进行舍入。

2.4.2.4 数据空间写饱和

除加法器 / 减法器会饱和外，写数据空间也会饱和但不影响源累加器的内容。数据空间写饱和和逻辑模块接收一个来自舍入逻辑模块的 16 位 1.15 小数值作为其输入，还接收来自源（累加器）和 16 位舍入加法器的溢出状态。将这些值组合在一起并用于选择适当的 1.15 小数值作为输出，写入数据存储空间。

如果 CORCON 寄存器的 SATDW 位置 1，则检测数据（舍入和截取后的值）是否溢出并作相应调整。针对大于 0x007FFF 的输入数据，写入存储器的数据被强制为最大的 1.15 值形式的正数 0x7FFF。针对小于 0xFF8000 的输入数据，写入存储器的数据被强制为最小的 1.15 值形式的负数 0x8000。源（bit 39）的最高位（MSb）被用于确定被测试操作数的符号。

如果 CORCON 寄存器的 SATDW 位未被置 1，则在所有情况下，不对输入数据进行任何修改。

2.4.3 桶形移位寄存器

桶形移位寄存器在单个周期内可将数据逻辑或算术右移或左移最多 16 位。源操作数可为任何两个 DSP 累加器之一或 X 总线（以支持对寄存器或存储器中的数据进行多位移位）。

移位寄存器需要一个有符号二进制值来确定移位操作的方向和幅度（位数）。正值将操作数右移。负值将操作数左移。0 值不改变操作数。

桶形移位寄存器的宽度为 40 位，从而可获得 DSP 移位操作的 40 位结果，或 MCU 移位操作的 16 位结果。对于来自 X 总线的的数据，右移时将其装入桶形移位器 bit 16 位到 bit 31，左移时将其装入桶形移位寄存器的 bit 0 至 bit 15。

dsPIC30F3014/4013

注:

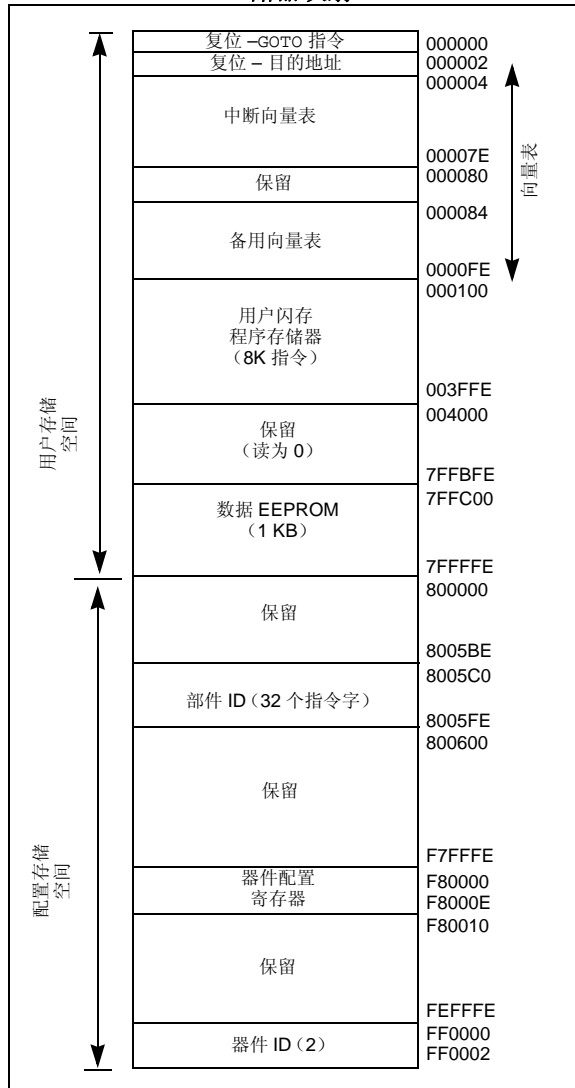
3.0 存储器构成

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

3.1 程序地址空间

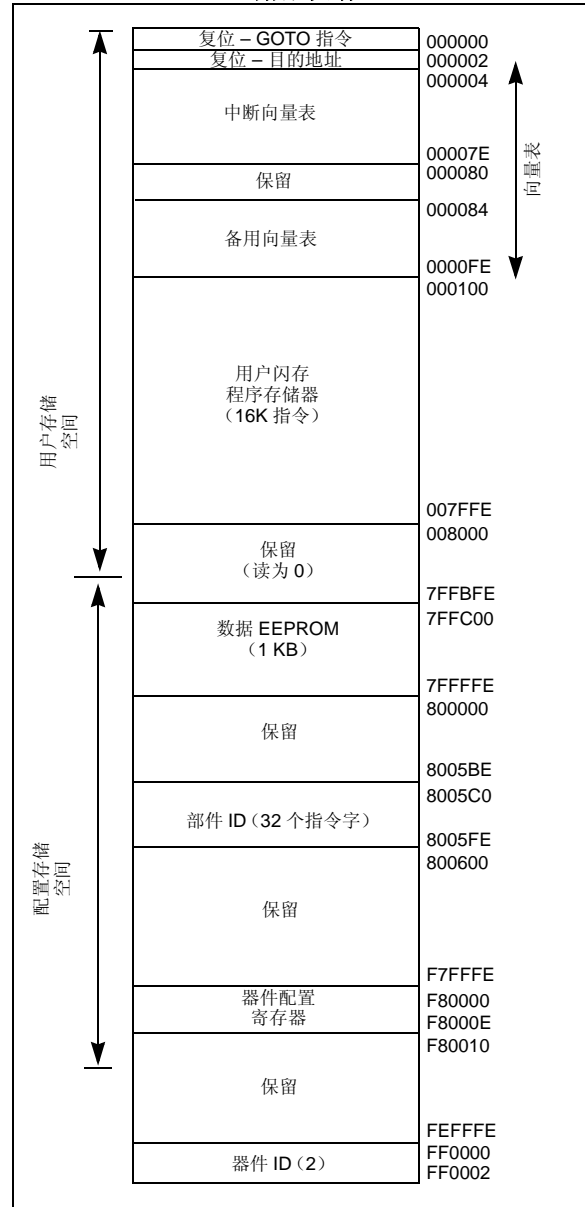
程序地址空间为 4M 指令字。当程序空间按表 3-1 中的定义映射到数据空间时，可由来自 23 位的 PC、表指令有效地址 (Effective Address, EA) 或数据空间 EA 的 24 位值寻址。注意在访问连续的程序字时，程序空间地址应该递增 2 以与数据空间寻址相一致。

图 3-1: dsPIC30F3014 程序空间存储器映射



所有对用户程序空间的访问都被限制在低 4M 指令字地址范围 (0x000000 至 0x7FFFFE) 内，TBLRD/TBLWT 指令除外，它们使用 TBLPAG<7> 来确定要访问的是用户空间还是配置空间。在表 3-1 中，bit 23 置 1 允许访问器件 ID、用户 ID 和配置位，否则 bit 23 始终清零。

图 3-2: dsPIC30F4013 程序空间存储器映射

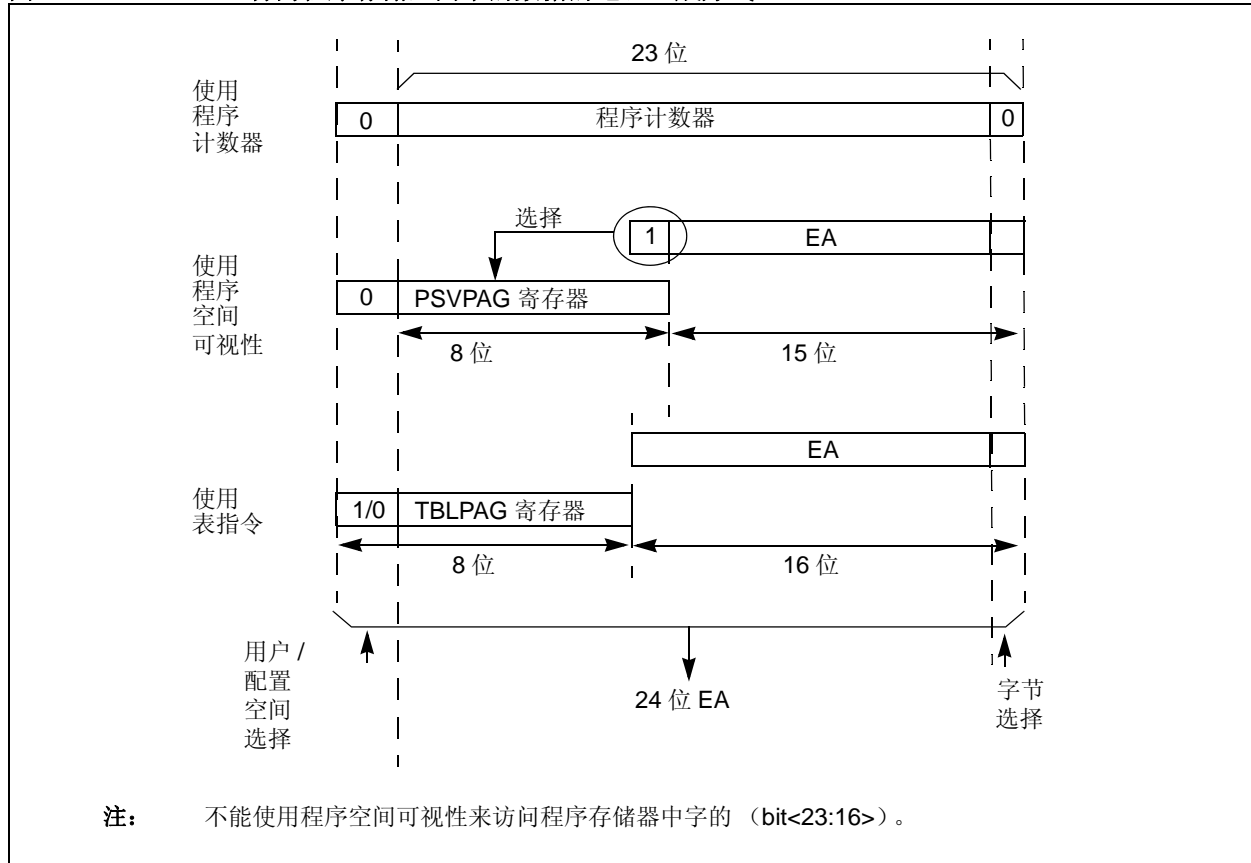


dsPIC30F3014/4013

表 3-1: 程序空间地址构成

访问类型	访问空间	程序空间地址				
		<23>	<22:16>	<15>	<14:1>	<0>
指令访问	用户	0	PC<22:1>			0
TBLRD/TBLWT	用户 (TBLPAG<7> = 0)	TBLPAG<7:0>		数据 EA<15:0>		
TBLRD/TBLWT	配置 (TBLPAG<7> = 1)	TBLPAG<7:0>		数据 EA<15:0>		
程序空间可视性	用户	0	PSVPAG<7:0>		数据 EA<14:0>	

图 3-3: 访问程序存储空间中的数据的数据的地址生成方式



3.1.1 使用表指令访问程序存储器中的数据

由于此类器件的架构对 24 位宽的程序存储器取指，因而指令始终是对齐的。由于采用的是改进的哈佛架构，因此也可在程序空间中存储数据。

可使用以下两种方法来访问程序空间：通过特殊的表指令或通过将 16K 字的程序空间页重映射到数据空间的高半部分（见第 3.1.2 节“使用程序空间可视性访问程序存储器中的数据”）。TBLRDL 和 TBLWTL 指令提供不通过数据空间就能直接读或写程序空间内任一地址最低字的方法。TBLRDH 和 TBLWTH 指令是将程序空间字的高 8 位作为数据访问的惟一方法。

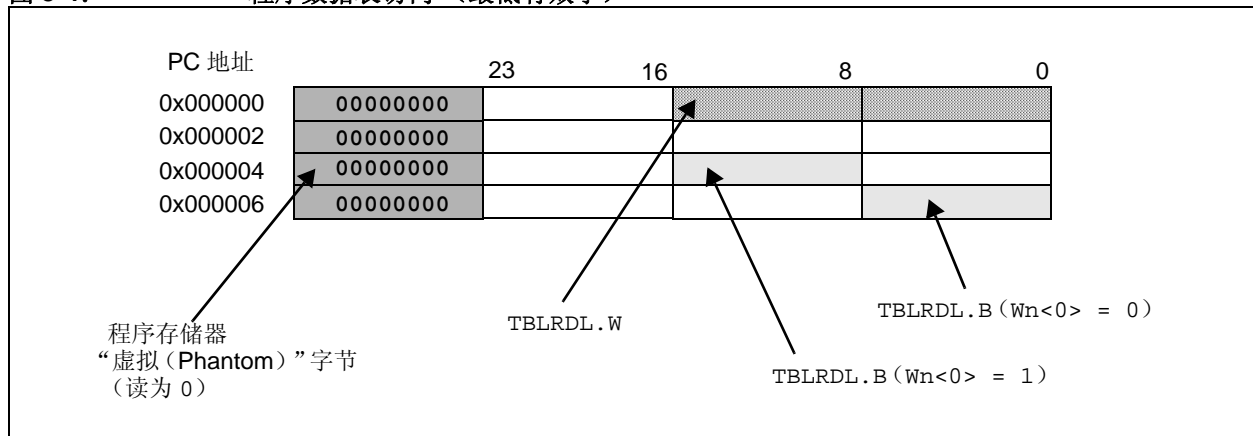
对于每个连续的 24 位程序字，PC 需要递增 2。表指令允许将程序存储地址单元直接映射到数据空间地址单元中。这样就可以把程序存储器看作是两个 16 位字宽的地址空间，两部分空间边界对齐并且大小相同。TBLRDL 和 TBLWTL 访问包含最低有效数据字的空间，而 TBLRDH 和 TBLWTH 访问包含最高字节的空间。

图 3-3 给出了表操作和数据空间访问 (PSV = 1) 时 EA 的构成方式。在该图中，P<23:0> 指程序空间字，而 D<15:0> 指数据空间字。

提供了一组表指令，可使用它们与程序空间之间进行传送字节或字大小的数据（见图 3-4 和图 3-5）。

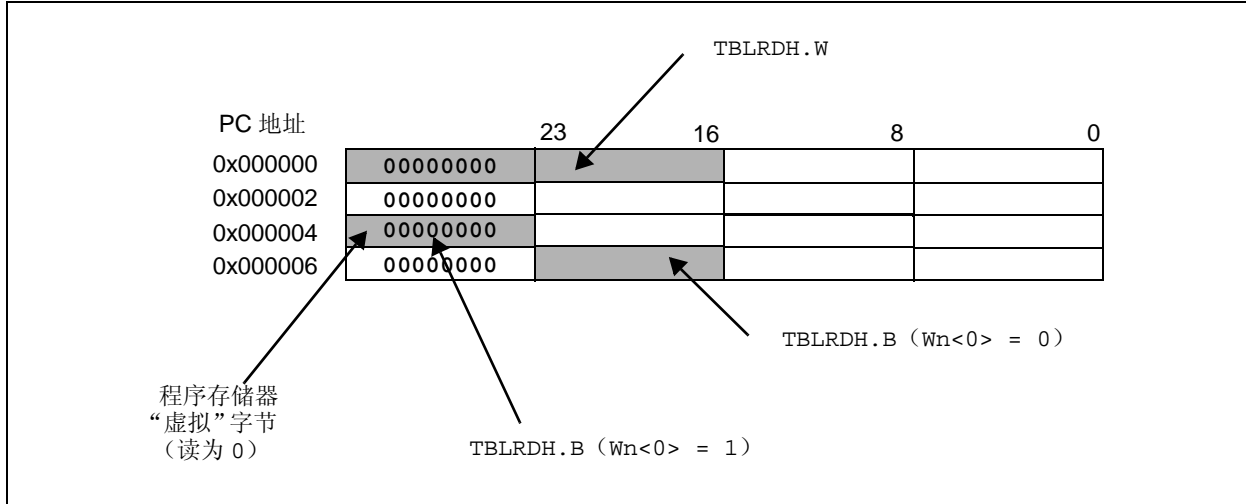
1. TBLRDL: 对低字部分执行表读操作
字: 读程序地址的低字部分;
P<15:0> 映射到 D<15:0>。
字节: 读程序地址的某个低字节;
当字节选择位 = 0 时, P<7:0> 映射到目标字节;
当字节选择位 = 1 时, P<15:8> 映射到目标字节。
2. TBLWTL: 对低字部分执行表写操作 (有关闪存编程的详细信息请参见第 5.0 节“闪存程序存储器”)
3. TBLRDH: 对高字部分执行表读操作
字: 读程序地址的最高字 (msw); P<23:16> 映射到 D<7:0>; D<15:8> 将始终为零 (= 0)。
字节: 读程序地址的某个高字节;
当字节选择位 = 0 时, P<23:16> 映射到目标字节;
当字节选择位 = 1 时, 目标字节将始终为零 (= 0)。
4. TBLWTH: 对高字部分执行表写操作 (有关闪存编程的详细信息请参见第 5.0 节“闪存程序存储器”)

图 3-4: 程序数据表访问 (最低有效字)



dsPIC30F3014/4013

图 3-5: 程序数据表访问 (MSB)



3.1.2 使用程序空间可视性访问程序存储器中的数据

可选择将数据空间的高 32 KB 映射到任何 16K 字的程序空间页中。这样无需使用特殊指令 (即 TBLRDH/H 和 TBLWTL/H 指令), 即可提供对存储在 X 数据空间的常量进行透明访问。

如果数据空间 EA 的最高位置 1 并且通过将内核控制寄存器 (CORCON) 中的 PSV 位置 1 使能程序空间可视性, 就会发生通过数据空间进行程序空间访问。在第 2.4 节 “DSP 引擎” 中对 CORCON 的功能进行了讨论。

由于访问该区域内的数据需要取两次程序存储字, 因而会使指令的执行时间多出一个周期。

注意可寻址数据空间的高半部分始终位于 X 数据空间内。因此, 当 DSP 操作使用程序空间映射来访问该存储区域时, Y 数据空间通常应包含 DSP 操作的状态 (变量) 数据, 而 X 数据空间通常应包含系数 (常量) 数据。

虽然每个数据空间中地址大于等于 0x8000h 的部分直接映射到相应的程序存储地址单元中 (见图 3-6), 但只有 24 位程序字的低 16 位用于存储数据。应将高 8 位编程为强制执行一条无效指令以维持器件的可靠性。有关指令编码的详细信息请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

注意: 每访问一个程序字, PC 就会递增 2, 数据空间的低 15 位直接映射到相应程序空间地址的低 15 位。其余位由程序空间可视性页寄存器 PSVPAG<7:0> 提供, 如图 3-6 所示。

注: 在读/写操作时暂时禁止 PSV 访问。

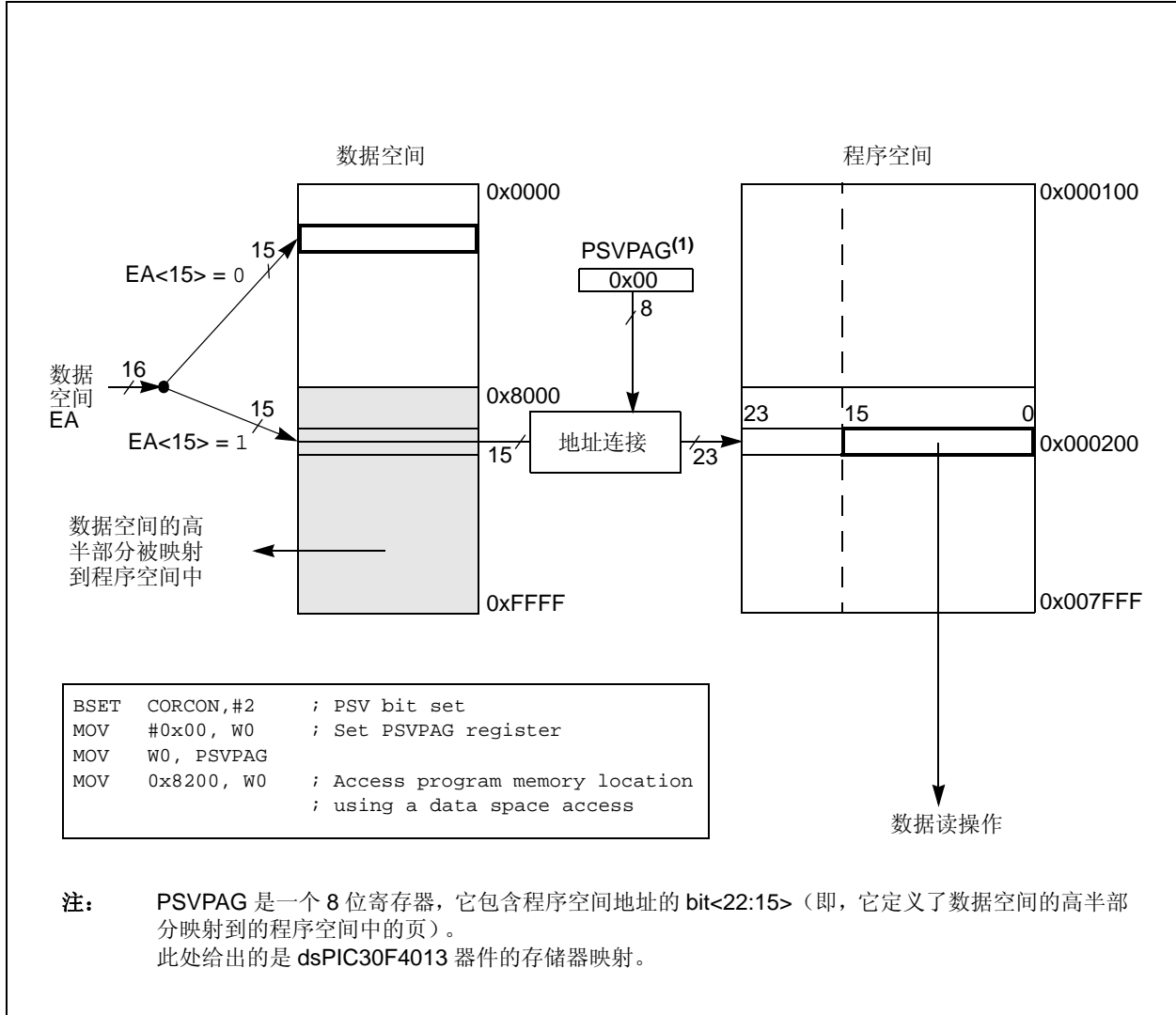
对于在 REPEAT 循环外执行的使用 PSV 的指令:

- 以下指令除规定的执行时间外还需一个额外的指令周期:
 - 带数据操作数预取操作的 MAC 类指令
 - MOV 指令
 - MOV.D 指令
- 所有其他指令除规定的执行时间外还需两个额外的指令周期。

对于在 REPEAT 循环内执行的使用 PSV 的指令:

- 以下指令除规定的执行时间外还需两个额外的指令周期:
 - 在第一次迭代时执行的指令
 - 在最后一次迭代时执行的指令
 - 在由于中断而退出循环前执行的指令
 - 中断被处理后, 在重新进入循环时执行的指令
- 允许在 REPEAT 循环的任何其他迭代中使用 PSV 访问数据, 该操作的执行时间为一个周期。

图 3-6: 将数据空间映射到程序空间的操作



dsPIC30F3014/4013

3.2 数据地址空间

内核具有两个数据空间。可将这两个数据空间视为两个独立的空间（对于某些 DSP 指令）或是一个统一的线性寻址空间（对于 MCU 指令）。使用两个地址发生单元（AGU）和相互独立的数据存取路径对这两个数据空间进行访问。

3.2.1 数据空间存储器映射

数据存储空间被分为两块，分别为 X 数据空间和 Y 数据空间。该架构的要素就在于 Y 空间是 X 空间的子集，它完全包含在 X 空间内。X 空间和 Y 空间具有连续的地址，以提供明显的线性寻址空间。

当执行除 MAC 类指令外的任何指令时，X 空间由 64 KB 的数据地址空间组成，其中包含整个 Y 地址空间。当执行某条 MAC 类指令时，X 空间由 64 KB 的数据地址空间组成，但不包括 Y 地址空间（只用于数据读操作）。换言之，除 MAC 类指令外的所有其他指令均将整个数据存储空间视为一个组合的地址空间。MAC 类指令将 Y 地址空间从数据空间中抽取出来，它使用 W10 和 W11 的内容构成的 EA 来寻址这一个空间。而通过使用 W8 和 W9 来寻址其余的 X 数据空间。只有使用 MAC 类指令才能同时访问 X 和 Y 这两个地址空间。

数据空间存储器映射如图 3-7 所示。

图 3-7: dsPIC30F3014/dsPIC30F4013 数据空间存储器映射

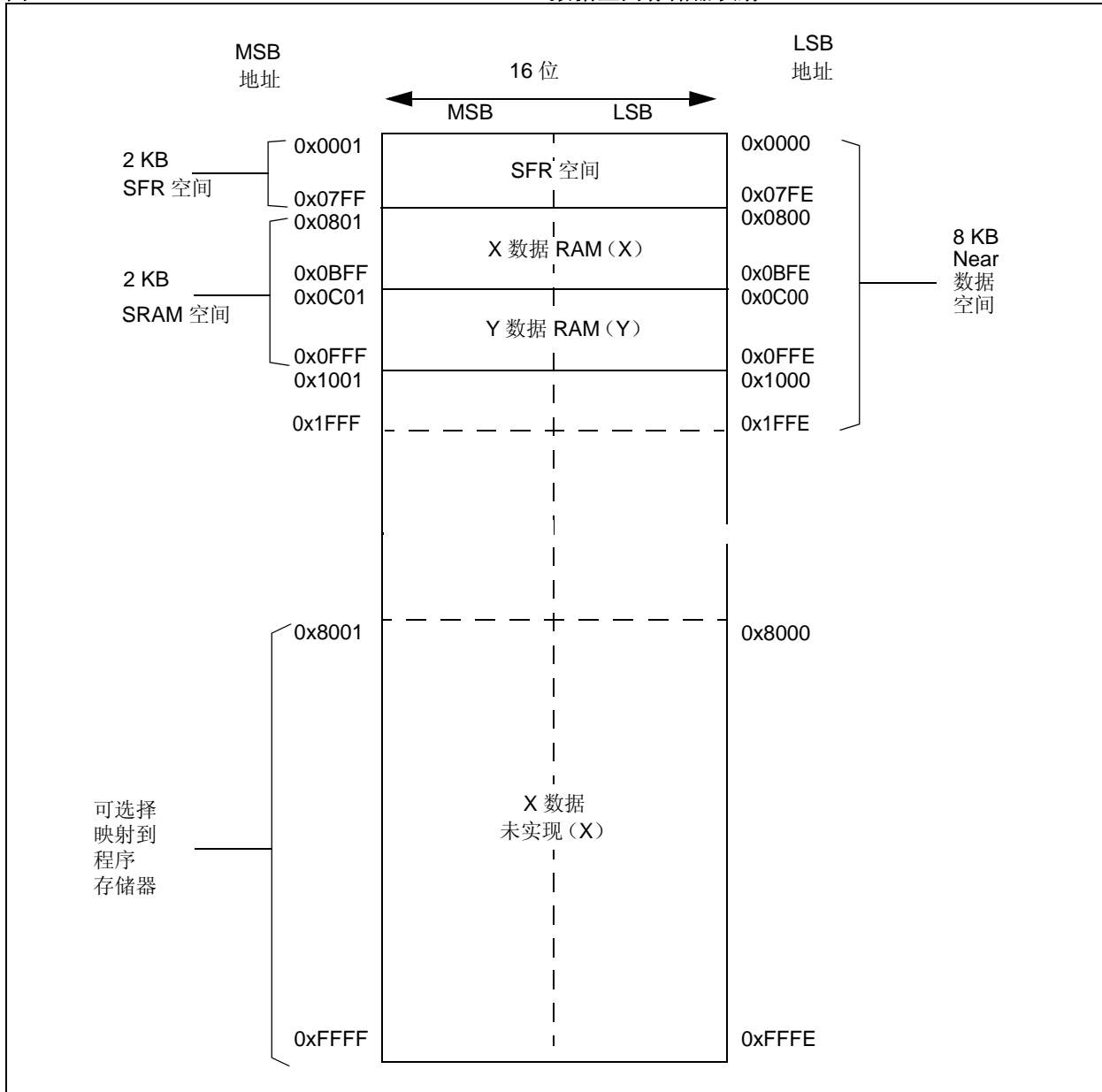
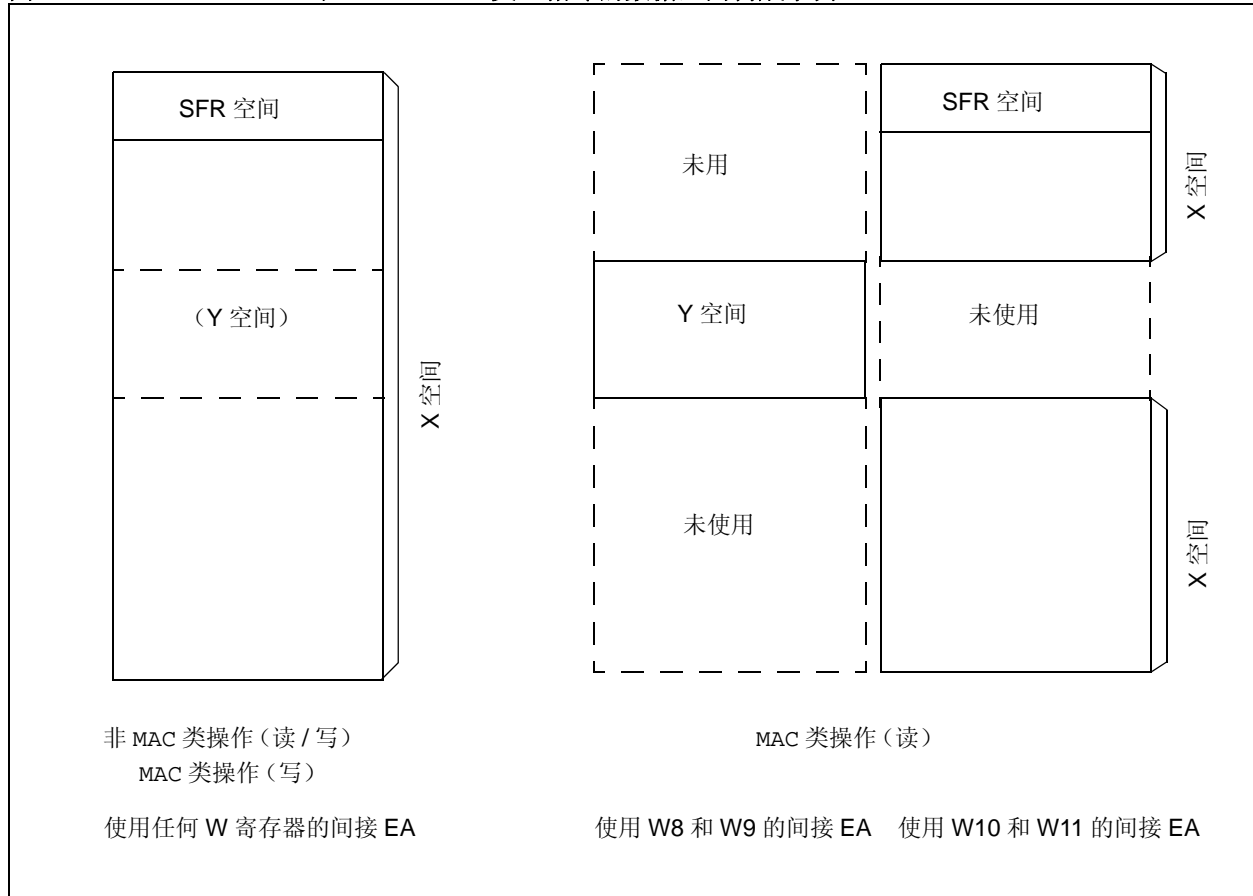


图 3-8: MCU 和 DSP (MAC 类) 指令的数据空间构成示例



3.2.2 数据空间

X 数据空间可被所有指令使用且支持所有寻址模式。其读写数据总线是相互独立的。X 读数据总线是所有将数据空间视为一个组合的 X 和 Y 地址空间的指令的数据返回路径。它也是双操作数读指令（MAC 类）的 X 地址空间数据路径。X 写数据总线是所有指令对数据空间执行写操作的唯一路径。

X 数据空间还支持所有指令的模寻址，但要受到寻址模式的限制。只有写 X 数据空间的操作才支持位反转寻址模式。

MAC 类指令（CLR、ED、EDAC、MAC、MOVSAC、MPY、MPY.N 和 MSC）将 Y 数据空间与 X 数据空间配合使用以提供两条可同时读取数据的路径。不能通过 Y 总线执行任何写操作。此类指令专门指定 W10 和 W11 作为寻址 Y 数据空间的 W 寄存器指针，而指定 W8 和 W9 作为寻址 X 数据空间的 W 寄存器指针。注意，在累加器回写操作期间，数据地址空间被认为是一个组合的 X 和 Y 数据空间，因此写操作通过 X 总线发生。从而可以对整个地址空间中的任何地址单元执行写操作。

Y 数据空间只能用于与 MAC 类指令相关的数据预取操作。它也支持自动循环缓冲区的模寻址。当然，所有其他指令将 Y 数据地址空间视作组合的线性空间的一部分，通过 X 数据总线对其进行访问。

X 数据空间和 Y 数据空间之间的边界定义如图 3-7，它不能由用户编程。如果 EA 指向已为其分配的数据空间之外的数据或是指向物理存储器之外的单元，则将返回全零的字或字节。例如，虽然所有使用任何寻址模式的非 MAC 指令都能访问 Y 地址空间，但使用 W8 或 W9（X 空间指针）作为地址指针的 MAC 指令从 Y 数据空间取数据将返回 0x0000。

表 3-2: 无效存储器访问的结果

尝试的操作	返回的数据
EA = 未实现的地址	0x0000
MAC 指令中使用 W8 或 W9 来访问 Y 数据空间	0x0000
MAC 指令中使用 W10 或 W11 来访问 X 数据空间	0x0000

所有有效地址均为 16 位宽，指向数据空间中的字节。因此，数据空间的地址范围为 64 KB 或 32K 字。

3.2.3 数据空间宽度

内核的数据宽度为 16 位。所有内部寄存器都以 16 位宽的字的形式构成。数据存储空间以可按字节寻址的 16 位宽的块构成。

3.2.4 数据对齐

为了便于与 PIC® MCU 器件向后兼容并提高数据存储空间的使用效率，dsPIC30F 指令集支持字和字节操作。数据存储器和寄存器中的数据以字对齐，但数据空间的所有 EA 都被解析成字节。数据字节读操作将读取包含此字节的整个字，再使用有效地址的最低位（LSb）决定要选择哪个字节。选定的字节被放置到 X 数据总线的 LSB 中（由于 MAC 类指令只能取字，因而不可能通过 Y 数据总线进行字节访问）。即，数据存储器和寄存器以两个并列的字节宽度的实体构成，它们共享（字）地址译码，但写操作线相互独立。数据字节写操作只写入阵列或寄存器中与字节地址匹配的那一半。

这类字节访问的结果是，所有有效地址的计算（包括由 DSP 操作生成的只限字大小数据的地址）在内部进行调整以逐次访问字对齐的存储器。例如，被内核所识别的后修改寄存器间接寻址模式 [Ws++]，字节操作结果为 Ws + 1，而字操作的结果则是 Ws + 2。

所有字访问必须与一个偶地址对齐，不支持违背上述原则的字数据取操作，所以在字节和字的混合操作或转换 8 位 MCU 代码时必须十分小心。若试图执行不与偶地址对齐的读或写操作，就会产生地址错误陷阱。如果在读操作时发生该错误，则当前指令将立即停止；如果在写操作时发生该错误，指令会继续执行，但不会真正写入。无论哪种情况，都会执行陷阱程序，以允许系统和/或用户在执行地址故障恢复程序前检查机器状态。

图 3-9: 数据对齐

	15	MSB	8	7	LSB	0	
0001	字节 1		字节 0				0000
0003	字节 3		字节 2				0002
0005	字节 5		字节 4				0004

装载到任何W寄存器的所有字节都只装载到LSB。MSB不被修改。

符号扩展指令（SE）用于允许用户将 8 位有符号数据转换为 16 位有符号值。而对于 16 位无符号数据，用户通过在相应地址上执行零扩展（ZE）指令可以清零任一 W 寄存器的高字节。

虽然大多数指令都可以对字或字节数据执行操作，但应该注意有些指令（包含 DSP 指令）只能针对字进行操作。

3.2.5 NEAR 数据空间

在 X 地址存储空间的 0x0000 和 0x1FFF 之间保留了一个 8 KB 的“near”数据空间，可通过所有存储器直接寻址指令中的 13 位绝对地址字段来直接寻址该空间。其余的 X 地址空间和所有的 Y 地址空间都采用间接寻址。此外，使用 MOV 指令可寻址整个 X 数据空间，该指令带有 16 位地址字段，支持存储器直接寻址模式。

3.2.6 软件堆栈

dsPIC DSC 器件包含软件堆栈。使用 W15 作为堆栈指针。

堆栈指针始终指向第一个空字，并从低地址向高地址递增。出栈操作该指针预减，压栈操作该指针预加，如图 3-10 所示。注意对于任意 CALL 指令期间的 PC 压栈操作，PC 的高字节部分在压栈前要经过零扩展以保证高字节始终清零。

注： 异常事件处理期间的PC压栈操作会在操作前将 SRL 寄存器与 PC 的 MSB 连接起来。

有一个与堆栈指针相关的堆栈指针限制寄存器（Stack Pointer Limit register, SPLIM）。复位时 SPLIM 不会被初始化。由于所有栈操作都必须字对齐，因此复位时 SPLIM<0> 被强制为 0，这与堆栈指针相同。使用 W15 作为源或目标指针生成有效地址（EA）后，生成的地址要与 SPLIM 中的值做比较。如果堆栈指针（W15）的内容与 SPLIM 寄存器的值相等，则执行压栈操作时不会产生堆栈错误陷阱。而在下一次压栈操作时产生陷阱。例如，如果希望在栈地址超过 RAM 中的 0x2000 时产生堆栈错误陷阱，则用 0x1FFE 初始化 SPLIM。

同样，在栈指针地址小于 0x0800 时会产生栈指针下溢（堆栈错误）陷阱，从而防止堆栈与特殊功能寄存器（SFR）空间重叠。

对 SPLIM 寄存器执行写操作后，不要立即使用 W15 对该寄存器执行间接读操作。

图 3-10： 执行 CALL 指令的堆栈帧

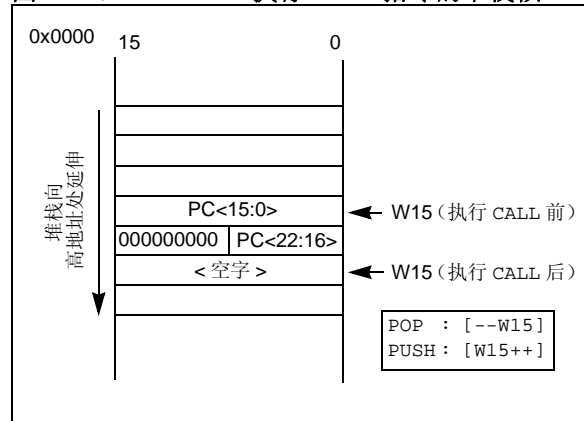


表 3-3: 内核寄存器映射 (1)

SFR 名称	地址 (低地址)	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态	
W0	0000	W0/WREG																0000 0000 0000 0000	
W1	0002	W1																0000 0000 0000 0000	
W2	0004	W2																0000 0000 0000 0000	
W3	0006	W3																0000 0000 0000 0000	
W4	0008	W4																0000 0000 0000 0000	
W5	000A	W5																0000 0000 0000 0000	
W6	000C	W6																0000 0000 0000 0000	
W7	000E	W7																0000 0000 0000 0000	
W8	0010	W8																0000 0000 0000 0000	
W9	0012	W9																0000 0000 0000 0000	
W10	0014	W10																0000 0000 0000 0000	
W11	0016	W11																0000 0000 0000 0000	
W12	0018	W12																0000 0000 0000 0000	
W13	001A	W13																0000 0000 0000 0000	
W14	001C	W14																0000 0000 0000 0000	
W15	001E	W15																0000 1000 0000 0000	
SPLIM	0020	SPLIM																0000 0000 0000 0000	
ACCAL	0022	ACCAL																0000 0000 0000 0000	
ACCAH	0024	ACCAH																0000 0000 0000 0000	
ACCAU	0026	符号扩展 (ACCA<39>)								ACCAU								0000 0000 0000 0000	
ACCBL	0028	ACCBL																0000 0000 0000 0000	
ACCBH	002A	ACCBH																0000 0000 0000 0000	
ACCBU	002C	符号扩展 (ACCB<39>)								ACCBU								0000 0000 0000 0000	
PCL	002E	PCL																0000 0000 0000 0000	
PCH	0030	—	—	—	—	—	—	—	—	—	PCH							0000 0000 0000 0000	
TBLPAG	0032	—	—	—	—	—	—	—	—	TBLPAG							0000 0000 0000 0000		
PSVPAG	0034	—	—	—	—	—	—	—	—	PSVPAG							0000 0000 0000 0000		
RCOUNT	0036	RCOUNT																uuuu uuuu uuuu uuuu	
DCOUNT	0038	DCOUNT																uuuu uuuu uuuu uuuu	
DOSTARTL	003A	DOSTARTL																0	uuuu uuuu uuuu uu0
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	DOSTARTH							0000 0000 0uuu uuuu	
DOENDL	003E	DOENDL																0	uuuu uuuu uuuu uu0
DOENDH	0040	—	—	—	—	—	—	—	—	—	DOENDH							0000 0000 0uuu uuuu	
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000 0000 0000 0000	

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 3-3: 内核寄存器映射 (1) (续)

SFR 名称	地址 (低地址)	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
CORCON	0044	—	—	—	US	EDT	DL2	DL1	DL0	SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0000 0000 0010 0000
MODCON	0046	XMODEN	YMODEN	—	—	BWM<3:0>			YWM<3:0>			XWM<3:0>			0000 0000 0000 0000			
XMODSRT	0048	XS<15:1>															0	uuuu uuuu uuuu uuu0
XMODEND	004A	XE<15:1>															1	uuuu uuuu uuuu uuu1
YMODSRT	004C	YS<15:1>															0	uuuu uuuu uuuu uuu0
YMODEND	004E	YE<15:1>															1	uuuu uuuu uuuu uuu1
XBREV	0050	BREN	XB<14:0>															uuuu uuuu uuuu uuuu
DISICNT	0052	—	—	DISICNT<13:0>													0000 0000 0000 0000	

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

4.0 地址发生器单元

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

dsPIC DSC 内核包含两个独立的地址发生器单元：X AGU 和 Y AGU。Y AGU 仅支持 DSP MAC 类指令字数据读取。dsPIC DSC AGU 支持如下三类数据寻址模式：

- 线性寻址
- 模（循环）寻址
- 位反转寻址

线性寻址和模数据寻址模式可用于数据空间或程序空间。位反转寻址只适用于数据空间内的地址单元。

4.1 指令寻址模式

表 4-1 中给出的寻址模式均为基本寻址模式，它们已经过优化，可支持各条指令的特殊功能。MAC 类指令的寻址模式与其他指令类型中的寻址模式略有不同。

4.1.1 文件寄存器指令

大部分文件寄存器指令使用 13 位地址字段（f）直接对数据存储器的前 8192 个字节（near 数据空间）中的数据进行寻址。大多数文件寄存器指令使用工作寄存器 W0，W0 在这些指令中表示为 WREG。目的寄存器通常是同一个文件寄存器或 WREG（MUL 指令除外），它将结果写入一个寄存器或一对寄存器。使用 MOV 指令能够获得更大的灵活性，在文件寄存器操作期间可以访问整个数据空间。

4.1.2 MCU 指令

三操作数 MCU 指令的形式为：

$$\text{操作数 3} = \text{操作数 1} < \text{功能} > \text{操作数 2}$$

其中，操作数 1 始终是工作寄存器（即，寻址模式只能为寄存器直接寻址），称为 Wb。操作数 2 可以是 W 寄存器，其内容取自数据存储器或为 5 位立即数。结果可以存放在 W 寄存器中或一个地址单元中。MCU 指令支持如下寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 5 位或 10 位立即数寻址

注： 并非所有的指令都支持上述全部寻址模式。不同指令可能支持这些寻址模式中的不同模式。

表 4-1: 支持的基本寻址模式

寻址模式	说明
文件寄存器直接寻址	明确指定文件寄存器的地址。
寄存器直接寻址	直接访问寄存器中的内容。
寄存器间接寻址	由 Wn 的内容构成 EA。
执行后修改的寄存器间接寻址	由 Wn 的内容构成 EA，然后用一个常量值修改 Wn（递增或递减）。
执行前修改的寄存器间接寻址	先用一个有符号常量值修改 Wn（递增或递减），再由此时的 Wn 内容构成 EA。
带寄存器偏移量的寄存器间接寻址	Wn 和 Wb 的和构成 EA。
带立即数偏移量的寄存器间接寻址	Wn 和立即数的和构成 EA。

4.1.3 传送和累加器指令

与其他指令相比，传送指令和 DSP 累加类指令可提供更为灵活的寻址模式。除大多数 MCU 指令支持的寻址模式之外，传送和累加指令还支持带寄存器偏移的寄存器间接寻址模式，也称为寄存器变址寻址模式。

注： 对于 MOV 指令，指令中指定的寻址模式对于源寄存器和目的寄存器 EA 可以是不同的。然而，4 位 Wb（寄存器偏移量）字段为源寄存器和目的寄存器所共用（但通常只由源寄存器或目的寄存器之一使用）。

概括地说，传送和累加指令支持如下寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 带寄存器偏移的寄存器间接寻址（变址寻址）
- 带立即数偏移的寄存器间接寻址
- 8 位立即数寻址
- 16 位立即数寻址

注： 并非所有的指令都支持上述全部寻址模式。单个指令可能支持上述寻址模式中的部分模式。

4.1.4 MAC 指令

双源操作数 DSP 指令（CLR, ED, EDAC, MAC, MPY, MPY.N, MOVSAAC 和 MSC），也称为 MAC 类指令，使用一组简化的寻址模式供用户通过寄存器间接寻址表有效地对数据指针进行操作。

双源操作数预取寄存器必须是 {W8,W9,W10,W11} 集合的成员。对于数据读取操作，W8 和 W9 始终用于 X RAGU，而 W10 和 W11 始终用于 Y AGU。因此，产生的有效地址（修改前和修改后）对于 W8 和 W9 必须是 X 数据空间中的有效地址，对于 W10 和 W11 则必须是 Y 数据空间中的有效地址。

注： 带寄存器偏移量的寄存器间接寻址只适用于 W9（在 X 空间中）和 W11（在 Y 空间中）。

概括地说，MAC 类指令支持如下寻址模式：

- 寄存器间接寻址
- 执行后修改（修改量为 2）的寄存器间接寻址
- 执行后修改（修改量为 4）的寄存器间接寻址
- 执行后修改（修改量为 6）的寄存器间接寻址
- 带寄存器偏移的寄存器间接寻址（变址寻址）

4.1.5 其他指令

除了上述列出的寻址模式外，某些指令使用不同长度的立即数。例如，BRA（转移）指令使用 16 位有符号立即数来直接指定转移目的地址，而 DISI 指令使用 14 位无符号立即数字段。对于某些指令，如 ADD Acc，操作数的源地址或目的地址由操作码本身指定。某些操作不包含任何操作数，如 NOP。

4.2 模寻址

模寻址模式是一种使用硬件来自动支持循环数据缓冲区的方法。其目的是在执行紧密循环代码时（这在许多 DSP 算法中很常见），不需要用软件来执行数据地址边界检查。

可以在数据空间或程序空间中进行模寻址（因为两种空间的数据指针机制基本相同）。X（也提供指向程序空间的指针）和 Y 数据空间中都支持一个循环缓冲区。模寻址可以对任何 W 寄存器指针进行操作。然而，最好不要将 W14 或 W15 用于模寻址，因为这两个寄存器分别用作堆栈帧指针和堆栈指针。

通常，任何特定的循环缓冲区只能配置为单向工作，因为根据缓冲区的方向，对缓冲区起始地址（对于递增缓冲区）或结束地址（对于递减缓冲区）有某些限制。

使用限制的惟一例外是那些长度为 2 的幂的缓冲区。这些缓冲区满足起始和结束地址标准，因此它们可以双向工作（即，在低地址边界和高地址边界上都将进行地址边界检查）。

4.2.1 起始和结束地址

模寻址机制要求指定起始和结束地址，并将其载入 16 位模缓冲地址寄存器：XMODSRT、XMODEND、YMODSRT 和 YMODEND（见 表 3-3）。

注： Y 空间模寻址 EA 计算使用字大小的数据（每个 EA 的 LSb 始终清零）。

循环缓冲区的长度未直接指定。该长度由相应的起始地址和结束地址的差决定。循环缓冲区最大长度为 32K 字（64 KB）。

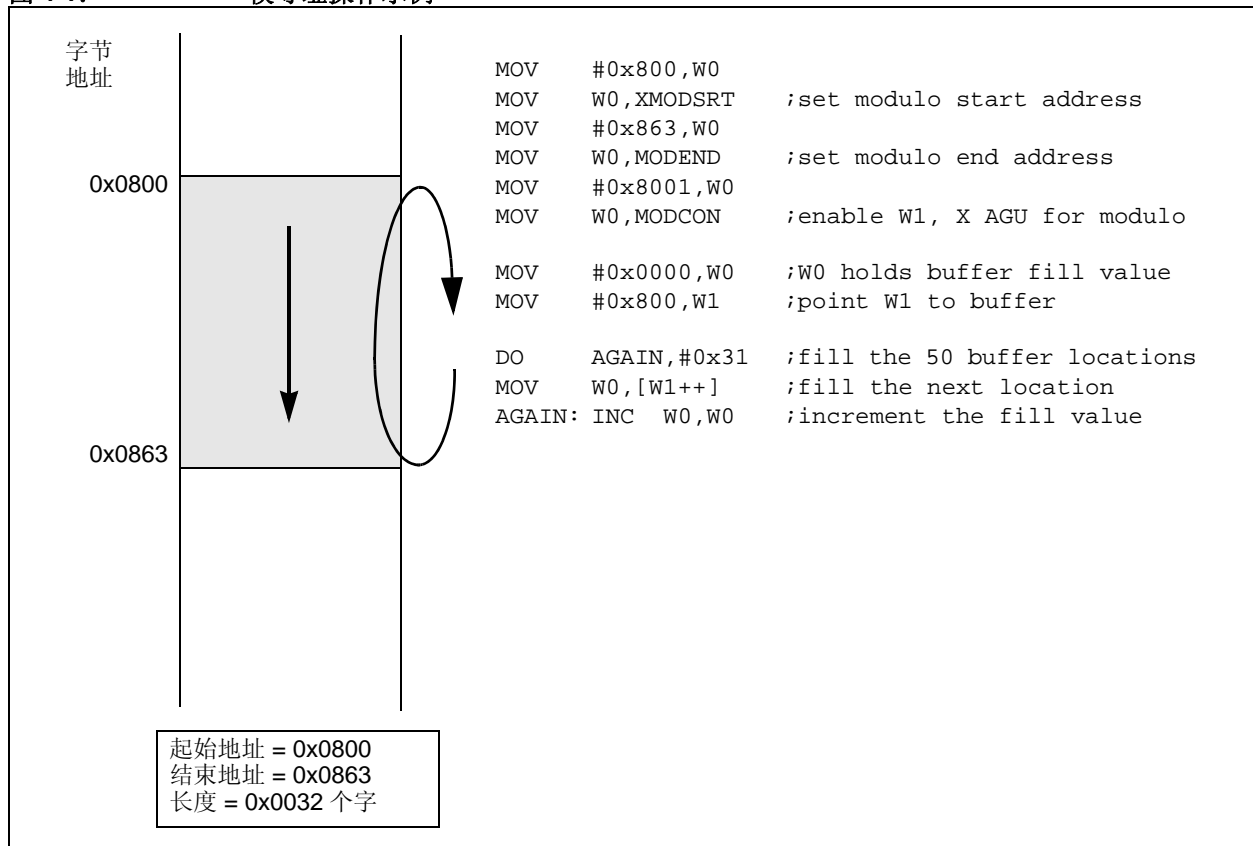
4.2.2 W 地址寄存器选择

模寻址和位反转控制寄存器的 MODCON<15:0> 中包含使能标志位以及指定 W 地址寄存器的 W 寄存器字段。由 XWM 和 YWM 字段选择模寻址时所使用的寄存器。如果 XWM = 15，则禁止 X RAGU 和 X WAGU 模寻址。类似地，如果 YWM = 15，禁止 Y AGU 模寻址。

要对其进行模寻址的 X 地址空间指针 W 寄存器（XWM）位于 MODCON<3:0> 中（见 表 3-3）。当 XWM 被设置为除 15 之外的任何值且 XMODEN 位（MODCON<15>）置 1 时，使能 X 数据空间的模寻址。

要对其进行模寻址的 Y 地址空间指针 W 寄存器（YWM）位于 MODCON<7:4> 中。当 YWM 被设置为除 15 之外的任何值且 XMODEN 位（MODCON<14>）置 1 时，使能 Y 数据空间的模寻址。

图 4-1： 模寻址操作示例



4.2.3 模寻址的应用

模寻址可以应用于与任何 W 寄存器相关的有效地址 (EA) 的计算中。应意识到, 地址边界检查功能不仅会检查地址是否正好在地址边界上, 而且会检查地址是否小于或大于上限 (对于递增缓冲区) 和下限 (对于递减缓冲区), 这是很重要的。因此, 地址变化可能会越过边界, 但仍然可以正确调整。

注: 只有使用预修改或后修改的寻址模式来计算有效地址时, 模修正有效地址才被写入寄存器。如果使用了地址偏移量 (例如, [W7 + W2]), 会进行模地址修正, 但寄存器的内容保持不变。

4.3 位反转寻址

位反转寻址用来简化基 2 FFT 算法的数据重新排序。X WAGU 支持位反转寻址, 并只能用于数据写入。

地址修改量可以是常数或寄存器的内容, 它被视为将其位顺序反转。源地址和目的地址为正常顺序。于是, 惟一需要反转的操作数就是地址修改量。

4.3.1 位反转寻址的实现

在当发生下列情况时, 使能位反转寻址:

1. 当 MODCON 寄存器的 BWM (W 寄存器选择位) 为除 15 之外的任何值 (使用位反转寻址无法访问堆栈), 且
2. XBREV 寄存器的 BREN 位置 1, 且
3. 所用的寻址模式为带预递增或后递增的寄存器间接寻址。

如果位反转缓冲区的长度为 $M = 2^N$ 字节, 则数据缓冲区起始地址的最后 “N” 位必须为零。

XB<14:0> 为位反转地址修饰符, 也称 “中心点 (pivot point)” 通常是一个常数。在进行 FFT 计算时, 它的值等于 FFT 数据缓冲区大小的一半。

注: 所有位反转 EA 的计算都使用字大小的数据 (每个 EA 的 LSb 始终为零)。为了产生兼容 (字节) 地址, 要相应地调整 XB 的值。

位反转寻址使能时, 将仅用于执行预增或后增寻址的寄存器间接寻址和字大小数据写操作。而对于任何其他寻址模式或字节数据, 不会进行位反转寻址, 而是生成正常顺序的地址。在进行位反转寻址时, W 地址指针将始终加上地址修饰符 (XB), 与寄存器间接寻址模式相关的偏移量将被忽略。此外, 由于要求是字数据, EA 的 LSb 被忽略 (且始终被清零)。

注: 不应同时使能模寻址和位反转寻址。如果用户试图这么做的话, 对于 X WAGU, 位反转寻址将优先, X WAGU 模寻址将被禁止。然而在 X RAGU 中, 模寻址将继续起作用。

如果已经通过置 1 BREN (XBREV<15>) 位使能了位反转寻址, 那么, 在写 XBREV 寄存器之后, 不应马上跟一个使用 W 寄存器的间接读操作, 而此时 W 寄存器已被指定为位反转指针。

图 4-2: 位反转寻址示例

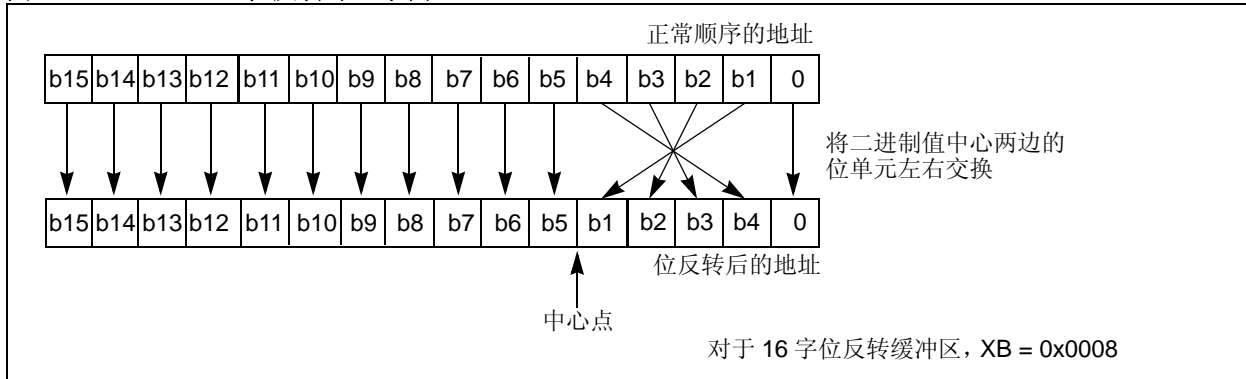


表 4-2: 位反转地址序列 (16 项)

正常地址					位反转后的地址				
A3	A2	A1	A0	十进制	A3	A2	A1	A0	十进制
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	8
0	0	1	0	2	0	1	0	0	4
0	0	1	1	3	1	1	0	0	12
0	1	0	0	4	0	0	1	0	2
0	1	0	1	5	1	0	1	0	10
0	1	1	0	6	0	1	1	0	6
0	1	1	1	7	1	1	1	0	14
1	0	0	0	8	0	0	0	1	1
1	0	0	1	9	1	0	0	1	9
1	0	1	0	10	0	1	0	1	5
1	0	1	1	11	1	1	0	1	13
1	1	0	0	12	0	0	1	1	3
1	1	0	1	13	1	0	1	1	11
1	1	1	0	14	0	1	1	1	7
1	1	1	1	15	1	1	1	1	15

表 4-3: XBREV 寄存器的位反转地址修改量的值

缓冲区大小 (字)	XB<14:0> 位反转地址修改量
1024	0x0200
512	0x0100
256	0x0080
128	0x0040
64	0x0020
32	0x0010
16	0x0008
8	0x0004
4	0x0002
2	0x0001

dsPIC30F3014/4013

注:

5.0 闪存程序存储器

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

dsPIC30F 系列器件包含用以执行用户代码的内部闪存程序存储器。用户可使用两种方法对该存储器编程：

1. 运行时自编程 (Run-Time Self-Programming, RTSP)
2. 在线串行编程 (In-Circuit Serial Programming, ICSP™)

5.1 在线串行编程 (ICSP)

可以在最终的应用电路中对 dsPIC30F 器件进行串行编程。只需要 5 根线即可实现这一操作，其中编程时钟线和编程数据线各一根（分别称为 PGC 和 PGD），其余 3 根分别是电源线 (VDD)、接地线 (VSS) 和主复位线 (MCLR)。这使用户可在制造电路板时使用未编程器件，而仅在产品交付前才对数字信号控制器进行编程。这还能保证将最新版本的固件或者定制固件烧写到器件中。

5.2 运行时自编程 (RTSP)

使用 TBLRD (表读) 和 TBLWT (表写) 指令完成 RTSP。通过 RTSP，用户可以一次擦除程序存储器中的 32 条指令 (96 个字节) 或一次向程序存储器写入 32 条指令 (96 个字节)。

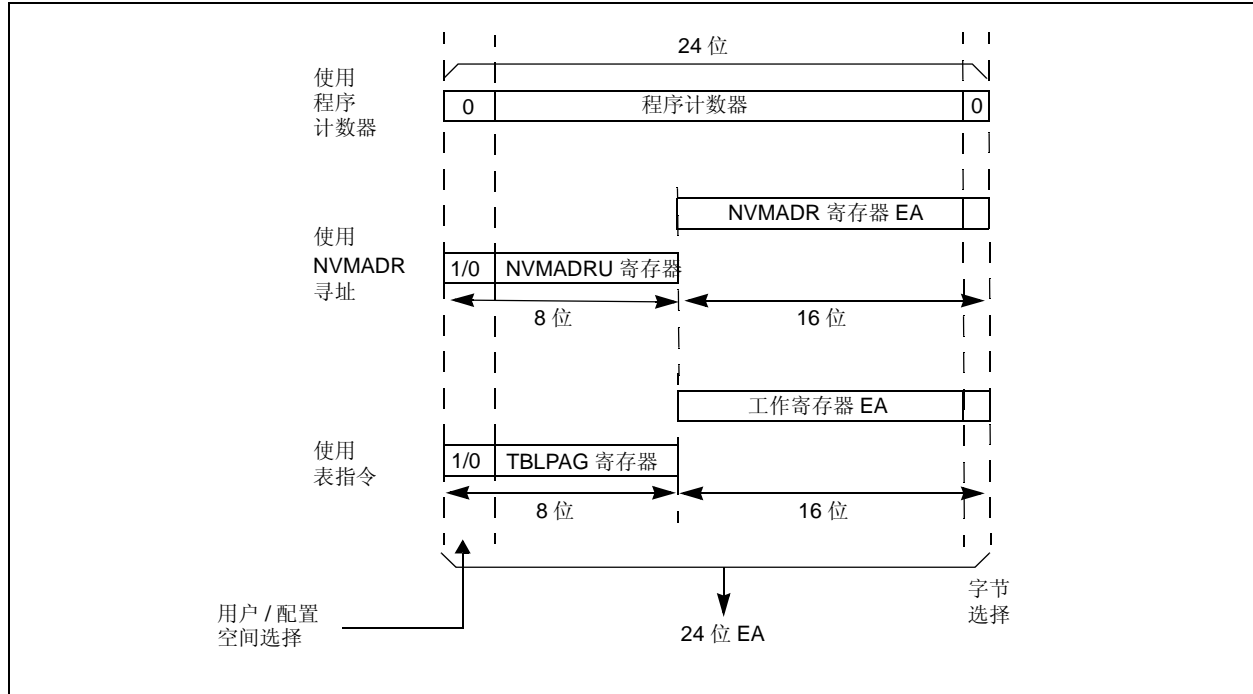
5.3 表指令操作汇总

TBLRDL 和 TBLWTL 指令用于从程序存储器的 bit <15:0> 读取数据，或将数据写入程序存储器的 bit <15:0>。TBLRDL 和 TBLWTL 可以字或字节的方式访问程序存储器。

TBLRDH 和 TBLWTH 指令用于从程序存储器的 bit <23:16> 读取数据或将数据写入程序存储器的 bit <23:16>。TBLRDH 和 TBLWTH 可以字或字节的方式访问程序存储器。

使用 TBLPAG 寄存器的 bit <7:0> 和来自表指令中指定的 W 寄存器的有效地址 (EA) 构成 24 位程序存储器地址，如图 5-1 所示。

图 5-1: 对表和 NVM 寄存器进行寻址



dsPIC30F3014/4013

5.4 RTSP 操作

dsPIC30F 闪存程序存储器以行 (row) 和板 (panel) 为单位构成。每行由 32 条指令或 96 个字节组成。每板由 128 行或 4K x 24 条指令组成。RTSP 使用户可一次擦除一行 (32 条指令) 或一次编程 4 条指令。可使用 RTSP 来编程多个程序存储器板, 但必须在每个板边界更改表指针。

程序存储器的每板均包含一些写锁存器, 可用来存储 32 条指令的编程数据。在实际的编程操作开始之前, 写数据必须被装载到板写锁存器中。要编程到板中的数据按如下顺序写入写锁存器: 指令 0、指令 1 等。被装载的指令字必须始终来自一个 32 位的地址边界。

RTSP 编程的基本次序是: 设置表指针, 然后执行一系列 TBLWT 指令来装载写锁存器。通过设置 NVMCON 寄存器中的特定位执行编程。要装载 32 条指令, 需要执行 32 条 TBLWTL 指令和 4 条 TBLWTH 指令。如果需要多个板进行编程, 则需要更改表指针并对下一组写锁存器执行写操作。

由于所有的表写操作都只写入表锁存器, 因而它们都是单字写操作 (2 个指令周期)。对每一行编程都需要一个编程周期。

在整个 VDD 范围内, 在正常工作状态下闪存程序存储器都是可读可写且可擦除的。

5.5 控制寄存器

有 4 个 SFR 用来读写闪存程序存储器, 它们是:

- NVMCON
- NVMADR
- NVMADRU
- NVMKEY

5.5.1 NVMCON 寄存器

NVMCON 寄存器控制要擦除的块和要编程的存储器类型, 以及编程周期的开始。

5.5.2 NVMADR 寄存器

NVMADR 寄存器用来保持有效地址的两个低字节。NVMADR 寄存器捕捉已被执行的上一个表指令的 EA<15:0>, 选择要写入的行。

5.5.3 NVMADRU 寄存器

NVMADRU 寄存器用来保持有效地址的高字节。NVMADRU 寄存器捕捉已被执行的上一个表指令的 EA<23:16>。

5.5.4 NVMKEY 寄存器

NVMKEY 是一个只写寄存器, 用于写保护。要启动编程或擦除序列, 用户必须连续写 0x55 和 0xAA 到 NVMKEY 寄存器。更多详细信息请参见第 5.6 节 “编程操作”。

注: 用户也可直接写 NVMADR 和 NVMADRU 寄存器来指定要擦除或编程的程序存储器地址。

5.6 编程操作

在 RTSP 模式下，编程或擦除内部闪存时需要执行完整的编程步骤。编程操作的持续时间通常为 2 ms，在此期间处理器暂停（等待）直到操作完成。将 WR 位（NVMCON<15>）置 1 启动操作，操作完成时 WR 位会自动清零。

5.6.1 对闪存程序存储器编程的算法

用户可以一次擦除或编程闪存程序存储器的一行。一般步骤如下：

1. 读取闪存程序存储器的一行（32 个指令字）并将读到的内容作为数据“映像”存储到数据 RAM 中。
2. 使用所需的新数据更新数据映像。
3. 擦除闪存行。
 - a) 将 NVMCON 寄存器设置为擦除闪存程序存储器的多个字，并将 WREN 位置 1。
 - b) 将要被擦除的行的地址写入 NVMADRU/NVMADR。
 - c) 将 0x55 写入 NVMKEY。
 - d) 将 0xAA 写入 NVMKEY。
 - e) 将 WR 位置 1。这将启动擦除周期。
 - f) CPU 停止，等待擦除周期完成。
 - g) WR 位在擦除周期结束时清零。

4. 将数据 RAM “映像”中的前 32 个指令字写入闪存程序存储器写锁存器。
5. 将 32 个指令字编程到闪存程序存储器中。
 - a) 将 NVMCON 寄存器设置为将多个字编程到闪存程序存储器中，并将 WREN 位置 1。
 - b) 将 0x55 写入 NVMKEY。
 - c) 将 0xAA 写入 NVMKEY。
 - d) 将 WR 位置 1。这将启动编程周期。
 - e) CPU 停止，等待编程周期完成。
 - f) 当编程周期结束时，WR 位由硬件自动清零。
6. 根据需要重复步骤 1 至步骤 5 以对所需容量的闪存程序存储区编程。

5.6.2 擦除程序存储器的一行

例 5-1 给出了可用来擦除一行（32 条指令）程序存储器的代码序列。

例 5-1: 擦除程序存储器的一行

```

; Setup NVMCON for erase operation, multi word write
; program memory selected, and writes enabled
MOV    #0x4041,W0
MOV    W0,NVMCON
; Init NVMCON SFR
; Init pointer to row to be ERASED
MOV    #tblpage(PROG_ADDR),W0
MOV    W0,NVMADRU
MOV    #tbloffset(PROG_ADDR),W0
MOV    W0,NVMADR
DISI   #5
; Block all interrupts with priority <7 for
; next 5 instructions

MOV    #0x55,W0
MOV    W0,NVMKEY
MOV    #0xAA,W1
MOV    W1,NVMKEY
BSET   NVMCON,#WR
NOP
NOP
; Insert two NOPs after the erase
; command is asserted

```

dsPIC30F3014/4013

5.6.3 装载写锁存器

例 5-2 给出了可用来装载 96 字节的写锁存器的指令序列。需要 32 条 TBLWTL 指令和 32 条 TBLWTH 指令来装载由表指针选定的写锁存器。

5.6.4 启动编程序列

作为保护措施，必须使用 NVMKEY 的写启动序列，以允许进行任何的写或编程操作。当执行完编程命令后，用户必须等待一段时间直至编程完成。编程启动序列后应紧跟两条 NOP 指令，如例 5-3 所示。

例 5-2: 装载写锁存器

```
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
MOV    #0x0000,W0                ;
MOV    W0,TBLPAG                 ; Initialize PM Page Boundary SFR
MOV    #0x6000,W0                ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
MOV    #LOW_WORD_0,W2            ;
MOV    #HIGH_BYTE_0,W3          ;
TBLWTL W2,[W0]                  ; Write PM low word into program latch
TBLWTH W3,[W0++]                ; Write PM high byte into program latch
; 1st_program_word
MOV    #LOW_WORD_1,W2            ;
MOV    #HIGH_BYTE_1,W3          ;
TBLWTL W2,[W0]                  ; Write PM low word into program latch
TBLWTH W3,[W0++]                ; Write PM high byte into program latch
; 2nd_program_word
MOV    #LOW_WORD_2,W2            ;
MOV    #HIGH_BYTE_2,W3          ;
TBLWTL W2,[W0]                  ; Write PM low word into program latch
TBLWTH W3,[W0++]                ; Write PM high byte into program latch
.
.
.
; 31st_program_word
MOV    #LOW_WORD_31,W2           ;
MOV    #HIGH_BYTE_31,W3         ;
TBLWTL W2,[W0]                  ; Write PM low word into program latch
TBLWTH W3,[W0++]                ; Write PM high byte into program latch
```

注：在例 5-2 中，W3 中高字节的内容无效。

例 5-3: 启动编程序列

```
DISI   #5                        ; Block all interrupts with priority <7 for
                                           ; next 5 instructions
MOV    #0x55,W0                  ;
MOV    W0,NVMKEY                 ; Write the 0x55 key
MOV    #0xAA,W1                  ;
MOV    W1,NVMKEY                 ; Write the 0xAA key
BSET   NVMCON,#WR                ; Start the erase sequence
NOP                                         ; Insert two NOPs after the erase
NOP                                         ; command is asserted
```

表 5-1: NVM 寄存器映射 (1)

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
NVMCON	0760	WR	WREN	WRERR	—	—	—	—	TWRI	—	PROGOP<6:0>							0000 0000 0000 0000
NVMADR	0762	NVMADR<15:0>																uuuu uuuu uuuu uuuu
NVMADRU	0764	—	—	—	—	—	—	—	—	NVMADR<23:16>							0000 0000 uuuu uuuu	
NVMKEY	0766	—	—	—	—	—	—	—	—	KEY<7:0>							0000 0000 0000 0000	

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

6.0 数据 EEPROM 存储器

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

正常操作期间，数据 EEPROM 在整个 V_{DD} 范围内都是可读写的。数据 EEPROM 存储器直接映射到程序存储地址空间中。

用来读写闪存程序存储器的四个 SFR 也可用来访问数据 EEPROM 存储器。如第 5.5 节“控制寄存器”所述，这些寄存器是：

- NVMCON
- NVMADR
- NVMADRU
- NVMKEY

EEPROM 数据存储器允许读写单个字和 16 个字的块。当与数据存储器接口时，NVMADR 寄存器与 NVMADRU 寄存器一起来寻址要访问的 EEPROM 单元。TBLRD 和 TBLWTL 指令用来读写数据 EEPROM。dsPIC30F 器件最多具有 8 KB (4 K 字) 的数据 EEPROM 空间，地址范围为 0x7FF000 至 0x7FFFE。

在字写操作之前应该先擦除相应的存储单元。通常完成写操作需要 2 ms，但实际需要的时间随电压和温度变化。

对数据 EEPROM 执行编程或擦除操作不会停止指令流的执行。由用户负责在开始另一次数据 EEPROM 写 / 擦除操作之前等待一段适当的时间。尝试在编程或擦除操作进行过程中读数据 EEPROM 将导致读到不确定的数据。

控制位 WR 启动数据 EEPROM 的写操作，这与写闪存程序存储器类似。使用软件不能将该位清零而只能置 1。在写操作完成后由硬件将该位清零。不能由软件将 WR 位清零的特性阻止写操作意外或过早终止。

WREN 位置 1 允许写操作。上电时，WREN 位清零。在正常工作状态下，当写操作被 MCLR 复位或 WDT 超时复位中断时，WRERR 位置 1。在这些情形下，复位后，用户可以检查 WRERR 位并重写相应的单元。地址寄存器 NVMADR 保持不变。

注： 当写操作完成时，IFS0 寄存器中的中断标志位 NVMIF 置 1。它必须在软件中清零。

6.1 读数据 EEPROM

TBLRD 指令读取当前程序字地址中的一个字。本示例采用 W0 作为指向数据 EEPROM 的指针。如例 6-1 所示，结果保存在寄存器 W4 中。

例 6-1: 读数据 EEPROM

```
MOV    #LOW_ADDR_WORD,W0    ; Init Pointer
MOV    #HIGH_ADDR_WORD,W1
MOV    W1,TBLPAG
TBLRD  [ W0 ], W4           ; read data EEPROM
```

dsPIC30F3014/4013

6.2 擦除数据 EEPROM

6.2.1 擦除数据 EEPROM 中的一块

要擦除数据 EEPROM 中的一块，NVMADRU 和 NVMADR 寄存器必须首先指向要擦除的存储块。配置 NVMCON 以擦除数据 EEPROM 中的一块并将 NVMCON 寄存器中的 WR 和 WREN 位置 1。将 WR 位置 1 启动擦除，如例 6-2 所示。

6.2.2 擦除数据 EEPROM 中的一个字

NVMADRU 和 NVMADR 寄存器必须指向一个存储块。选择一块数据闪存并将 NVMCON 寄存器中的 WR 和 WREN 位置 1。将 WR 位置 1 启动擦除，如例 6-3 所示。

例 6-2: 擦除数据 EEPROM 块

```
; Select data EEPROM block, WR, WREN bits
MOV    #4045,W0
MOV    W0,NVMCON           ; Initialize NVMCON SFR

; Start erase cycle by setting WR after writing key sequence
DISI   #5                  ; Block all interrupts with priority <7 for
                           ; next 5 instructions
MOV    #0x55,W0           ;
MOV    W0,NVMKEY          ; Write the 0x55 key
MOV    #0xAA,W1           ;
MOV    W1,NVMKEY          ; Write the 0xAA key
BSET   NVMCON,#WR        ; Initiate erase sequence
NOP
NOP

; Erase cycle will complete in 2mS. CPU is not stalled for the Data Erase Cycle
; User can poll WR bit, use NVMIF or Timer IRQ to determine erasure complete
```

例 6-3: 擦除数据 EEPROM 字

```
; Select data EEPROM word, WR, WREN bits
MOV    #4044,W0
MOV    W0,NVMCON

; Start erase cycle by setting WR after writing key sequence
DISI   #5                  ; Block all interrupts with priority <7 for
                           ; next 5 instructions
MOV    #0x55,W0           ;
MOV    W0,NVMKEY          ; Write the 0x55 key
MOV    #0xAA,W1           ;
MOV    W1,NVMKEY          ; Write the 0xAA key
BSET   NVMCON,#WR        ; Initiate erase sequence
NOP
NOP

; Erase cycle will complete in 2mS. CPU is not stalled for the Data Erase Cycle
; User can poll WR bit, use NVMIF or Timer IRQ to determine erasure complete
```

6.3 写入数据 EEPROM

要写入 EEPROM 数据单元，必须执行以下步骤：

1. 擦除数据 EEPROM 字。
 - a) 选择对数据 EEPROM 执行字擦除操作，并将 NVMCON 寄存器中的 WREN 位置 1。
 - b) 将要被擦除的字的地址写入 NVMADR。
 - c) 允许 NVM 中断（可选）。
 - d) 将 0x55 写入 NVMKEY。
 - e) 将 0xAA 写入 NVMKEY。
 - f) 将 WR 位置 1。这将启动擦除周期。
 - g) 查询 NVMIF 位或等待 NVMIF 中断。
 - h) WR 位在擦除周期结束时清零。
2. 将数据字写入数据 EEPROM 写锁存器。
3. 将一个数据字编程入数据 EEPROM。
 - a) 选择对数据 EEPROM 执行字编程操作，并将 NVMCON 寄存器中的 WREN 位置 1。
 - b) 允许 NVM 写完成中断（可选）。
 - c) 将 0x55 写入 NVMKEY。
 - d) 将 0xAA 写入 NVMKEY。
 - e) 将 WR 位置 1。这将启动编程周期。
 - f) 查询 NVMIF 位或等待 NVM 中断。
 - g) WR 位在写周期结束时清零。

若未完全遵照上述步骤（将 0x55 写入 NVMKEY，将 0xAA 写入 NVMCON，然后将 WR 位置 1）写入每个字，则无法启动写操作。强烈建议在该代码段执行期间禁止中断。

此外，NVMCON 中的 WREN 位必须置 1 以能使写操作。该机制阻止了由于执行不期望的代码而导致的对数据 EEPROM 的意外写入。除更新 EEPROM 外，WREN 位应始终保持清零。WREN 位不会由硬件清零。

写序列开始以后，将 WREN 位清零不会影响当前的写周期。除非 WREN 位置 1，否则无法将 WR 位置 1。必须在上一条指令中将 WREN 位置 1。无法在同一条指令中将 WR 位和 WREN 位清零。

在写周期完成时，WR 位由硬件清零，并且非易失性存储器写操作完成中断标志位（NVMIF）置 1。用户可以允许写操作完成中断，也可以查询 NVMIF 位。NVMIF 必须在软件中清零。

6.3.1 写入数据 EEPROM 中的一个字

一旦用户擦除了要编程的字，就可使用表写指令写入一个写锁存器，如例 6-4 所示。

6.3.2 写入数据 EEPROM 中的一块

要写入数据 EEPROM 中的一块，首先要写入所有 16 个锁存器，然后设置 NVMCON 寄存器并编程相应的块，如例 6-5 所示。

例 6-4: 数据 EEPROM 字写操作

```

; Point to data memory
MOV     #LOW_ADDR_WORD,W0           ; Init pointer
MOV     #HIGH_ADDR_WORD,W1
MOV     W1,TBLPAG
MOV     #LOW(WORD),W2              ; Get data
TBLWTL  W2,[ W0 ]                  ; Write data
; The NVMADR captures last table access address
; Select data EEPROM for 1 word op
MOV     #0x4004,W0
MOV     W0,NVMCON

; Operate key to allow write operation
DISI    #5                          ; Block all interrupts with priority <7 for
                                        ; next 5 instructions

MOV     #0x55,W0                    ; Write the 0x55 key
MOV     W0,NVMKEY
MOV     #0xAA,W1                    ; Write the 0xAA key
MOV     W1,NVMKEY
BSET    NVMCON,#WR                  ; Initiate program sequence
NOP
NOP

; Write cycle will complete in 2mS. CPU is not stalled for the Data Write Cycle
; User can poll WR bit, use NVMIF or Timer IRQ to determine write complete

```

dsPIC30F3014/4013

例 6-5: 数据 EEPROM 块写操作

```
MOV      #LOW_ADDR_WORD,W0 ; Init pointer
MOV      #HIGH_ADDR_WORD,W1
MOV      W1,TBLPAG
MOV      #data1,W2          ; Get 1st data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data2,W2          ; Get 2nd data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data3,W2          ; Get 3rd data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data4,W2          ; Get 4th data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data5,W2          ; Get 5th data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data6,W2          ; Get 6th data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data7,W2          ; Get 7th data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data8,W2          ; Get 8th data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data9,W2          ; Get 9th data
TBLWTL  W2,[ W0]++         ; write data
MOV      #data10,W2         ; Get 10th data
TBLWTL  W2,[ W0]++        ; write data
MOV      #data11,W2         ; Get 11th data
TBLWTL  W2,[ W0]++        ; write data
MOV      #data12,W2         ; Get 12th data
TBLWTL  W2,[ W0]++        ; write data
MOV      #data13,W2         ; Get 13th data
TBLWTL  W2,[ W0]++        ; write data
MOV      #data14,W2         ; Get 14th data
TBLWTL  W2,[ W0]++        ; write data
MOV      #data15,W2         ; Get 15th data
TBLWTL  W2,[ W0]++        ; write data
MOV      #data16,W2         ; Get 16th data
TBLWTL  W2,[ W0]++        ; write data. The NVMADR captures last table access address.
MOV      #0x400A,W0         ; Select data EEPROM for multi word op
MOV      W0,NVMCON          ; Operate Key to allow program operation
DISI    #5                  ; Block all interrupts with priority <7 for
                                ; next 5 instructions

MOV      #0x55,W0           ; Write the 0x55 key
MOV      W0,NVMKEY
MOV      #0xAA,W1
MOV      W1,NVMKEY          ; Write the 0xAA key
BSET    NVMCON,#WR         ; Start write cycle
NOP
NOP
```

6.4 写校验

根据具体的应用，一般要求对写入存储器的值比照原始值进行校验，这样才是好的编程习惯。当写入次数过多已接近规范极限值时，就应该采用写校验。

6.5 防止误写的保护措施

在有些情况下，用户可能不希望向数据 EEPROM 存储器写入数据。为了防止误写 EEPROM，器件内建了各种保护机制。上电时，WREN 位清零；上电延时定时器也可用来防止误写 EEPROM。

在欠压、电源毛刺或软件故障期间，写操作启动序列和 WREN 位可共同防止误写操作的发生。

7.0 I/O 端口

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

所有的器件引脚 (VDD、VSS、MCLR 和 OSC1/CLKI 除外) 均由外设和并行 I/O 端口共用。

所有 I/O 输入端口都采用施密特触发器输入，以增强抗噪声能力。

7.1 并行 I/O (PIO) 端口

当使能某外设并且该外设正有效驱动相关引脚时，将禁止此引脚的通用输出功能。可以读该 I/O 引脚，但并行端口位的输出驱动器将被禁止。若使能某外设但不由该外设驱动引脚时，则相应引脚可由端口驱动。

所有的端口引脚都有三个与端口引脚的操作直接相关的寄存器。数据方向寄存器 (TRISx) 决定引脚是输入引脚还是输出引脚。如果数据方向位为 1，则为输入引脚。复位以后，所有端口引脚被定义为输入引脚。

可以直接读写锁存器 (LATx)。读取端口 (PORTx) 时，读的是端口引脚的值；而写入端口引脚时，写入的是相应的锁存器 (LATx)。

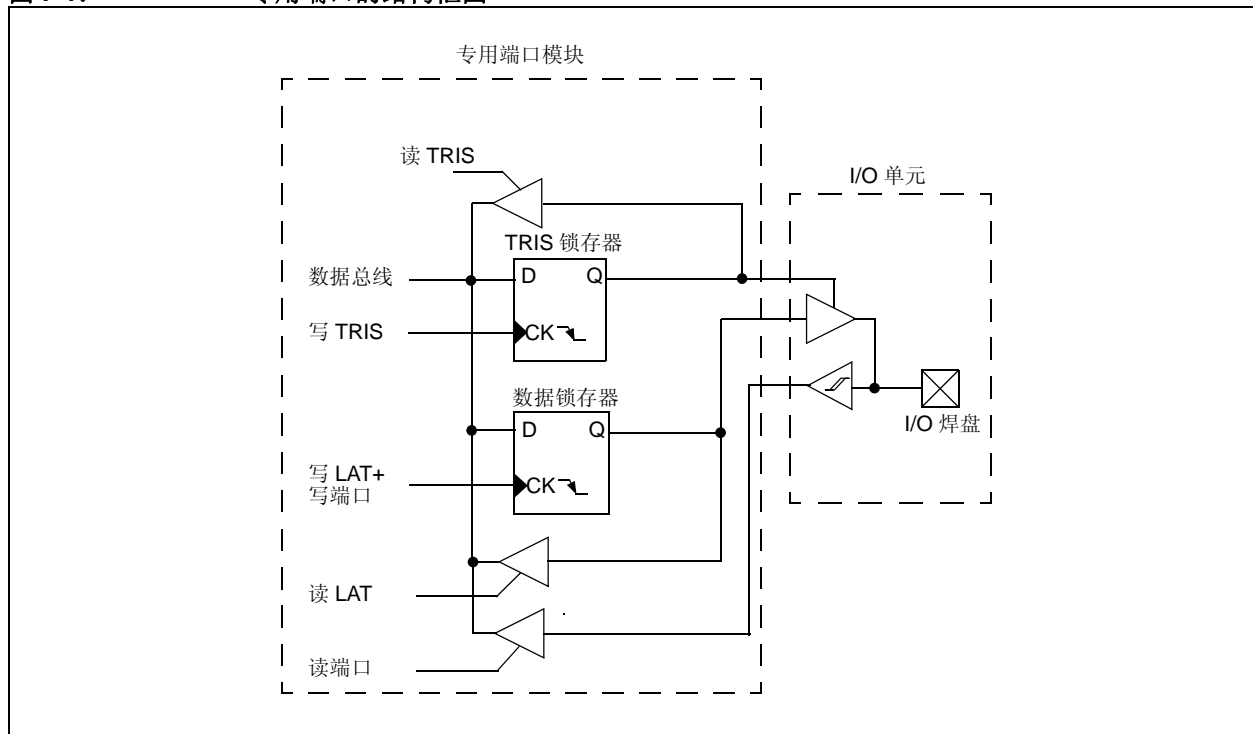
对特定器件无效的任何位及其相关的数据和控制寄存器均被禁止，这表明读相应的 LATx 和 TRISx 寄存器以及端口引脚将得到零。

当一个只用作输入的引脚与另一个外设或功能复用，由于没有其他与其竞争的输出源，它仍然被视作一个专用端口。INT4 引脚就是这样一个例子。

通常，与外设复用某个引脚的并行 I/O (PIO) 端口服从于该外设。外设的输出缓冲器数据和控制信号被提供给一对多路开关，由这对多路开关选择由外设还是相关的端口使用 I/O 焊盘单元的输出数据和控制信号。图 7-2 给出了端口与其他外设及连接到外设的相关 I/O 单元 (焊盘) 的共用情况。表 7-1 显示了与共用端口 PORTB 到 PORTF 相关的寄存器的格式。

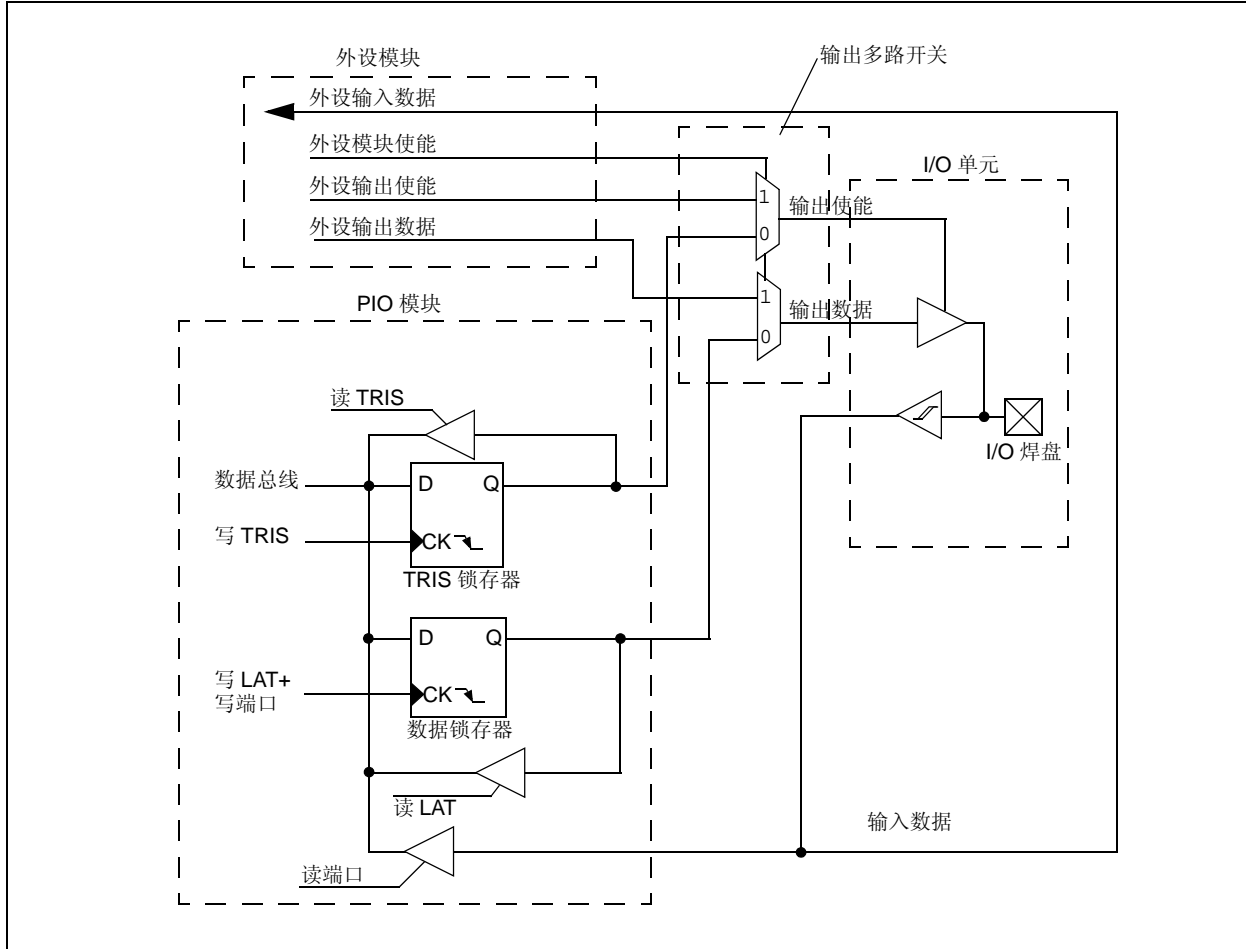
注： 不同器件所用的实际位也将不同。

图 7-1: 专用端口的结构框图



dsPIC30F3014/4013

图 7-2: 共享端口的结构框图



7.2 配置模拟端口引脚

ADPCFG 和 TRIS 寄存器控制 A/D 端口引脚的操作。若希望端口引脚为模拟输入引脚，则必须将相应的 TRIS 位置 1（输入）。如果将 TRIS 位清零（输出），则将转换数字输出电平（VOH 或 VOL）。

读端口寄存器时，所有配置为模拟输入通道的引脚均读为 0（低电平）。

配置为数字输入的引脚将不对模拟输入信号进行转换。对任何定义为数字输入的引脚（包括 ANx 引脚）施加模拟电平可能导致输入缓冲器的电流消耗超出规范值。

7.2.1 I/O 端口写 / 读时序

改变端口方向，或对端口执行写操作然后对同一端口执行读操作之间需要一个指令周期。通常在这段时间内执行一条 NOP 指令。

例 7-1: 端口写 / 读示例

```

MOV    0xFF00, W0 ; Configure PORTB<15:8>
                ; as inputs
MOV    W0, TRISB ; and PORTB<7:0> as outputs
NOP                    ; additional instruction
                    ; cycle
BTSS   PORTB, #11 ; bit test RB11 and skip if
                    ; set
    
```

表 7-1: dsPIC30F3014/4013 端口寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
TRISA	02C0	—	—	—	—	TRISA11	—	—	—	—	—	—	—	—	—	—	—	0000 1000 0000 0000
PORTA	02C2	—	—	—	—	RA11	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
LATA	02C4	—	—	—	—	LATA11	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
TRISB	02C6	—	—	—	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	0001 1111 1111 1111
PORTB	02C8	—	—	—	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	0000 0000 0000 0000
LATB	02CB	—	—	—	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	0000 0000 0000 0000
TRISC	02CC	TRISC15	TRISC14	TRISC13	—	—	—	—	—	—	—	—	—	—	—	—	—	1110 0000 0000 0000
PORTC	02CE	RC15	RC14	RC13	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
LATC	02D0	LATC15	LATC14	LATC13	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
TRISD	02D2	—	—	—	—	—	—	TRISD9	TRISD8	—	—	—	—	TRISD3	TRISD2	TRISD1	TRISD0	0000 0011 0000 1111
PORTD	02D4	—	—	—	—	—	—	RD9	RD8	—	—	—	—	RD3	RD2	RD1	RD0	0000 0000 0000 0000
LATD	02D6	—	—	—	—	—	—	LATD9	LATD8	—	—	—	—	LATD3	LATD2	LATD1	LATD0	0000 0000 0000 0000
TRISF	02DE	—	—	—	—	—	—	—	—	—	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	0000 0000 0111 1111
PORTF	02E0	—	—	—	—	—	—	—	—	—	RF6	RF5	RF4	RF3	RF2	RF1	RF0	0000 0000 0000 0000
LATF	02E2	—	—	—	—	—	—	—	—	—	LATF6	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	0000 0000 0000 0000

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

7.3 输入电平变化通知模块

输入电平变化通知模块赋予 dsPIC30F 器件向处理器产生中断请求的能力，以响应某些选定输入引脚上的电平变化（Change-Of-State, COS）。即便在禁止时钟时的休眠模式下，该模块也可检测到输入电平的状态变化。最多可选择（使能）10 个外部信号（CN0 到 CN7，CN17 和 CN18），用以在状态变化时产生中断请求。

表 7-2: dsPIC30F3014 (BIT 15-0) 的输入电平变化通知寄存器映射⁽¹⁾

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
CNEN1	00C0	—	—	—	—	—	—	—	—	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000 0000 0000 0000
CNEN2	00C2	—	—	—	—	—	—	—	—	—	—	—	—	—	CN18IE	CN17IE	—	0000 0000 0000 0000
CNPU1	00C4	—	—	—	—	—	—	—	—	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000 0000 0000 0000
CNPU2	00C6	—	—	—	—	—	—	—	—	—	—	—	—	—	CN18PUE	CN17PUE	—	0000 0000 0000 0000

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

8.0 中断

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

dsPIC30F 传感器和通用系列具有最多 41 个中断源和 4 个处理器异常（陷阱），必须根据优先级机制进行仲裁。

CPU 负责读取中断向量表（Interrupt Vector Table, IVT）并将中断向量中的地址传递给程序计数器。将中断向量从程序数据总线，经由程序计数器输入端上的 24 位多路开关传递到程序计数器。

中断向量表（IVT）和备用中断向量表（Alternate Interrupt Vector Table, AIVT）位于程序存储器（0x000004）的开头部分。图 8-1 给出了 IVT 和 AIVT。

中断控制器负责预处理中断和处理器异常，然后再将中断和异常送入处理器内核。使用集中的特殊功能寄存器允许和控制外设中断和陷阱，并设置这些中断和陷阱的优先级：

- IFS0<15:0>、IFS1<15:0> 和 IFS2<15:0>
所有中断请求标志位都位于这三个寄存器中。这些标志位被各自的外设信号或外部信号置 1，并通过软件清零。
- IEC0<15:0>、IEC1<15:0> 和 IEC2<15:0>
所有中断允许控制位都位于这三个寄存器中。这些控制位分别用于允许外设或外部信号的中断。
- IPC0<15:0>...IPC10<7:0>
与 41 个中断相关联的可由用户分配的优先级可集中在这 11 个寄存器中设置。
- IPL<3:0>
CPU 的当前优先级被存储在 IPL 位中。IPL<3> 位于 CORCON 寄存器中，而 IPL<2:0> 则位于处理器内核的状态寄存器中（SR）。

- INTCON1<15:0> 和 INTCON2<15:0>
全局中断控制功能来源于这两个寄存器。INTCON1 寄存器包含处理器异常的控制位和状态标志位。INTCON2 寄存器控制外部中断请求信号的行为和备用中断向量的使用。

注： 当中断条件产生时，无论相关中断允许位的状态如何，中断标志位都会置 1。用户软件要确保在允许中断之前将相应的中断标志位清零。

用户可以通过 IPCx 寄存器给所有中断源分配 1 至 7 级优先级。每个中断源都与一个中断向量相关联，如表 8-1 所示。7 级和 1 级分别代表可屏蔽中断的最高优先级和最低优先级。

注： 给某个中断源分配优先级 0 就相当于禁止该中断。

如果将 NSTDIS 位（INTCON1<15>）置 1，就可以禁止中断嵌套。因此在处理一个中断时，即使产生新中断的优先级高于正在处理的中断的优先级，新的中断还是不会被处理。

注： 只要将 NSTDIS 位置 1 就只能读而不能写 IPL 位。

某些中断具有专用的控制位来控制边沿触发中断、电平触发中断和电平变化中断等特性。在产生中断的外设模块中控制这些特性。

DISI 指令可用于禁止对周期内优先级小于等于 6 的中断进行处理，在此期间 DISI 位（INTCON2<14>）保持置 1。

处理一个中断时，将程序存储器中与中断相对应的向量单元中的地址装载到 PC。IVT（见表 8-1）中有 63 个不同的向量。这些向量位于程序存储器中地址从 0x000004 到 0x0000FE 的单元中（见表 8-1）。这些单元中包含 24 位地址。为了保证可靠性，在正常执行期间，如果 PC 试图取位于向量单元中的任何字就会产生地址错误陷阱。这样可以避免执行随机数据，而此随机数据是由于意外递减 PC 使其位于向量空间、意外将数据空间地址映射到向量空间中或在到达实现的程序存储空间末尾后 PC 返回 0x000000 而导致的。执行跳转到向量空间的 GOTO 指令也会产生地址错误陷阱。

dsPIC30F3014/4013

8.1 中断优先级

每个中断源的用户可分配中断优先级（IP<2:0>）位均位于 IPCx 寄存器中每个半字节的低 3 位。每个半字节的 bit 3 均不使用，读为 0。用户通过这些位为特定的中断分配优先级。

注： 可由用户分配的优先级为 0（最低优先级）至 7（最高优先级）。

由于多个中断请求源可能被分配给一个特定的用户分配优先级，有一个可以提供在给定优先级内再次分配优先级的方法，称为“自然顺序优先级”，从而最终确定中断请求源的优先级。

自然顺序优先级由中断在向量表中的位置确定，并只在具有相同的用户分配优先级的多个中断同时处于等待状态时才会影响中断操作。

表 8-1 和表 8-2 分别列举了与 dsPIC30F3014 和 dsPIC30F4013 器件相关的中断编号、相应中断源以及向量编号。

注 1： 自然顺序优先级机制为 0（最高优先级）至 53（最低优先级）。

注 2： 自然顺序优先级的编号与 INT 相同。

用户可以为每个中断源分配 7 个优先级之一，这表示用户可以给低自然优先级的中断分配非常高的总优先级，例如，可以给可编程低电压检测（Programmable Low-Voltage Detect, PLVD）分配优先级 7。还可以给 INTO（外部中断 0）分配优先级 1，从而将其的有效优先级设置的非常低。

表 8-1: dsPIC30F3014 中断向量表

中断编号	向量编号	中断源
最高自然顺序优先级		
0	8	INT0 —— 外部中断 0
1	9	IC1 —— 输入捕捉 1
2	10	OC1 —— 输出比较 1
3	11	T1 —— Timer1
4	12	IC2 —— 输入捕捉 2
5	13	OC2 —— 输出比较 2
6	14	T2 —— Timer2
7	15	T3 —— Timer3
8	16	SPI1
9	17	U1RX —— UART1 接收器
10	18	U1TX —— UART1 发送器
11	19	ADC —— ADC 转换完成
12	20	NVM —— NVM 写操作完成
13	21	SI2C —— I ² C™ 从中断
14	22	MI2C —— I ² C 主中断
15	23	输入变化中断
16	24	INT1 —— 外部中断 1
17-22	25-30	保留
23	31	INT2 —— 外部中断 2
24	32	U2RX —— UART2 接收器
25	33	U2TX —— UART2 发送器
26	34	保留
27	35	C1 —— CAN1 的组合 IRQ
28-41	36-49	保留
42	50	LVD —— 低电压检测
43-53	51-61	保留
最低自然顺序优先级		

表 8-2: dsPIC30F4013 中断向量表

中断编号	向量编号	中断源
最高自然顺序优先级		
0	8	INT0 —— 外部中断 0
1	9	IC1 —— 输入捕捉 1
2	10	OC1 —— 输出比较 1
3	11	T1 —— Timer1
4	12	IC2 —— 输入捕捉 2
5	13	OC2 —— 输出比较 2
6	14	T2 —— Timer2
7	15	T3 —— Timer3
8	16	SPI1
9	17	U1RX —— UART1 接收器
10	18	U1TX —— UART1 发送器
11	19	ADC —— ADC 转换完成
12	20	NVM —— NVM 写操作完成
13	21	SI2C —— I ² C™ 从中断
14	22	MI2C —— I ² C 主中断
15	23	输入电平变化中断
16	24	INT1 —— 外部中断 1
17	25	IC7 —— 输入捕捉 7
18	26	IC8 —— 输入捕捉 8
19	27	OC3 —— 输出比较 3
20	28	OC4 —— 输出比较 4
21	29	T4 —— Timer4
22	30	T5 —— Timer5
23	31	INT2 —— 外部中断 2
24	32	U2RX —— UART2 接收器
25	33	U2TX —— UART2 发送器
26	34	保留
27	35	C1 —— CAN1 的组合 IRQ
28-40	36-48	保留
41	49	DCI —— 编解码器传输完成
42	50	LVD —— 低电压检测
43-53	51-61	保留
最低自然顺序优先级		

8.2 复位的过程

因为复位过程不会使用中断控制器，所以复位并不是真正的异常。复位时处理器会初始化寄存器作为响应，这会强制 PC 为零。然后处理器会在地址为 0x000000 的单元处开始执行程序。存储在程序存储器第一个单元中的 GOTO 指令后紧跟要跳转到的目标地址。处理器执行 GOTO 指令跳转到指定的地址，然后在指定的目标（开始）地址开始操作。

8.2.1 复位源

除了外部复位和上电复位（POR），以下错误源也会使程序跳转到复位向量处：

- 看门狗超时：
看门狗定时器超时，表示处理器不再执行正确的代码流。
- 未初始化的 W 寄存器陷阱：
试图将一个未初始化的 W 寄存器用作地址指针会导致复位。
- 非法指令陷阱：
试图执行任何未用的操作码会导致产生非法的指令陷阱。注意，如果在执行非法指令前由于指令流的改变而造成该指令被清除，则取该指令不会产生非法指令陷阱。
- 欠压复位（BOR）：
检测到可能导致故障的器件电源的瞬时下降。
- 陷阱锁定：
多个陷阱条件同时发生会导致复位。

8.3 陷阱

可以将陷阱视为表示软件错误或硬件错误的不可屏蔽的中断，它必须符合图 8-1 给出的预定义优先级。陷阱旨在为用户提供一种在调试和运行应用时纠正错误操作的方法。

注： 如果用户不想在出现陷阱错误条件时采取纠正措施，则必须将一个只包含 `RESET` 指令的默认处理程序的地址装入这些向量中。另一方面，如果调用其中一个包含无效地址的向量，则将产生地址错误陷阱。

注意，许多陷阱条件只能在发生时才能被检测到。其结果是存在问题的指令会在陷阱异常处理前被允许执行。如果用户选择从错误恢复，则必须纠正导致陷阱的错误操作的结果。

陷阱有 8 个固定的优先级：优先级 8 到优先级 15，这意味着在处理陷阱时，`IPL3` 始终为 1。

如果用户当前不在执行陷阱操作，并且将 `IPL<3:0>` 位设置为 0111（优先级 7），那么将禁止所有中断，但仍可处理陷阱。

8.3.1 陷阱源

以下陷阱以优先级递增的顺序给出。但是，由于所有陷阱都能被嵌套，所以优先级不起什么作用。

算术错误陷阱：

在以下情况下执行算术错误陷阱：

1. 试图执行除数为零的除法运算，除法运算会在一个循环的边界处停止，且产生陷阱。
2. 允许算术错误陷阱时，当对累加器 A 或 B 执行算术运算导致累加器从 bit 31 溢出，并且没有使用累加器警戒位时，将产生算术错误陷阱。
3. 允许算术错误陷阱时，当对累加器 A 或 B 执行算术运算导致累加器从 bit 39 发生灾难性溢出，并且禁止所有把饱和时，将产生算术错误陷阱。
4. 如果在移位指令中指定的移位数量大于最大允许移位数量，会产生陷阱。

地址错误陷阱：

当发生以下任何一种情况时就会产生陷阱：

1. 尝试访问不对齐的数据字。
2. 尝试从未实现的数据存储单元中取数据。
3. 尝试访问未实现的程序存储单元中的数据。
4. 尝试从向量空间取指。

注： 在 MAC 类指令中数据空间被分为 X 和 Y 数据空间，未实现的 X 空间包含所有的 Y 空间，未实现的 Y 空间包含所有的 X 空间。

5. 执行“`BRA #literal`”指令或“`GOTO #literal`”指令，其中 `literal` 是一个未实现的程序存储地址。
6. 在修改 PC 使其指向未实现的程序存储地址单元后执行指令。可通过将一个值装入堆栈并执行 `RETURN` 指令修改 PC。

堆栈错误陷阱：

在以下情况下产生该陷阱：

1. 将一个大于写入 `SPLIM` 寄存器中的极限值的值装入堆栈指针，极限值可由用户设置（堆栈上溢）。
2. 将一个小于 0x0800 的值装入堆栈指针（堆栈下溢）。

振荡器故障陷阱：

如果外部振荡器发生故障，工作要依赖内部备用 RC 振荡器运行就会产生该陷阱。

8.3.2 硬陷阱和软陷阱

多个陷阱可能在同一周期内出现（例如，对一个已溢出的地址单元执行不对齐的字堆栈写操作）。在这种情况下，使用图 8-2 中所示的固定优先级，这种情况要求用户检查是否有其余陷阱等待处理，以彻底更正故障。

“软”陷阱包含从优先级 8 到优先级 11（含优先级 8 和 11）的异常。算术错误陷阱（优先级 11）属于该类陷阱。

“硬”陷阱包含从优先级 12 到优先级 15（含优先级 12 和 15）的异常。地址错误（优先级 12）、堆栈错误（优先级 13）和振荡器错误（优先级 14）陷阱属于该类陷阱。

8.5 备用向量表

在程序存储器中，备用中断向量表（IVT）位于中断向量表（AIVT）之后，如图 8-1 所示。由 INTCON2 寄存器中的 ALTIVT 位提供对备用向量表的访问。如果 ALTIVT 位置 1，所有中断和异常处理将使用备用向量而不是默认的向量。备用向量的构成方式与默认向量相同。AIVT 支持仿真和调试功能，它提供了一种不需要将中断向量再编程就可以在应用和支持环境之间切换的方法。此功能也使用户可在不同应用之间切换，以便评估运行时各种软件算法。

如果不需要使用 AIVT，那么分配到 AIVT 的程序存储区就可以用于其他用途。AIVT 不受保护，用户可以自由对其进行编程。

8.6 快速现场保护

使用影子寄存器可以实现现场保护。SR 中的 DC、N、OV、Z 和 C 位以及寄存器 W0 到 W3 提供影子寄存器。影子寄存器只有一级深。只可使用 PUSH.S 和 POP.S 指令访问影子寄存器。

当处理器转向中断向量处执行程序时，可以使用 PUSH.S 指令将上述寄存器的当前值保存到各自的影子寄存器中。

如果一个具有确定优先级的 ISR 使用 PUSH.S 和 POP.S 指令进行快速现场保护，则优先级高于它的 ISR 就不能再使用相同的指令。如果较高优先级的 ISR 要使用快速现场保护，则用户必须在较低优先级中断期间将关键寄存器的值保存到软件中。

8.7 外部中断请求

该中断控制器支持最多 5 个外部中断请求信号 INT0-INT4。这些中断引脚以边沿方式触发中断，要求从低到高或从高到低的电平跳变来产生中断请求。INTCON2 寄存器有三个位：INT0EP 到 INT2EP，用来选择边沿检测电路的极性。

8.8 从休眠和空闲模式唤醒

产生中断时，如果器件处于休眠模式或空闲模式，则中断控制器可以将处理器从休眠模式或空闲模式唤醒。

如果中断控制器接收到一个允许的具有足够优先级的中断请求，则该中断请求就会交给处理器。同时，处理器将从休眠或空闲模式唤醒，然后开始执行处理该中断请求所需的 interrupt 服务程序（ISR）。

表 8-3: dsPIC30F3014 中断控制器寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态	
INTCON1	0080	NSTDIS	—	—	—	—	OVATE	OVATE	COVTE	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000 0000 0000 0000	
INTCON2	0082	ALTIPT	DISI	—	—	—	—	—	—	—	—	—	—	—	INT2EP	INT1EP	INT0EP	0000 0000 0000 0000	
IFS0	0084	CNIF	MI2CIF	SI2CIF	NVMIF	ADIF	U1TXIF	U1RXIF	SPI1IF	T3IF	T2IF	OC2IF	IC2IF	T1IF	OC1IF	IC1IF	INT0IF	0000 0000 0000 0000	
IFS1	0086	—	—	—	—	C1IF	—	U2TXIF	U2RXIF	INT2IF	—	—	—	—	—	—	INT1IF	0000 0000 0000 0000	
IFS2	0088	—	—	—	—	—	LVDIF	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000	
IEC0	008C	CNIE	MI2CIE	SI2CIE	NVMIE	ADIE	U1TXIE	U1RXIE	SPI1IE	T3IE	T2IE	OC2IE	IC2IE	T1IE	OC1IE	IC1IE	INT0IE	0000 0000 0000 0000	
IEC1	008E	—	—	—	—	C1IE	—	U2TXIE	U2RXIE	INT2IE	—	—	—	—	—	—	INT1IE	0000 0000 0000 0000	
IEC2	0090	—	—	—	—	—	LVDIE	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000	
IPC0	0094	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	INT0IP<2:0>			0100 0100 0100 0100	
IPC1	0096	—	T31P<2:0>			—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			0100 0100 0100 0100	
IPC2	0098	—	ADIP<2:0>			—	U1TXIP<2:0>			—	U1RXIP<2:0>			—	SPI1IP<2:0>			0100 0100 0100 0100	
IPC3	009A	—	CNIP<2:0>			—	MI2CIP<2:0>			—	SI2CIP<2:0>			—	NVMIP<2:0>			0100 0100 0100 0100	
IPC4	009C	—	—	—	—	—	—	—	—	—	—	—	—	—	INT1IP<2:0>			0100 0100 0100 0100	
IPC5	009E	—	INT2IP<2:0>			—	—	—	—	—	—	—	—	—	—	—	—	—	0100 0100 0100 0100
IPC6	00A0	—	C1IP<2:0>			—	—	—	—	—	U2TXIP<2:0>			—	U2RXIP<2:0>			0100 0100 0100 0100	
IPC7	00A2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0100 0100 0100 0100	
IPC8	00A4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0100 0100 0100 0100	
IPC9	00A6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0100 0100 0100	
IPC10	00A8	—	—	—	—	—	LVDIP<2:0>			—	DCIIP<2:0>			—	—	—	—	0000 0100 0100 0000	

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 8-4: dsPIC30F4013 中断控制器寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
INTCON1	0080	NSTDIS	—	—	—	—	OVATE	OVATE	COVTE	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000 0000 0000 0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	—	—	INT2EP	INT1EP	INT0EP	0000 0000 0000 0000
IFS0	0084	CNIF	MI2CIF	SI2CIF	NVMIF	ADIF	U1TXIF	U1RXIF	SPI1IF	T3IF	T2IF	OC2IF	IC2IF	T1IF	OC1IF	IC1IF	INT0IF	0000 0000 0000 0000
IFS1	0086	—	—	—	—	C1IF	—	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	IC8IF	IC7IF	INT1IF	0000 0000 0000 0000
IFS2	0088	—	—	—	—	—	LVDIF	DCIIF	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
IEC0	008C	CNIE	MI2CIE	SI2CIE	NVMIE	ADIE	U1TXIE	U1RXIE	SPI1IE	T3IE	T2IE	OC2IE	IC2IE	T1IE	OC1IE	IC1IE	INT0IE	0000 0000 0000 0000
IEC1	008E	—	—	—	—	C1IE	—	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	IC8IE	IC7IE	INT1IE	0000 0000 0000 0000
IEC2	0090	—	—	—	—	—	LVDIE	DCIIE	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
IPC0	0094	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	INT0IP<2:0>			0100 0100 0100 0100
IPC1	0096	—	T31P<2:0>			—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			0100 0100 0100 0100
IPC2	0098	—	ADIP<2:0>			—	U1TXIP<2:0>			—	U1RXIP<2:0>			—	SPI1IP<2:0>			0100 0100 0100 0100
IPC3	009A	—	CNIP<2:0>			—	MI2CIP<2:0>			—	SI2CIP<2:0>			—	NVMIP<2:0>			0100 0100 0100 0100
IPC4	009C	—	OC3IP<2:0>			—	IC8IP<2:0>			—	IC7IP<2:0>			—	INT1IP<2:0>			0100 0100 0100 0100
IPC5	009E	—	INT2IP<2:0>			—	T5IP<2:0>			—	T4IP<2:0>			—	OC4IP<2:0>			0100 0100 0100 0100
IPC6	00A0	—	C1IP<2:0>			—	SPI2IP<2:0>			—	U2TXIP<2:0>			—	U2RXIP<2:0>			0100 0100 0100 0100
IPC7	00A2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0100 0100 0100 0100
IPC8	00A4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0100 0100 0100 0100
IPC9	00A6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0100 0100 0100
IPC10	00A8	—	—	—	—	—	LVDIP<2:0>			—	DCIIP<2:0>			—	—	—	—	0000 0100 0100 0000

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

9.0 TIMER1 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本节介绍了 16 位通用 Timer1 模块及其相关工作模式。图 9-1 所示为 16 位 Timer1 模块的简化框图。

对于定时器工作模式，下面各节详细描述了设置和控制寄存器，以及相应的框图。

Timer1 模块是一个 16 位定时器，可用作实时时钟 (Real-Time Clock, RTC) 的时间计数器，或用作自由运行的时隙定时器 / 计数器。16 位定时器有如下模式：

- 16 位定时器
- 16 位同步计数器
- 16 位异步计数器

另外，还支持下列操作特性：

- 定时器门控操作
- 可选择的预分频比设置
- CPU 空闲和休眠模式下的定时器工作
- 16 位周期寄存器匹配时或在外部门控信号下降沿产生中断

通过在 16 位 SFR T1CON 中设置相应位来确定工作模式。图 9-1 所示为 16 位定时器模块的框图。

16 位定时器模式：在 16 位定时器模式下，定时器在每个指令周期递增，直到达到周期寄存器 PR1 中预装入的匹配值，然后复位到 0，并继续计数。

当 CPU 进入空闲模式时，除非 TSIDL (T1CON<13>) 位 = 0，否则定时器停止递增。如果 TSIDL = 1，定时器模块逻辑在 CPU 空闲模式结束后恢复递增过程。

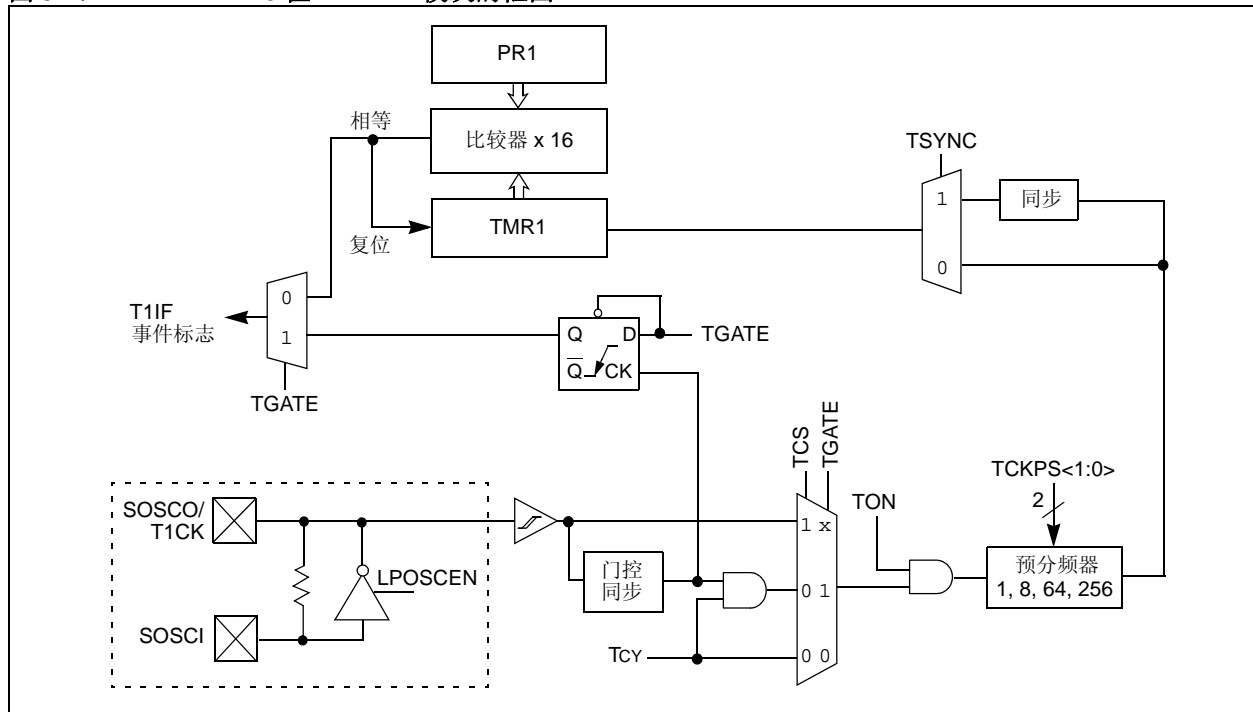
16 位同步计数器模式：在 16 位同步计数器模式下，定时器在应用的外部时钟信号（与内部相位时钟同步）的上升沿递增。定时器计数直到 PR1 中预装入的匹配值，然后复位到 0，并继续计数。

当 CPU 进入空闲模式时，除非对应的 TSIDL 位 = 0，否则定时器停止递增。如果 TSIDL = 1，定时器模块逻辑在 CPU 空闲模式结束后恢复递增过程。

16 位异步计数器模式：在 16 位异步计数器模式下，定时器在应用的外部时钟信号的上升沿递增。该定时器计数直到 PR1 中预装入的匹配值，然后复位到 0，并继续计数。

当定时器被配置为异步工作模式，且 CPU 进入空闲模式时，如果 TSIDL = 1，则定时器停止递增。

图 9-1: 16 位 TIMER1 模块的框图



dsPIC30F3014/4013

9.1 定时器门控工作

16 位定时器可以置为门控时间累加模式。当门控输入信号 (T1CK 引脚) 被设置为高电平时, 该模式允许内部 TcY 递增相应的定时器。必须将控制位 TGATE (T1CON<6>) 置 1 以启用该模式。必须使能该定时器 (TON = 1), 并将定时器时钟源设置为内部时钟源 (TCS = 0)。

当 CPU 进入空闲模式时, 除非 TSIDL = 0, 否则定时器停止递增。如果 TSIDL = 1, 定时器在 CPU 空闲模式结束后恢复递增过程。

9.2 定时器预分频器

16 位定时器的输入时钟 (Fosc/4 或外部时钟) 有 1:1、1:8、1:64 和 1:256 的预分频比选项, 可通过控制位 TCKPS<1:0> (T1CON<5:4>) 进行选择。当发生以下任何一种情况时, 预分频器计数器将会被清零:

- 写 TMR1 寄存器
- 写 T1CON 寄存器
- 器件复位, 例如 POR 和 BOR

但是, 如果禁止该定时器 (TON = 0), 由于预分频器时钟停止了, 则定时器预分频器不会复位。

写 T1CON 时, TMR1 不会被清零, 而是通过写 TMR1 寄存器将 TMR1 清零。

9.3 休眠模式下的定时器工作

在 CPU 休眠模式下, 如果发生下列情况定时器将继续工作:

- 使能定时器模块 (TON = 1), 且
- 选择外部时钟作为定时器的时钟源 (TCS = 1), 且
- 将 TSYNC 位 (T1CON<2>) 设置为逻辑 0, 从而将外部时钟源定义为异步。

当上述三种情况都成立时, 定时器将继续计数直到达到周期寄存器中的值, 然后复位到 0x0000。

当定时器和周期寄存器之间发生匹配时, 如果将相应的定时器中断允许位设置为有效, 则会产生中断。

9.4 定时器中断

16 位定时器具有在周期匹配时产生中断的能力。当定时器计数与周期寄存器相匹配时, 将 T1IF 位置 1 并且如果允许了中断还将产生中断。必须由软件将 T1IF 位清零。该定时器的中断标志位 T1IF 位于中断控制器的 IFS0 控制寄存器中。

当使能了门控时间累加模式时, 还会在门控信号 (在累加周期的末端) 的下降沿产生中断。

通过相应的定时器中断允许位 T1IE 允许中断。该定时器中断允许位位于中断控制器的 IECO 控制寄存器中。

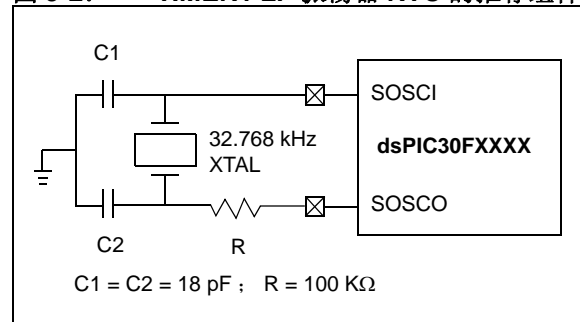
9.5 实时时钟

Timer1 在实时时钟 (RTC) 模式下工作时, 提供了时间和事件时间戳功能。RTC 的主要工作特性有:

- 使用 32 kHz LP 振荡器工作
- 8 位预分频器
- 低功耗
- 实时时钟中断

这些工作模式通过设置 T1CON 控制寄存器中的相应位来确定。

图 9-2: TIMER1 LP 振荡器 RTC 的推荐组件



9.5.1 RTC 振荡器工作

当 TON = 1、TCS = 1 且 TGATE = 0 时, 定时器在 32 kHz LP 振荡器输出信号的上升沿递增, 并在达到周期寄存器中指定的值后复位为 0。

必须将 TSYNC 位设置为逻辑 0 (异步模式) 以确保正确工作。

使能 LPOSCEN (OSCCON<1>) 将禁止正常的定时器和计数器模式, 而使能定时器溢出唤醒事件。

当 CPU 进入休眠模式时, 如果 32 kHz 外部晶振是激活的, 而且控制位未发生变化, 则 RTC 将继续工作。要使 RTC 在空闲模式下继续工作, TSIDL 位应清零。

9.5.2 RTC 中断

当发生中断事件时, 会将相应的中断标志位 T1IF 置 1, 并在允许中断时将产生中断。T1IF 位必须用软件清零。相应的定时器的中断标志位 T1IF 位于中断控制器的 IFS0 寄存器中。

通过相应的定时器中断允许位 T1IE 允许中断。该定时器中断允许位位于中断控制器的 IECO 控制寄存器中。

表 9-1: dsPIC30F3014/4013 TIMER1 寄存器映射 (1)

SFR 名称	地址	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位状态
TMR1	0100	Timer1 寄存器																uuuu uuuu uuuu uuuu
PR1	0102	周期寄存器 1																1111 1111 1111 1111
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—	0000 0000 0000 0000

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 关于寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

10.0 TIMER2/3 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本节介绍 32 位通用定时器模块 (Timer2/3) 及其相关的工作模式。图 10-1 所示为 32 位 Timer2/3 模块的简化框图。图 10-2 和图 10-3 分别给出了将 Timer2/3 配置为两个独立的 16 位定时器 Timer2 和 Timer3 的原理图。

Timer2/3 模块是一个可选工作模式的 32 位定时器 (可配置为两个 16 位定时器)。这两个定时器可被其他外设模块使用，如下：

- 输入捕捉
- 输出比较 / 简单 PWM

下面的章节提供了包括设置和控制寄存器及定时器工作模式相关框图在内的详细描述。

32 位定时器有如下模式：

- 两个独立的 16 位定时器 (Timer2 和 Timer3)，可实现所有 16 位工作模式 (异步计数器模式除外)
- 单一 32 位定时器工作模式
- 单一 32 位同步计数器工作模式

此外，还支持以下工作特性：

- ADC 事件触发
- 定时器门控操作
- 可选预分频器设置
- 空闲和休眠模式下的定时器操作
- 在 32 位周期寄存器匹配时产生中断

通过在 16 位 T2CON 和 T3CON SFR 中设置相应位确定工作模式。

对于 32 位定时器 / 计数器工作模式，Timer2 和 Timer3 分别为 32 位定时器的低有效字和高有效字。

注： 在 32 位定时器工作模式下，T3CON 中的控制位被忽略。只使用 T2CON 中的控制位来进行设置和控制。32 位定时器模块使用 Timer2 的时钟和门控输入信号，但产生中断时会将 Timer3 的中断标志位 (T3IF) 置 1，并且由 Timer3 的中断允许位 (T3IE) 允许该模块的中断。

16 位定时器模式： 在 16 位模式中，可以将 Timer2 和 Timer3 配置为两个独立的 16 位定时器。每个定时器都可以设置为 16 位定时器模式或 16 位同步计数器模式。关于这两种工作模式的详细信息，请参见第 9.0 节“Timer1 模块”。

Timer2 和 Timer3 在功能上的惟一不同是 Timer2 可以提供时钟预分频器输出的同步。这对于高频外部钟输入很有用。

32 位定时器模式： 在 32 位定时器模式中，定时器在每个指令周期递增，当其中的值与预装载到组合的 32 位周期寄存器 PR3/PR2 中的值匹配时，定时器复位为 0 后继续计数。

同步读取 Timer2/Timer3 对的 32 位时，读取最低有效字 (TMR2 寄存器) 会使读取到的值为最高有效字并使该值锁存到名为 TMR3HLD 的 16 位保持寄存器中。

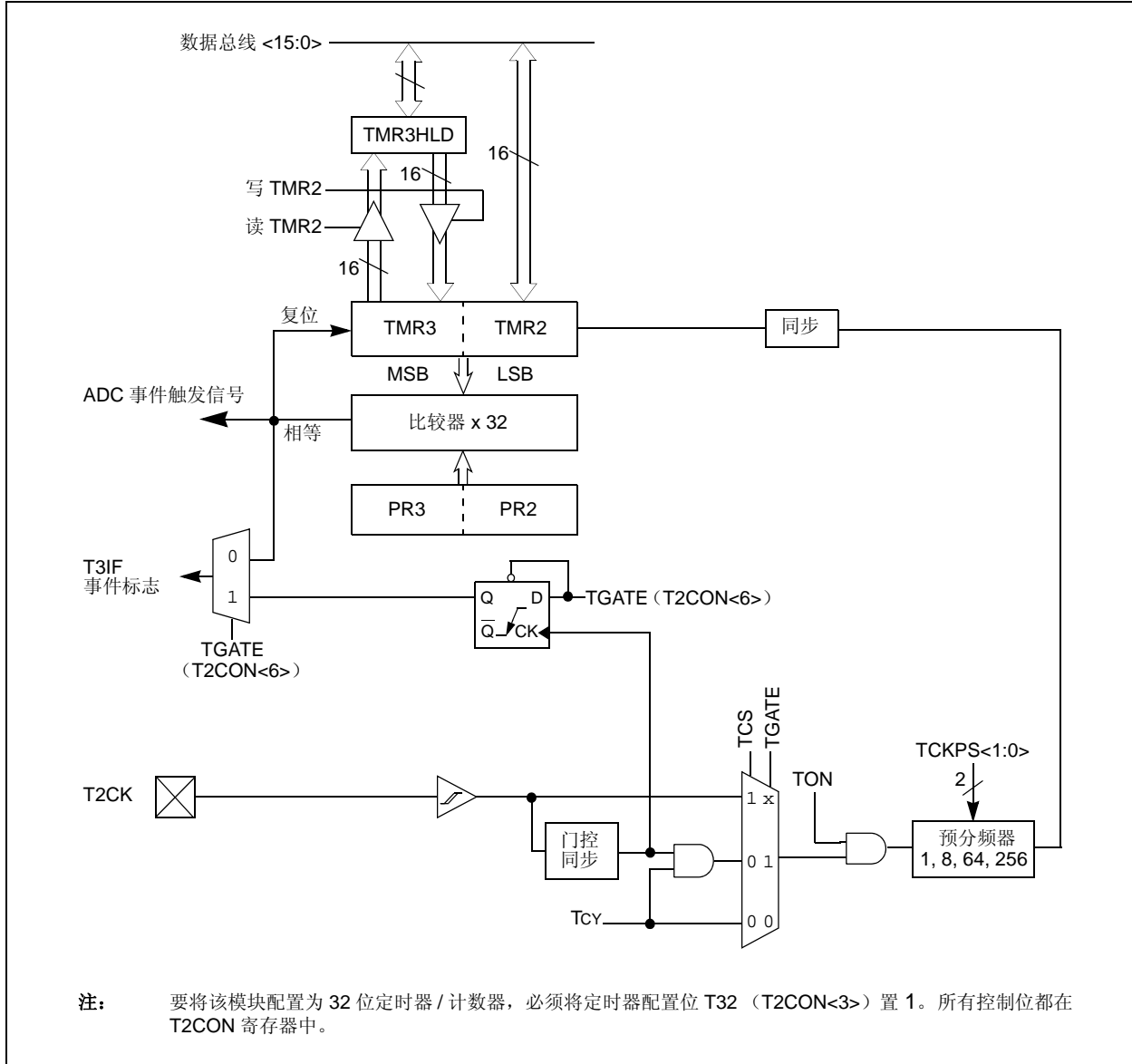
进行 32 位同步写操作时，必须首先写保持寄存器 (TMR3HLD)。写入 TMR2 寄存器之后，TMR3HLD 的内容被传输并锁存到 32 位定时器 (TMR3) 的 MSB 中。

32 位同步计数器模式： 在 32 位同步计数器模式中，定时器在与内部相位时钟同步的外部时钟信号的上升沿递增。定时器一直计数到与预装载到组合的 32 位周期寄存器 PR3/PR2 中的值匹配，然后复位为 0 并继续计数。

当定时器被配置为同步计数器工作模式，且 CPU 进入空闲模式时，除非 TSIDL (T2CON<13>) 位 = 0，否则定时器将停止递增。如果 TSIDL = 1，定时器模块逻辑将在 CPU 退出空闲模式后恢复递增。

dsPIC30F3014/4013

图 10-1: 32 位 TIMER2/3 框图



10.1 定时器门控操作

可将32位定时器设置为工作在门控时间累加模式。当门控输入信号（T2CK 引脚）被拉高时，该模式允许相应的定时器在内部 Tcy 控制下递增。控制位 TGATE（T2CON<6>）必须置 1 以使能该模式。在该模式下，由 Timer2 提供时钟。TGATE 的设置与 Timer3 无关。必须使能定时器（TON = 1）并将定时器时钟源设置为内部（TCS = 0）。

外部信号的下降沿终止计数操作但不复位定时器。要从零开始计数，用户必须先复位定时器。

10.2 ADC 事件触发信号

当 32 位定时器（TMR3/TMR2）和 32 位组合的周期寄存器（PR3/PR2）发生匹配时，Timer3 会产生一个特殊的 ADC 事件触发信号。

10.3 定时器预分频器

定时器的输入时钟（Fosc/4 或外部时钟）有 1:1、1:8、1:64 和 1:256 的预分频，具体的数值由控制位 TCKPS<1:0>（T2CON<5:4> 和 T3CON<5:4>）选择。对于 32 位定时器工作模式，起作用的时钟源为 Timer2，Timer3 的预分频操作不起作用。发生以下任何事件，都会将预分频器计数器清零：

- 写入 TMR2/TMR3 寄存器
- 写入 T2CON/T3CON 寄存器
- 器件复位，比如 POR 和 BOR

但是，如果禁止定时器（TON = 0），因为预分频时钟暂停，Timer2 预分频器将不会复位。

当写 T2CON/T3CON 时，TMR2/TMR3 不会被清零。

10.4 休眠模式下的定时器工作

在 CPU 休眠模式下，由于内部时钟被禁止，因而定时器不工作。

10.5 定时器中断

32 位定时器模块会在周期匹配或外部门控信号出现下降沿时产生一个中断。当 32 位定时器的计数值与相应的 32 位周期寄存器匹配，或检测到外部“门控”信号的下降沿时，将 T3IF 位（IFS0<7>）置 1 并产生中断（如果允许中断）。在该模式下，T3IF 中断标志作为中断源使用。必须在软件中将 T3IF 位清零。

通过相应的定时器中断允许位 T3IE（IEC0<7>）允许中断。

表 10-1: dsPIC30F3014/4013 TIMER2/3 寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
TMR2	0106	Timer2 寄存器																uuuu uuuu uuuu uuuu
TMR3HLD	0108	Timer3 保持寄存器 (仅用于 32 位定时器工作模式)																uuuu uuuu uuuu uuuu
TMR3	010A	Timer3 寄存器																uuuu uuuu uuuu uuuu
PR2	010C	周期寄存器 2																1111 1111 1111 1111
PR3	010E	周期寄存器 3																1111 1111 1111 1111
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	T32	—	TCS	—	0000 0000 0000 0000
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	—	TCS	—	0000 0000 0000 0000

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

11.0 TIMER4/5 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本节介绍辅助 32 位通用定时器模块 (Timer4/5) 及其相关工作模式。图 11-1 所示为 32 位 Timer4/5 模块的简化框图。图 11-2 和图 11-3 分别给出了将 Timer4/5 配置为两个独立的 16 位定时器 Timer4 和 Timer5 的结构图。

Timer4/5 模块的操作方式与 Timer2/3 模块类似。但仍存在一些差别：

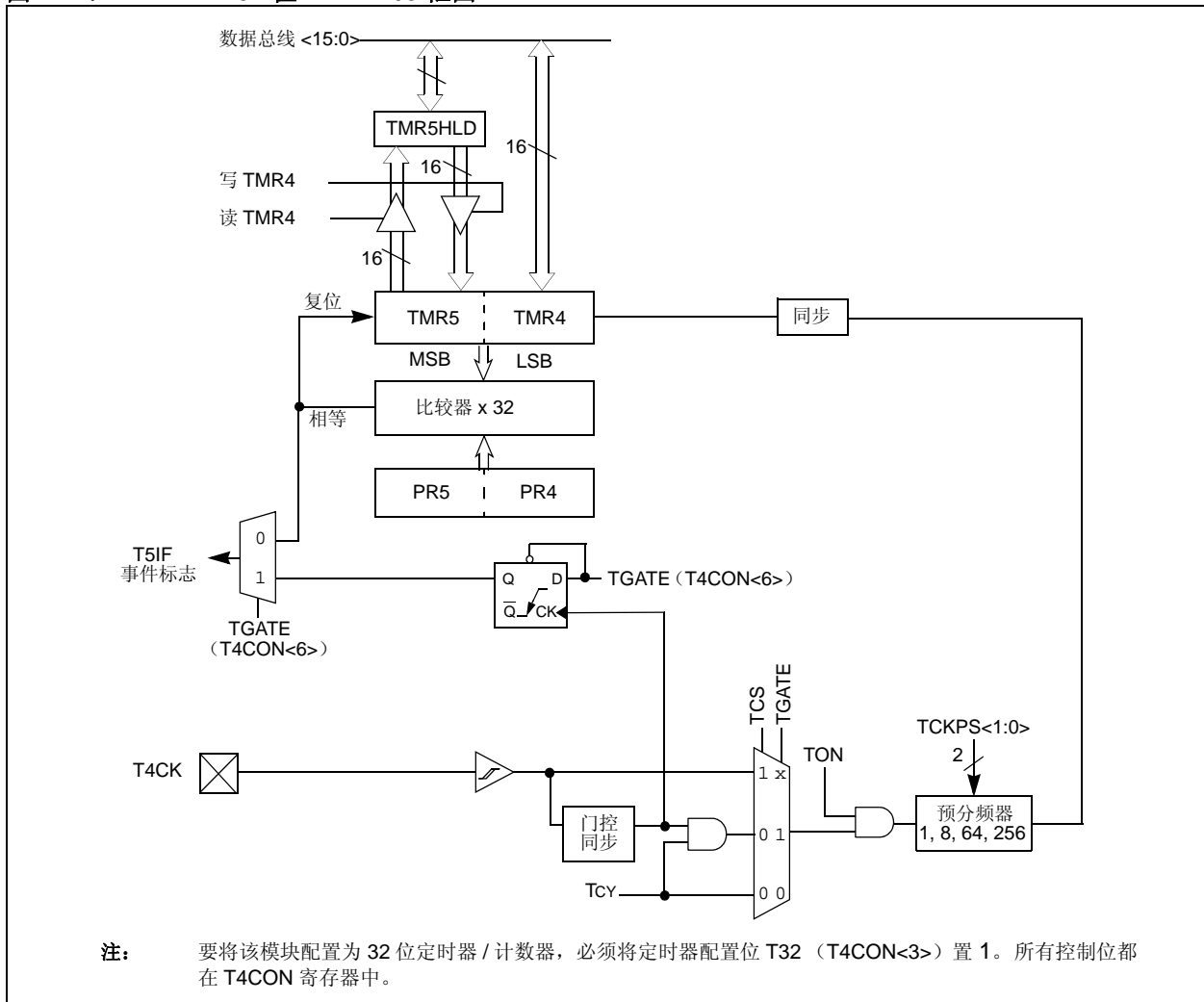
- Timer4/5 模块不支持 ADC 事件触发功能
- 输入捕捉和输出比较等其他外设模块不能使用 Timer4/5

通过设置 16 位 T4CON 和 T5CON SFR 中的相应位来确定 Timer4/5 模块的工作模式。

对于 32 位定时器 / 计数器操作，Timer4 和 Timer5 分别为 32 位定时器的最低有效数字和最高有效数字。

注： 在 32 位定时器操作模式下，T5CON 中的控制位被忽略。只使用 T4CON 中的控制位来进行设置和控制。32 位定时器模块使用 Timer4 的时钟和门控输入信号，但产生中断时会将 Timer5 的中断标志位 (T5IF) 置 1，并且由 Timer5 的中断允许位 (T5IE) 允许该模块的中断。

图 11-1: 32 位 TIMER4/5 框图



dsPIC30F3014/4013

图 11-2: 16 位 TIMER4 框图

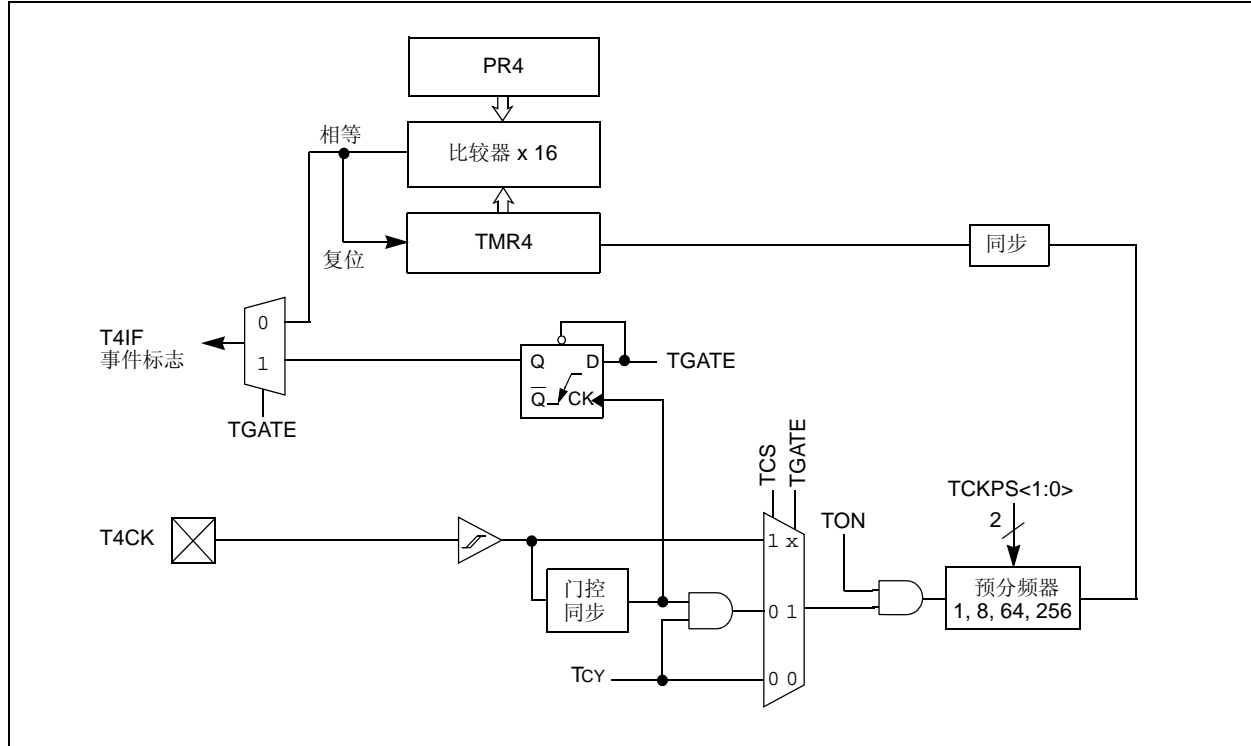


图 11-3: 16 位 TIMER5 框图

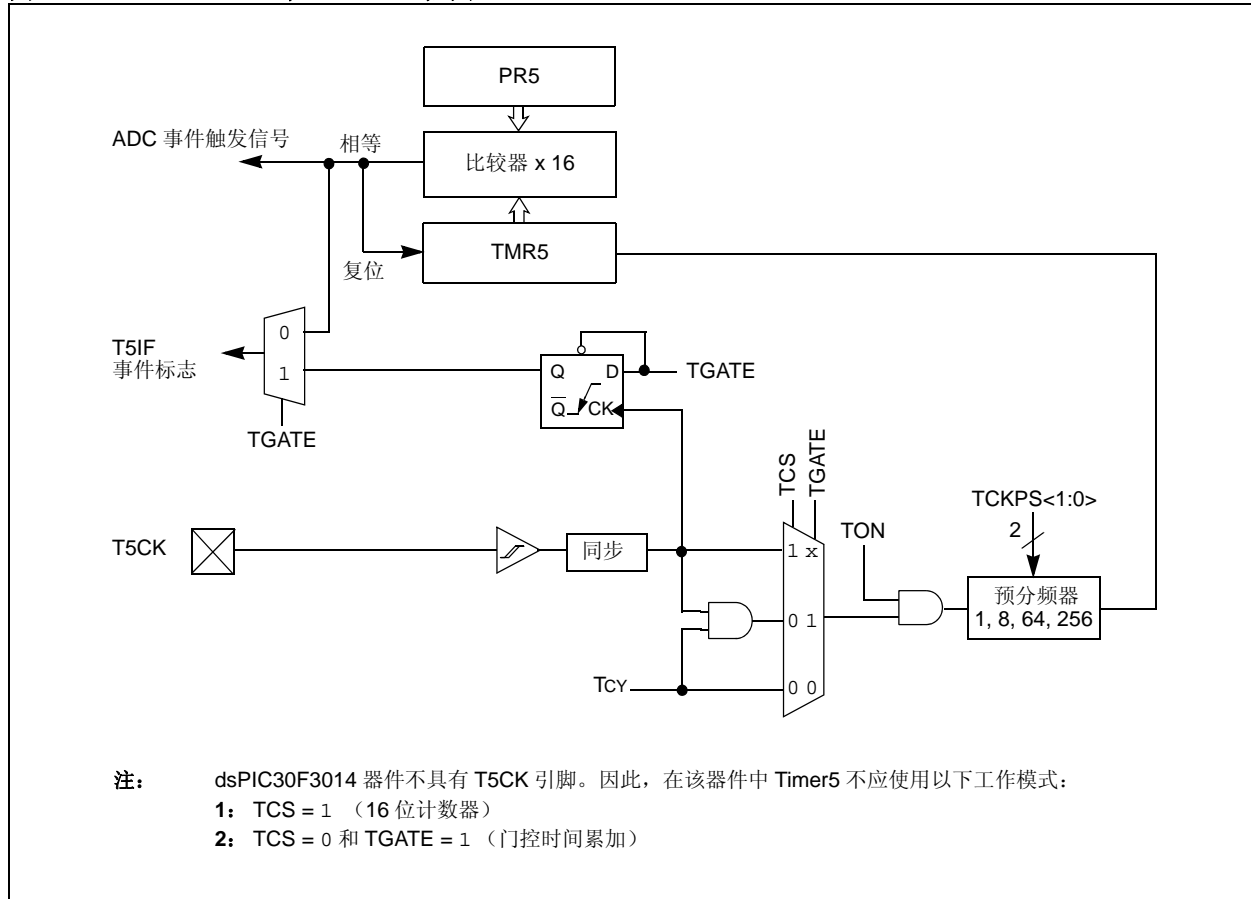


表 11-1: dsPIC30F4013 TIMER4/5 寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
TMR4	0114	Timer4 寄存器																uuuu uuuu uuuu uuuu
TMR5HLD	0116	Timer5 保持寄存器 (仅用于 32 位工作模式)																uuuu uuuu uuuu uuuu
TMR5	0118	Timer5 寄存器																uuuu uuuu uuuu uuuu
PR4	011A	周期寄存器 4																1111 1111 1111 1111
PR5	011C	周期寄存器 5																1111 1111 1111 1111
T4CON	011E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	T32	—	TCS	—	0000 0000 0000 0000
T5CON	0120	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS1	TCKPS0	—	—	TCS	—	0000 0000 0000 0000

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

12.0 输入捕捉模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本节描述输入捕捉模块及其相关工作模式。此模块提供的功能可用于需要频率（周期）和脉冲测量的应用。图 12-1 为输入捕捉模块的框图。输入捕捉可用于下列模式：

- 频率 / 周期 / 脉冲测量
- 额外的外部中断源

输入捕捉模块的主要工作特性有：

- 简单捕捉事件模式
- Timer2 和 Timer3 模式选择
- 发生输入捕捉事件时产生中断

通过设置 ICxCON (x = 1、2、... 或 N) 寄存器中的相应位选择上述工作模式。dsPIC DSC 器件最多可包含 8 个捕捉通道（即 N 的最大值为 8）。dsPIC30F3014 器件包含 2 个捕捉通道，而 dsPIC30F4013 器件包含 4 个捕捉通道。

12.1 简单捕捉事件模式

dsPIC30F 产品系列中的简单捕捉事件有：

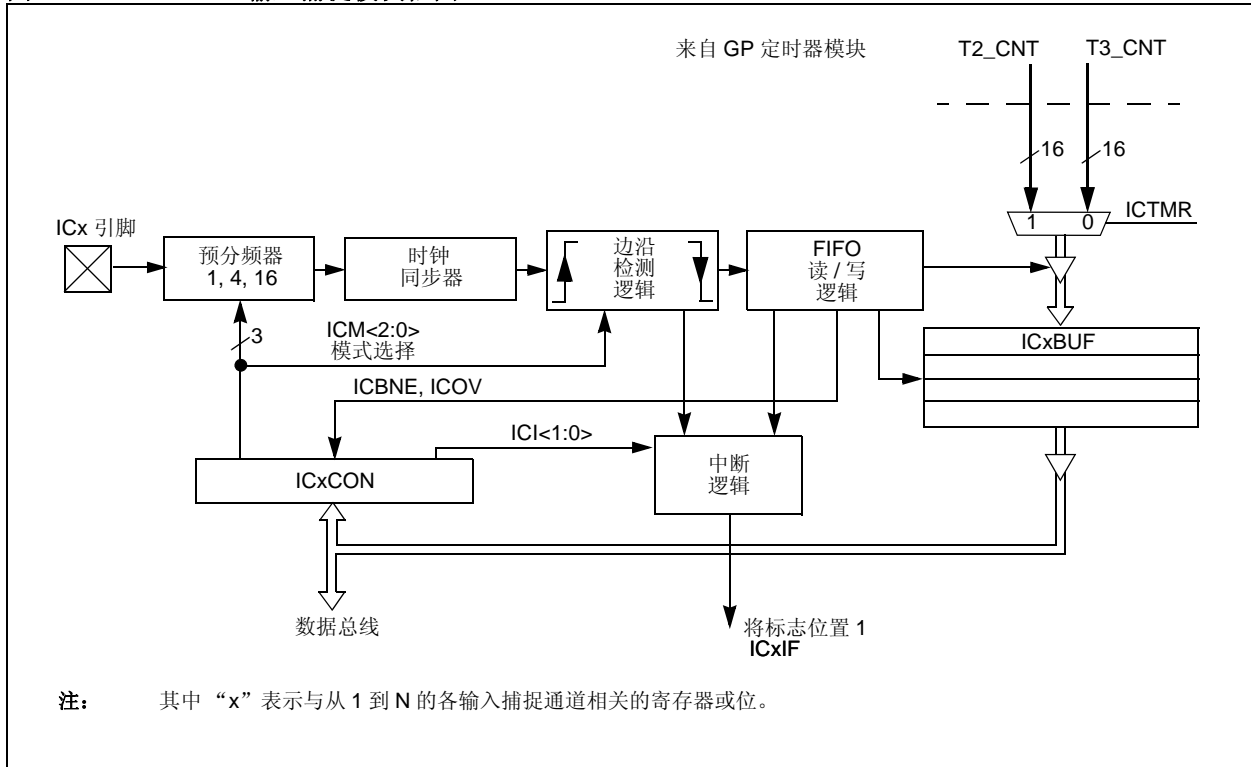
- 每个下降沿捕捉
- 每个上升沿捕捉
- 每 4 个上升沿捕捉
- 每 16 个上升沿捕捉
- 每个上升沿和下降沿捕捉

通过设置相应位 ICM<2:0> (ICxCON<2:0>) 配置简单输入捕捉模式。

12.1.1 捕捉预分频器

ICM<2:0> (ICxCON<2:0>) 位可指定 4 种输入捕捉预分频器设置。只要捕捉通道一关闭，预分频计数器就会清零。此外，任何复位操作都将清零预分频计数器。

图 12-1: 输入捕捉模块框图



12.1.2 捕捉缓冲区操作

每个捕捉通道都具有一个 4x16 字深度的相关 FIFO 缓冲区。FIFO 缓冲区具有两个表示状态的状态标志位。

- ICBNE —— 输入捕捉缓冲区非空
- ICOV —— 输入捕捉溢出

在第一个输入捕捉事件发生时将 ICBFNE 置 1，且保留其置 1 状态直到从 FIFO 读出所有的输入捕捉事件为止。每从 FIFO 中读出一个字，剩余各字就在缓冲区内前移一个位置。

当 FIFO 为满（其中有 4 个捕捉事件）且在读取 FIFO 前发生了第 5 个捕捉事件时，产生溢出，ICOV 位将被置为逻辑 1，第 5 个捕捉事件将丢失且不会储存在 FIFO 中。从缓冲区读出全部 4 个事件之前，无法捕捉其他事件。

如果完成最后一次读取后执行 FIFO 读操作，且此时未接收到新的捕捉事件，则产生一个不确定结果。

12.1.3 TIMER2 和 TIMER3 选择模式

输入捕捉模块由 8 个输入捕捉通道组成。每个通道可选择 Timer2 或 Timer3 作为时基定时器。

通过设置 SFR 位，ICTMR (ICxCON<7>) 选择定时器。输入捕捉模块的默认定时器为 Timer3。

12.1.4 霍尔传感器模式

当将输入捕捉模块设置为在每个上升或下降边沿捕捉 (ICM<2:0> = 001) 时，输入捕捉逻辑将完成如下操作：

- 在每个边沿（上升 / 下降沿）将输入捕捉标志位置 1。
- 由于每次捕捉都产生中断，忽略捕捉模式的中断设置位 ICI<1:0>
- 此模式下不会产生捕捉溢出条件。

12.2 休眠和空闲模式下的输入捕捉操作

如果器件处于 CPU 空闲或休眠模式下，则输入捕捉事件可唤醒器件或产生中断（如果允许中断）。

如果 ICM<2:0> = 111 且中断允许位置 1，无论是否使能定时器，输入捕捉模块都会将器件从 CPU 休眠或空闲模式中唤醒。如果满足中断处理条件，该唤醒将产生一个中断。唤醒功能是添加额外的外部引脚中断的一种有用的方法。

12.2.1 CPU 休眠模式下的输入捕捉

CPU 休眠模式允许输入捕捉模块继续工作，但功能减少了。在 CPU 休眠模式中，ICI<1:0> 位不可用且输入捕捉模块只能用作一个外部中断源。

如果要对处于休眠模式的器件使用输入捕捉模块，必须将捕捉模块配置为仅在上升沿中断 (ICM<2:0> = 111)。4:1 或 16:1 的预分频比设置在此模式下不可用。

12.2.2 CPU 空闲模式下的输入捕捉

CPU 空闲模式允许输入捕捉模块实现所有的功能。在此模式下，由 ICI<1:0> 位选择的的中断模式可用，由控制位 ICM<2:0> 定义的 4:1 和 16:1 的捕捉预分频比设置也可用。此模式要求所选定的定时器已使能。此外，ICSIDL 位必须设置为逻辑 0。

如果在 CPU 空闲模式下将输入捕捉模块定义为 ICM<2:0> = 111，输入捕捉引脚仅用作外部中断引脚。

12.3 输入捕捉中断

输入捕捉通道具有基于选中的捕捉事件数量而产生中断的功能。由控制位 ICI<1:0> (ICxCON<6:5>) 设置选中事件的数量。

每个通道提供一个中断标志 (ICxIF) 位。相应的捕捉通道中断标志位于相应的 IFSx 寄存器中。

通过相应的输入捕捉通道中断允许 (ICxIE) 位允许中断。捕捉中断允许位位于相应的 IEC 控制寄存器中。

表 12-1: dsPIC30F3014 输入捕捉寄存器映射⁽¹⁾

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
IC1BUF	0140	输入 1 捕捉寄存器																uuuu uuuu uuuu uuuu
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000	
IC2BUF	0144	输入 2 捕捉寄存器																uuuu uuuu uuuu uuuu
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000	

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 欲知有关寄存器位字段的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 12-2: dsPIC30F4013 输入捕捉寄存器映射⁽¹⁾

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
IC1BUF	0140	输入 1 捕捉寄存器																uuuu uuuu uuuu uuuu
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000	
IC2BUF	0144	输入 2 捕捉寄存器																uuuu uuuu uuuu uuuu
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000	
IC7BUF	0158	输入 7 捕捉寄存器																uuuu uuuu uuuu uuuu
IC7CON	015A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000	
IC8BUF	015C	输入 8 捕捉寄存器																uuuu uuuu uuuu uuuu
IC8CON	015E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>			0000 0000 0000 0000	

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 欲知有关寄存器位字段的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

13.0 输出比较模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本节介绍输出比较模块及其相关工作模式。该模块提供的功能可用于需要以下工作模式的应用：

- 生成可变宽度输出脉冲
- 功率因数校正

图 13-1 所示为输出比较模块的框图。

输出比较模块的主要工作特性如下：

- Timer2 和 Timer3 选择模式
- 简单输出比较匹配模式
- 双重输出比较匹配模式
- 简单 PWM 模式
- 休眠和空闲模式下的输出比较
- 输出比较 /PWM 事件的中断

这些工作模式是通过设置 16 位 OCxCON SFR（其中 $x = 1, 2, 3, \dots$ 或 N ）中的相应位而确定的。dsPIC DSC 器件最多包括 8 个比较通道（即 N 的最大值为 8）。dsPIC30F3014 器件包含 2 个比较通道，而 dsPIC30F4013 器件包含 4 个比较通道。

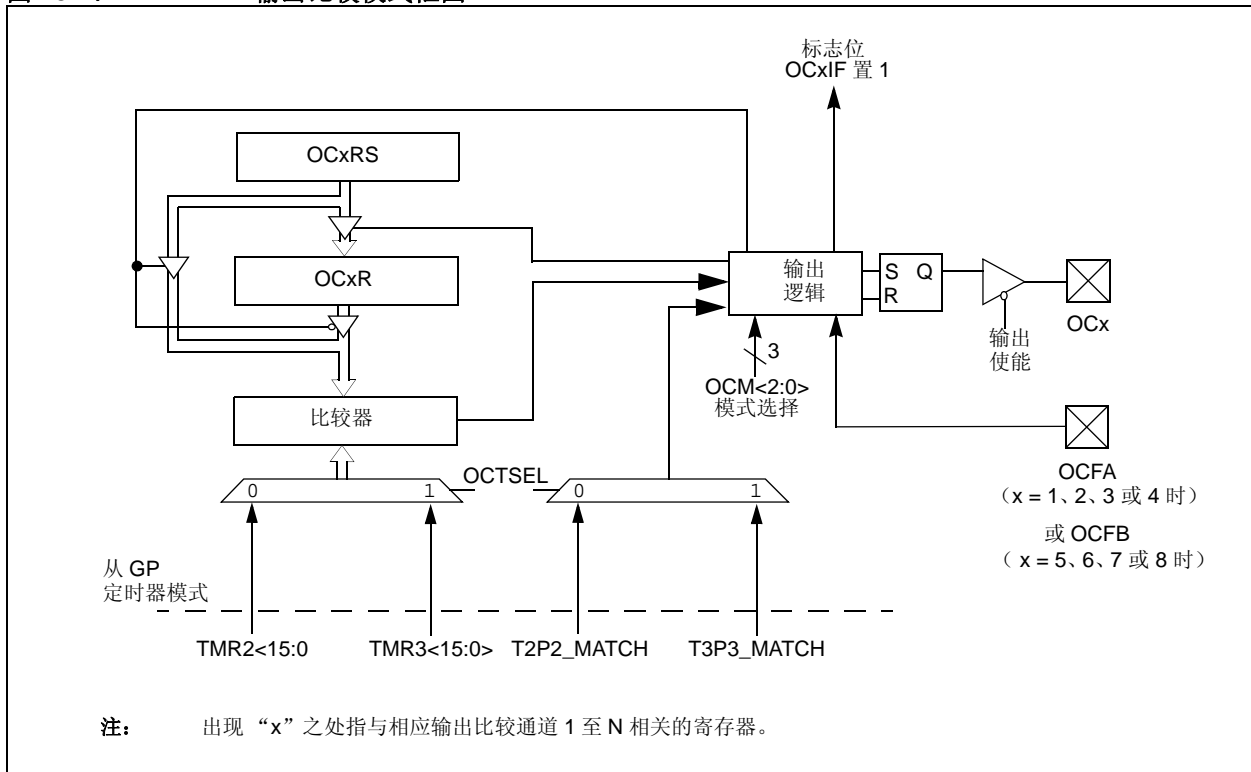
图 13-1 中的 OCxRS 和 OCxR 表示双重比较寄存器。在双重比较模式中，OCxR 寄存器用于第一次比较，OCxRS 寄存器用于第二次比较。

13.1 Timer2 和 Timer3 选择模式

每个输出比较通道都可在两个 16 位定时器，Timer2 和 Timer3 之间选择。

定时器的选择由 OCTSEL (OCxCON<3>) 位控制。Timer2 是输出比较模块的默认定时器资源。

图 13-1: 输出比较模式框图



13.2 简单输出比较匹配模式

控制位 $OCM<2:0>$ ($OCxCON<2:0>$) = 001、010 或 011 时，将所选的输出比较通道配置成如下三种简单的输出比较匹配模式中的一种：

- 比较匹配时强制 I/O 引脚为低电平
- 比较匹配时强制 I/O 引脚为高电平
- 比较匹配时翻转 I/O 引脚的电平

$OCxR$ 寄存器用于这些模式。在 $OCxR$ 寄存器中装载一个值并将它与选择的递增计数器的计数值做比较。比较时，会发生其中一个比较匹配模式。如果在达到 $OCxR$ 中的值之前计数器复位为零， OCx 引脚的状态将保持不变。

13.3 双重输出比较匹配模式

当控制位 $OCM<2:0>$ ($OCxCON<2:0>$) = 100 或 101 时，将所选的输出比较通道配置为以下两种双重输出比较模式中的一种：

- 单输出脉冲模式
- 连续输出脉冲模式

13.3.1 单脉冲模式

用户需要执行以下步骤来将该模块配置为产生单个输出脉冲（假设定时器关闭）：

- 确定指令周期时间 T_{CY} 。
- 以 T_{CY} 为基础计算所需的脉宽值。
- 从定时器起始值 $0x0000$ 计算启动脉冲的时间。
- 将脉冲宽度启动和停止时间写入 $OCxR$ 和 $OCxRS$ 比较寄存器（ x 表示通道 1、2、... 或 N ）。
- 设置定时器周期寄存器的值大于或等于 $OCxRS$ 比较寄存器的值。
- 设置 $OCM<2:0> = 100$
- 使能定时器， TON ($TxCON<15>$) = 1。

要启动另外一个脉冲，则重新将 $OCM<2:0>$ 设置为 100。

13.3.2 连续脉冲模式

用户需要执行以下步骤来将该模块配置为产生连续的输出脉冲流：

- 确定指令周期时间 T_{CY} 。
- 以 T_{CY} 为基础计算所需的脉宽值。
- 从定时器起始值 $0x0000$ 计算启动脉冲的时间。
- 将脉宽启动时间和停止时间写入 $OCxR$ 和 $OCxRS$ 比较寄存器（ x 表示通道 1、2、... 或 N ）。
- 设置定时器周期寄存器的值大于或等于 $OCxRS$ 比较寄存器中的值。
- 设置 $OCM<2:0> = 101$ 。
- 使能定时器， TON ($TxCON<15>$) = 1。

13.4 简单 PWM 模式

当控制位 $OCM<2:0>$ ($OCxCON<2:0>$) = 110 或 111 时，将所选的输出比较通道配置为 PWM 工作模式。配置 PWM 工作模式时， $OCxR$ 为主锁存器（只读）， $OCxRS$ 为辅助锁存器。这样就避免了在 PWM 电平转换时出现毛刺。

用户必须执行下列步骤将输出比较模块配置为 PWM 操作：

1. 通过写相应的周期寄存器设置 PWM 周期。
2. 通过写 $OCxRS$ 寄存器设置 PWM 占空比。
3. 将输出比较模块配置为 PWM 操作。
4. 设置 $TMRx$ 预分频值并使能定时器， TON ($TxCON<15>$) = 1。

13.4.1 PWM 的输入引脚故障保护

当控制位 $OCM<2:0>$ ($OCxCON<2:0>$) = 111 时，选定的输出比较通道被配置为带有额外的输入故障保护功能的 PWM 工作模式。在此模式下，如果在 $OCFA/B$ 引脚上检测到逻辑 0，就会将相应的 PWM 输出引脚置为高阻态。 $OCFLT$ 位 ($OCxCON<4>$) 指示是否出现故障条件。该状态会一直持续到发生下面两个事件为止：

- 外部故障条件已清除。
- 已经通过写相应的控制位重新使能了 PWM 模式。

13.4.2 PWM 周期

通过写 PRx 寄存器指定 PWM 周期。可以使用公式 13-1 计算 PWM 周期。

公式 13-1:

$$\text{PWM 周期} = [(\text{PRx}) + 1] \cdot 4 \cdot \text{TOSC} \cdot (\text{TMRx 预分频值})$$

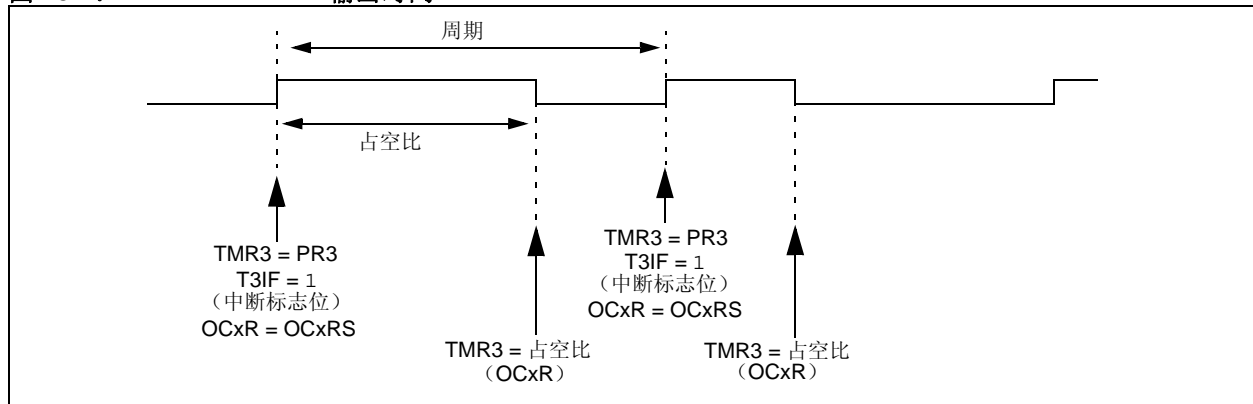
PWM 频率定义为 $1/[\text{PWM 周期}]$ 。

当选定的 TMRx 与自身的周期寄存器 PRx 相等时，在下一个递增周期会发生下列四个事件：

- 清零 TMRx。
- 将 OCx 引脚拉为高电平。
 - 例外 1: 若 PWM 占空比为 0x0000，则 OCx 引脚将保持低电平。
 - 例外 2: 如果占空比大于 PRx 中的值，则 OCx 引脚将保持高电平。
- 将 PWM 占空比从 OCxRS 锁存到 OCxR。
- 相应的定时器中断标志位置 1。

请参见图 13-2 了解主要的 PWM 周期比较。为了起见，在图 13-2 中引用了 Timer3。

图 13-2: PWM 输出时间



13.5 CPU 休眠模式下的输出比较操作

当 CPU 进入休眠模式时，所有内部时钟都停止。因此，当 CPU 进入休眠状态时，输出比较通道将引脚驱动为在 CPU 进入休眠状态之前观察到的有效状态。

例如，当 CPU 进入休眠状态时，若引脚为高电平，则保持高电平。同样，当 CPU 进入休眠状态时，若引脚为低电平，则保持低电平。在这两种情况下，当器件被唤醒时，输出比较模块都将恢复正常工作。

13.6 CPU 空闲模式下的输出比较操作

CPU 进入空闲模式时，输出比较模块的所有功能都可以正常工作。

如果 OCSIDL 位 (CON<13>) 为逻辑 0，且使能选定的时基 (Timer2 或 Timer3)，并且选定定时器的 TSIDL 位被设置为逻辑 0，则输出比较通道可在 CPU 空闲模式下工作。

13.7 输出比较中断

无论选择哪种匹配模式，输出比较通道均有在比较匹配时产生中断的功能。

对于除 PWM 模式外的所有模式，当发生比较事件时，相应的中断标志位 (OCxIF) 置 1，并在允许中断时还将产生中断。OCxIF 位位于相应的 IFSx 寄存器中并且必须用软件清零。通过相应的比较中断允许位 (OCxIE) 允许中断，该允许位位于相应的 IEC 寄存器中。

对于 PWM 模式，当有事件发生时，会将相应的定时器中断标志位 (T2IF 或 T3IF) 置 1，并且如果允许中断还将产生中断。TxIF 位位于 IFS0 寄存器中并且必须用软件清零。通过 IEC0 寄存器中的相应定时器中断允许位 (T2IE 或 T3IE) 允许中断。在 PWM 工作模式下，输出比较中断标志永远不会置 1。

表 13-1: dsPIC30F3014 输出比较寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
OC1RS	0180	输出比较 1 辅助寄存器																0000 0000 0000 0000
OC1R	0182	输出比较 1 主寄存器																0000 0000 0000 0000
OC1CON	0184	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	
OC2RS	0186	输出比较 2 辅助寄存器																0000 0000 0000 0000
OC2R	0188	输出比较 2 主寄存器																0000 0000 0000 0000
OC2CON	018A	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSE	OCM<2:0>		0000 0000 0000 0000	

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 13-2: dsPIC30F4013 输出比较寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
OC1RS	0180	输出比较 1 辅助寄存器																0000 0000 0000 0000
OC1R	0182	输出比较 1 主寄存器																0000 0000 0000 0000
OC1CON	0184	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	
OC2RS	0186	输出比较 2 辅助寄存器																0000 0000 0000 0000
OC2R	0188	输出比较 2 主寄存器																0000 0000 0000 0000
OC2CON	018A	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSE	OCM<2:0>		0000 0000 0000 0000	
OC3RS	018C	输出比较 3 辅助寄存器																0000 0000 0000 0000
OC3R	018E	输出比较 3 主寄存器																0000 0000 0000 0000
OC3CON	0190	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	
OC4RS	0192	输出比较 4 辅助寄存器																0000 0000 0000 0000
OC4R	0194	输出比较 4 主寄存器																0000 0000 0000 0000
OC4CON	0196	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

14.0 I²C™ 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

I²C™ 模块能够为 I²C 串行通信标准的从模式和多主器件模式提供完整的硬件支持，且该模块带有一个 16 位接口。

此模块的主要特性如下：

- I²C 接口支持主 / 从工作模式。
- I²C 从模式支持 7 位和 10 位寻址。
- I²C 主模式支持 7 位和 10 位寻址。
- I²C 端口允许主器件和从器件之间进行双向传输。
- I²C 端口的串行时钟同步可被用作握手机制以暂停或恢复串行传输（SCLREL 控制）。
- I²C 支持多主器件工作；检测总线冲突并作出相应的仲裁。

14.1 工作功能描述

硬件完全实现了 I²C 标准和快速模式规范下的全部主从功能，以及 7 位和 10 位寻址。

因此，I²C 模块既可作为 I²C 总线上的主器件工作，也可作为从器件工作。

14.1.1 各种 I²C 模式

器件支持以下 I²C 工作模式：

- 7 位寻址的 I²C 从模式
- 10 位寻址的 I²C 从模式
- 7 位或 10 位寻址的 I²C 主模式

请参见图 14-1 中的 I²C 编程模型。

14.1.2 I²C 模式下的引脚配置

I²C 接口有 2 个引脚：SCL 引脚为时钟引脚，SDA 引脚为数据引脚。

14.1.3 I²C 寄存器

I2CCON 和 I2CSTAT 分别为控制寄存器和状态寄存器。I2CCON 为可读写寄存器。I2CSTAT 寄存器的低 6 位为只读位，其他位为可读 / 可写位。

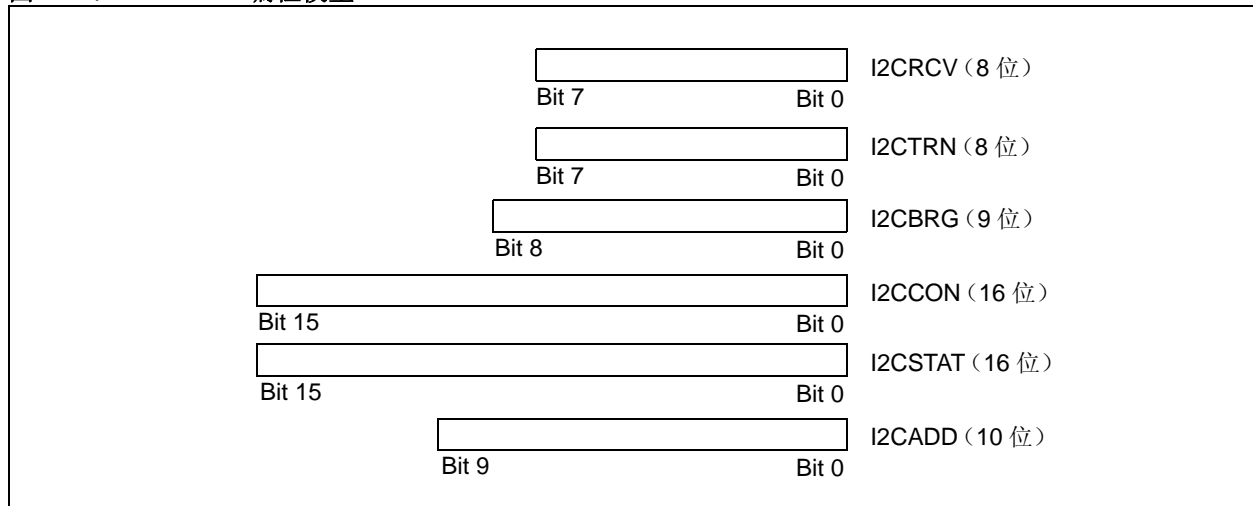
I2CRSR 为移位寄存器，用于数据移位，而 I2CRCV 为缓冲寄存器，可写入或读取数据字节。如图 14-1 所示，I2CRCV 为接收缓冲区。如图 14-2 所示，I2CTRN 为发送寄存器，在执行发送操作时会向其中写入字节。

I2CADD 寄存器用于保存从器件地址。状态位 ADD10 表示是否为 10 位寻址模式。I2CBRG 用作波特率发生器重载值。

在进行接收时，I2CRSR 和 I2CRCV 一起形成双重缓冲接收器。I2CRSR 接收到一个完整的字节后，将把它传输到 I2CRCV 并产生中断脉冲。在发送操作中，I2CTRN 不是双重缓冲的。

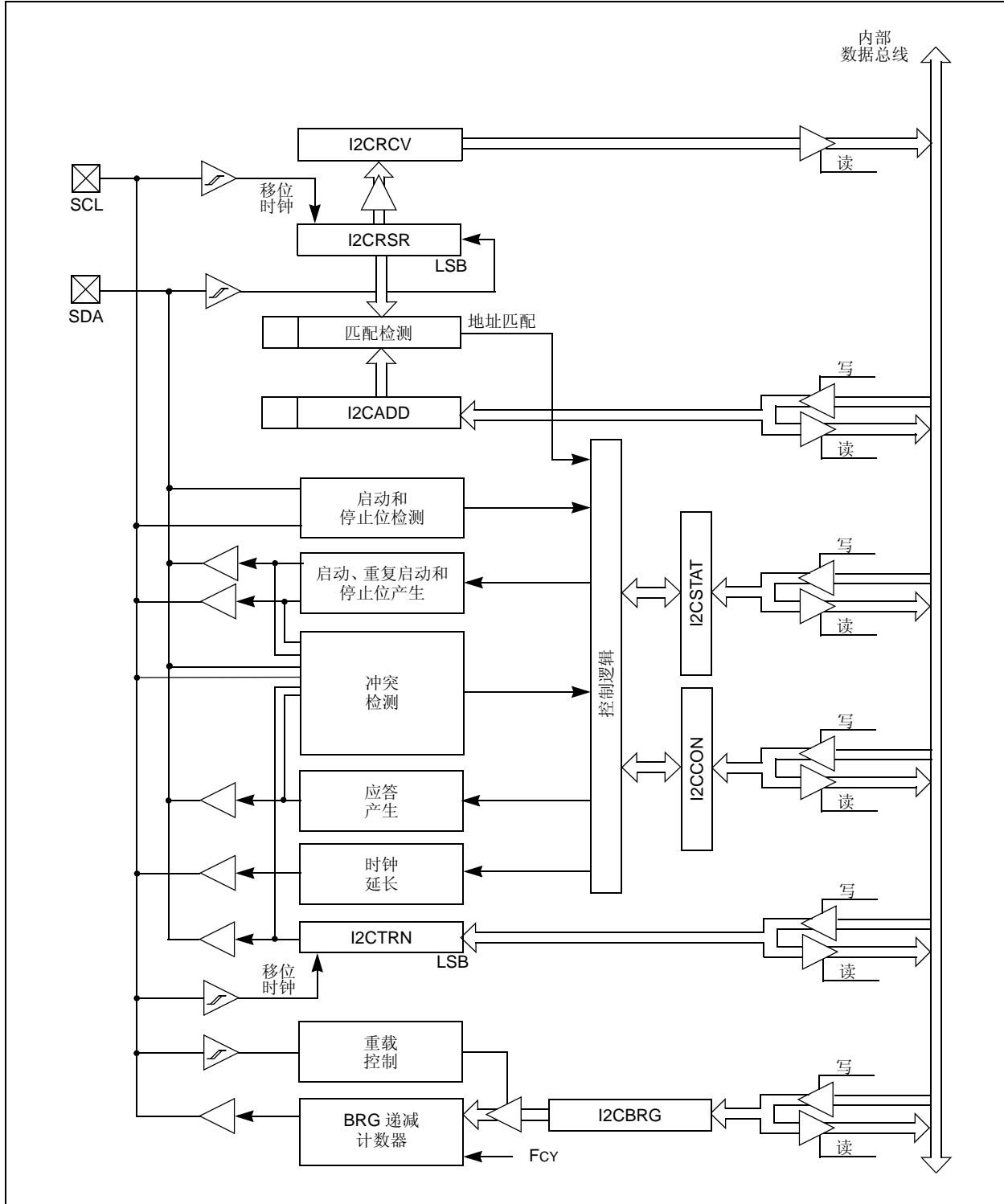
注： 在 10 位寻址模式下，重复启动条件发生后，用户只需匹配前 7 位地址。

图 14-1: 编程模型



dsPIC30F3014/4013

图 14-2: I²C™ 框图



14.2 I²C 模块地址

I2CADD 寄存器包含从模式地址，它是一个 10 位寄存器。

当 A10M 位 (I2CCON<10>) 为 0 时，模块把地址解释为 7 位地址。在接收地址时，将接收到的地址与 I2CADD 寄存器的低 7 位进行比较。

当 A10M 位为 1 时，模块将认为地址是 10 位地址。在接收地址时，将接收到的地址与二进制值 11110 A9 A8 (其中 A9 和 A8 为 I2CADD 的高 2 位) 进行比较。如果两者相匹配，则将下一个地址与 I2CADD 的低 8 位进行比较，这是 10 位寻址协议规定的。

表 14-1: dsPIC30F 支持的 7 位 I²C™ 从地址

0x00	广播呼叫地址或起始字节
0x01-0x03	保留
0x04-0x07	Hs 模式主代码
0x08-0x77	7 位有效地址
0x78-0x7b	10 位有效地址 (低 7 位)
0x7c-0x7f	保留

14.3 I²C 7 位从模式工作原理

一旦使能 (I2CEN = 1)，从模块将等待起始位出现 (如，I²C 模块处于“空闲”模式)。检测到起始位后，会将 8 个位移入 I2CRSR，并将地址与 I2CADD 进行比较。在 7 位模式下 (A10M = 0)，将 I2CADD<6:0> 位和 I2CRSR<7:1> 进行比较，而 I2CRSR<0> 为 R_W 位。所有输入位均在 SCL 引脚的上升沿进行采样。

如果发生地址匹配，则发送应答，且在第 9 位 (ACK) 的下降沿将从事件中断标志位 (SI2CIF) 置 1。地址匹配不会影响 I2CRCV 缓冲区的内容或 RBF 位。

14.3.1 从模式发送

如果接收到的 R_W 位为 1，则串行端口进入发送模式。它在第 9 位发送 ACK 信号，然后将 SCL 引脚保持在 0 状态，直至通过写 I2CTRN 寄存器引起 CPU 响应。通过将 SCLREL 位置 1，释放 SCL 引脚并移出 8 位数据。数据在 SCL 的下降沿移出，因此，当 SCL 为高电平时，SDA 引脚信号有效。无论从主器件收到的 ACK 状态如何，都将在第 9 个时钟脉冲的下降沿发送中断脉冲。

14.3.2 从模式接收

在地址匹配过程中，如果接收到的 R_W 位为 0，则启动接收模式。所有输入位均在 SCL 引脚的上升沿进行采样。接收 8 位数据后，若 I2CRCV 未满载或 I2COV 未置 1，则将 I2CRSR 中的数据传送到 I2CRCV。在第九个时钟发送 ACK 信号。

如果 RBF 标志位被置 1，表示 I2CRCV 中仍存有前一次操作的数据 (RBF = 1)，则不发送 ACK 信号；但仍会产生中断脉冲。溢出时，I2CRSR 寄存器的内容不会装入到 I2CRCV。

注： 当 I2COV 位为 1 且 RBF 标志位为 0 时，I2CRCV 寄存器装有数据。在这种情形下，已对 I2CRCV 寄存器执行了读操作，但在下一次接收操作开始之前，用户没有清零 I2COV 位的状态。此时将不发送应答信号 (ACK = 1)，但会更新 I2CRCV 寄存器。

14.4 I²C 10 位从模式工作原理

在 10 位模式下，基本的接收和发送操作与 7 位模式下的操作相同，但地址匹配的条件更加复杂。

I²C 规范指出，对于写操作，必须在起始位后跟两个地址字节来对从器件进行寻址。

由 A10M 控制位表明 I2CADD 寄存器中的地址为 10 位地址而非 7 位地址。报文地址首字节的地址检测协议，对于 7 位和 10 位报文而言是完全相同的，但被比较的位不同。

I2CADD 中保存了完整的 10 位地址。在接收到起始位后的地址后，将 I2CRSR <7:3> 与立即数 11110 (默认 10 位地址) 进行比较，而将 I2CRSR <2:1> 与 I2CADD <9:8> 进行比较。如果发生匹配且 R_W = 0，则发送中断脉冲。清零 ADD10 位以表示部分地址匹配。如果匹配失败或 R_W = 1，则清零 ADD10 位且模块返回到空闲状态。

随后接收地址的低字节，并将该字节与 I2CADD <7:0> 进行比较。如果发生地址匹配，则产生中断脉冲，并将 ADD10 位置 1，表示 10 位地址完全匹配。如果未发生地址匹配，则清零 ADD10 位且模块返回到空闲状态。

14.4.1 10 位模式下的从发送

如果使用完整的 10 位地址方式对从器件进行寻址 (我们将此状态称为“PRIOR_ADDR_MATCH”)，主器件就可以开始发送数据字节以供器件接收。

dsPIC30F3014/4013

14.4.2 10 位模式下的从接收

找到从器件后，主器件会产生一个重复启动信号、复位地址的高字节并将 R_W 位置 1，但不产生停止位，从而启动从器件发送操作。

14.5 自动时钟延长

在从模式下，模块可通过时钟延长，同步主器件的缓冲区读写操作。

14.5.1 发送时钟延长

在 7 位和 10 位发送模式下，如果 TBF 位被清零（表明缓冲区为空），则可在第 9 个时钟的下降沿之后将 SCLREL 清零，以实现时钟延长。

在从发送模式下，始终执行时钟延长，而与 STREN 位的状态无关。

在发送序列的第 9 个时钟后发生时钟同步。如果器件在第 9 个时钟信号的下降沿采样到 ACK 且 TBF 位仍为零，则自动清零 SCLREL 位，从而使 SCL 线变为低电平。用户 ISR 必须在允许继续发送之前将 SCLREL 位置 1。将 SCL 线保持为低电平，使用户在主器件启动另一次发送序列之前有时间执行 ISR 并装载 I2CTRN 的内容。

- 注 1:** 如果用户载入 I2CTRN 寄存器的内容，并在第 9 个时钟的下降沿之前将 TBF 位置 1，则 SCLREL 位不会清零，且不会发生时钟延长。
- 2:** SCLREL 位可由软件置 1，而无需考虑 TBF 位的状态。

14.5.2 接收时钟延长

在从接收模式下，I2CCON 寄存器的 STREN 位可用于使能时钟延长。将 STREN 位置 1 时，SCL 引脚在每个数据接收序列的末端均保持低电平。

14.5.3 7 位寻址模式下的时钟延长 (STREN = 1)

在从接收模式下，当 STREN 位置 1 时，若缓冲寄存器满，则 SCL 线保持为低电平。对于 7 位和 10 位寻址模式，延长 SCL 输出的方法是相同的。

在接收序列的第 9 个时钟后发生时钟延长。在 ACK 序列末端的第 9 个时钟的下降沿，如果 RBF 位被置 1，则 SCLREL 位会自动清零，从而强制将 SCL 输出拉为低电平。用户 ISR 必须在允许继续接收之前将 SCLREL 位置 1。通过将 SCL 线保持为低电平，使用户有时间执行 ISR 并读取 I2CRCV 的内容，之后主器件才会启动下一个接收序列。这将防止发生缓冲区溢出。

- 注 1:** 如果用户读取 I2CRCV 寄存器的内容，并在第 9 个时钟的下降沿之前将 TBF 位清零，则 SCLREL 位不会清零，且不会发生时钟延长。
- 2:** 无论 RBF 位的状态如何，均可用软件将 SCLREL 位置 1。在下一个接收序列开始之前，用户清零 ISR 中的 RBF 位时必须小心，以避免溢出。

14.5.4 10 位寻址模式下的时钟延长 (STREN = 1)

在该寻址过程中自动发生时钟延长。由于此模块带有一个存放整个地址的寄存器，协议无需等待地址更新。

寻址阶段完成后，在每个数据接收或发送序列上发生时钟延长，如前所述。

14.6 软件控制的时钟延时 (STREN = 1)

当 STREN 位为 1 时，可使用软件清零 SCLREL 位，以允许通过软件控制时钟延长。程序逻辑将写 SCLREL 位与 SCL 时钟同步。清零 SCLREL 位不会将 SCL 输出拉为低电平，直至模块检测到 SCL 输出的一个下降沿且 SCL 采样为低电平。如果 SCLREL 位被用户清零，且在 SCL 线采样到低电平，则 SCL 输出被拉为低电平（保持低电平）。SCL 输出将保持低电平直至 SCLREL 位被置 1，且 I²C 总线上的所有其他器件都将 SCL 拉为高电平，此操作确保写 SCLREL 位的操作不会违反 SCL 所需的最短高电平时间要求。

如果 STREN 位为 0，则忽略由软件对 SCLREL 位的写操作，且不影响 SCLREL 位。

14.7 中断

I²C 模块可产生 2 个中断标志：MI2CIF（I²C 主中断标志）和 SI2CIF（I²C 从中断标志）。当完成主器件报文事件后，激活 MI2CIF 中断标志。当检测到发送给从器件的报文时，激活 SI2CIF 中断标志。

14.8 斜率控制

I²C 标准需要对快速模式（400 kHz）下的 SDA 和 SCL 信号进行斜率控制。如果需要的话，DISSLW 控制位可供用户禁止斜率控制。对于 1 MHz 模式必须禁止斜率控制。

14.9 IPMI 支持

IPMIEN 控制位使模块能够支持智能外设管理接口（Intelligent Peripheral Management Interface, IPMI）。当此位被置 1 时，模块接受所有地址并对这些地址进行操作。

14.10 广播呼叫地址支持

广播呼叫地址可寻址所有器件。当使用此地址时，理论上所有器件都应对此地址作出应答。

广播呼叫地址是 I²C 协议为特定目的保留的 8 个地址之一。它由全 0 组成，且 R_W = 0。

当广播呼叫使能位（GCEN）置 1（I2CCON<7> = 1）时，则识别为广播呼叫地址。检测到起始位后，将 8 位数据移入 I2CRSR 寄存器，并将地址与 I2CADD 进行比较，同时也与广播呼叫地址（固化在硬件中）进行比较。

如果发生广播呼叫地址匹配，则在第 8 个时钟之后，I2CRSR 的内容被传输到 I2CRCV，在第 9 位（ACK 位）的下降沿将 RBF 标志位置 1，主事件中断标志（MI2CIF）被置 1。

响应中断时，可通过读 I2CRCV 寄存器的内容检查中断源，从而判断该地址是特定于器件的还是广播呼叫地址。

14.11 I²C 主器件支持

用作主器件时，支持如下六种操作：

- 在 SDA 和 SCL 上产生启动条件。
- 在 SDA 和 SCL 上产生重复启动条件。
- 写 I2CTRN 寄存器以启动数据 / 地址发送。
- 在 SDA 和 SCL 上产生停止条件。
- 配置 I²C 端口以接收数据。
- 在数据字节接收完成后产生 $\overline{\text{ACK}}$ 条件。

14.12 I²C 主模式工作原理

由主器件生成所有的串行时钟脉冲以及启动和停止条件。由停止条件或重复启动条件停止数据传输。由于重复启动条件是下一个串行传输的开始，所以此时不会释放 I²C 总线。

在主发送模式下，由 SDA 引脚输出串行数据，而 SCL 引脚输出串行时钟。发送的第一个字节中包含接收器件的从地址（7 位）和数据方向位。此时，数据方向位（R_W）为逻辑 0。每次发送 8 位串行数据。在每个字节发送完之后会收到一个 $\overline{\text{ACK}}$ 位。输出的启动和停止条件表明串行传输的开始和结束。

在主接收模式下，发送的第一个字节中包含发送器件的从地址（7 位）和数据方向位。此时，数据方向位（R_W）为逻辑 1。这样，发送的第一个字节为 7 位从地址，后跟一个表明接收位的 1。通过 SDA 接收串行数据，而 SCL 输出串行时钟。每次接收 8 位串行数据。在每个字节接收之后，会发送一个 $\overline{\text{ACK}}$ 位。启动和停止条件表示发送的开始和结束。

14.12.1 I²C 主发送

直接把值写入 I2CTRN 寄存器即可完成数据字节或 7 位地址，或 10 位地址的后半部分的发送。当模块处于等待（Wait）状态时，用户只能写 I2CTRN 寄存器。此操作会将缓冲区满标志位（TBF）置 1，允许波特率发生器开始计数并开始下一次发送。SCL 信号出现下降沿之后，将地址 / 数据的每一位都移出到 SDA 引脚。发送状态标志位 TRSTAT（I2CSTAT<14>）显示主器件是否正在发送。

14.12.2 I²C 主接收

可通过编程 RCEN (I2CCON<3>) 接收使能位来使能主模式接收。将 RCEN 位置 1 之前, I²C 模块必须处于空闲状态, 否则将忽略 RCEN 位。波特率发生器开始计数, 每次计满返回时, SCL 引脚上的 ACK 位状态和数据就会在每个时钟的上升沿移入 I2CRSR 寄存器。

14.12.3 波特率发生器

在 I²C 主模式下, BRG 的重载值位于 I2CBRG 寄存器中。当 BRG 载入此值后, BRG 计数器递减计数到 0 并停止计数, 直至下一次重载发生。如果发生时钟仲裁, 例如 BRG 在 SCL 引脚采样为高电平时被重载。

按照 I²C 标准, F_{SCK} 的值为 100 kHz 或 400 kHz。但用户可指定最高达 1 MHz 的任何波特率。0 或 1 在 I2CBRG 寄存器中为非法值。

公式 14-1: 串行时钟速率

$$I2CBRG = \left(\frac{F_{CY}}{F_{SCK}} - \frac{F_{CY}}{1,111,111} \right) - 1$$

14.12.4 时钟仲裁

在任何接收、发送或重复启动 / 停止条件期间, 主器件拉高 SCL 引脚的电平 (允许 SCL 引脚悬空为高电平) 就会发生时钟仲裁。当 SCL 引脚悬空为高电平时, 波特率发生器 (BRG) 停止计数, 直至 SCL 引脚被实际采样到高电平为止。此时, 波特率发生器重新装入 I2CBRG 寄存器的内容并开始计数。从而保证当时钟被外部器件拉为低电平时, SCL 高电平时间至少能够保证一个 BRG 计满返回周期。

14.12.5 多主机通信、总线冲突和总线仲裁

多主机工作支持是通过总线仲裁实现的。主器件在 SDA 引脚上输出地址 / 数据位时, 如果一个主器件通过使 SDA 悬空为高电平从而在 SDA 上输出一个 1, 而另一个主器件在 SDA 线上输出 0, 就会发生仲裁。SCL 引脚悬空为高电平时, 数据应该稳定。如果 SDA 上的预期数据值是 1, 但从 SDA 引脚采样到的值为 0, 则表示已经发生总线冲突。主器件将发出 MI2CIF 脉冲, 且将主器件使用的那部分 I²C 端口复位到其空闲状态。

如果正在进行数据发送时发生总线冲突, 则会中止发送, TBF 标志位被清零, SDA 和 SCL 线被拉为高电平, 此时可向 I2CTRN 寄存器中写入值。当用户执行 I²C 主事件中断服务程序时, 如果 I²C 总线空闲 (即 P 位被置 1), 则用户可通过产生一个启动条件恢复通信。

如果总线冲突发生时遇到启动、重复启动、停止或应答条件, 则这些条件将被中止。SDA 和 SCL 线被拉为高电平, 且 I2CCON 寄存器中的相应控制位被清零。当用户执行总线冲突中断服务程序时, 如果 I²C 总线空闲, 则用户可通过产生一个启动条件恢复通信。

主器件继续监视 SDA 和 SCL 引脚, 当产生停止条件时, 将 MI2CIF 位置 1。

写 I2CTRN 寄存器的操作将从第一个数据位开始发送数据, 而与总线冲突发生时发送器的状态无关。

在多主机环境下, 在检测到启动和停止条件时产生中断, 从而允许确定总线何时空闲。当 I2CSTAT 寄存器的 P 位被置 1 时, 可控制 I²C 总线, 否则, 总线处于空闲状态且 S 和 P 位为零。

14.13 CPU 休眠和空闲模式下的 I²C 模块工作原理

14.13.1 CPU 休眠模式下的 I²C 工作原理

器件进入休眠模式时, 模块的所有时钟源将关闭并保持为逻辑 0。如果在发送过程中发生休眠, 且当时钟停止时, 状态机部分进入发送状态, 则中止发送。类似地, 如果在接收的过程中发生休眠, 接收也将中止。

14.13.2 CPU 空闲模式下的 I²C 工作原理

对于 I²C, 由 I2CSIDL 位决定模块在空闲模式时继续工作还是停止工作。如果 I2CSIDL = 0, 模块在空闲模式下继续工作。如果 I2CSIDL = 1, 模块在空闲模式下停止工作。

表 14-2: dsPIC30F3014/4013 I²C™ 寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
I2CRCV	0200	—	—	—	—	—	—	—	—	接收寄存器								0000 0000 0000 0000
I2CTRN	0202	—	—	—	—	—	—	—	—	发送寄存器								0000 0000 1111 1111
I2CBRG	0204	—	—	—	—	—	—	—	波特率发生器								0000 0000 0000 0000	
I2CCON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0001 0000 0000 0000
I2CSTAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF	0000 0000 0000 0000
I2CADD	020A	—	—	—	—	—	—	地址寄存器								0000 0000 0000 0000		

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

15.0 SPI 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

串行外设接口 (Serial Peripheral Interface, SPI) 模块是一个同步串行接口。SPI 模块用于与其他外设 (EEPROM、移位寄存器、显示驱动器、A/D 转换器等) 或单片机进行通信。该模块与 Motorola 的 SPI 和 SIOP 接口兼容。dsPIC30F3014 和 dsPIC30F4013 器件具有一个 SPI 模块，即 SPI1。

15.1 功能描述

每个 SPI 模块的组成如下：1 个 16 位的移位寄存器 SPIxSR (其中 x = 1 或 2)，用于移入和移出数据；一个缓冲寄存器 SPIxBUF；一个控制寄存器 SPIxCON，用于配置模块；还有一个状态寄存器 SPIxSTAT，用于指示各种状态条件。

串行接口包含 4 个引脚：SDIx (串行数据输入)、SDOx (串行数据输出)、SCKx (移位时钟输入或输出) 和 SSx (低电平有效从选择)。

在主模式下工作时，SCKx 为时钟输出，而在从模式下为时钟输入。

一组 8 或 16 个时钟脉冲将位从 SPIxSR 移出到 SDOx 引脚，同时从 SDIx 引脚移入数据。传输完成后将产生中断，相应的中断标志位 (SPI1IF 或 SPI2IF) 将置 1。通过中断允许位 (SPI1IE 或 SPI2IE) 可禁止该中断。

接收操作是双重缓冲的。当接收到一个完整的字节时，将该字节从 SPIxSR 传输到 SPIxBUF。

当新数据从 SPIxSR 传输到 SPIxBUF 时，如果接收缓冲区满，模块将 SPIROV 位置 1，表示产生溢出条件。数据从 SPIxSR 向 SPIxBUF 的传输未完成，新数据将丢失。当 SPIROV 为 1 时，模块不会响应 SCL 电平跳变，这样便在用户软件读取 SPIxBUF 之前有效地禁止了模块。

发送写操作也是双重缓冲的。用户写入到 SPIxBUF。当主器件或从器件传输完成时，移位寄存器 (SPIxSR) 中的内容将送至接收缓冲区。如果已向缓冲寄存器写入了任何发送数据，则发送缓冲区的内容将送至 SPIxSR。因此，接收到的数据存放在 SPIxBUF 中，SPIxSR 中的发送数据已准备就绪，可进行下次传输。

注： 发送缓冲区 (SPIxTXB) 和接收缓冲区 (SPIxRXB) 均映射到相同的寄存器地址 SPIxBUF。

在主模式下，通过对系统时钟进行预分频产生时钟。数据一写入到 SPIxBUF，数据就开始发送。在最后一位传输的过程中产生中断。

在从模式下，当外部时钟脉冲出现在 SCKx 上时，开始发送和接收数据。同样，当最后一位被锁存时产生中断。如果使能了 SSx 控制，则仅当 SSx 为低电平时，发送和接收才被使能。在 SSx 模式下，SSx 为高电平时，SDOx 输出被禁止。

提供给模块的时钟为 (Fosc/4)。随后由主 (PPRE<1:0>) 和辅助 (SPRE<2:0>) 预分频系数对此时钟进行预分频。CKE 位决定是从有效时钟状态转变为空闲时钟状态时开始发送，还是相反。CKP 位选择时钟的空闲状态 (高电平或低电平)。

15.1.1 字和字节通信

控制位 MODE16 (SPIxCON<10>) 可允许模块在 16 位或 8 位模式下通信。16 位操作除了发送的位数为 16 而不是 8 以外，与 8 位操作相同。

用户软件必须在更改 MODE16 位前禁止模块。用户更改 MODE16 位时，SPI 模块将复位。

8 位和 16 位操作的基本差异是：8 位操作中数据是从 SPIxSR 的 bit 7 发送的，而在 16 位操作中数据是从 SPIxSR 的 bit 15 发送的。在这两种模式下，数据都会移入 SPIxSR 的 bit 0。

15.1.2 禁止 SDOx

SPIxCON 寄存器中 DISSDO 控制位，用于禁止 SDOx 输出。这将允许在仅输入的配置中连接 SPI 模块。SDOx 也可用作通用 I/O。

dsPIC30F3014/4013

15.2 帧 SPI 支持

在主模式或从模式下，模块均支持基本的帧 SPI 协议。控制位 FRMEN 使能帧 SPI 支持，并使得 SSx 引脚执行帧同步脉冲（FSYNC）功能。控制位 PIFSD 决定 SSx

引脚是输入还是输出（即，模块是接收还是产生帧同步脉冲）。对单 SPI 时钟周期，帧脉冲是高电平有效脉冲。当使能了帧同步时，仅在 SPI 时钟的后续发送边沿开始发送数据。

图 15-1: SPI 框图

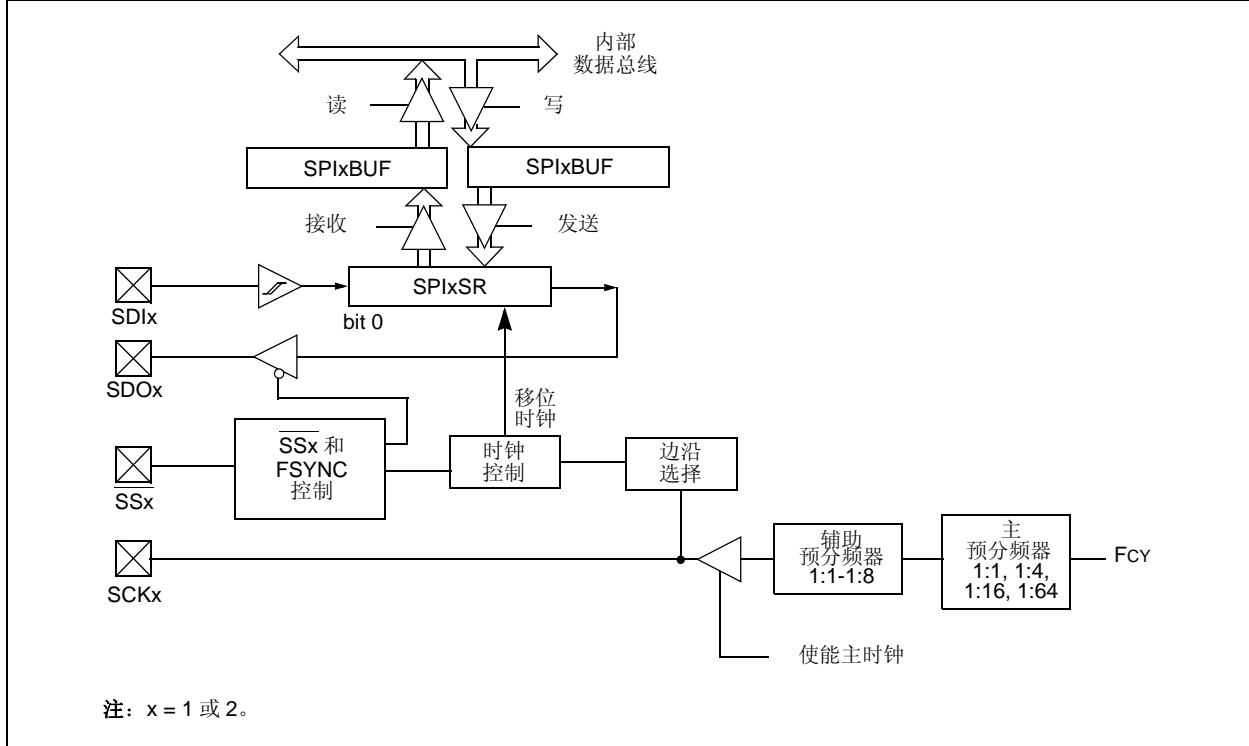
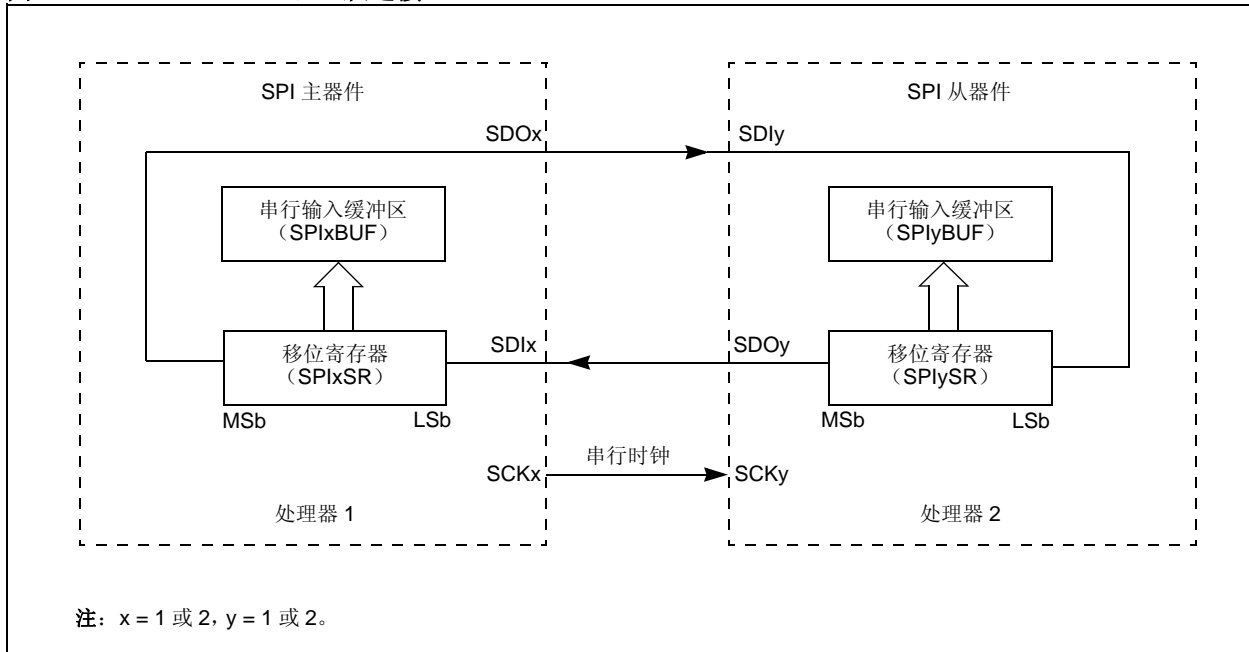


图 15-2: SPI 主 / 从连接



15.3 从选择同步

\overline{SSx} 引脚允许同步从模式。必须用 \overline{SSx} 引脚控制使能位 ($SSEN = 1$) 将 SPI 配置为 SPI 从模式。当 \overline{SSx} 引脚为低电平时, 使能发送和接收, 且驱动 $SDOx$ 引脚。当 \overline{SSx} 引脚变为高电平时, 不再驱动 $SDOx$ 引脚。另外, 可以重新同步 SPI 模块, 所有的计数器 / 控制电路都将复位。因此, 当 \overline{SSx} 引脚再次拉为低电平时, 即使 \overline{SSx} 在发送 / 接收过程中被拉为高电平, 发送 / 接收也将从 MSb 开始。

15.4 CPU 休眠模式下的 SPI 工作

在休眠模式期间, SPI 模块关闭。若在 SPI 传输过程中, CPU 进入了休眠模式, 则发送和接收中止。

在休眠模式下, 发送器和接收器都将停止。但是, 进入或退出休眠模式不影响寄存器的内容。

15.5 CPU 空闲模式下的 SPI 工作

器件进入空闲模式时, 所有时钟源都继续工作。SPISIDL 位 ($SPIxSTAT<13>$) 决定 SPI 模块在空闲模式下是停止还是继续工作。如果 $SPISIDL = 0$, 当 CPU 进入空闲模式时模块将继续工作。如果 $SPISIDL = 1$, 当 CPU 进入空闲模式时模块将停止工作。

表 15-1: dsPIC30F3014/4013 SPI1 寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
SP1STAT	0220	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SP1CON	0222	—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SP1BUF	0224	发送和接收缓冲区																0000 0000 0000 0000

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

16.0 通用异步收发器 (UART) 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

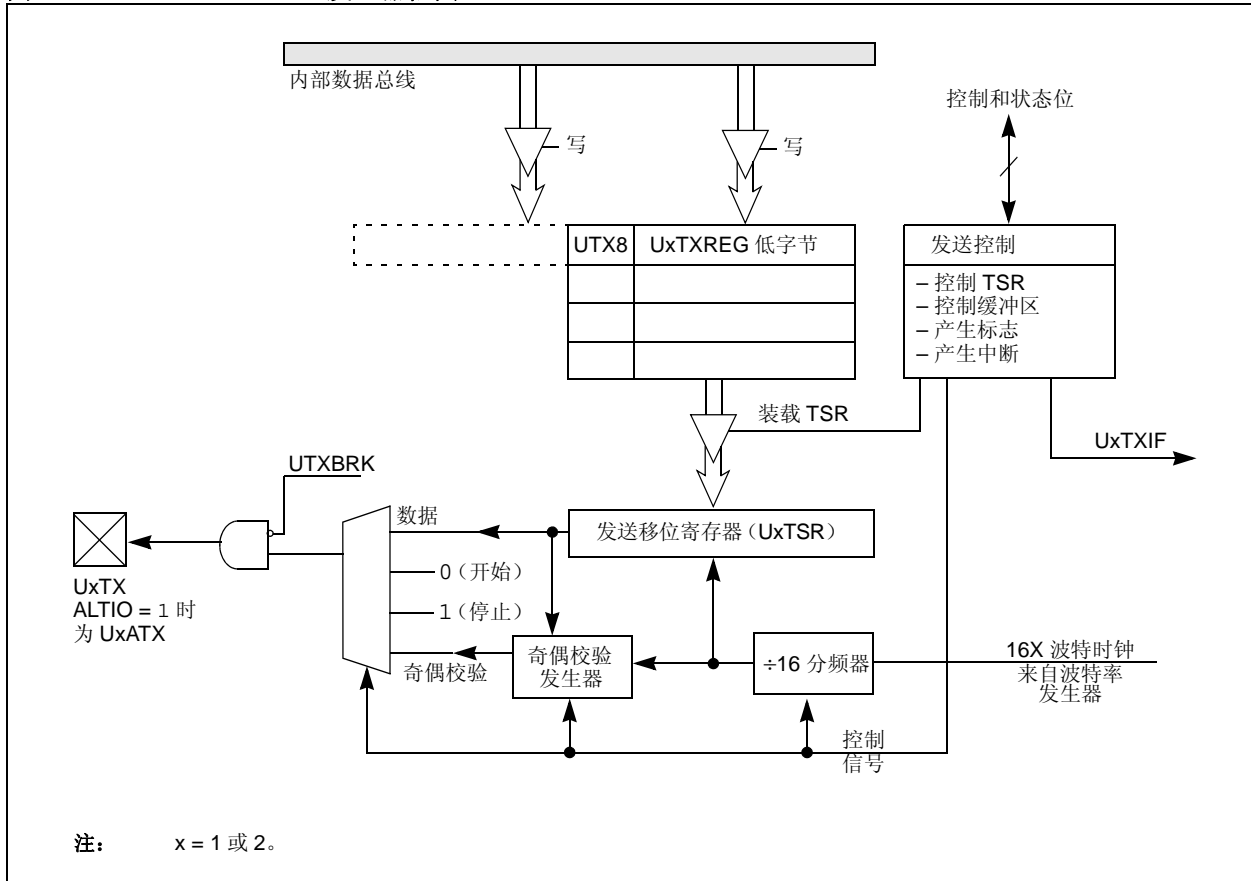
本节描述了通用异步收发器通信模块。

16.1 UART 模块概述

UART 模块的主要特性有：

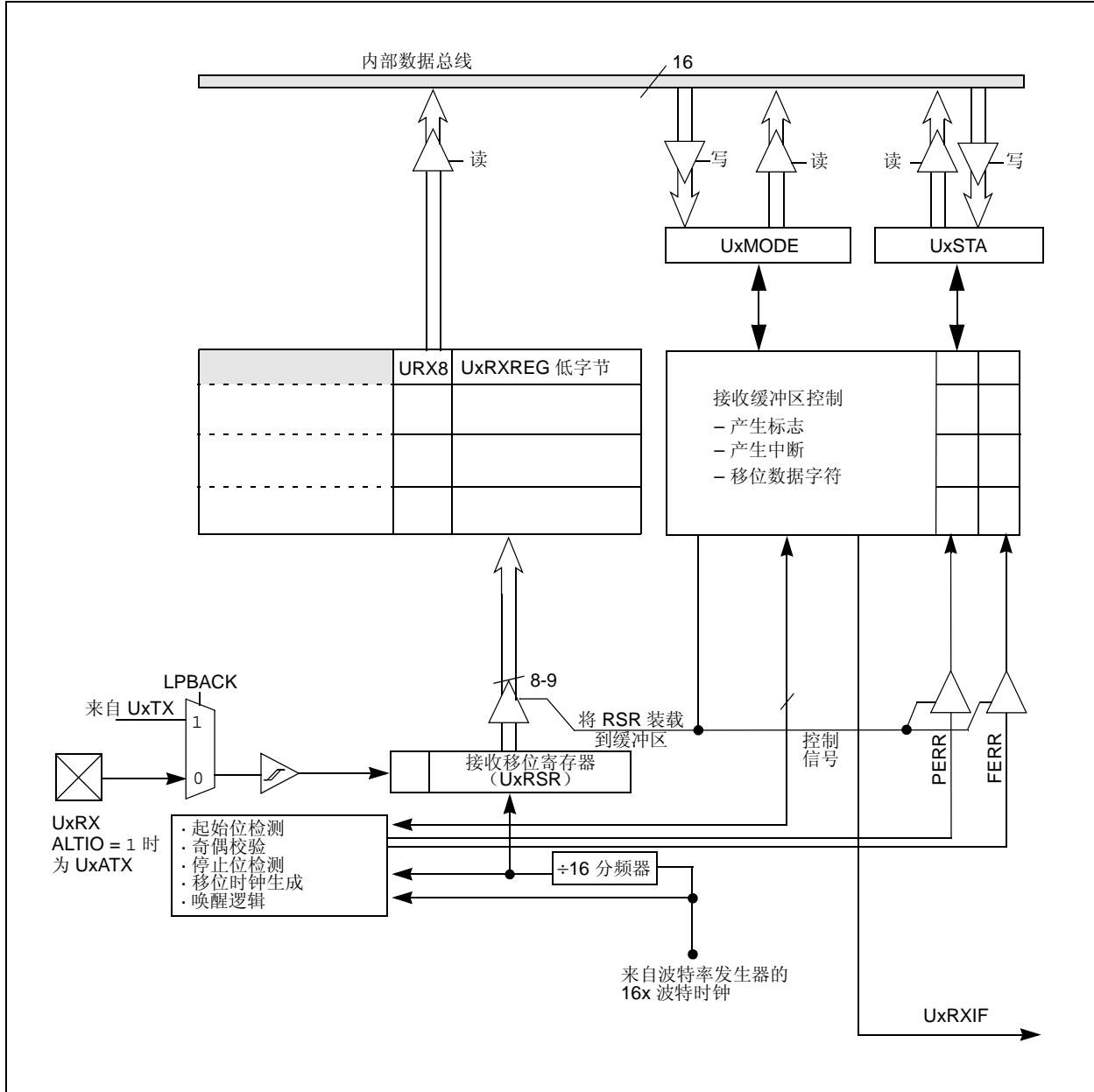
- 全双工，8 位或 9 位数据通信
- 偶校验、奇校验或无奇偶校验选项（对于 8 位数据）
- 一个或两个停止位
- 完全集成的、具有 16 位预分频器的波特率发生器
- 指令速率为 30 MHz 时，波特率范围为 38 bps 到 1.875 Mbps
- 深度为 4 字的发送数据缓冲区
- 深度为 4 字的接收数据缓冲区
- 奇偶校验错误、帧错误和缓冲区溢出错误检测
- 支持仅在检测地址时（第 9 位 = 1）中断
- 独立的发送和接收中断
- 支持诊断的环回模式
- UART1 模块的 TX/RX 引脚具有两种选择

图 16-1: UART 发送器框图



dsPIC30F3014/4013

图 16-2: UART 接收器框图



16.2 使能并设置 UART

16.2.1 使能 UART

通过将 UxMODE 寄存器 (x = 1 或 2) 中的 UARTEN 位置 1 来使能 UART 模块。一旦被使能, UxTX 和 UxRX 引脚将分别配置为输出和输入, 并将覆盖相应 I/O 端口引脚的 TRIS 和 LAT 寄存器位的设置。当不进行任何发送时, UxTX 引脚为逻辑 1。

16.2.2 禁止 UART

通过将 UxMODE 寄存器中的 UARTEN 位清零来禁止 UART 模块。这是执行任何复位操作后的默认状态。如果禁止了 UART, 则所有 I/O 引脚均用作端口引脚, 受相应端口引脚的 LAT 和 TRIS 位控制。

禁止 UART 模块将会使缓冲区复位为空状态。缓冲区中的所有数据字符将丢失且波特率计数器将复位。

当禁止模块时, 将复位所有与 UART 模块相关的错误和状态标志。URXDA、OERR、FERR、PERR、UTXEN、UTXBRK 和 UTXBF 位将被清零, RIDLE 和 TRMT 将置 1。其他控制位 (包括 ADDEN、URXISEL<1:0>、UTXISEL 以及 UxMODE 和 UxBRG) 将不受影响。

UART 活动时, 清零 UARTEN 位将中止所有等待的发送和接收并将复位模块 (如上所述)。重新使能 UART 将以相同的配置重启 UART。

16.2.3 备用 I/O

将 ALTIO 位 (UxMODE<10>) 置 1 可使能备用 I/O 功能。如果 ALTIO = 1, 则 UART 模块将使用 UxATX 和 UxARX 引脚 (分别为备用发送和备用接收引脚) 而不使用 UxTX 和 UxRX 引脚。如果 ALTIO = 0, 则 UART 模块使用 UxTX 和 UxRX 引脚。

16.2.4 设置数据、奇偶校验以及停止位选项

使用 UxMODE 寄存器中的控制位 PDSEL<1:0> 来选择发送过程中使用的数据长度和奇偶校验。数据长度可以为 8 位 (带有奇校验、偶校验或无奇偶校验) 或 9 位 (不使用奇偶校验)。

STSEL 位决定在数据发送过程中使用 1 个还是 2 个停止位。

UART 的默认 (上电时) 设置为 8 位, 无奇偶校验, 带有 1 个停止位 (通常表示为 8, N, 1)。

16.3 发送数据

16.3.1 8 位数据模式下的发送

要发送 8 位数据, 需执行以下步骤:

1. 设置 UART:
首先, 必须选择数据长度、奇偶校验和停止位数。然后, 在 UxMODE 和 UxSTA 寄存器中设置发送和接收中断允许位和优先级位。另外还必须向 UxBRG 寄存器写入适当的波特率值。
2. 通过设置 UARTEN 位 (UxMODE<15>) 使能 UART。
3. 将 UTXEN 位 (UxSTA<10>) 置 1, 从而使能发送。
4. 将要发送的字节写入 UxTXREG 的低字节。该值将立即被传输到发送移位寄存器 (UxTSR), 并且在波特率时钟的下一个上升沿开始移出串行位流。当 UTXEN = 0 时, 可以选择写入数据字节, 此后, 用户可以将 UTXEN 置 1。这将导致立即开始传输串行位流, 原因是波特率时钟从清零状态启动。
5. 根据中断控制位 UTXISEL (UxSTA<15>) 的值, 将产生一个发送中断。

16.3.2 9 位数据模式下的发送

9 位数据发送的步骤与 8 位发送基本相同, 只是 16 位数据字 (其高 7 位总是清零) 必须写入到 UxTXREG 寄存器中。

16.3.3 发送缓冲区 (UxTXB)

发送缓冲区的宽度为 9 位, 深度为 4 个字符。包括发送移位寄存器 (UxTSR), 用户实际上具有一个深度为 5 的 FIFO (先进先出) 缓冲区。UTXBF 状态位 (UxSTA<9>) 指示发送缓冲区是否为满。

如果用户试图对已满的缓冲区进行写入操作, 则 FIFO 将不会接收新的数据, 并且不会有数据移入缓冲区。这确保了能够从缓冲区溢出条件中恢复。

任何器件复位都将复位 FIFO, 但当器件进入节能模式或从节能模式唤醒时, FIFO 将不受影响。

16.3.4 发送中断

发送中断标志 (U1TXIF 或 U2TXIF) 位于相应的中断标志寄存器中。

发送器产生一个边沿来将 UxTXIF 位置 1。产生中断的条件取决于 UTXISEL 控制位:

- 如果 UTXISEL = 0, 当从发送缓冲区将一个字传输到发送移位寄存器 (UxTSR) 时, 将产生一个中断。这意味着发送缓冲区至少具有一个空字。
- 如果 UTXISEL = 1, 当从发送缓冲区将一个字传输到发送移位寄存器 (UxTSR) 并且发送缓冲区为空时, 将产生一个中断。

在操作过程中, 可在两个中断模式之间进行切换, 这有时能提供更大的灵活性。

16.3.5 发送间隔

将 UTXBRK 位 (UxSTA<11>) 置 1 将导致 UxTX 线驱动为逻辑 0。UTXBRK 位将覆盖所有的发送活动。因此, 用户通常会等到发送器空闲后才将 UTXBRK 置 1。

要发送一个间隔字符, 必须通过软件将 UTXBRK 置 1 并且必须至少保持 13 个波特率时钟周期。然后, 通过软件将 UTXBRK 位清零产生停止位。用户必须等待至少一个或两个波特率时钟周期以确保停止位有效, 再重新装载 UxTXB 或启动其他发送活动。发送间隔字符不会产生发送中断。

16.4 接收数据

16.4.1 8 位或 9 位数据模式下的接收

接收 8 位或 9 位数据时必须执行以下步骤:

- 设置 UART (见第 16.3.1 节 “8 位数据模式下的发送”)。
- 使能 UART (见第 16.3.1 节 “8 位数据模式下的发送”)。
- 当接收到一个或多个字时, 将根据由 URXISEL 位 (UxSTA<7:6>) 指定的接收中断设置产生接收中断。
- 读取 OERR 位以确定是否发生溢出错误。OERR 位必须用软件复位。
- 从 UxRXREG 读取接收的数据。读取 UxRXREG 的操作将下一个字移动到接收 FIFO 的顶端, 并且更新 PERR 和 FERR 值。

16.4.2 接收缓冲区 (UxRXB)

接收缓冲区的深度为 4 字。包括接收移位寄存器 (UxRSR) 在内, 用户实际上具有深度为 5 字的 FIFO 缓冲区。

URXDA (UxSTA<0>) = 1 表明接收缓冲区具有可用的数据。URXDA = 0 意味着缓冲区为空。如果用户试图读取空缓冲区, 则将读取缓冲区中的旧值并且 FIFO 中将不会发生数据移位。

任何器件复位将复位 FIFO。当器件进入节能模式或从节能模式唤醒时, 不会影响 FIFO。

16.4.3 接收中断:

可从相应的中断标志寄存器读取接收中断标志 (U1RXIF 或 U2RXIF)。通过接收器产生的边沿来设置中断标志。接收中断标志的设置情况取决于由 URXISEL<1:0> (UxSTA<7:6>) 控制位指定的设置。

- 如果 URXISEL<1:0> = 00 或 01, 每次从接收移位寄存器将一个数据字传输到接收缓冲区时将产生中断。接收缓冲区中可能具有一个或多个字符。
- 如果 URXISEL<1:0> = 10, 当从接收移位寄存器 (UxRSR) 将一个字传输到接收缓冲区时产生中断, 传输后, 接收缓冲区包含 3 个字符。
- 如果 URXISEL<1:0> = 11, 当从接收移位寄存器 (UxRSR) 将一个字传输到接收缓冲区时产生中断, 传输后, 接收缓冲区包含 4 个字符 (即为满状态)。

可以在操作过程中切换中断模式, 但在正常操作时通常不建议这样做。

16.5 接收错误处理

16.5.1 接收缓冲区溢出错误 (OERR 位)

如果发生以下情况, 则 OERR 位 (UxSTA<1>) 将被置 1:

- 接收缓冲区为满。
- 接收移位寄存器为满, 但是无法将字符传输到接收缓冲区。
- 检测到 UxRSR 中字符的停止位, 这表示 UxRSR 需要将字符传输到缓冲区。

一旦将 OERR 置 1, 将不会有数据移入到 UxRSR (直到用软件将 OERR 位清零或发生复位为止)。UxRSR 和 UxRXREG 中的数据保持有效。

16.5.2 帧错误 (FERR)

如果检测到 0 而不是停止位，FERR 位 (UxSTA<2>) 将被置 1。如果选择了两个停止位，则两个停止位都必须为 1，否则 FERR 将被置 1。FERR 只读位将与接收到的数据一起缓存。发生任何复位时，FERR 位将被清零。

16.5.3 奇偶校验错误 (PERR)

如果接收到的字的奇偶性不正确，则 PERR 位 (UxSTA<3>) 将被置 1。仅当选择奇偶校验模式 (奇校验或偶校验) 时，才可使用此错误位。PERR 只读位将与接收到的数据字节一起缓存。发生任何复位时，它将被清零。

16.5.4 空闲状态

当接收器活动时 (即，从最初检测到起始位始到停止位完成期间)，RIDLE 位 (UxSTA<4>) 为 0。完成停止位到检测到下一个起始位期间，RIDLE 位为 1，表示 UART 空闲。

16.5.5 接收间隔

接收器根据在 PDSEL (UxMODE<2:1>) 和 STSEL (UxMODE<0>) 位中编程的值计数并等待一定数量的位时间。

如果间隔时间大于 13 个位时间，则经过 PDSEL 和 STSEL 指定的位时间数量后，可认为接收完成。URXDA 和 FERR 将被置 1，接收 FIFO 将装载 0，如果允许中断则还将产生中断，RIDLE 位将置 1。

当模块接收长间隔信号并且接收器检测到起始位、数据位和无效停止位 (该位将 FERR 置 1) 时，接收器必须等到一个有效停止位后才能查找下一个起始位。不能假设线路上的间隔条件为下一个起始位。

间隔被认为是一个包含全 0 且 FERR 置 1 的字符。间隔字符将被装载到缓冲区。在接收到停止位之前将不会接收任何后续数据。注意，在未接收到停止位之前，RIDLE 将为高电平。

16.6 地址检测模式

将 ADDEN 位 (UxSTA<5>) 置 1 使能该特殊模式，在该模式中，第 9 位 (URX8) 的值为 1 指示接收字为地址，而不是数据。该模式仅适用于 9 位数据通信。由于每当接收到的字的第 9 位置 1 时都将产生中断 (如果允许了中断)，因此，URXISEL 控制位对该模式的中断产生不造成任何影响。

16.7 环回模式

将 LPBACK 位置 1 使能该特殊模式，在该模式中，UxTX 引脚内部连接到 UxRX 引脚。当配置为环回模式时，UxRX 引脚与内部 UART 接收逻辑断开。但在正常操作中，UxTX 引脚仍然正常工作。

要选择该模式，需要：

- 将 UART 配置为所需的工作模式。
- 将 LPBACK 设置为 1 以使能环回模式。
- 按第 16.3 节 “发送数据” 中的定义使能发送。

16.8 波特率发生器

UART 具有 16 位波特率发生器以使波特率发生时具有最大的灵活性。波特率发生器寄存器 (UxBRG) 可读写。根据下述公式计算波特率：

BRG = 保存在 UxBRG 寄存器中的 16 位值 (0 到 65535)

Fcy = 指令时钟速率 (1/Tcy)

波特率由公式 16-1 给出。

公式 16-1: 波特率

$$\text{波特率} = \text{Fcy} / (16 * (\text{BRG} + 1))$$

因此，可达到的最大波特率为：

Fcy / 16 (如果 BRG = 0)，

并且，可达到的最小波特率为：

Fcy / (16 * 65536)。

当器件工作在 30 MIPS 下时使用完整的 16 位波特率发生器，可达到的最小波特率为 28.5 bps。

16.9 自动波特率支持

要允许系统确定接收字符的波特率，可以选择将输入连接到所选捕捉输入（UART1 的捕捉输入为 IC1，UART2 的捕捉输入为 IC2）。要使用该模式，用户必须对输入捕捉模块进行编程以检测起始位的下降沿和上升沿。

16.10 CPU 休眠和空闲模式下的 UART 工作原理

16.10.1 CPU 休眠模式下的 UART 工作原理

当器件进入休眠模式时，模块的所有时钟源关闭并且保持为逻辑 0。如果在发送过程中进入休眠模式，则发送将中止。UxTX 引脚驱动为逻辑 1。类似地，如果在接收过程中进入休眠模式，则接收将中止。UxSTA、UxMODE、发送和接收寄存器以及缓冲区和 UxBRG 寄存器不受休眠模式的影响。

如果在器件进入休眠模式前 WAKE 位（UxMODE<7>）已置 1，则在 UxRX 引脚的下降沿产生接收中断。接收中断选择模式位（URXISEL）对此功能无效。如果允许接收中断，则器件将从休眠模式唤醒。UARTEN 位必须置 1 才能产生唤醒中断。

16.10.2 CPU 空闲模式下的 UART 工作原理

器件进入空闲模式后，由 USIDL 位决定 UART 模块停止还是继续工作。当 USIDL = 0 时，模块在空闲模式下继续工作。当 USIDL = 1 时，模块在空闲模式下停止工作。

表 16-1: dsPIC30F3014/4013 UART1 寄存器映射⁽¹⁾

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
U1MODE	020C	UARTEN	—	USIDL	—	—	ALTIO	—	—	WAKE	LPBACK	ABAUD	—	—	PDSEL1	PDSEL0	STSEL	0000 0000 0000 0000
U1STA	020E	UTXISEL	—	—	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0000 0001 0001 0000
U1TXREG	0210	—	—	—	—	—	—	—	UTX8	发送寄存器								0000 000u uuuu uuuu
U1RXREG	0212	—	—	—	—	—	—	—	URX8	接收寄存器								0000 0000 0000 0000
U1BRG	0214	波特率发生器预分频器																0000 0000 0000 0000

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 16-2: dsPIC30F3014/4013 UART2 寄存器映射⁽¹⁾

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
U2MODE	0216	UARTEN	—	USIDL	—	—	—	—	—	WAKE	LPBACK	ABAUD	—	—	PDSEL1	PDSEL0	STSEL	0000 0000 0000 0000
U2STA	0218	UTXISEL	—	—	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0000 0001 0001 0000
U2TXREG	021A	—	—	—	—	—	—	—	UTX8	发送寄存器								0000 000u uuuu uuuu
U2RXREG	021C	—	—	—	—	—	—	—	URX8	接收寄存器								0000 0000 0000 0000
U2BRG	021E	波特率发生器预分频器																0000 0000 0000 0000

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

17.0 CAN 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

17.1 概述

控制器局域网 (Controller Area Network, CAN) 模块是一种串行接口，用于与其他 CAN 模块或单片机通信。这一接口 / 协议是针对在噪声环境下进行通信而设计的。

CAN 模块就是一个通信控制器，实现了 BOSCH 规范中定义的 CAN 2.0A 或 CAN 2.0B 协议。该模块还支持 CAN 1.2、CAN 2.0A、CAN 2.0B Passive 和 CAN 2.0B Active 版本，实现了一种完整的 CAN 系统。但是本数据手册不讨论 CAN 规范。更多详细信息请参见 BOSCH CAN 规范。

该模块具有以下特性：

- 实现了 CAN 协议 CAN 1.2、CAN 2.0A 和 CAN 2.0B
- 标准和扩展的数据帧
- 0 至 8 字节数据长度
- 高达 1 Mb/s 的可编程比特率
- 支持远程帧
- 双重缓冲接收器，带有两个区分优先级的接收报文存储缓冲区（每个缓冲区最多可包含 8 字节的数据）
- 6 个完全（标准 / 扩展标识符）接收过滤器，其中 2 个与高优先级的接收缓冲区相关联，另外 4 个与低优先级的接收缓冲区相关联
- 2 个完全接收过滤屏蔽寄存器，其中一个与高优先级的接收缓冲区相关联，另一个与低优先级的接收缓冲区相关联
- 三个发送缓冲区，可由应用指定优先级并带有中止功能（每个缓冲区最多可包含 8 字节的数据）
- 集成了低通滤波器的可编程唤醒功能
- 支持自检操作的可编程环回模式
- 通过中断功能，为所有 CAN 接收器和发送器错误状态发送信号
- 可编程时钟源
- 到输入捕捉模块（对于 CAN1 和 CAN2 为 IC2）的可编程链接，可标记时间戳和进行网络同步
- 低功耗休眠和空闲模式

CAN 总线模块由一个协议引擎和报文缓冲 / 控制组成。CAN 协议引擎自动处理在 CAN 总线接收和发送报文的所有功能。必须首先装载相应的数据寄存器才可以开始发送报文。通过读取相应的寄存器可以检测总线的状态和判断是否发生了错误。对 CAN 总线上检测到的任何报文进行错误检查，随后将其与过滤条件进行比对，以确定该报文是否应该被接收并存储在两个接收寄存器之一中。

17.2 帧类型

CAN 模块发送各种类型的帧，这些帧包括由用户发出的数据报文或远程发送请求，以及其他在控制过程自动生成的帧。支持下列帧类型：

- **标准数据帧：**
当节点希望发送数据时会产生一个标准数据帧。该帧包含一个 11 位的标准标识符 (Standard Identifier, SID)，而不是一个 18 位的扩展标识符 (Extend Identifier, EID)。
- **扩展数据帧：**
扩展数据帧与标准数据帧类似，但它还包括一个扩展标识符。
- **远程帧：**
也可能发生目标节点向源节点请求发送数据的情况。要做到这一点，目标节点应先发送一个带标识符的远程帧，此标识符与所需的数据帧相匹配，然后，相应的数据源节点发送一个数据帧，作为对该远程请求的响应。
- **错误帧：**
错误帧由检测到总线错误的任一节点生成。错误帧包含 2 个字段：即错误标志字段和错误定界符字段。
- **过载帧：**
节点在两种情况下会产生过载帧。第一种情况，节点在帧间间隔检测到非法显性位。第二种情况，由于内部原因，节点尚无法开始接收下一条报文。节点最多可产生两条连续过载帧来延迟下一条报文的接收。
- **帧间间隔：**
帧间间隔将前一个帧（无论何种类型）与其后的数据帧或远程帧分隔开来。

17.3 工作模式

CAN 模块可工作在以下几种由用户选择的工作模式下。这些模式包括：

- 初始化模式
- 禁止模式
- 正常工作模式
- 监听模式
- 环回模式
- 错误识别模式

通过设置 REQOP<2:0> 位 (CiCTRL<10:8>) 可以选择所需的模式。通过监视 OPMODE<2:0> 位 (CiCTRL<7:5>) 判断是否已经进入某模式。在允许模式改变之前，通常在由至少 11 位连续的隐性位确定的总线空闲时间内，模块不会改变模式和 OPMODE 位。

17.3.1 初始化模式

在初始化模式下，该模块不执行发送或接收操作。此时，错误计数器被清零，且中断标志位保持不变。编程器访问在其它模式下访问受限的配置寄存器。此模块可防止用户因编程错误而意外违反 CAN 协议。当模块在线时，所有控制模块配置的寄存器都不能被修改。发送时，不允许 CAN 模块进入配置模式。配置模式作为保护以下寄存器的锁。

- 所有模块控制寄存器
- 波特率和中断配置寄存器
- 总线时序寄存器
- 标识符接收过滤寄存器
- 标识符接收屏蔽寄存器

17.3.2 禁止模式

在禁止模式下，模块不执行发送或接收操作。当总线上有活动时，模块可将 WAKIF 位置 1，但是等待处理的中断将继续等待且错误计数器也将保持不变。

如果 REQOP<2:0> 位 (CiCTRL<10:8>) 被设置为 001，模块将进入模块禁止模式。如果模块处于活动状态，它将等待 CAN 总线上出现 11 位隐性位，检测空闲总线的条件，然后接受模块禁止命令。OPMODE<2:0> 位 (CiCTRL<7:5>) = 001 表明模块成功进入了模块禁止模式。当模块处于模块禁止模式时，I/O 引脚将恢复为普通 I/O 功能。

当模块或 CPU 处于休眠模式时，通过对模块进行编程可以在 CiRX 输入线路中加入低通滤波功能。WAKFIL 位 (CiCFG2<14>) 使能或禁止该滤波器。

注： 通常，如果允许 CAN 模块在某种工作模式下进行发送操作，并且在 CAN 模块进入该操作模式后立即执行发送操作，则模块等待总线上出现 11 个连续隐性位后开始发送。如果用户在此 11 位周期内切换到禁止模式，则发送会被中止，同时将相应的 TXABT 位置 1，TXREQ 位清零。

17.3.3 正常工作模式

当 REQOP<2:0> = 000 时，选择正常工作模式。在该模式下，模块是激活的，且 I/O 引脚承担 CAN 总线功能。该模块通过 CxTX 和 CxRX 引脚发送和接收 CAN 总线报文。

17.3.4 监听模式

如果激活监听模式，则 CAN 总线上的模块处于被动状态。发送缓冲区恢复为端口 I/O 功能。接收引脚保持在输入状态。对于接收器，不发出错误标志或应答信号。该状态下，错误计数器失效。监听模式可用于检测 CAN 总线上的波特率。要使用此模式，至少必须有两个以上可以相互通信的节点。

17.3.5 监听所有报文模式

此模块可被设置为忽略所有错误并接受任何报文。通过将 REQOP<2:0> 位设置为 111 来激活监听所有报文模式。在此模式下，在发生错误之前位于报文组装缓冲区中的数据被复制到接收缓冲区中，并且可通过 CPU 接口读取。

17.3.6 环回模式

如果激活环回模式，则该模块将在模块边界将内部发送信号连接到内部接收信号。发送和接收引脚恢复为它们的端口 I/O 功能。

17.4 报文接收

17.4.1 接收缓冲区

CAN 总线模块有 3 个接收缓冲区。但其中的一个总是用于监视总线上是否有进入的报文。此缓冲区被称为报文组装缓冲区 (MAB)。因此, 只可见 2 个接收缓冲区 RXB0 和 RXB1, 它们基本上可以瞬间接收来自协议引擎的完整报文。

所有报文只有在符合接收过滤器标准时才被 MAB 组装并传送到 RXBn 缓冲区。接收到报文时, RXnIF 标志位 (CiINRF<0> 或 CiINRF<1>) 被置 1。只有在接收到报文时模块才能将此位置 1。处理完缓冲区中的报文后, 由 CPU 将此位清零。如果将 RXnIE 位 (CiINTE<0> 或 CiINTE<1>) 置 1, 则当接收报文时会产生中断。

过滤器 RXF0、RXF1 和过滤器屏蔽寄存器 RXM0 与 RXB0 相关联。过滤器 RXF2、RXF3、RXF4 和 RXF5 和屏蔽寄存器 RXM1 与 RXB1 相关联。

17.4.2 报文接收过滤器

报文接收过滤器和屏蔽寄存器用于决定报文组装缓冲区中的报文是否应该被装入两个接收缓冲区中的任何一个。一旦报文组装缓冲区 (MAB) 接收到有效的报文, 就会将该报文的标识符字段与过滤值进行比较。如果匹配的话, 该报文就会被装入相应的接收缓冲区。

接收过滤器通过 RXIDE 位 (CiRXnSID<0>) 检查进入的报文以决定如何比较标识符。如果 RXIDE 位清零, 该报文就是一个标准帧, 而且只与 EXIDEN 位 (CiRXFnSID<0>) 清零的过滤器相比较。如果 RXIDE 位置 1, 该报文就是一个扩展帧, 而且只与 EXIDEN 位置 1 的过滤器相比较。

17.4.3 报文接收过滤屏蔽寄存器

屏蔽位可以决定对哪些位应用过滤器。如果任何屏蔽位被置为零, 则无论过滤位为何值该位都会被自动接收。有两个与接收缓冲区相关的可编程接收过滤屏蔽寄存器, 它们分别与两个接收缓冲区相对应。

17.4.4 接收溢出

当报文组装缓冲区 (MAB) 组装了一个有效的接收报文后, 报文通过接收过滤器被接收, 此时如果与该过滤器相关的接收缓冲区还未清除上一次的报文, 将会发生溢出。

此时溢出错误标志位 RXnOVR (CiINTF<15> 或 CiINTF<14>) 和 ERRIF 位 (CiINTF<5>) 被置 1, 并且丢弃 MAB 中的报文。

如果 DBEN 位被清零, 则 RXB1 和 RXB0 独立工作。这种情况下, 即使 RXB0 包含未读报文, 为 RXB0 指定的报文也不会被发送到 RXB1, 同时将 RX0OVR 位置 1。

如果 DBEN 位被置 1, 则采用不同的方法处理 RXB0 溢出。如果 RXB0 接收为有效报文且 RXFUL = 1, 则表明 RXB0 为满, 如果 RXFUL = 0, 则表明 RXB1 为空, 将 RXB0 的报文装入 RXB1 中。RXB0 不产生溢出错误。如果 RXB0 接收为有效报文且 RXFUL = 1, 则表明 RXB0 和 RXB1 都为满, 报文丢失且 RXB1 产生溢出。

17.4.5 接收错误

CAN 模块将会检测到以下接收错误:

- 循环冗余校验 (Cyclic Redundancy Check, CRC) 错误
- 位填充错误
- 无效报文接收错误

发生上述错误之中的任何一种错误, 接收错误计数器的值就会加 1。RXWAR 位 (CiINTF<9>) 表明接收错误计数器已经达到 CPU 的警告上限值 96, 并产生中断。

17.4.6 接收中断

接收中断主要可以分成 3 类, 每类包含了产生中断的不同情况:

- 接收中断:
报文已被成功接收, 并装入到其中一个接收缓冲区。接收到帧结束 (End-of-Frame, EOF) 字段之后立即激活中断。读取 RXnIF 标志位可知是由哪个接收缓冲区所引起的中断。
- 唤醒中断:
从禁止模式下唤醒 CAN 模块, 或从休眠模式下唤醒器件。

- 接收错误中断：
ERRIF 位表示接收错误中断。由它显示有错误条件发生。通过检查 CAN 中断寄存器 CiINTF 中的相应位，就可以确定错误源。
 - 收到无效报文：
如果在接收上一个报文期间发生了任何类型的错误，IVRIF 位都将指出有错误发生。
 - 接收溢出：
RXnOVR 位表明有溢出情况发生。
 - 接收器警告：
RXWAR 位表明接收错误计数器 (RERRCNT<7:0>) 已达到了警告上限值 96。
 - 接收器错误被动：
RXBP 位表明接收错误计数器已经超过了错误的上限值 127，且该模块已经进入错误被动状态。

17.5 报文发送

17.5.1 发送缓冲区

CAN 模块有 3 个发送缓冲区。每个缓冲区都可容纳 14 字节的数据。其中 8 个字节用于存放发送的报文（最大 8 个字节），还有 5 个字节用于存放标准或扩展标识符以及其他报文仲裁信息。

17.5.2 发送报文优先级

发送优先级指在各个节点内待发送报文的优先级。发送优先级有 4 级。如果某个报文缓冲区的 TXPRI<1:0> (CiTXnCON<1:0>)，其中 n = 0、1 或 2，表示特定发送缓冲区) 被置为 11，则该缓冲区具有最高的优先级。如果某个报文缓冲区的 TXPRI<1:0> 被置为 10 或 01，则该缓冲区具有中等优先级。如果某个报文缓冲区的 TXPRI<1:0> 被置为 00，则该缓冲区具有最低优先级。

17.5.3 发送过程

要启动报文的发送，必须将 TXREQ 位 (CiTXnCON<3>) 置 1。CAN 总线模块解决了置 1 TXREQ 位与帧起始 (Start-of-Frame, SOF) 之间的时序冲突，确保当优先级改变时，能在发送 SOF 之前正确解决时序冲突。当 TXREQ 置 1 时，TXABT (CiTXnCON<6>)、TXLARB (CiTXnCON<5>) 和 TXERR (CiTXnCON<4>) 标志位自动清零。

将 TXREQ 位置 1 即标志一个报文缓冲区正在排队等待发送。当模块检测到可用的总线时，开始发送设定为具有最高优先级的报文。

如果第一次尝试发送就成功完成，则 TXREQ 位将自动清零，且如果 TX1IE 位被置 1，将产生中断。

如果报文发送失败，其中一个错误状态标志位将被置 1，TXREQ 位将保持置 1，这表明该报文仍处于等待发送状态。如果在尝试发送报文时遇到出错情况，TXERR 位将被置 1，且此出错情况可能引起中断。如果在尝试发送的过程中，报文丢失仲裁，TXLARB 位将被置 1，此时不会产生任何告知仲裁失败的中断。

17.5.4 中止报文发送

系统可通过清零与每个报文缓冲区相关的 TXREQ 位中止报文发送。将 ABAT 位 (CiCTRL<12>) 置 1 将请求中止所有待发送的报文。如果报文还未开始发送或者报文已开始发送但由于丢失仲裁或错误而被中断，那么会执行中止。当模块将 TXABT 位置 1 且 TXnIF 标志位不自动置 1 时，表明发生了中止。

17.5.5 发送错误

CAN 模块检测以下发送错误：

- 应答错误
- 格式错误
- 位错误

这些发送错误不一定会产生中断，但是发送错误计数器会进行计数以表明有发送错误产生。上述每个错误都将引起错误计数器加 1。一旦错误计数器的值超过 96，ERRIF (CiINTF<5>) 和 TXWAR 位 (CiINTF<10>) 将被置 1。一旦错误计数器的值超过 96，将产生中断且将错误标志寄存器中的 TXWAR 位置 1。

17.5.6 发送中断

发送中断主要分成 2 类，每类包含了产生中断的不同情况：

- 发送中断：
三个发送缓冲区中至少有一个为空（非预定的），并且可以装入预定发送的报文。读 TXnIF 标志可以确定可用的发送缓冲区和引起中断的缓冲区。
- 发送错误中断：
发送错误中断由 ERRIF 标志位指出。此标志位显示有错误情况发生。通过检查 CAN 中断寄存器 CiINTF 中的错误标志位，就可以确定错误源。该寄存器的标志位与接收和发送错误有关。
 - 发送器警告中断：
TXWAR 位指示发送错误计数器已达到 CPU 警告上限值 96。
 - 发送器错误被动：
TXEP 位（CiINTF<12>）指示发送错误计数器已经超过了错误被动的上限值 127，且该模块已经进入错误被动状态。
 - 总线关断：
TXBO 位（CiINTF<13>）指示发送错误计数器（TERRCNT<7:0>）已超过了 255，且该模块已进入了总线关断状态。

17.6 波特率设置

任何一条 CAN 总线上的所有节点都必须有相同的标称比特率。要设置波特率，必须初始化以下参数：

- 同步跳转宽度
- 波特率预分频比
- 相位段
- 相位段 2 的长度决定位
- 采样点
- 传播段位

17.6.1 位时序

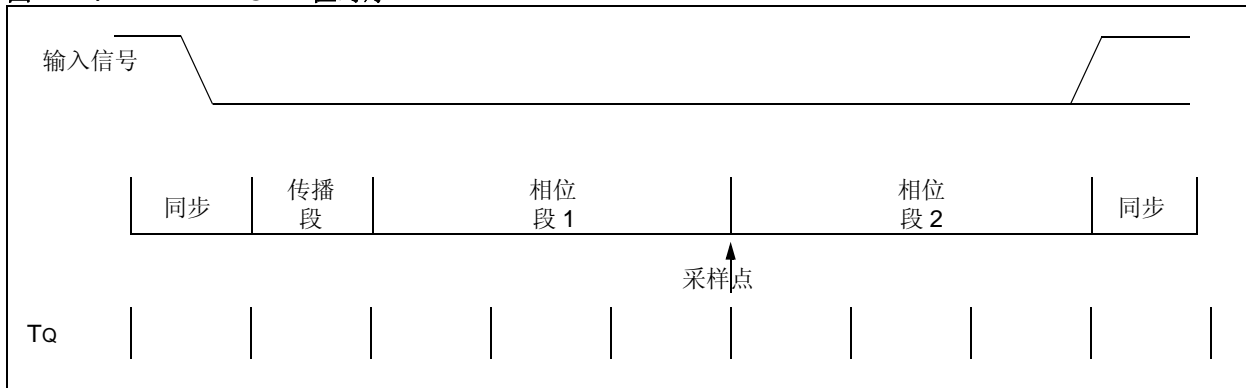
CAN 总线上的所有控制器必须具有相同的波特率和位长度。但不同的控制器并不要求使用相同的主振荡器时钟。由于各个控制器的时钟频率不同，必须通过调节各个段的时间份额数来调节波特率。

可以认为标称位时间分成几个不重叠的时间段。这些段如图 17-2 中所示。

- 同步段（Sync Seg）
- 传播时间段（Prop Seg）
- 相位段 1（Phase1 Seg）
- 相位段 2（Phase2 Seg）

时间段以及标称位时间都是由整数个时间单元组成，这些单元称作时间份额或 Tq。根据定义，标称位时间最少由 8 个 Tq 组成，最多由 25 个 Tq 组成。而且根据定义，最小标称位时间为 1 μs，而最大位速率为 1 MHz。

图 17-2: CAN 位时序



17.6.2 预分频器设置

模块带有一个可编程预分频器，其整数预分频值范围为 1 至 64，除对时钟进行固定的二分频外还可提供其他时钟选项。时间份额 (Tq) 是振荡器周期的固定单元，如公式 17-1 中所示，其中，FCAN 等于 Fcy (CANCKs 位置 1 时) 或 4Fcy (CANCKs 位清零时)。

注： FCAN 不能超过 30 MHz。当 CANCKs = 0 时，Fcy 不能超过 7.5 MHz。

公式 17-1: 时钟发生器的时间份额

$$Tq = 2 (BRP<5:0> + 1) / FCAN$$

17.6.3 传播段

这部分的位时间用来补偿网络内的物理延迟时间。这些延迟时间包括总线线路上的信号传播时间以及节点的内部延迟时间。通过设置 PRSEG<2:0> 位 (CiCFG2<2:0>) 可将传播段设置为 1 Tq 至 8 Tq。

17.6.4 相位段

相位缓冲段用于将接收位的采样点定位在标称位时间内的最佳位置。采样点发生在相位段 1 与相位段 2 之间。可以通过重新同步延长或缩短这两个段。相位段 1 的末尾决定一个比特时间内的采样点。相位段 1 的持续时间可编程为 1 Tq 到 8 Tq。相位段 2 提供在发送下一数据前的延时，其持续时间也可被编程为 1 Tq 到 8 Tq。或被定义为相位段 1 与信息处理时间 (2 Tq) 两者中的较大者。通过将 SEG1PH<2:0> 位 (CiCFG2<5:3>) 置 1 来初始化相位段 1，将 SEG2PH<2:0> 位 (CiCFG2<10:8>) 置 1 来初始化相位段 2。

设置相位段的长度时必须满足以下要求：

$$\text{传播段} + \text{相位段 1} \geq \text{相位段 2}$$

17.6.5 采样点

采样点是读总线电平并确定接收位的值的一个时间点。采样点发生在相位段 1 的末尾。如果比特时序较慢且包含很多 Tq，可以在采样点对总线线路进行多次采样。接收位的值被确定为采样到的三个值中出现次数最多的那个值。出现次数最多的采样发生在采样点，另外两次采样发生在采样点之前且采样相隔为 Tq/2。通过置 1 或清零 SAM 位 (CiCFG2<6>)，CAN 模块允许用户选择在同一点采样 3 次或 1 次。

通常，位采样应发生在位时间的 60-70% 附近，这取决于系统参数。

17.6.6 同步

为了补偿不同总线站振荡器频率的相移，每个 CAN 控制器必须能够与输入信号的相关信号沿同步。当检测到发送数据中的一个沿时，逻辑电路会将该沿与预期时间 (同步段) 比较。随后，电路将根据需要调整相位段 1 和相位段 2 的值。使用两种机制进行同步。

17.6.6.1 硬同步

仅当总线空闲期间有一个从隐性到显性的边沿出现时，才执行硬同步，它指示报文传输的开始。硬同步后，位时间计数器从同步段重新开始计数。硬同步强制该边沿处于重新开始的位时间的同步段之内。如果产生硬同步，则在相应的位时间内不能再有重新同步。

17.6.6.2 重新同步

重新同步可能使相位段 1 延长或使相位段 2 缩短。相位缓冲段延长或缩短的上限由同步跳转宽度给出，该值由 SJW<1:0> 位 (CiCFG1<7:6>) 指定。同步跳转宽度的值添加到相位段 1 或从相位段 2 中减去。重新同步跳转宽度可编程为 1 Tq 到 4 Tq。

设置 SJW<1:0> 位时，必须满足以下要求：

$$\text{相位段 2} > \text{同步跳转宽度}$$

表 17-1: dsPIC30F4013 CAN1 寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态	
C1RXF0SID	0300	—	—	—	接收过滤器 0 标准标识符 <10:0>												—	EXIDE	000u uuuu uuuu uu0u
C1RXF0EIDH	0302	—	—	—	接收过滤器 0 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXF0EIDL	0304	接收过滤器 0 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF1SID	0308	—	—	—	接收过滤器 1 标准标识符 <10:0>												—	EXIDE	000u uuuu uuuu uu0u
C1RXF1EIDH	030A	—	—	—	接收过滤器 1 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXF1EIDL	030C	接收过滤器 1 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF2SID	0310	—	—	—	接收过滤器 2 标准标识符 <10:0>												—	EXIDE	000u uuuu uuuu uu0u
C1RXF2EIDH	0312	—	—	—	接收过滤器 2 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXF2EIDL	0314	接收过滤器 2 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF3SID	0318	—	—	—	接收过滤器 3 标准标识符 <10:0>												—	EXIDE	000u uuuu uuuu uu0u
C1RXF3EIDH	031A	—	—	—	接收过滤器 3 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXF3EIDL	031C	接收过滤器 3 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF4SID	0320	—	—	—	接收过滤器 4 标准标识符 <10:0>												—	EXIDE	000u uuuu uuuu uu0u
C1RXF4EIDH	0322	—	—	—	接收过滤器 4 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXF4EIDL	0324	接收过滤器 4 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF5SID	0328	—	—	—	接收过滤器 5 标准标识符 <10:0>												—	EXIDE	000u uuuu uuuu uu0u
C1RXF5EIDH	032A	—	—	—	接收过滤器 5 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXF5EIDL	032C	接收过滤器 5 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM0SID	0330	—	—	—	接收屏蔽寄存器 0 标准标识符 <10:0>												—	MIDE	000u uuuu uuuu uu0u
C1RXM0EIDH	0332	—	—	—	接收屏蔽寄存器 0 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXM0EIDL	0334	接收屏蔽寄存器 0 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM1SID	0338	—	—	—	接收屏蔽寄存器 1 标准标识符 <10:0>												—	MIDE	000u uuuu uuuu uu0u
C1RXM1EIDH	033A	—	—	—	接收屏蔽寄存器 1 扩展标识符 <17:6>														0000 uuuu uuuu uuuu
C1RXM1EIDL	033C	接收屏蔽寄存器 1 扩展标识符 <5:0>						—	—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1TX2SID	0340	发送缓冲区 2 标准标识符 <10:6>						—	—	—	发送缓冲区 2 标准标识符 <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX2EID	0342	发送缓冲区 2 扩展标识符 <17:14>						—	—	发送缓冲区 2 扩展标识符 <13:6>								uuuu 0000 uuuu uuuu	
C1TX2DLC	0344	发送缓冲区 2 扩展标识符 <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>			—	—	—			uuuu uuuu uuuu u000
C1TX2B1	0346	发送缓冲区 2 字节 1						发送缓冲区 2 字节 0										uuuu uuuu uuuu uuuu	
C1TX2B2	0348	发送缓冲区 2 字节 3						发送缓冲区 2 字节 2										uuuu uuuu uuuu uuuu	
C1TX2B3	034A	发送缓冲区 2 字节 5						发送缓冲区 2 字节 4										uuuu uuuu uuuu uuuu	
C1TX2B4	034C	发送缓冲区 2 字节 7						发送缓冲区 2 字节 6										uuuu uuuu uuuu uuuu	
C1TX2CON	034E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>		0000 0000 0000 0000	
C1TX1SID	0350	发送缓冲区 1 标准标识符 <10:6>						—	—	—	发送缓冲区 1 标准标识符 <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX1EID	0352	发送缓冲区 1 扩展标识符 <17:14>						—	—	发送缓冲区 1 扩展标识符 <13:6>								uuuu 0000 uuuu uuuu	
C1TX1DLC	0354	发送缓冲区 1 扩展标识符 <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>			—	—	—			uuuu uuuu uuuu u000

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 欲知更多有关寄存器位域的信息, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 17-1: dsPIC30F4013 CAN1 寄存器映射⁽¹⁾ (续)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态	
C1TX1B1	0356	发送缓冲区 1 字节 1								发送缓冲区 1 字节 0								uuuu uuuu uuuu uuuu	
C1TX1B2	0358	发送缓冲区 1 字节 3								发送缓冲区 1 字节 2								uuuu uuuu uuuu uuuu	
C1TX1B3	035A	发送缓冲区 1 字节 5								发送缓冲区 1 字节 4								uuuu uuuu uuuu uuuu	
C1TX1B4	035C	发送缓冲区 1 字节 7								发送缓冲区 1 字节 6								uuuu uuuu uuuu uuuu	
C1TX1CON	035E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>	0000 0000 0000 0000		
C1TX0SID	0360	发送缓冲区 0 标准标识符 <10:6>						—	—	—	发送缓冲区 0 标准标识符 <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX0EID	0362	发送缓冲区 0 扩展标识符 <17:14>				—	—	—	—	发送缓冲区 0 扩展标识符 <13:6>								uuuu 0000 uuuu uuuu	
C1TX0DLC	0364	发送缓冲区 0 扩展标识符 <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>				—	—	—	uuuu uuuu uuuu u000	
C1TX0B1	0366	发送缓冲区 0 字节 1								发送缓冲区 0 字节 0								uuuu uuuu uuuu uuuu	
C1TX0B2	0368	发送缓冲区 0 字节 3								发送缓冲区 0 字节 2								uuuu uuuu uuuu uuuu	
C1TX0B3	036A	发送缓冲区 0 字节 5								发送缓冲区 0 字节 4								uuuu uuuu uuuu uuuu	
C1TX0B4	036C	发送缓冲区 0 字节 7								发送缓冲区 0 字节 6								uuuu uuuu uuuu uuuu	
C1TX0CON	036E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>	0000 0000 0000 0000		
C1RX1SID	0370	—	—	—	接收缓冲区 1 标准标识符 <10:0>										SRR	RXIDE	000u uuuu uuuu uuuu		
C1RX1EID	0372	—	—	—	接收缓冲区 1 扩展标识符 <17:6>										0000 uuuu uuuu uuuu				
C1RX1DLC	0374	接收缓冲区 1 扩展标识符 <5:0>						RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>				uuuu uuuu 000u uuuu	
C1RX1B1	0376	接收缓冲区 1 字节 1								接收缓冲区 1 字节 0								uuuu uuuu uuuu uuuu	
C1RX1B2	0378	接收缓冲区 1 字节 3								接收缓冲区 1 字节 2								uuuu uuuu uuuu uuuu	
C1RX1B3	037A	接收缓冲区 1 字节 5								接收缓冲区 1 字节 4								uuuu uuuu uuuu uuuu	
C1RX1B4	037C	接收缓冲区 1 字节 7								接收缓冲区 1 字节 6								uuuu uuuu uuuu uuuu	
C1RX1CON	037E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	FILHIT<2:0>		0000 0000 0000 0000		
C1RX0SID	0380	—	—	—	接收缓冲区 0 标准标识符 <10:0>										SRR	RXIDE	000u uuuu uuuu uuuu		
C1RX0EID	0382	—	—	—	接收缓冲区 0 扩展标识符 <17:6>										0000 uuuu uuuu uuuu				
C1RX0DLC	0384	接收缓冲区 0 扩展标识符 <5:0>						RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>				uuuu uuuu 000u uuuu	
C1RX0B1	0386	接收缓冲区 0 字节 1								接收缓冲区 0 字节 0								uuuu uuuu uuuu uuuu	
C1RX0B2	0388	接收缓冲区 0 字节 3								接收缓冲区 0 字节 2								uuuu uuuu uuuu uuuu	
C1RX0B3	038A	接收缓冲区 0 字节 5								接收缓冲区 0 字节 4								uuuu uuuu uuuu uuuu	
C1RX0B4	038C	接收缓冲区 0 字节 7								接收缓冲区 0 字节 6								uuuu uuuu uuuu uuuu	
C1RX0CON	038E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	DBEN	JTOFF	FILHITO	0000 0000 0000 0000	
C1CTRL	0390	CANCAP	—	CSIDL	ABAT	CANCKS	REQOP<2:0>			OPMODE<2:0>			—	ICODE<2:0>			—	0000 0100 1000 0000	
C1CFG1	0392	—	—	—	—	—	—	—	—	SJW<1:0>			BRP<5:0>					0000 0000 0000 0000	
C1CFG2	0394	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>			0u00 0uuu uuuu uuuu	
C1INTF	0396	RX0OVR	RX1OVR	TXBO	TXEP	RXEP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	TX2IF	TX1IF	TX0IF	RX1IF	RX0IF	0000 0000 0000 0000	
C1INTE	0398	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	TX2IE	TX1IE	TX0IE	RX1IE	RX0IE	0000 0000 0000 0000	
C1EC	039A	TERRCNT<7:0>								RERRCNT<7:0>								0000 0000 0000 0000	

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 欲知更多有关寄存器位域的信息, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

18.0 数据转换器接口 (DCI) 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能, 但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

18.1 模块介绍

dsPIC30F 数据转换器接口 (Data Converter Interface, DCI) 模块允许与其他设备简单接口, 这些设备包括音频编码/解码器 (编解码器)、A/D 转换器和 D/A 转换器等。支持以下接口:

- 帧同步串行传输 (单通道或多通道)
- Inter-IC Sound (I²S) 接口
- AC-Link 兼容模式

DCI 模块提供以下通用特性:

- 可编程字长度, 最大为 16 位
- 最多支持 16 个时隙, 帧长最大为 256 位
- 数据缓冲最高可接受 4 次采样, 无 CPU 开销

18.2 模块 I/O 引脚

有 4 个 I/O 引脚与此模块相关。使能时, 模块可控制这 4 个引脚的数据方向。

18.2.1 CSCK 引脚

CSCK 引脚为 DCI 模块提供串行时钟。使用 DCICON1 SFR 中的 CSCKD 控制位可将 CSCK 引脚配置为输入或输出。配置为输出时, 串行时钟由 dsPIC30F 提供。配置为输入时, 串行时钟必须由外部器件提供。

18.2.2 CSDO 引脚

使能此模块时, 串行数据输出 (CSDO) 引脚被配置为只能输出的引脚。发送数据时, 由 CSDO 引脚驱动串行总线。不发送数据时, 在 CSCK 周期内, CSDO 引脚是三态或驱动为 0, 这取决于 CSDOM 控制位的状态。这将允许其他器件在 DCI 模块不使用的发送周期内, 将数据放置到串行总线上。

18.2.3 CSDI 引脚

使能此模块时, 串行数据输入 (CSDI) 引脚被配置为只能输入的引脚。

18.2.3.1 COFS 引脚

编解码帧同步 (Codec Frame Synchronization, COFS) 引脚用于同步发生在 CSDO 和 CSDI 引脚上的数据传输。COFS 引脚可配置为输入或输出。COFS 引脚的数据方向由 DCICON1 寄存器中的 COFSD 控制位确定。

当 CPU 正在访问由存储器映射的缓冲寄存器时, DCI 模块访问影子寄存器。

18.2.4 缓冲区数据对齐

因为大多数编解码数据是用有符号二进制补码小数表示的, 所以数据值在缓冲区中总是以左对齐方式储存。如果接收到的字长度小于 16 位, 则接收缓冲寄存器中未使用的低位由模块设置为 0。如果发送的字长度小于 16 位, 则模块将忽略发送缓冲寄存器中未使用的低位。有关字长的设置将在本文档中的随后部分中介绍。

18.2.5 发送/接收移位寄存器

DCI 模块有一个 16 位的移位寄存器, 用于将串行数据移入或移出模块。因为音频 PCM 数据是以有符号二进制补码格式发送的, 所以数据首先移入/移出移位寄存器的高位。

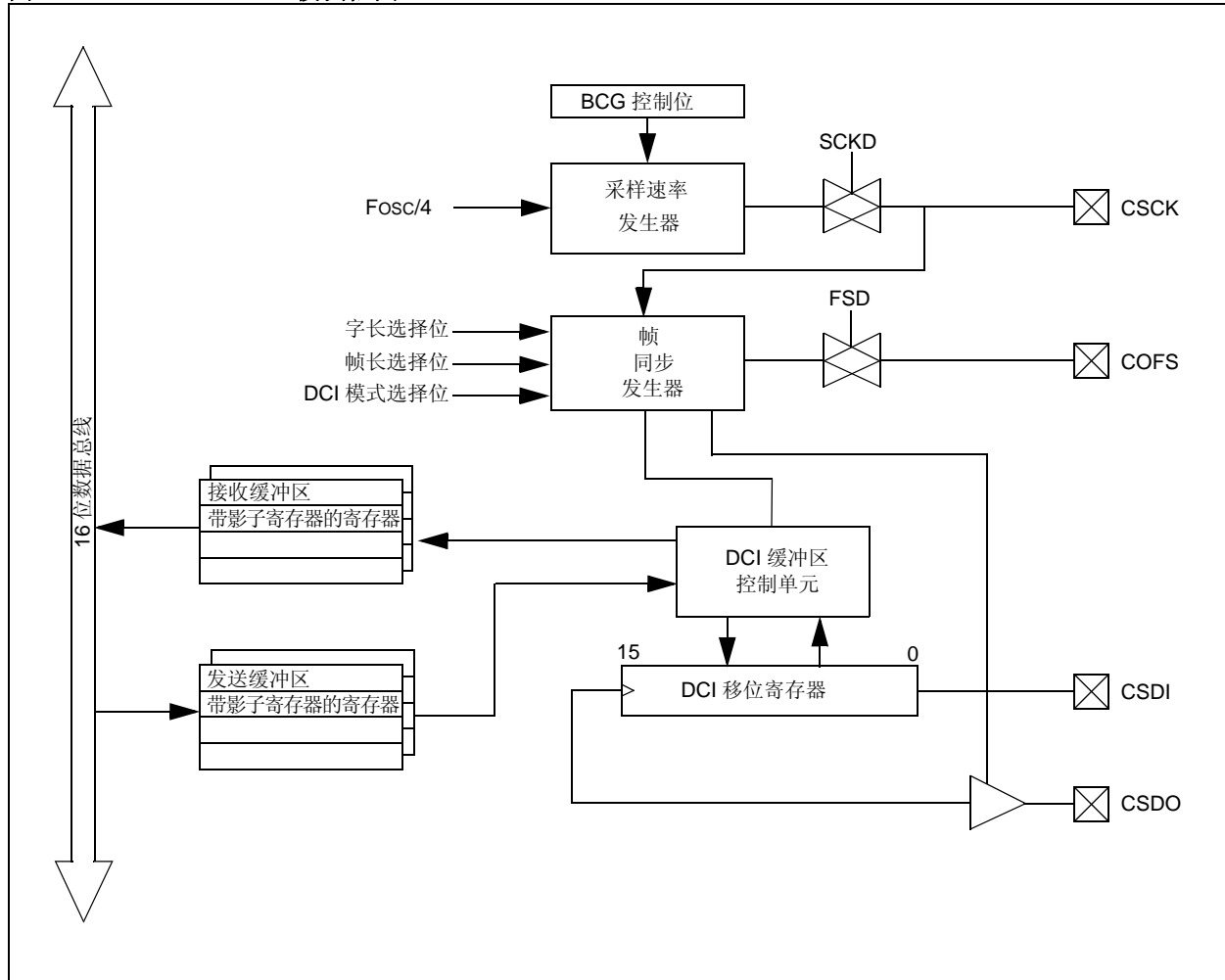
18.2.6 DCI 缓冲区控制

DCI 模块包括一个缓冲区控制单元, 用于在影子缓冲存储区和串行移位寄存器之间传输数据。缓冲控制单元是一个简单的 2 位地址计数器, 指向影子缓冲存储区的字存储单元。对于接收存储空间 (DCI 缓冲存储区的高地址部分), 地址计数器与位于 MSb 位置的 0 连接以构成一个 3 位的地址。对于发送存储空间 (DCI 缓冲存储器的高地址部分), 地址计数器与位于 MSb 位置的 1 连接。

注: DCI 缓冲区控制单元始终访问发送和接收缓冲区中的相同相对位置, 因此仅提供一个地址计数器。

dsPIC30F3014/4013

图 18-1: DCI 模块框图



18.3 DCI 模块操作

18.3.1 模块使能

通过置 1/清零 DCICON1 SFR 中的 DCIEN 控制位使能或禁止 DCI 模块。清零 DCIEN 控制位能复位模块。尤其是，所有与 CSCK 发生、帧同步和 DCI 缓冲控制单元相关的计数器都将被复位。

清零 DCIEN 位时 DCI 时钟将被关闭。

使能时，DCI 控制与模块相关的 4 个 I/O 引脚的数据方向。当 DCIEN 位置 1 时，这些引脚的端口、LAT 和 TRIS 寄存器值都会被 DCI 模块改写。

使能位时钟发生器时，有可能单独改写 CSCK 引脚。这将允许不使能 DCI 模块的其他部分而让位时钟发生器工作。

18.3.2 字长选择位

DCICON2 SFR 中的 WS<3:0> 字长选择位确定各 DCI 数据字中的位数。本质上，WS<3:0> 位确定了以 CSCK 信号作为时钟的 4 位计数器的计数周期。

可选择任何数据长度，最长为 16 位。载入到 WS<3:0> 位的值比期望字长小 1。例如，当 WS<3:0> = 1111 时，选择 16 位的数据字长。

注： WS<3:0> 控制位仅用在多通道和 I²S 模式中。在 AC-Link 模式下，因为数据间隙大小由协议固定，所以这些位在 AC-Link 模式下不起作用。

18.3.3 帧同步发生器

帧同步发生器 (COFSG) 是一个 4 位计数器，设置数据字中的帧长。帧同步发生器在字长计数器每次复位时加 1 (见第 18.3.2 节“字长选择位”)。通过写 DCICON2 SFR 中的 COFSG<3:0> 控制位设置帧同步发生器的周期。时钟周期中的 COFSG 周期由以下公式确定：

公式 18-1: COFSG 周期

$$\text{帧长} = \text{字长} \cdot (\text{FSG 值} + 1)$$

可选的帧长度最大为 16 个数据字。CSCK 周期中的帧长随所选的字长而变化，最大可以为 256。

注： 因为协议将帧长设定为 256 个 CSCK 周期，所以 COFSG 控制位在 AC-Link 模式下不起作用。

18.3.4 帧同步模式控制位

通过使用 DCICON1 SFR 中的帧同步模式控制位 (COFSM<1:0>) 选择帧同步信号的类型。可选择以下工作模式：

- 多通道模式
- I²S 模式
- AC-Link 模式 (16 位)
- AC-Link 模式 (20 位)

COFSM 控制位的操作取决于 DCI 模块是作为主器件产生帧同步信号，还是作为从器件接收帧同步信号。

一对 DSP/编解码器中的主器件是产生帧同步信号的器件。帧同步信号启动 CSDI 和 CSDO 引脚上的数据传输，并且通常具有与数据采样速率 (COFS) 相同的频率。

如果 COFSD 控制位被清零，则 DCI 模块是一个帧同步主器件；如果 COFSD 控制位置 1，则 DCI 模块是一个帧同步从器件。

18.3.5 主器件帧同步操作

当 DCI 模块作为帧同步主器件工作 (COFSD = 0) 时，COFSM 模式位确定了由帧同步发生器逻辑产生的帧同步脉冲的类型。

当帧同步发生器复位为 0 时，会产生一个新的 COFS 信号。

在多通道模式下，帧同步脉冲在 CSCK 周期内被驱动为高电平，以启动数据传输。连续的帧同步脉冲间的 CSCK 周期数取决于字长和帧同步发生器控制位。多通道模式下帧同步信号的时序图如图 18-2 所示。

在 AC-Link 工作模式下，帧同步信号具有固定的周期和占空比。AC-Link 帧同步信号在 16 个 CSCK 周期内为高电平，在 240 个周期内为低电平。图 18-3 显示了带有 AC-Link 帧启动详细时序的时序图。

dsPIC30F3014/4013

在 I²S 模式下，产生占空比为 50% 的帧同步信号。CSCK 周期中的 I²S 帧同步信号周期由帧同步发生器控制位确定。新的 I²S 数据传输边界由 COFS 引脚上的由高到低或由低到高的跳变沿标记。

18.3.6 从器件帧同步操作

当 DCI 模块作为帧同步从器件工作 (COFSD = 1) 时，数据传输由连接到 DCI 模块的编解码器控制。COFSM 控制位控制 DCI 模块如何响应输入的 COFS 信号。

在多通道模式下，在 COFS 引脚采样到高电平后再经过一个 CSCK 周期后，开始传输新的数据帧 (见图 18-2)。COFS 引脚上的脉冲复位帧同步发生器逻辑。

在 I²S 模式下，在 COFS 引脚上采样到由低到高或由高到低的跳变后再经过一个 CSCK 周期后，传输新数据字。COFS 引脚上的上升沿或下降沿复位帧同步发生器逻辑。

在 AC-Link 模式下，在 COFS 引脚上采样到高电平后再经过一个 CSCK 周期后，传输下一个帧的标记时隙和随后的数据时隙。

当模块工作在从模式下时，必须配置 COFSG 和 WS 位以提供恰当的帧长。一旦模块在 COFS 引脚上采样到一个有效的帧同步脉冲，就会发生一个完整的数据帧传输。在数据帧传输完成前，模块将不再响应其他帧同步脉冲。

图 18-2: 帧同步时序，多通道模式

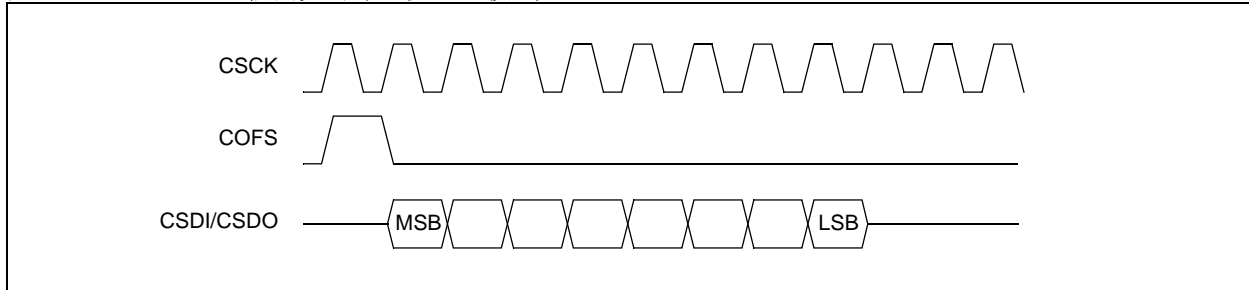


图 18-3: 帧同步时序，AC-LINK 帧起始

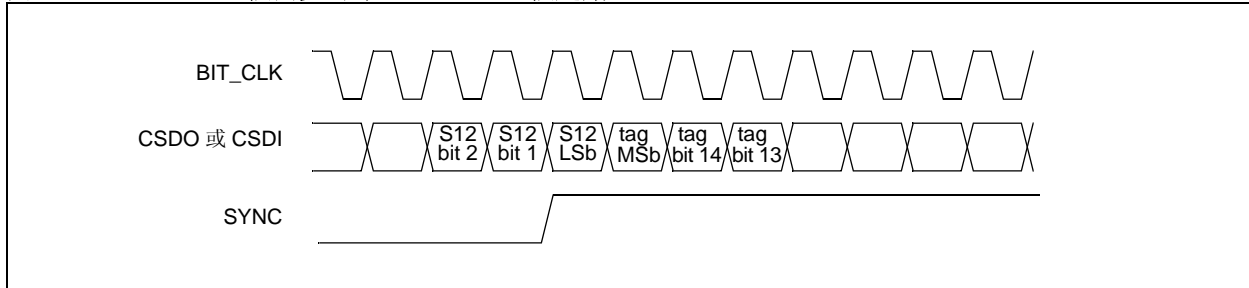
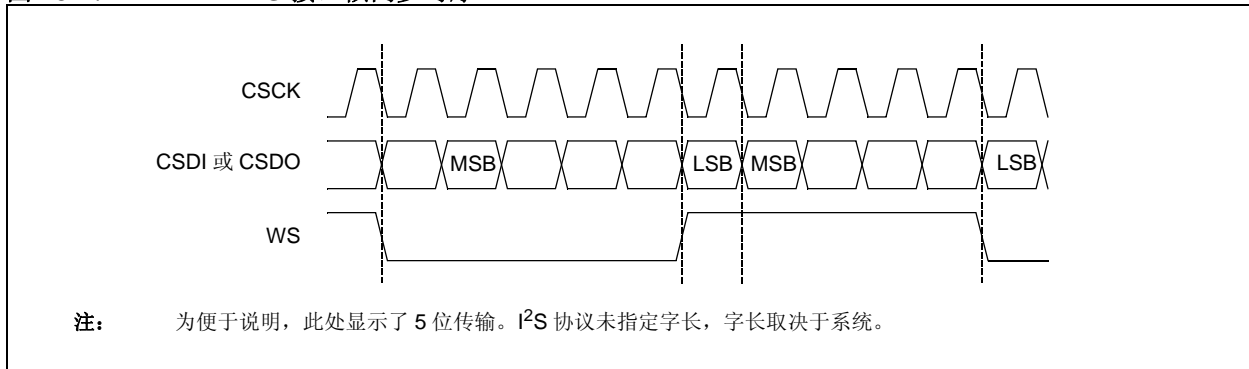


图 18-4: I²S 接口帧同步时序



18.3.7 位时钟发生器

DCI 模块具有一个专用的 12 位时基，用来产生位时钟。通过向 DCICON3 SFR 中的 BCG<11:0> 控制位写一个非零的 12 位值来设置时钟速率（周期）。

当 BCG<11:0> 位被设置为零时，位时钟被禁止。当 BCG<11:0> 位被设置为非零值时，位时钟发生器被使能。如果 DCI 的串行时钟是从外部器件接收的，则这些位应设置为 0，且 CSCKD 位应设置为 1。

公式 18-2 给出了位时钟频率的公式。

公式 18-2: 位时钟频率

$$F_{BCK} = \frac{F_{CY}}{2 \cdot (BCG + 1)}$$

所需的位时钟频率由系统采样速率和帧大小确定。根据数据转换器和所使用的通信协议，典型的位时钟频率范围在 16x 到 512x 转换器采样速率之间。

要获得与常见音频采样速率相关的位时钟频率，用户需要选择一个具有“偶”数二进制值的晶振频率。表 18-1 列出了这类晶振频率的示例。

表 18-1: 常见编解码器 CSCK 频率的器件频率

Fs (kHz)	Fcscck/Fs	Fcscck (MHz) ⁽¹⁾	Fosc (MHz)	PLL	Fcy (MIPS)	BCG ⁽²⁾
8	256	2.048	8.192	4	8.192	1
12	256	3.072	6.144	8	12.288	1
32	32	1.024	8.192	8	16.384	7
44.1	32	1.4112	5.6448	8	11.2896	3
48	64	3.072	6.144	16	24.576	3

- 注 1: 当从外部施加 CSCK 信号 (CSCKD = 1) 时，BCG<11:0> 位对 DCI 模块的工作不起作用。
- 注 2: 当从外部施加 CSCK 信号 (CSCKD = 1) 时，外部时钟高电平和低电平时间必须满足器件时序的要求。

18.3.8 采样时钟边沿控制位

采样时钟边沿 (CSCKE) 控制位确定 CSCK 信号的采样边沿。如果清零 CSCK 位 (默认), 则在 CSCK 信号的下降沿采样数据。AC-Link 协议和大多数多通道格式要求在 CSCK 信号的下降沿采样数据。如果 CSCK 位置 1, 则在 CSCK 信号的上升沿采样数据。I²S 协议要求在 CSCK 信号的上升沿采样数据。

18.3.9 数据对齐控制位

在大多数应用中, 在 COFS 信号采样为有效后再经过一个 CSCK 周期后开始数据传输。这是 DCI 模块的默认配置。通过将 DCICON1 SFR 中的 DJST 控制位置 1, 来选择其他数据对齐。DJST = 1 时, 若 COFS 信号采样为有效, 则将在同一 CSCK 周期内开始数据传输。

18.3.10 发送时隙使能位

TSCON SFR 用于使能最多 16 个发送时隙的控制位。这些控制位是 TSE<15:0> 位。各时隙的大小由 WS<3:0> 字长选择位确定, 且最大为 16 位。

如果通过一个 TSE 位 (TSE_x = 1) 使能了发送时隙, 则当前发送影子缓冲区中的内容就被装载到 CSDO 移位寄存器中, 且 DCI 缓冲区控制单元加 1, 指向下一单元。

在未使用的发送时隙内, CSDO 引脚驱动为 0, 或在所有禁止的时隙内呈现三态, 这取决于 DCICON1 SFR 中 CSDOM 位的状态。

位中的数据帧长取决于所选数据字长和帧中的数据字元素的数量。如果所选帧长的元素小于 16, 则额外的时隙使能位不起作用。

各发送数据字以左对齐方式写入 16 位发送缓冲区。如果所选字长小于 16 位, 则发送缓冲存储区的 LSB 对发送的数据不起作用。用户应该向各发送缓冲单元中未使用的 LSB 写入 0。

18.3.11 接收时隙使能位

RSCON SFR 包括用于使能最多 16 个接收时隙的控制位。这些控制位是 RSE<15:0> 位。各接收时隙的大小由 WS<3:0> 字长选择位确定, 且大小可在 1 到 16 位之间。

如果通过一个 RSE 位 (RSE_x = 1) 使能接收时隙, 则移位寄存器的内容写入到当前 DCI 接收影子缓冲单元, 且缓冲控制单元加 1, 指向下一缓冲单元。

如果所选择的字长小于 16 位, 则数据在接收存储器缓冲单元中不被打包。每个接收到的时隙数据字都储存在一个独立的 16 位缓冲单元中。数据总是以左对齐格式储存在接收存储器缓冲区中。

18.3.12 帧同步的时隙使能位操作

TSE 和 RSE 控制位与 DCI 帧同步发生器协同工作。在主模式下, 只要帧同步发生器复位, 就会产生 COFS 信号。在从模式下, 只要接收到 COFS 脉冲, 帧同步发生器就会复位。

TSE 和 RSE 控制位允许为发送或接收使能最多 16 个连续的时隙。在最后一个使能的时隙发送/接收完后, DCI 停止缓冲数据, 直到下一次 COFS 脉冲发生为止。

18.3.13 同步数据传输

一旦使能了给定的发送或接收时隙, DCI 缓冲区控制位就会递增一个字单元。大多数情况下, 数据输入和数据输出传输是同步的, 这意味着对于同一个通道, 数据采样的发送和接收是同时进行的。因此, 当产生 DCI 中断时, 发送和接收缓冲区将装填相同数量的数据。

在有些情况下, 一个数据帧中发送和接收的数据量可能不相同。例如, 假设使用了一个双字数据帧。此外, 假设仅在时隙 #0 内接收数据, 在时隙 #0 和 #1 内发送数据。这种情况下, 在一个数据帧期内, 缓冲区控制位计数器将递增两次, 但仅有一个接收寄存器单元会填入数据。

18.3.14 缓冲区长度控制

中断期间缓冲的数据量由 DCICON2 SFR 中的缓冲长度控制位 (BLEN<1:0>) 确定。使用 BLEN 控制位, 可将发送和接收缓冲区的大小设置为 1 至 4 个数据字。BLEN 控制位与 DCI 缓冲区控制单元地址计数器中的当前值进行比较。当 DCI 地址计数器的两个低位与 BLEN<1:0> 值相匹配时, 缓冲区控制单元将复位到 0。另外, 接收影子寄存器的内容将传输到接收缓冲寄存器中, 而发送缓冲寄存器的内容将传输到发送影子寄存器中。

18.3.15 数据帧与缓冲区的对齐

AGU 地址指针的位置和数据帧边界之间没有直接的耦合。这意味着各发送和接收缓冲区有一个隐含的分配, 这是 BLEN 控制位和使能的数据时隙数量 (由 TSE 和 RSE 控制位确定) 的函数。

例如, 假设选择了一个 4 字数据帧, 要在帧的全部四个时隙上发送。可通过置 1 TSCON SFR 中的 TSE0、TSE1、TSE2 和 TSE3 控制位来实现此配置。这样设置模块后, TXBUFO 寄存器将被自动地分配给时隙 #0, 而 TXBUF1 寄存器将自动地分配给时隙 #1, 以此类推。

注: 当一个数据帧内有四个以上的时隙时, 用户代码必须跟踪在发生各个中断时要读写的时隙。有时, 发送 / 接收缓冲区和相应的时隙分配之间的对齐可能会丢失。例如仿真断点或硬件陷阱等情况。这种情况下, 用户应查询 SLOT 状态位以确定应载入到缓冲寄存器的数据, 以重新同步软件和 DCI 模块。

18.3.16 发送状态位

DCISTAT SFR 中有两个发送状态位。

当发送缓冲寄存器中的内容传输到发送影子寄存器时, TMPTY 位将置 1。可在软件中查询 TMPTY 位以确定何时写发送缓冲寄存器。当向四个发送缓冲区之一进行写入时, TMPTY 位将由硬件自动清零。

TUNF 位是只读的, 并表示使用中的发送缓冲寄存器中至少有一个发生了发送下溢。当发送缓冲寄存器的内容被传输到发送影子寄存器时, TMPTY 位将置 1。当 CPU 对发生下溢的缓冲寄存器进行写入时, TUNF 状态位将自动清零。

注: 发送状态位仅表示模块所使用的缓冲单元的状态。例如, 如果缓冲长度设置为小于 4 个字, 则未使用的缓冲单元不会影响发送状态位。

18.3.17 接收状态位

DCISTAT SFR 中有两个接收状态位。

RFUL 状态位是只读的, 且表示接收缓冲区中是否有新数据。在 CPU 读取所有正在使用的接收缓冲区后, RFUL 位将自动清零。

ROV 位是只读的, 且表示至少有一个接收缓冲单元发生了接收溢出。如果在从影子寄存器传输新数据之前, CPU 未读取缓冲单元, 则将发生接收溢出。当 CPU 读取了导致溢出的缓冲寄存器时, ROV 状态位将自动清零。

当特定缓冲单元发生接收溢出后, 缓冲区的原有内容将被覆盖。

注: 接收状态位仅表示模块所使用的缓冲单元的状态。例如, 如果缓冲长度设置为小于 4 个字, 则未使用的缓冲单元不会影响发送状态位。

18.3.18 时隙状态位

DCI STAT SFR 中的 SLOT<3:0> 状态位表示当前有效时隙。这些位与帧同步发生器计数器相对应。发生 DCI 中断时，用户可在软件中查询这些状态位，以确定上次接收了哪些时隙数据，以及哪些时隙数据应被装入 TXBUF 寄存器。

18.3.19 CSDO 模式位

CSDOM 控制位控制 CSDO 引脚在未使用的发送时隙中的状态。如果 TSCON SFR 中相应的 TSE_x 位被清零，则不使用给定的时隙。

如果清零 CSDOM 位（默认），则 CSDO 引脚在未使用的时隙周期中为低电平。当仅有两个器件连接到串行总线时才使用此模式。

如果 CSDOM 位置 1，则 CSDO 引脚在未使用的时隙周期内为三态。在多通道应用中，此模式允许多个器件共享同一根 CSDO 线路。将 CSDO 线路上的每个器件都配置为仅在特定的时隙内发送数据。同一时隙内，不会有两个器件发送数据。

18.3.20 数字环回模式

通过置 1 DCICON1 SFR 中的 DLOOP 控制位使能数字环回模式。当 DLOOP 位置 1 时，模块在内部将 CSDP 信号连接到 CSDI。在数字环回模式下，CSDI I/O 引脚上实际输入的数据被忽略。

18.3.21 下溢模式控制位

发生下溢时，根据 DCICON1 SFR 中的下溢模式 (UNFM) 控制位的状态，会执行两种操作之一。如果 UNFM 位被清零（默认），则在缓冲单元有效时隙内，模块将在 CSDO 引脚上发送 0。在此工作模式下，连接到 DCI 模块的编解码器只接收到数字“静音”信号。如果 UNFM 控制位置 1，则模块发送最后写入缓冲单元的数据。此工作模式允许用户向编解码器发送连续的数据，而无需 CPU 开销。

18.4 DCI 模块中断

DCI 模块中断的频率由 DCICON2 SFR 中的 BLEN<1:0> 控制位确定。每当达到设置的缓冲长度且影子寄存器开始传输时，就会产生一个到 CPU 的中断。影子寄存器传输定义为之前写入的 TXBUF 值传输到发送影子寄存器且接收影子寄存器中新接收的值传输到 RXBUF 寄存器。

18.5 CPU 休眠和空闲模式下的 DCI 模块操作

18.5.1 CPU 休眠模式下的 DCI 模块操作

DCI 模块能在休眠模式下工作，并在外部器件提供 CSCK 信号 (CSCKD = 1) 时唤醒 CPU。当 DCI 缓冲区传输完成且 CPU 处于休眠模式时，DCI 模块产生一个异步中断。

18.5.2 CPU 空闲模式下的 DCI 模块操作

如果 DCISIDL 控制位清零（默认），则模块将在空闲模式下继续正常工作。如果 DCISIDL 位置 1，则当进入空闲模式时，模块将中止工作。

18.6 AC-Link 模式操作

AC-Link 协议是一个 256 位的帧，可分为一个 16 位的数据时隙和 12 个 20 位的数据时隙。DCI 模块有两个针对 AC-Link 协议的工作模式。这些工作模式可通过 DCICON1 SFR 中的 COFSM<1:0> 控制位进行选择。第一个 AC-Link 模式称为“16 位 AC-Link 模式”，可通过设置 COFSM<1:0> = 10 来选择。第二个 AC-Link 模式称为“20 位 AC-Link 模式”，可通过设置 COFSM<1:0> = 11 来选择。

18.6.1 16 位 AC-LINK 模式

在 16 位 AC-Link 模式中，数据字长限制为 16 位。注意此限制仅影响 AC-Link 协议的 20 位数据时隙。对于接收时隙，输入数据被简单地截取为 16 位。对于输出时隙，数字字的低 4 位被模块设置为 0。时隙的这种截取将 A/D 和 DAC 数据限制为 16 位，但允许在 TXBUF 和 RXBUF 寄存器中进行正确的数据对齐。各 RXBUF 和 TXBUF 寄存器均包含一个数据时隙值。

18.6.2 20 位 AC-LINK 模式

20 位 AC-Link 模式允许发送或接收数据时隙中的所有位，但不能保持 TXBUF 和 RXBUF 寄存器中的数据对齐。

20 位 AC-Link 模式与 DCI 模块的多通道模式的原理相似，只是帧同步信号的占空比有所不同。AC-Link 帧同步信号应在 16 个 CSCK 周期中为高电平，而在接下来的 240 个周期中为低电平。

20 位模式将每个 256 位 AC-link 帧作为 16 个 16 位时隙进行处理。在 20 位 AC-Link 模式下，模块就好像是在 COFSG<3:0> = 1111 且 WS<3:0> = 1111 的条件下工作。20 位数据时隙的数据对齐被忽略。例如，通过置 1 TSCON 和 RSCON SFR 中的所有位，可将整个 AC-Link 数据帧打包发送或接收。因为整个可用缓冲长度为 64 位，传输 AC-Link 帧需要 4 个连续的中断。应用软件必须跟踪当前 AC-Link 帧段。

18.7 I²S 模式操作

通过向 DCICON1 SFR 中的 COFSM<1:0> 控制位写入值 01，可将 DCI 模块配置为 I²S 模式。工作在 I²S 模式下时，DCI 模块将产生占空比为 50% 的帧同步信号。帧同步信号的各个边沿标记新数据字传输的边界。

用户也必须通过使用 DCICON2 SFR 中的 COFSG 和 WS 控制位选择帧长和数据字长。

18.7.1 I²S 帧和数据字长选择

将 WS 和 COFSG 控制位置 1 以产生 I²S 数据帧长一半的周期。即，帧长是左对齐或右对齐数据字传输所需的总的 CSCK 周期数。

必须置 1 BLEN 位，从而得到所需的缓冲区长度。设置 BLEN<1:0> = 01 以在每个 I²S 帧产生一个 CPU 中断。

18.7.2 I²S 数据对齐

如各 I²S 规范那样，默认情况下，数据字在 WS 信号跳变后的一个 CSCK 周期后开始传输。可使用 DCICON1 SFR 中的 DJST 控制位选择“MSb 左对齐”选项。

如果 DJST = 1，则 I²S 数据传输为 MSb 左对齐。在 COFS 信号的上升或下降沿的同一 CSCK 周期内，数据字的 MSb 出现在 CSDO 引脚上。数据字发送后，CSDO 引脚为三态。

表 18-2: dsPIC30F3014/4013 DCI 寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
DCICON1	0240	DCIEN	—	DCISIDL	—	DLOOP	CSCKD	CSCKE	COFSD	UNFM	CSDOM	DJST	—	—	—	COFSM1	COFSM0	0000 0000 0000 0000
DCICON2	0242	—	—	—	—	BLEN1	BLEN0	—	COFSG<3:0>			—	WS<3:0>					0000 0000 0000 0000
DCICON3	0244	—	—	—	—	BCG<11:0>											0000 0000 0000 0000	
DCISTAT	0246	—	—	—	—	SLOT3	SLOT2	SLOT1	SLOT0	—	—	—	—	ROV	RFUL	TUNF	TMPTY	0000 0000 0000 0000
TSCON	0248	TSE15	TSE14	TSE13	TSE12	TSE11	TSE10	TSE9	TSE8	TSE7	TSE6	TSE5	TSE4	TSE3	TSE2	TSE1	TSE0	0000 0000 0000 0000
RSCON	024C	RSE15	RSE14	RSE13	RSE12	RSE11	RSE10	RSE9	RSE8	RSE7	RSE6	RSE5	RSE4	RSE3	RSE2	RSE1	RSE0	0000 0000 0000 0000
RXBUF0	0250	接收缓冲区 0 数据寄存器																0000 0000 0000 0000
RXBUF1	0252	接收缓冲区 1 数据寄存器																0000 0000 0000 0000
RXBUF2	0254	接收缓冲区 2 数据寄存器																0000 0000 0000 0000
RXBUF3	0256	接收缓冲区 3 数据寄存器																0000 0000 0000 0000
TXBUF0	0258	发送缓冲区 0 数据寄存器																0000 0000 0000 0000
TXBUF1	025A	发送缓冲区 1 数据寄存器																0000 0000 0000 0000
TXBUF2	025C	发送缓冲区 2 数据寄存器																0000 0000 0000 0000
TXBUF3	025E	发送缓冲区 3 数据寄存器																0000 0000 0000 0000

图注: — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

19.0 12 位模数转换器 (ADC) 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能, 但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

12 位模数转换器 (ADC) 可将一个模拟输入信号转换为一个 12 位数字信号。该模块基于逐次逼近寄存器 (SAR) 架构并提供最高 200 ksp/s 的采样速率。该 A/D 模块最多具有 16 路模拟输入, 这些模拟输入与采样保持放大器通过多路开关连接。采样和保持输出是转换器的输入, 转换器产生转换结果。可通过软件将器件的供电电压 (AVDD/AVSS) 或 VREF+/VREF- 引脚上的电压选择作为模拟参考电压。A/D 转换器具有一个独特性能: 选择 RC 振荡器后, A/D 转换器可在器件休眠模式下正常工作。

A/D 模块具有 6 个 16 位寄存器:

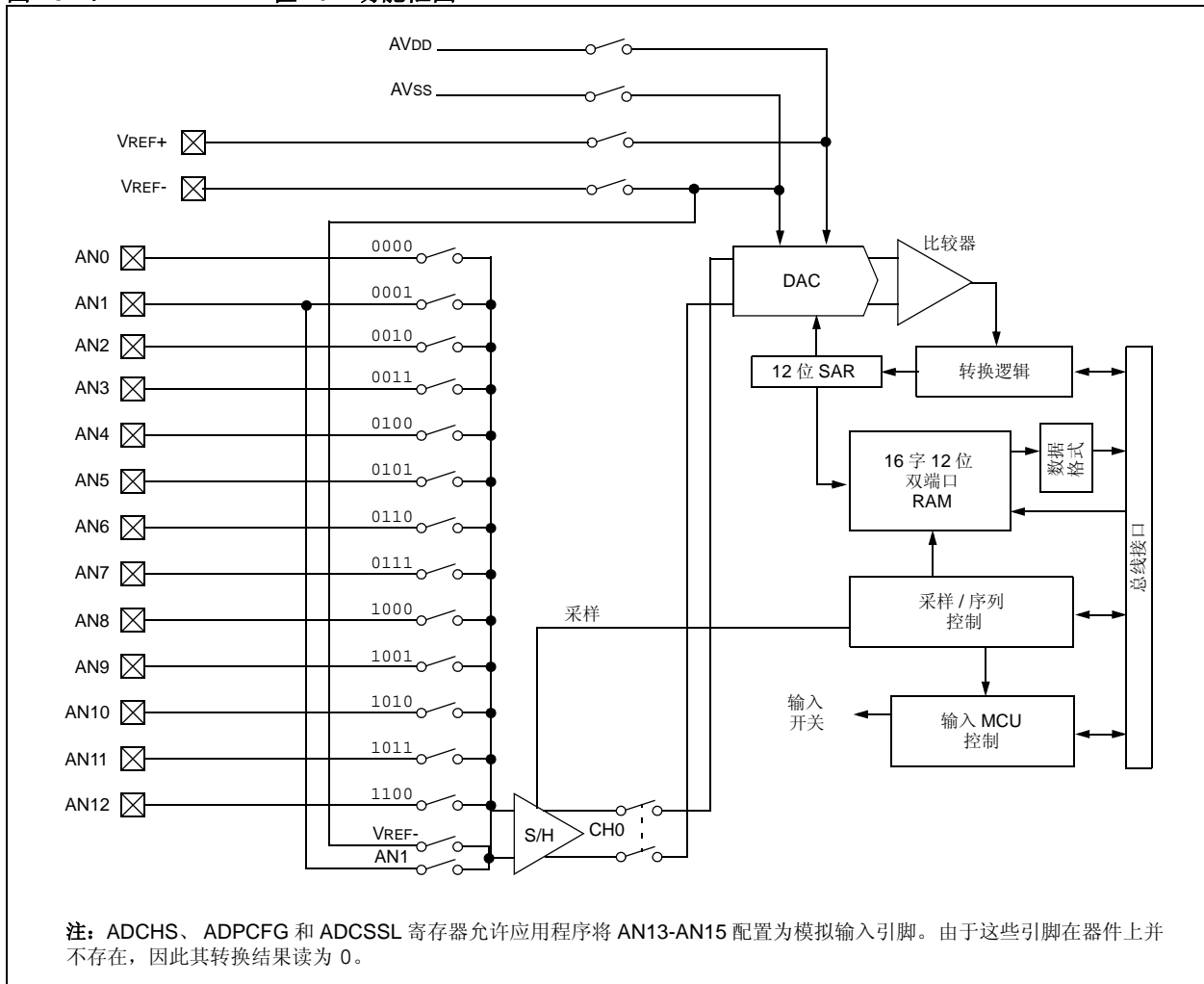
- A/D 控制寄存器 1 (ADCON1)
- A/D 控制寄存器 2 (ADCON2)
- A/D 控制寄存器 3 (ADCON3)
- A/D 输入选择寄存器 (ADCHS)
- A/D 端口配置寄存器 (ADPCFG)
- A/D 输入扫描选择寄存器 (ADCSSL)

ADCON1、ADCON2 和 ADCON3 寄存器控制 A/D 模块的操作。由 ADCHS 寄存器选择待转换的输入通道。ADPCFG 寄存器将端口引脚配置为模拟输入或数字 I/O。ADCSSL 寄存器选择待扫描的输入。

注: 当 ADON = 1 时, 不能对 SSRC<2:0>、ASAM、SMPI<3:0>、BUFM 和 ALTS 位以及 ADCON3 和 ADCSSL 寄存器执行写操作, 否则将可能导致不确定的结果。

12 位 A/D 模块的框图如图 19-1 所示。

图 19-1: 12 位 A/D 功能框图



19.1 A/D 结果缓冲区

该模块包含一个名为 ADCBUF0...ADCBUFF 的 16 字双端口只读缓冲区，用于缓冲 A/D 结果。RAM 为 12 位宽，但获取的数据可用 4 种不同的 16 位数据格式之一表示。16 个 A/D 转换结果缓冲寄存器（从 ADCBUF0 到 ADCBUFF）的内容均不能由用户软件写入。

19.2 转换操作

配置完 A/D 模块之后，通过将 SAMP 位置 1 启动采样采集。有多种原因可终止采集并启动转换，例如可编程位、定时器超时或外部事件。A/D 转换完成后，转换结果将被装入 ADCBUF0...ADCBUFF，由 SMPI 位指定的采样数达到后，DONE 位及 A/D 中断标志位 ADIF 将被置 1。可按第 19.3 节“选择转换过程”中介绍的那样为 ADC 模块配置不同的中断频率。

按照以下步骤执行 A/D 转换：

1. 配置 A/D 模块：
 - 配置模拟引脚、参考电压和数字 I/O
 - 选择 A/D 输入通道
 - 选择 A/D 转换时钟
 - 选择 A/D 转换触发器
 - 启动 A/D 模块
2. 配置 A/D 中断（如果需要）：
 - 清零 ADIF 位
 - 选择 A/D 中断优先级
 - 将 ADIE 位置 1（以允许进行 IRS 处理）
3. 开始采样
4. 等待所需采集时间
5. 触发采集结束，启动转换：
6. 等待 A/D 转换完成，有两种方法：
 - 等待 A/D 中断，或
 - 等待 DONE 位置 1。
7. 读取 A/D 结果缓冲区，如果需要，清零 ADIF

19.3 选择转换过程

可用若干组控制位选择转换过程，在此过程中，A/D 将输入连接到采样 / 保持通道，对通道进行转换，将转换结果写入缓冲存储器并产生中断。

该过程由采样时钟控制。

SMPI 位选择在中断发生前可执行的采集/转换过程的次数。其范围是：每次中断采样 1 次到 16 次。

BUFm 位将一个 16 字结果缓冲区分成两个 8 字组。在每次中断事件时，交替填写这两个 8 字缓冲区。

BUFm 的使用取决于有多少时间可用于中断之后移出缓冲区中的数据。

如果处理器可以在其采集和转换一个通道的时间内快速地卸空缓冲区，则 BUFm 位可以为 0 且一次中断最多可完成 16 次转换（对应 16 路输入通道）。处理器使用一个采集和转换时间即可移出 16 个转换结果。

如果处理器不能在其采样和转换时间内卸空缓冲区，则应将 BUFm 位置 1。例如，如果 $SMPI<3:0> = 0111$ ，则先将 8 个转换装入其中一个 8 字缓冲区，发生中断后再将其他 8 个转换装入另一个 8 字缓冲区。两次中断之间的所有时间均可供处理器用来移出 8 个转换结果。

可以使用 ALTS 位在采样过程中交替所选择的输入。输入多路开关具有两组采样输入：MUX A 和 MUX B。如果 ALTS 位是 0，只选择 MUX A 的输入用于采样。如果在第一次采样 / 转换过程中，ALTS 位为 1， $SMPI<3:0> = 0000$ ，则 MUX A 被选作输入，而在下一次采样 / 转换过程中，MUX B 被选作输入。

CSCNA 位 ($ADCON2<10>$) 允许在 MUX A 组的选定数量的模拟输入上按顺序扫描 S/H 输入。通过 ADCSSL 寄存器选择输入。如果 ADCSSL 寄存器中的一个特定位是 1，则选择相应的输入通道。总是按从编号低到高的顺序扫描输入，每次中断后开始扫描。如果选择的输入数量大于每次中断的采样数量，则编号较高的输入不会被使用。

注： ADCHS、ADPCFG 和 ADCSSL 寄存器允许应用程序将 AN13-AN15 配置为模拟输入引脚。由于这些引脚在器件上并不存在，故其转换结果读为 0。

19.4 编程转换触发器的启动

转换触发信号可终止采集并启动所请求的转换。

SSRC<2:0> 位选择转换触发源。SSRC 位最多可提供 4 个备用转换触发源。

当 SSRC<2:0> = 000 时，转换触发器由软件控制。清零 SAMP 位导致触发转换。

当 SSRC<2:0> = 111 (自动转换模式) 时，转换触发器由 A/D 时钟控制。SAMC 位选择启动采集和启动转换之间的 A/D 时钟数量。这样便在多个通道上提供了最快的转换速率。SAMC 位必须总是不小于一个时钟周期。

其他触发源可来自定时器模块或外部中断。

19.5 中止转换

转换期间清零 ADON 位将中止当前转换并停止采样序列，直到下一次采样被触发为止。ADCBUF 不会由部分完成的 A/D 转换采样更新。也就是说，ADCBUF 将继续包含上次完成的转换值 (或最后写入 ADCBUF 寄存器的值)。

如果清零 ADON 位和自动启动碰巧同时发生，则由于清零具有更高优先级而不启动新的转换。

19.6 选择 ADC 转换时钟

ADC 转换需要 14 个 TAD。ADC 的转换时钟源可由软件通过一个 6 位计数器选择。TAD 有 64 个可能的选择。

公式 19-1: ADC 转换时钟

$$T_{AD} = T_{CY} * (0.5 * (ADCS<5:0> + 1))$$

通过将 ADRC 位置 1 可选择内部 RC 振荡器。

要获得正确的 ADC 转换结果，必须选择 ADC 转换时钟 (TAD) 以确保 TAD 最小时间为 334 ns (VDD = 5V)。其他工作条件下的最小 TAD 请参见第 23.0 节“电气特性”。

例 19-1 显示了 ADCS<5:0> 位的计算示例，假设器件的工作速度为 30 MIPS。

例 19-1: ADC 转换时钟和采样速率计算

$$\begin{aligned} \text{最小 } T_{AD} &= 334 \text{ ns} \\ T_{CY} &= 33.33 \text{ ns (30 MIPS)} \end{aligned}$$

$$\begin{aligned} ADCS<5:0> &= 2 \frac{T_{AD}}{T_{CY}} - 1 \\ &= 2 \cdot \frac{334 \text{ ns}}{33.33 \text{ ns}} - 1 \\ &= 19 \end{aligned}$$

所以，

设置 ADCS<5:0> = 19

$$\begin{aligned} \text{实际的 } T_{AD} &= \frac{T_{CY}}{2} (ADCS<5:0> + 1) \\ &= \frac{33.33 \text{ ns}}{2} (19 + 1) \\ &= 334 \text{ ns} \end{aligned}$$

如果 SSRC<2:0> = 111 且 SAMC<4:0> = 00001

因为，

$$\begin{aligned} \text{采样时间} &= \text{采集时间} + \text{转换时间} \\ &= 1 T_{AD} + 14 T_{AD} \\ &= 15 \times 334 \text{ ns} \end{aligned}$$

所以，

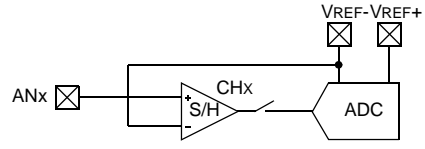
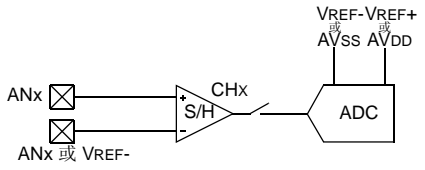
$$\begin{aligned} \text{采样速率} &= \frac{1}{(15 \times 334 \text{ ns})} \\ &= \sim 200 \text{ kHz} \end{aligned}$$

dsPIC30F3014/4013

19.7 ADC 速率

dsPIC30F 12 位 ADC 规范所允许的最高采样速率为 200 ksp/s。下表总结了 dsPIC30F 12 位 ADC 的转换速率和所要求的工作条件。

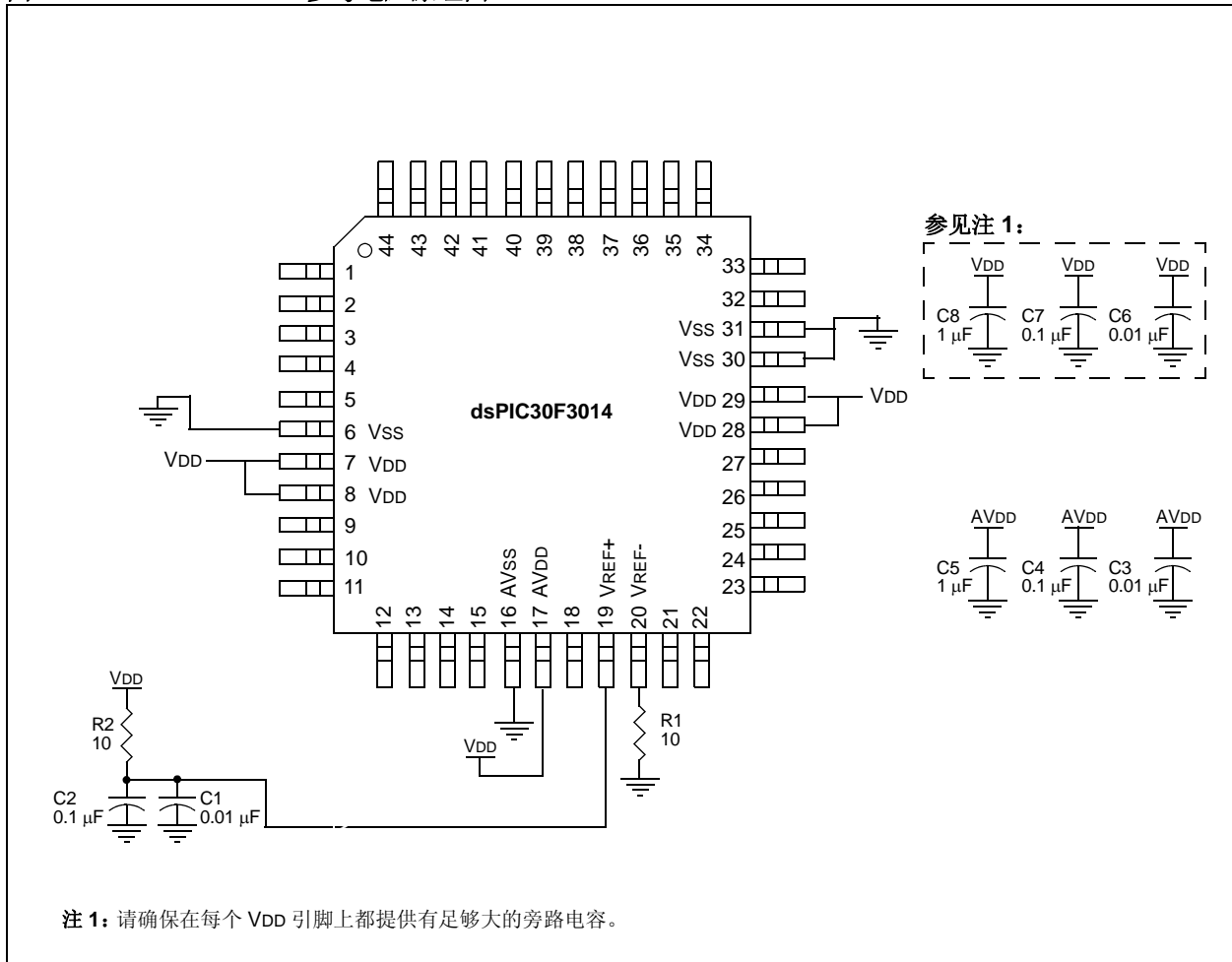
表 19-1: 12 位 ADC 扩展转换速率

dsPIC30F 12 位 ADC 转换速率						
速度	TAD 最小值	采样时间最小值	R_s 最大值	VDD	温度	通道配置
最高 200 ksp/s ⁽¹⁾	334 ns	1 TAD	2.5 k Ω	4.5V 至 5.5V	-40°C 至 +85°C	
最高 100 ksp/s	668 ns	1 TAD	2.5 k Ω	3.0V 至 5.5V	-40°C 至 +125°C	

注 1: 要使模块正常工作，必须使用外部 VREF- 和 VREF+ 引脚。建议采用的电路请参见图 19-2。

图 19-2 所示为转换速率大于 200 ksp/s 时的建议电路。该图以 dsPIC30F3014 为例。

图 19-2: ADC 参考电压原理图



下面的配置过程给出了大于 100 ksp/s 的转换速率所需的设置值。

19.7.1 200 ksp/s 配置指导

要达到 200 ksp/s 的转换速率需进行以下配置：

- 满足表 19-1 中的条件。
- 按图 19-2 所示的建议电路连接外部 VREF+ 和 VREF- 引脚。
- 设置 ADCON1 寄存器中的 SSRC<2:0> = 111，以启用自动转换选项。
- 通过将 ADCON1 寄存器中的 ASAM 控制位置 1 启用自动采样。
- 向 ADCON2 寄存器中的 SMPI<3:0> 控制位写入所需的两次中断间的转换次数。

- 通过写入 ADCON3 寄存器中的 ADCS<5:0> 控制位，将 ADC 的时钟周期配置为：

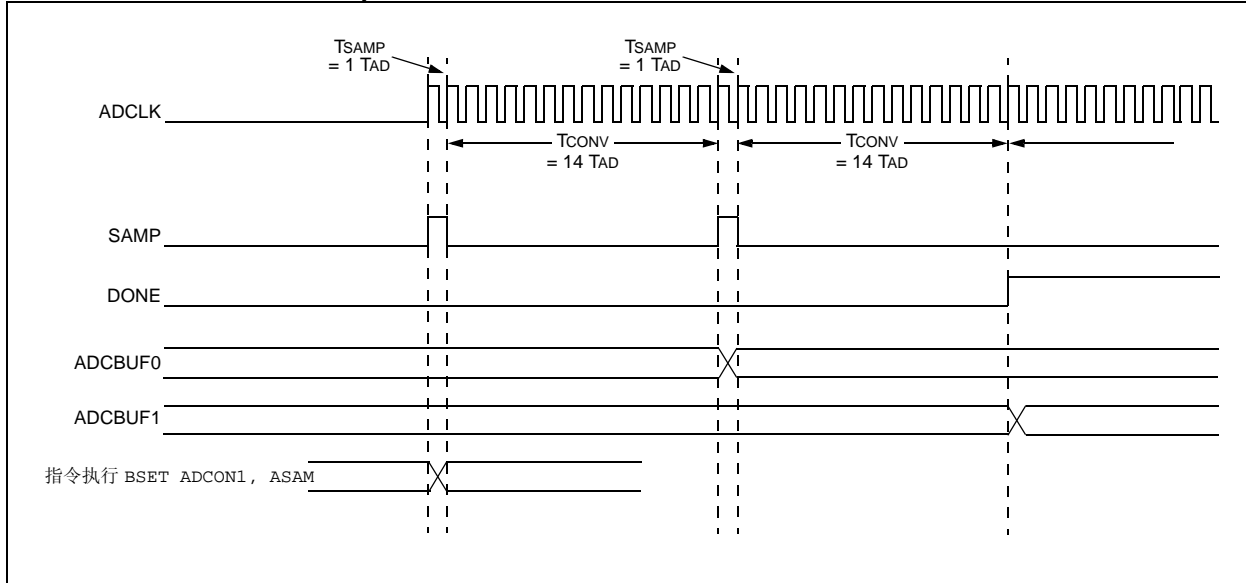
$$\frac{1}{(14 + 1) \times 200,000} = 334 \text{ ns}$$

- 通过向 SAMC<4:0> 写入 00001，将采样时间配置为 1 TAD。

下图显示了以 200 ksp/s 运行的 ADC 时序图。上述指导再加正确的 TAD 选择便可实现 200 ksp/s 的转换速率。代码示例请参见例 19-1。

dsPIC30F3014/4013

图 19-3: 以 200 ksps 转换 1 个通道，自动采样启动和 1 TAD 采样时间



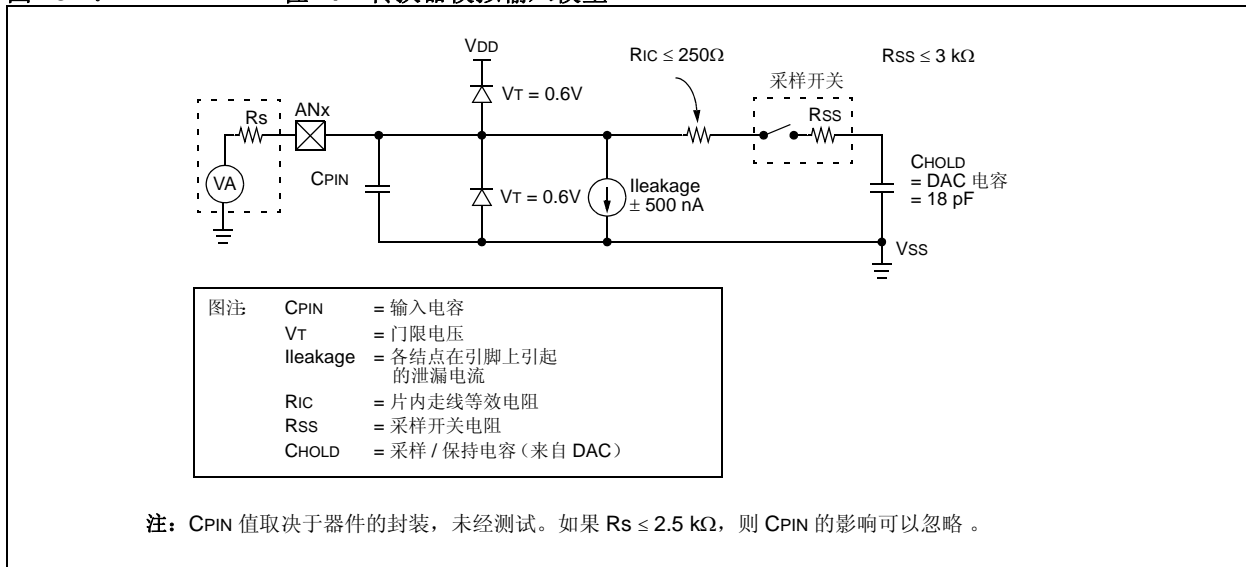
19.8 A/D 采集要求

12 位 A/D 转换器的模拟输入模型如图 19-4 所示。A/D 的总采样时间是内部放大器稳定时间和保持电容充电时间的函数。

为了使 A/D 转换达到规定精度，必须使充电保持电容 (CHOLD) 充满至模拟输入通道的电平。信号源阻抗 (R_s)、片内走线的等效电阻 (R_{ic}) 和内部采样开关

阻抗 (R_{ss}) 直接影响为电容 CHOLD 充电所需要的时间。因此，模拟信号源的总阻抗必须足够的小，以保持电容在所选的采样时间内充满电。要最大程度地降低泄漏电流对 A/D 转换器精度的影响，信号源阻抗 R_s 的最大建议值为 2.5 k Ω 。选择 (更改) 模拟输入通道以后，必须在启动转换前完成采样。每次采样前，内部保持电容都处于放电状态。

图 19-4: 12 位 A/D 转换器模拟输入模型



19.9 模块掉电模式

模块有两种内部功耗模式。

当 ADON 位为 1 时，该模块在有效 (Active) 模式下；其处于全功耗模式且功能完善。

当 ADON 位为 0 时，该模块在关闭 (Off) 模式下。为了最大程度地省电，电路的数字和模拟部分被禁止。

为了从关闭模式返回到有效模式，用户必须等待 ADC 电路稳定下来。第 23.0 节“电气特性”中规定了稳定所需的时间。

19.10 CPU 休眠模式和空闲模式下 A/D 的操作

19.10.1 CPU 休眠模式和空闲模式下 A/D 的操作

当器件进入休眠模式时，模块的所有时钟源被关闭且保持为逻辑 0。

如果在转换中途发生休眠，则转换被中止。从休眠模式中退出以后，转换器将不能继续完成部分完成的转换。

器件进入休眠模式或从休眠模式退出不会影响寄存器的内容。

如果将 A/D 时钟源设置为 RC (ADRC = 1)，则 A/D 模块可在休眠模式下工作。选择 RC 时钟源时，启动转换前，A/D 模块会等待一个指令周期。这使 SLEEP 指令能

够被执行，从而消除所有来自转换的数字开关噪声（转换完成后，DONE 位置 1）。

如果允许 A/D 中断，器件会从休眠模式中唤醒。如果禁止 A/D 中断，虽然仍会将 ADONE 位置 1，但 A/D 模块会被关闭。

19.10.2 CPU 空闲模式下的 A/D 操作

由 ADSIDL 位确定在空闲模式下模块是停止还是继续。如果 ADSIDL = 0，进入空闲模式时 A/D 模块会继续工作。如果 ADSIDL = 1，在空闲模式下 A/D 模块会停止操作。

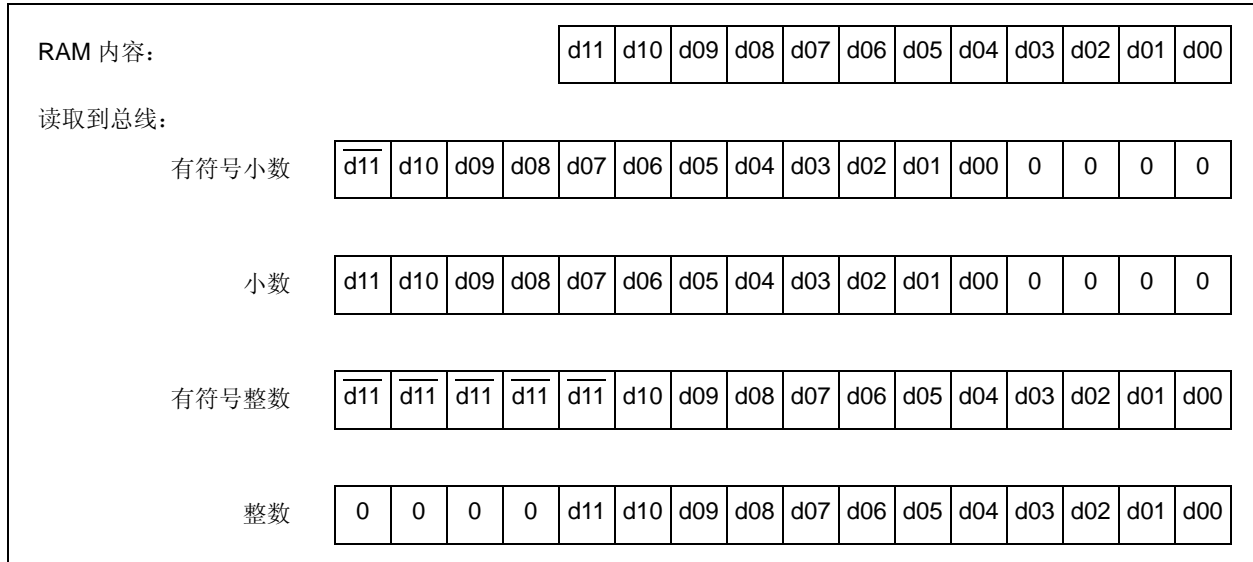
19.11 复位的影响

器件复位强制所有寄存器进入复位状态。器件复位会强制关闭 A/D 模块并中止所有的转换和采样。ADCBUF 寄存器中的值不会被修改。上电复位后，A/D 结果寄存器将包含未知数据。

19.12 输出格式

A/D 结果为 12 位宽。数据缓冲区 RAM 也是 12 位宽。可以用 4 种不同格式中的一种读取 12 位数据。FORM<1:0> 位选择格式。每种输出格式都将被转换到数据总线上的一个 16 位结果。写数据时总是按右对齐（整数）格式进行。

图 19-5: A/D 输出数据的格式



19.13 配置模拟端口引脚

使用 ADPCFG 和 TRIS 寄存器控制 A/D 端口引脚的操作。若要将端口引脚配置为模拟输入引脚，则必须将其相应的 TRIS 位置 1（输入）。如果将 TRIS 位清零（输出），则将转换该引脚的数字输出电平（VOH 或 VOL）。

A/D 的操作与 CH0SA<3:0>/CH0SB<3:0> 位和 TRIS 位的状态无关。

读取端口寄存器时，所有配置为模拟输入通道的引脚读为 0。

配置为数字输入的引脚将不转换模拟输入。若定义为数字输入的引脚（包括 ANx 引脚）上出现模拟电平，则可能导致输入缓冲器的电流消耗超出器件规范值。

19.14 连接注意事项

模拟输入与 VDD 和 VSS 之间连接有二极管作为 ESD 保护。该二极管将模拟输入限制在 VDD 和 VSS 之间。如果输入电压超出此范围 0.3 V（任一方向）以上，则其中一个二极管可能变为正偏，如果输入电流超出规范值，可能会损坏器件。

有时需要外接一个 RC 滤波器来防止输入信号混叠。应选择适当的 R 元件以确保满足采样时间要求。任何与模拟输入引脚连接（通过高阻抗）的外部元件（电容和齐纳二极管等）均应保证其在该引脚的泄漏电流极小。

表 19-2: A/D 转换器寄存器映射 (1)

SFR 名称	地址	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	复位状态
ADCBUF0	0280	—	—	—	—	ADC 数据缓冲区 0												0000 uuuu uuuu uuuu
ADCBUF1	0282	—	—	—	—	ADC 数据缓冲区 1												0000 uuuu uuuu uuuu
ADCBUF2	0284	—	—	—	—	ADC 数据缓冲区 2												0000 uuuu uuuu uuuu
ADCBUF3	0286	—	—	—	—	ADC 数据缓冲区 3												0000 uuuu uuuu uuuu
ADCBUF4	0288	—	—	—	—	ADC 数据缓冲区 4												0000 uuuu uuuu uuuu
ADCBUF5	028A	—	—	—	—	ADC 数据缓冲区 5												0000 uuuu uuuu uuuu
ADCBUF6	028C	—	—	—	—	ADC 数据缓冲区 6												0000 uuuu uuuu uuuu
ADCBUF7	028E	—	—	—	—	ADC 数据缓冲区 7												0000 uuuu uuuu uuuu
ADCBUF8	0290	—	—	—	—	ADC 数据缓冲区 8												0000 uuuu uuuu uuuu
ADCBUF9	0292	—	—	—	—	ADC 数据缓冲区 9												0000 uuuu uuuu uuuu
ADCBUFA	0294	—	—	—	—	ADC 数据缓冲区 10												0000 uuuu uuuu uuuu
ADCBUFB	0296	—	—	—	—	ADC 数据缓冲区 11												0000 uuuu uuuu uuuu
ADCBUFC	0298	—	—	—	—	ADC 数据缓冲区 12												0000 uuuu uuuu uuuu
ADCBUFD	029A	—	—	—	—	ADC 数据缓冲区 13												0000 uuuu uuuu uuuu
ADCBUFE	029C	—	—	—	—	ADC 数据缓冲区 14												0000 uuuu uuuu uuuu
ADCBUFF	029E	—	—	—	—	ADC 数据缓冲区 15												0000 uuuu uuuu uuuu
ADCON1	02A0	ADON	—	ADSIDL	—	—	—	FORM<1:0>		SSRC<2:0>			—	—	ASAM	SAMP	DONE	0000 0000 0000 0000
ADCON2	02A2	VCFG<2:0>			—	—	CSCNA	—	—	BUFS	—	SMPI<3:0>				BUFM	ALTS	0000 0000 0000 0000
ADCON3	02A4	—	—	—	SAMC<4:0>				ADRC	—	ADCS<5:0>						0000 0000 0000 0000	
ADCHS	02A6	—	—	—	CH0NB	CH0SB<3:0>			—	—	—	CH0NA	CH0SA<3:0>				0000 0000 0000 0000	
ADPCFG	02A8	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000 0000 0000 0000
ADCSSL	02AA	CSSL15	CSSL14	CSSL13	CSSL12	CSSL11	CSSL10	CSSL9	CSSL8	CSSL7	CSSL6	CSSL5	CSSL4	CSSL3	CSSL2	CSSL1	CSSL0	0000 0000 0000 0000

图注: u = 未初始化的位; — = 未实现位, 读为 0

注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F3014/4013

注:

20.0 系统集成

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

本系列器件的下列功能，旨在最大限度地提高系统可靠性、通过减少外部元件将成本降至最低、提供节能工作模式以及提供代码保护：

- 振荡器选择
- 复位
 - 上电复位 (POR)
 - 上电延时定时器 (PWRT)
 - 振荡器起振定时器 (OST)
 - 可编程欠压复位 (BOR)
- 看门狗定时器 (WDT)
- 低电压检测
- 节能模式 (休眠和空闲)
- 代码保护
- 器件 ID 单元
- 在线串行编程 (ICSP)

dsPIC30F 器件具有一个看门狗定时器，它可以通过配置位永久使能，或用软件进行控制。它凭借自身的 RC 振荡器来运行，可提高可靠性。有两个定时器在上电时提供必须的延时。一个是振荡器起振定时器 (OST)，它用于保持芯片复位，直至晶振稳定。另一个是上电延时定时器 (PWRT)，它仅在上电时提供延时，用来在电源稳定过程中使器件保持在复位状态。有了这两个片上定时器，大多数应用不再需要外部复位电路。

休眠模式旨在提供电流极低的掉电模式。用户可以通过外部复位、看门狗定时器唤醒或中断来从休眠模式唤醒器件。此外还提供了几个振荡器选项以允许器件得到更广泛应用。在空闲模式下，时钟源仍然处于工作状态，但 CPU 关闭。RC 振荡器节省系统成本，而 LP 晶振可降低功耗。

20.1 振荡器系统概述

dsPIC30F 振荡器系统包含以下模块和特性：

- 可选择多种外部和内部振荡器作为时钟源
- 片上 PLL 可提高内部工作频率
- 在多种时钟源之间进行切换的时钟切换机制
- 可编程时钟后分频器可节省系统功耗
- 故障保护时钟监视器 (FSCM) 可检测时钟故障并采取故障保护措施
- 时钟控制寄存器 (OSCCON)
- 用于主振荡器选择的配置位

配置位用于确定上电复位 (POR) 和欠压复位 (BOR) 时的时钟源。此后，可以在容许的时钟源之间切换。OSCCON 寄存器控制时钟切换并包含反映系统时钟的相关状态位。

表 20-1 汇总了 dsPIC30F 振荡器的工作模式。图 20-1 显示了振荡器系统的简化框图。

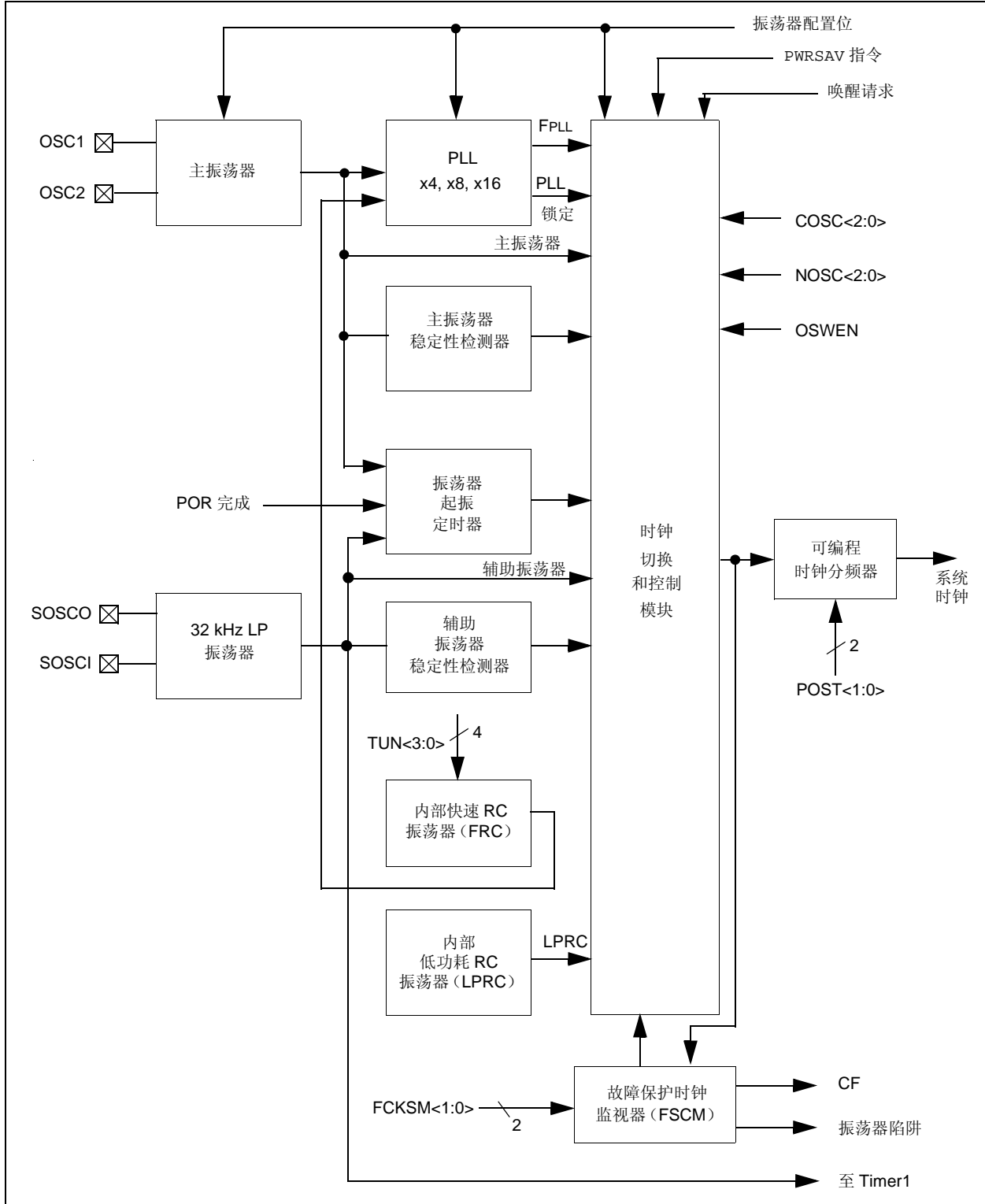
dsPIC30F3014/4013

表 20-1: 振荡器工作模式

振荡器模式	说明
XTL	OSC1:OSC2 上 400 kHz-4 MHz 晶振
XT	OSC1:OSC2 上 4 MHz-10 MHz 晶振
带 4 倍频 PLL 的 XT	OSC1:OSC2 上 4 MHz-10 MHz 晶振, 4x PLL 使能
带 8 倍频 PLL 的 XT	OSC1:OSC2 上 4 MHz-10 MHz 晶振, 8x PLL 使能
带 16 倍频 PLL 的 XT	OSC1:OSC2 上 4 MHz-7.5 MHz 晶振, 16x PLL 使能 ⁽¹⁾
LP	SOSCO:SOSCI 上 32 kHz 晶振 ⁽²⁾
HS	10 MHz-25 MHz 晶振
带 4 倍频 PLL 的 HS/2	10 MHz-20 MHz 晶振, 2 分频, 4x PLL 使能
带 8 倍频 PLL 的 HS/2	10 MHz-20 MHz 晶振, 2 分频, 8x PLL 使能
带 16 倍频 PLL 的 HS/2	10 MHz-15 MHz 晶振, 2 分频, 16x PLL 使能
带 4 倍频 PLL 的 HS/3	12 MHz-25 MHz 晶振, 3 分频, 4x PLL 使能
带 8 倍频 PLL 的 HS/3	12 MHz-25 MHz 晶振, 3 分频, 8x PLL 使能
带 16 倍频 PLL 的 HS/3	12 MHz-22.5 MHz 晶振, 3 分频, 16x PLL 使能
EC	外部时钟输入 (0-40 MHz)
ECIO	外部时钟输入 (0-40 MHz), OSC2 引脚为 I/O
带 4 倍频 PLL 的 EC	外部时钟输入 (4-10 MHz), OSC2 引脚为 I/O, 4x PLL 使能 ⁽¹⁾
带 8 倍频 PLL 的 EC	外部时钟输入 (4-10 MHz), OSC2 引脚为 I/O, 8x PLL 使能 ⁽¹⁾
带 16 倍频 PLL 的 EC	外部时钟输入 (4-7.5 MHz), OSC2 引脚为 I/O, 16x PLL 使能 ⁽¹⁾
ERC	外部 RC 振荡器, OSC2 引脚为 Fosc/4 输出 ⁽³⁾
ERCIO	外部 RC 振荡器, OSC2 引脚为 I/O ⁽³⁾
FRC	7.37 MHz 内部 RC 振荡器
带 4 倍频 PLL 的 FRC	7.37 MHz 内部 RC 振荡器, 4x PLL 使能
带 8 倍频 PLL 的 FRC	7.37 MHz 内部 RC 振荡器, 8x PLL 使能
带 16 倍频 PLL 的 FRC	7.37 MHz 内部 RC 振荡器, 16x PLL 使能
LPRC	512 kHz 内部 RC 振荡器

- 注 1: 必须满足 dsPIC30F 最大工作频率为 120 MHz 的要求。
2: LP 振荡器可方便地同时作为系统时钟和 Timer1 的实时时钟。
3: 需要外部 R 和 C 元件。工作频率最高为 4 MHz。

图 20-1: 振荡器系统框图



dsPIC30F3014/4013

20.2 振荡器配置

20.2.1 初始时钟源选择

当退出上电复位或欠压复位时，器件将根据以下原则来选择时钟源：

- 通过 FOS<2:0> 配置位选择四种振荡器组中的一种。
- 通过 FPR<4:0> 配置位在主振荡器组中的 13 个振荡器中选择一个。

表 20-2 显示了时钟选择。

20.2.2 振荡器起振定时器 (OST)

为了确保晶振（或陶瓷谐振器）已起振且达到稳定状态，将需要使用一个振荡器起振定时器。该定时器就是一个 10 位计数器，在释放振荡器时钟以应用于系统的其他部分之前，该计数器将计数 1024 个 T_{osc} 周期。超时周期称为 T_{OST}。每次振荡器重启时（即在 POR、BOR 以及从休眠模式唤醒时）都需要经过 T_{OST} 时间。振荡器起振定时器应用于 LP 振荡器和主振荡器的 XT、XTL 及 HS 振荡器模式（当从休眠模式唤醒、POR 以及 BOR 时）。

表 20-2: 时钟选择配置位的值

振荡器模式	振荡器源	FOS<2:0>			FPR<4:0>					OSC2 功能
带 PLL 4 倍频的 ECIO	PLL	1	1	1	0	1	1	0	1	I/O
带 PLL 8 倍频的 ECIO	PLL	1	1	1	0	1	1	1	0	I/O
带 PLL 16 倍频的 ECIO	PLL	1	1	1	0	1	1	1	1	I/O
带 PLL 4 倍频的 FRC	PLL	1	1	1	0	0	0	0	1	I/O
带 PLL 8 倍频的 FRC	PLL	1	1	1	0	1	0	1	0	I/O
带 PLL 16 倍频的 FRC	PLL	1	1	1	0	0	0	1	1	I/O
带 PLL 4 倍频的 XT	PLL	1	1	1	0	0	1	0	1	OSC2
带 PLL 8 倍频的 XT	PLL	1	1	1	0	0	1	1	0	OSC2
带 PLL 16 倍频的 XT	PLL	1	1	1	0	0	1	1	1	OSC2
带 PLL 4 倍频的 HS2	PLL	1	1	1	1	0	0	0	1	OSC2
带 PLL 8 倍频的 HS2	PLL	1	1	1	1	0	0	1	0	OSC2
带 PLL 16 倍频的 HS2	PLL	1	1	1	1	0	0	1	1	OSC2
带 PLL 4 倍频的 HS3	PLL	1	1	1	1	0	1	0	1	OSC2
带 PLL 8 倍频的 HS3	PLL	1	1	1	1	0	1	1	0	OSC2
带 PLL 16 倍频的 HS3	PLL	1	1	1	1	0	1	1	1	OSC2
ECIO	外部	0	1	1	0	1	1	0	0	I/O
XT	外部	0	1	1	0	0	1	0	0	OSC2
HS	外部	0	1	1	0	0	0	1	0	OSC2
EXT	外部	0	1	1	0	1	0	1	1	CLKO
ERC	外部	0	1	1	0	1	0	0	1	CLKO
ERCIO	外部	0	1	1	0	1	0	0	0	I/O
XTL	外部	0	1	1	0	0	0	0	0	OSC2
LP	辅助	0	0	0	X	X	X	X	X	(注 1 和 2)
FRC	内部 FRC	0	0	1	X	X	X	X	X	(注 1 和 2)
LPRC	内部 LPRC	0	1	0	X	X	X	X	X	(注 1 和 2)

注 1: OSC2 引脚是否可用作通用 I/O 引脚功能取决于主振荡器模式选择位 FPR<4:0>。

注 2: 注意 OSC1 引脚不能用作 I/O 引脚，即使始终选择辅助振荡器或内部时钟源也是如此。

20.2.3 LP 振荡器控制

使用以下两项可以启用 LP 振荡器：

- 当前振荡器组控制位 COSC<2:0>。
- LPOSCEN 位（OSCCON 寄存器）。

如果 LPOSCEN = 1，则启用 LP 振荡器（即使处于休眠模式）。在下列条件下，LP 振荡器用作器件时钟：

- COSC<2:0> = 00（选择 LP 作为主振荡器）且
- LPOSCEN = 1

保持 LP 振荡器始终启用可使器件快速切换至 32 kHz 系统时钟以实现低功耗工作。返回至更快速的主振荡器时仍然需要起振延时。

20.2.4 锁相环（PLL）

PLL 对主振荡器产生的时钟进行倍频处理。PLL 的增益可选择为 4 倍、8 倍和 16 倍。表 20-3 汇总了输入和输出频率的范围。

表 20-3: PLL 频率范围

FIN	PLL 倍频器	FOUT
4 MHz-10 MHz	x4	16 MHz-40 MHz
4 MHz-10 MHz	x8	32 MHz-80 MHz
4 MHz-7.5 MHz	x16	64 MHz-120 MHz

PLL 具有一个锁定输出，当 PLL 进入锁相状态时，锁定输出有效。如果锁相环锁定失败（例如，由于噪声），锁定信号将变为无效。由 OSCCON 寄存器中的只读 LOCK 位来反映此信号的状态。

20.2.5 快速 RC 振荡器（FRC）

FRC 振荡器是一个快速（7.37 MHz ±2% 标称值）的内部 RC 振荡器。此振荡器旨在提供合理的器件工作速度，而无需使用外部晶振、陶瓷谐振器或 RC 网络。FRC 振荡器可与 PLL 结合使用以获取更高的时钟频率。

只要 OSCCON 寄存器（OSCCON<14:12>）中的当前振荡器选择控制位被设置为 001，dsPIC30F 就将使用 FRC 振荡器进行工作。

用户可通过 TUN<3:0>（OSCTUN<3:0>）指定的 4 位位域对内部快速 RC 振荡器（标称值为 7.37 MHz）进行微调。用户可以在工厂校准设定值的 +10.5%（840 kHz）至 -12%（960 kHz）范围内微调 FRC 振荡器，调节步长为设定值的 1.50%（见表 20-4）。

注： 提供了 OSCTUN 功能来帮助客户补偿在较宽的温度范围内对 FRC 频率的温度影响。调节步长是近似值，既不是特性值也未进行测试。

如果将 OSCCON<14:12> 设置为 111，则将 FPR<4:0> 设置为 00101、00110 或 00111 时，则相应可以分别使用 4、8 或 16 倍的 PLL 倍频器。

注： 当使用了 16 倍 PLL 时，禁止将 FRC 频率调节至 7.5 MHz 以上。

表 20-4: FRC 调节

TUN<3:0> 位	FRC 频率
0111	+ 10.5%
0110	+ 9.0%
0101	+ 7.5%
0100	+ 6.0%
0011	+ 4.5%
0010	+ 3.0%
0001	+ 1.5%
0000	中心频率（振荡器以校准频率运行）
1111	- 1.5%
1110	- 3.0%
1101	- 4.5%
1100	- 6.0%
1011	- 7.5%
1010	- 9.0%
1001	- 10.5%
1000	- 12.0%

20.2.6 低功耗 RC 振荡器（LPRC）

LPRC 振荡器是看门狗定时器（WDT）的组成部分，其标称振荡频率为 512 kHz。同时 LPRC 振荡器也是上电延时定时器（PWRT）电路、WDT 和时钟监视器电路的时钟源。在对低功耗要求高而对时序精度要求不高的场合下，LPRC 振荡器也可以用作低频时钟源。

由于 LPRC 振荡器是 PWRT 的时钟源，所以该振荡器在上电复位时总是启用。当 PWRT 超时时，如果下列条件中有一个成立，那么 LPRC 振荡器将仍然保持启用状态：

- 故障保护时钟监视器启用
- WDT 启用
- 通过 OSCCON 寄存器中的 COSC<2:0> 控制位选择 LPRC 振荡器作为系统时钟

如果上述任何一个条件不成立，则在 PWRT 超时后 LPRC 将关闭。

注 1： OSC2 引脚功能由主振荡器模式选择位（FPR<4:0>）决定。

注 2： OSC1 引脚不能用作 I/O 引脚，即使始终选择辅助振荡器或内部时钟源也是如此。

20.2.7 故障保护时钟监视器

故障保护时钟监视器 (FSCM) 使器件即使在振荡器出现故障的情况下仍能够继续工作。适当地对 FOSC 器件配置寄存器中的 FCKSM 配置位 (时钟切换和监视器选择位) 进行编程, 即可使能 FSCM 功能。如果使能了 FSCM 功能, 则 LPRC 内部振荡器将始终保持运行 (休眠模式除外), 不再受 SWDTEN 位控制。

在发生振荡器故障时, FSCM 将产生时钟故障陷阱事件, 并将系统时钟切换到 FRC 振荡器。然后, 用户可以选择尝试重启振荡器或执行受控关闭。用户只需将复位地址装入振荡器故障陷阱向量中, 就可将陷阱作为热复位处理。在这种情况下, 只要识别出时钟故障, CF (时钟故障) 状态位 (OSCCON<3>) 也将置 1。

在发生时钟故障时, WDT 不受影响, 继续靠 LPRC 时钟运行。

在退出 POR、BOR 或休眠模式后, 如果振荡器起振时间非常长, 则可能出现的情况是, 在振荡器完成启动之前, PWRT 定时器可能已超时。在这样的情形下, FSCM 将被激活, 并产生一个时钟故障陷阱, 同时 COSC<2:0> 位将被装入选定 FRC 振荡器的值。这实际上关闭了先前尝试启动的振荡器。

用户可在检测到此种情形时在时钟故障陷阱 ISR 中重启振荡器。

检测到时钟故障时, FSCM 模块会将时钟切换到 FRC 振荡器, 如下:

1. 将 FRC 振荡器选择值装入 COSC 位 (OSCCON<14:12>)。
2. 将 CF 位置 1 (OSCCON<3>)。
3. 将 OSWEN 控制位 (OSCCON<0>) 清零。

为了实现时钟切换, 时钟源被划分为四组:

1. 主振荡器
2. 辅助振荡器
3. 内部 FRC
4. 内部 LPRC

用户可以在这些功能组之间切换, 但是不能在同一组内各选项之间切换。如果选择了主振荡器组, 则在该组内部的选择将由 FPR<4:0> 配置位决定。

OSCCON 寄存器保存与时钟切换相关的控制位和状态位。

- COSC<2:0>: 只读状态位, 始终反映当前有效的振荡器组。

- NOSC<2:0>: 控制位, 对其进行写操作可选择新振荡器组。

- 在 POR 和 BOR 时, 将用配置位值 FOS<2:0> 装载 COSC<2:0> 和 NOSC<2:0>。

- LOCK: LOCK 状态位, 指示 PLL 锁定。
- CF: 只读状态位, 指示是否检测到时钟故障。
- OSWEN: 控制位, 当时钟切换过程开始时, 从 0 变为 1。将 OSWEN 控制位清零将中止正在进行的时钟切换 (用于挂起情形)。

如果配置位 FCKSM<1:0> = 1x, 则时钟切换和故障保护时钟监视器功能将被禁止。这是默认的配置位设置。

如果禁止了时钟切换, 则 FOS<2:0> 和 FPR<4:0> 位将直接控制振荡器选择, 而 COSC<2:0> 位将不再控制时钟选择。但是, 这些位将反映时钟源选择。

注: 当使能了故障保护时钟监视器时, 在应用中不应尝试切换至频率低于 100 kHz 的时钟。如果执行这样的时钟切换, 器件可能产生振荡器故障陷阱并切换至快速 RC 振荡器。

20.2.8 防止意外写入 OSCCON

由于 OSCCON 寄存器控制着时钟切换和时钟分频, 因此特意将该寄存器的写操作过程设计得较为复杂。

要写入 OSCCON 的低字节, 必须执行以下代码序列且其间不准插入其他任何指令:

```
Byte Write 0x46 to OSCCON low
Byte Write 0x57 to OSCCON low
```

允许在一个指令周期内对 OSCCON 进行字节写操作。写入所需值或使用位操作指令。

要写入 OSCCON 的高字节, 必须执行以下指令且其间不准插入其他任何指令:

```
Byte Write 0x78 to OSCCON high
Byte Write 0x9A to OSCCON high
```

允许在一个指令周期内对 OSCCON 进行字节写操作。写入所需值或使用位操作指令。

20.3 振荡器控制寄存器

振荡器由两个特殊功能寄存器 OSCCON 与 OSCTUN 以及一个配置寄存器 FOSC 来控制。

注： 本节中对 OSCCON 和 OSCTUN 特殊功能寄存器以及 FOSC 配置寄存器的说明只适用于 dsPIC30F 产品系列中的 dsPIC30F3014 和 dsPIC30F4013 器件。

寄存器 20-1: OSCCON: 振荡器控制寄存器

U-0	R-y	R-y	R-y	U-0	R/W-y	R/W-y	R/W-y	
—	COSC<2:0>			—	NOSC<2:0>			
bit 15								bit 8
R/W-0	R/W-0	R-0	U-0	R/W-0	U-0	R/W-0	R/W-0	
POST<1:0>		LOCK	—	CF	—	LPOSCEN	OSWEN	
bit 7								bit 0

图注：

R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15 **未实现：** 读为 0
- bit 14-12 **COSC<2:0>：** 当前振荡器组选择位（只读）
 - 111 = PLL 振荡器；PLL 源由 FPR<4:0> 位选择
 - 011 = 外部振荡器；OSC1/OSC2 引脚；外部振荡器配置由 FPR<4:0> 位选择
 - 010 = LPRC 内部低功耗 RC
 - 001 = FRC 内部快速 RC
 - 000 = LP 晶振；SOSCI/SOSCO 引脚
 在 POR 或 BOR 时设置为 FOS<2:0> 的值。在成功完成时钟切换时，装入 NOSC<2:0> 的值。当 FSCM 检测到故障时，设置为 FRC，并将时钟切换到 FRC。
- bit 11 **未实现：** 读为 0
- bit 10-8 **NOSC<2:0>：** 新振荡器组选择位
 - 111 = PLL 振荡器；PLL 源由 FPR<4:0> 位选择
 - 011 = 外部振荡器；OSC1/OSC2 引脚；外部振荡器配置由 FPR<4:0> 位选择
 - 010 = LPRC 内部低功耗 RC
 - 001 = FRC 内部快速 RC
 - 000 = LP 晶振；SOSCI/SOSCO 引脚
 在 POR 或 BOR 时设置为 FOS<2:0> 的值。
- bit 7-6 **POST<1:0>：** 振荡器后分频比选择位
 - 11 = 振荡器后分频器对时钟进行 64 分频
 - 10 = 振荡器后分频器对时钟进行 16 分频
 - 01 = 振荡器后分频器对时钟进行 4 分频
 - 00 = 振荡器后分频器不改变时钟

dsPIC30F3014/4013

寄存器 20-1: OSCCON: 振荡器控制寄存器 (续)

- bit 5 **LOCK:** PLL 锁定状态位 (只读)
1 = 表示 PLL 处于锁定状态
0 = 表示 PLL 处于未锁定 (禁止) 状态
POR 或 BOR 时复位。启动一个有效的时钟切换序列时复位。当 PLL 启动后达到 PLL 锁定状态时置 1。
锁定丢失时复位。不选择 PLL 作为系统时钟时读为 0。
- bit 4 **未实现:** 读为 0
- bit 3 **CF:** 时钟故障检测位 (可由应用程序读 / 清零)
1 = FSCM 已检测到时钟故障
0 = FSCM 未检测到时钟故障
POR 或 BOR 时复位。启动一个有效的时钟切换序列时复位。检测到时钟故障后置 1。
- bit 2 **未实现:** 读为 0
- bit 1 **LPOSCEN:** 32 kHz 辅助 (LP) 振荡器使能位
1 = 使能辅助振荡器
0 = 禁止辅助振荡器
POR 或 BOR 时复位。
- bit 0 **OSWEN:** 振荡器切换使能位
1 = 请求振荡器切换到由 NOSC<2:0> 位指定的选择
0 = 振荡器切换完成
POR 或 BOR 时复位。时钟切换成功后复位。冗余时钟切换后复位。
FSCM 将振荡器切换到第一组 (Group 1) 中的 FRC 后复位。

寄存器 20-2: OSCTUN: FRC 振荡器调节寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	TUN<3:0>			
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = 上电复位时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-4

未实现: 读为 0

bit 3-0

TUN<3:0>: TUN 位域的低两位。

由 TUN<3:0> 指定的 4 位位域指定用户对内部快速 RC 振荡器 (标称值为 7.37 MHz) 进行调节的能力。

0111 = 最大频率

0110 =

0101 =

0100 =

0011 =

0010 =

0001 =

0000 = 中间频率, 振荡器运行在校准频率上

1111 =

1110 =

1101 =

1100 =

1011 =

1010 =

1001 =

1000 = 最小频率

dsPIC30F3014/4013

寄存器 20-3: FOSC: 振荡器配置寄存器

U	U	U	U	U	U	U	U
—	—	—	—	—	—	—	—
bit 23						bit 16	

R/P	R/P	U	U	U	R/P	R/P	R/P
FCKSM<1:0>		—	—	—	FOS<2:0>		
bit 15						bit 8	

U	U	U	R/P	R/P	R/P	R/P	R/P
—	—	—	FPR<4:0>				
bit 7						bit 0	

图注:

R = 可读位 W = 可写位 U = 未实现位, 读为 0
 -n = 上电复位时的值 1 = 置 1 0 = 清零 x = 未知

bit 23-16 **未实现:** 读为 0

bit 15-14 **FCKSM<1:0>:** 时钟切换和监视器选择配置位
 1x = 禁止时钟切换, 禁止故障保护时钟监视器
 01 = 使能时钟切换, 禁止故障保护时钟监视器
 00 = 使能时钟切换, 使能故障保护时钟监视器

bit 13-11 **未实现:** 读为 0

bit 10-8 **FOS<2:0>:** POR 时的振荡器组选择位
 111 = PLL 振荡器; PLL 源由 FPR<4:0> 位选择 (见表 20-2)
 011 = EXT: 外部振荡器; OSC1/OSC2 引脚; 外部振荡器配置由 FPR<4:0> 位选择
 010 = LPRC: 内部低功耗 RC
 001 = FRC: 内部快速 RC
 000 = LPOSC: 低功耗晶振; SOSCI/SOSCO 引脚

bit 7-5 **未实现:** 读为 0

bit 4-0 **FPR<4:0>:** 主振荡器组内的振荡器选择位 (见表 20-2)

20.4 复位

dsPIC30F3014/4013 的各种复位有所不同:

- 上电复位 (POR)
- 正常工作期间的 $\overline{\text{MCLR}}$ 复位
- 休眠期间的 $\overline{\text{MCLR}}$ 复位
- 看门狗定时器 (WDT) 复位 (正常工作期间)
- 可编程欠压复位 (BOR)
- RESET 指令
- 由陷阱锁定导致的复位 (TRAPR)
- 由非法操作码或将未初始化的 W 寄存器用作地址指针而引起的复位 (IOPUWR)

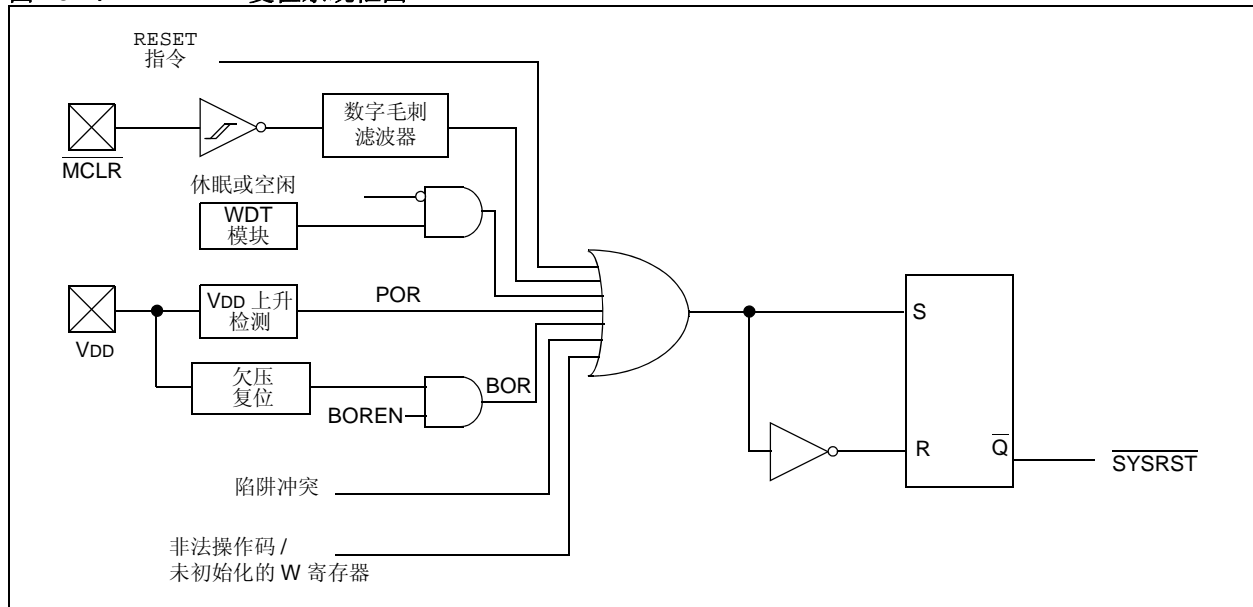
不同寄存器在各种复位条件下所受影响也不同。大多数寄存器不受 WDT 唤醒的影响, 因为 WDT 唤醒被认为是恢复正常工作。RCON 寄存器中的状态位在不同复位条件下的置 1 或清零状态也不相同, 如表 20-5 中所示。可在软件中使用这些位来确定复位的性质。

图 20-2 给出了片上复位电路的框图。

$\overline{\text{MCLR}}$ 复位路径上装有一个 $\overline{\text{MCLR}}$ 噪声滤波器。滤波器检测并滤除小脉冲。

内部产生的复位不会将 $\overline{\text{MCLR}}$ 引脚驱动为低电平。

图 20-2: 复位系统框图



20.4.1 POR: 上电复位

上电事件会在检测到 VDD 上升时产生一个内部 POR 脉冲。复位脉冲发生在标称值为 1.85V 的 POR 电路门限电压 (VPOR) 处。器件的电源电压特征必须满足起始电压和上升速率规范的要求。POR 脉冲复位 POR 定时器并将器件置于复位状态。POR 也选择由振荡器配置熔丝标识的器件时钟源。

POR 电路会产生一小段延时 T_{POR} , 其标称值为 $10\ \mu\text{s}$, 用于确保器件偏置电路稳定。另外, 还将应用一段由用户选择的上电延时超时时间 (T_{PWRT})。PWRT 参数基于器件配置位, 可以是 0 ms (无延时)、4 ms、16 ms 或 64 ms。器件上电时的总延时为 $T_{\text{POR}} + T_{\text{PWRT}}$ 。这些延时超时后, $\overline{\text{SYSRST}}$ 会在 Q1 时钟的下一前沿翻转且 PC 跳转到复位向量。

$\overline{\text{SYSRST}}$ 信号的时序如图 20-3 到图 20-5 所示。

dsPIC30F3014/4013

图 20-3: 上电时的延时时序 ($\overline{\text{MCLR}}$ 连接到 VDD)

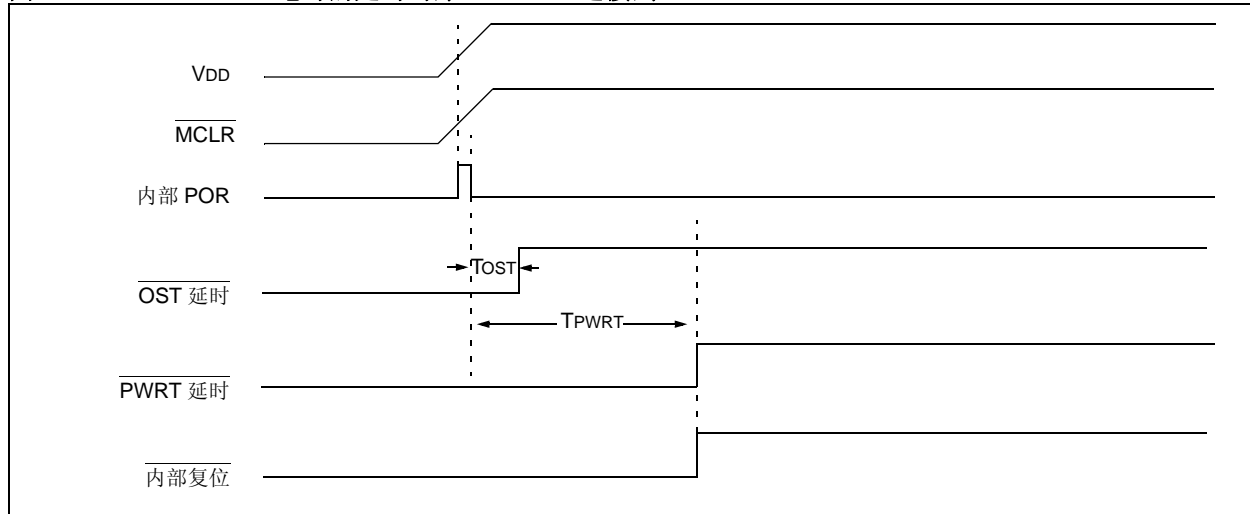


图 20-4: 上电时的延时时序 ($\overline{\text{MCLR}}$ 未连接到 VDD): 情形 1

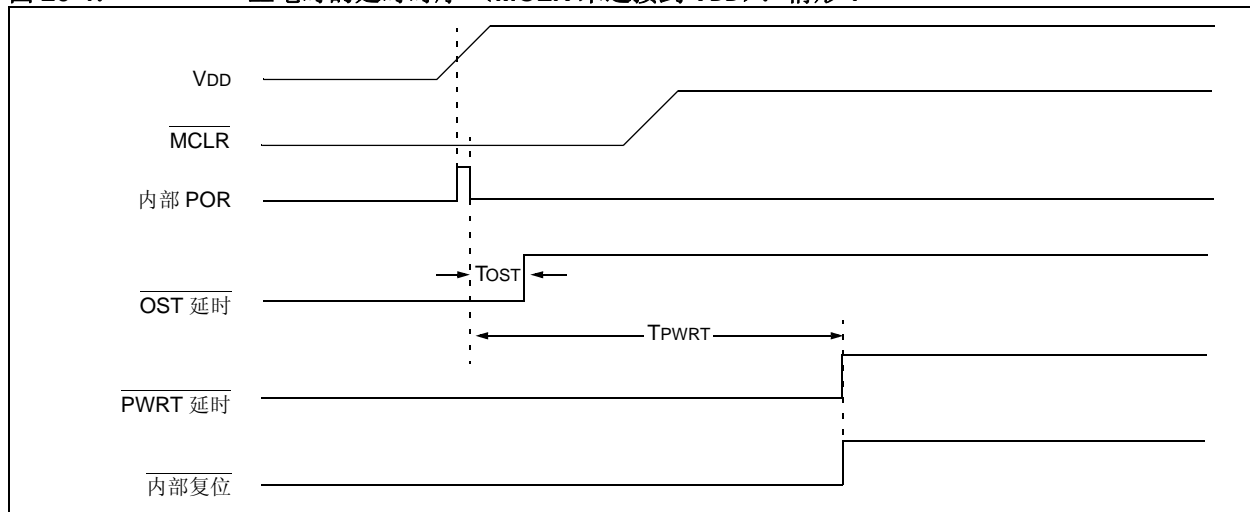
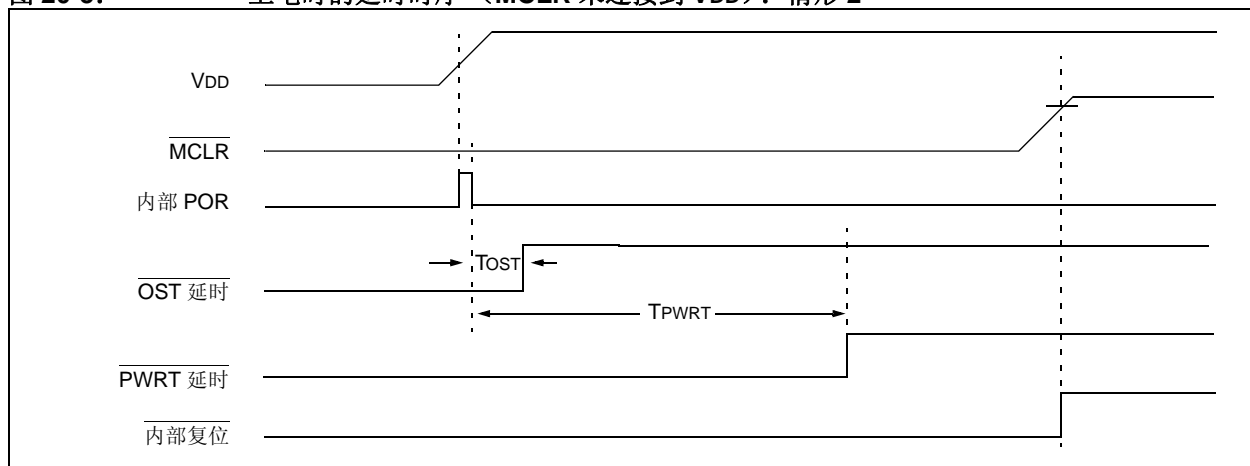


图 20-5: 上电时的延时时序 ($\overline{\text{MCLR}}$ 未连接到 VDD): 情形 2



20.4.1.1 具有长晶振起振时间的 POR (FSCM 已使能)

振荡器起振电路不与 POR 电路连接。某些晶振电路（尤其是低频晶振）的起振时间会相对较长。因此，POR 定时器和 PWRT 超时后可能会发生以下一种或多种情况：

- 振荡电路未起振。
- 振荡器起振定时器尚未超时（如果使用了晶振）。
- PLL 未实现锁定（如果使用了 PLL）。

如果使能了 FSCM 且以上情况之一成立，则发生时钟故障陷阱。器件自动将时钟切换为 FRC 振荡器，用户也可在陷阱 ISR 中将时钟切换为所需的晶振。

20.4.1.2 在无 FSCM 和 PWRT 的条件下工作

如果同时禁止了 FSCM 和上电延时定时器（PWRT），则器件在上电时会快速从复位中退出。如果时钟源为 FRC、LPRC、ERC 或 EC，器件将被立即激活。

如果 FSCM 禁止且系统时钟未起振，器件将冻结在复位状态，直到系统时钟起振为止。从用户的角度看，器件在系统时钟可用之前一直处于复位状态。

20.4.2 BOR：可编程欠压复位

BOR（欠压复位）模块基于一个内部参考电压电路。BOR 模块的主要目的在于发生欠压条件时产生器件复位。欠压条件一般是由交流电源线上的毛刺（即由于不良的电源传输线路导致的 AC 周期波形部分丢失，或导通一个大的感性负载时电流消耗过大而引起电压下降）引起的。

BOR 模块允许选择以下电压跳变点之一（见表 23-11）。

- 2.6V-2.71V
- 4.1V-4.4V
- 4.58V-4.73V

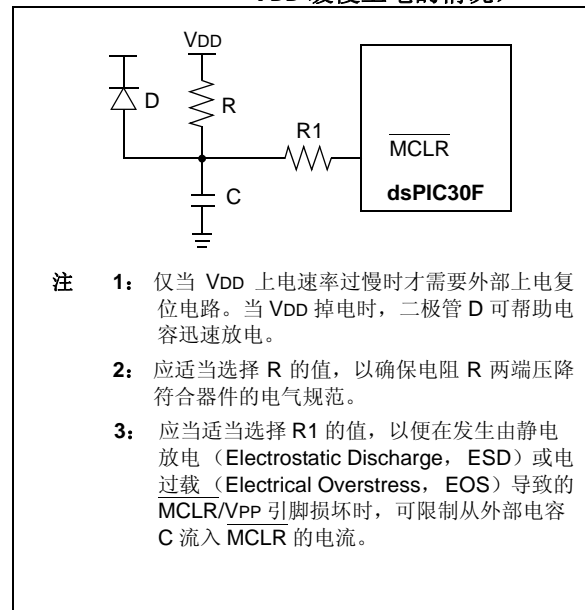
注： 此处所指的 BOR 电压跳变点为标称值，仅供设计参考。关于 BOR 电压范围规范，请参见特定器件数据手册中的电气规范。

BOR 产生复位脉冲，可将器件复位。BOR 根据器件配置位的值（FOS<2:0> 和 FPR<4:0>）选择时钟源。此外，如果选择了振荡器模式，则 BOR 激活振荡器起振定时器（OST）。系统时钟暂停，直到 OST 超时为止。如果使用了 PLL，则时钟暂停，直到 LOCK 位（OSCCON<5>）为 1 为止。

同时，在内部复位结束前应用 POR 延时（TPOR）和 PWRT 延时（TPWRT）。如果 TPWRT = 0 且正在使用晶振，则应用 TFSCM = 100 μs 的标称延时。该情形下的总延时为（TPOR + TFSCM）。

BOR 状态位（RCON<1>）置 1 表示已发生了 BOR。如果使能 BOR 电路，则它继续在休眠或空闲模式下运行，如果 VDD 下降到 BOR 门限电压以下，将复位器件。

图 20-6： 外部上电复位电路（用于 VDD 缓慢上电的情况）



注： 也可使用 MCP1XX 和 MCP8XX 等专用监视器件作为外部上电复位电路。

dsPIC30F3014/4013

表 20-5 显示了 RCON 寄存器的复位状态。由于 RCON 寄存器中的控制位都是可读写的，表中的信息意味着：所有位状态是在“条件”这一列指定操作之前的位状态的反码。

表 20-5: RCON 寄存器的初始状态: 情形 1

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	IDLE	SLEEP	POR	BOR
上电复位	0x000000	0	0	0	0	0	0	0	1	1
欠压复位	0x000000	0	0	0	0	0	0	0	0	1
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	0	0	0	0
正常工作期间的软件复位	0x000000	0	0	0	1	0	0	0	0	0
休眠期间的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	0	1	0	0
空闲期间的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	1	0	0	0
WDT 超时复位	0x000000	0	0	0	0	1	0	0	0	0
WDT 唤醒	PC + 2	0	0	0	0	1	0	1	0	0
通过中断从休眠唤醒	PC + 2 ⁽¹⁾	0	0	0	0	0	0	1	0	0
时钟故障陷阱	0x000004	0	0	0	0	0	0	0	0	0
陷阱复位	0x000000	1	0	0	0	0	0	0	0	0
非法操作陷阱	0x000000	0	1	0	0	0	0	0	0	0

图注: u = 不变, x = 未知, — = 未实现位, 读为 0

注 1: 当由于允许的中断而唤醒时, PC 将被装入相应的中断向量。

表 20-6 给出了 RCON 寄存器位状态的第二个示例。在此情形中, 假设用户在发生条件列中指定的动作前已将特定位置 1 或清零。

表 20-6: RCON 寄存器的初始状态: 情形 2

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	IDLE	SLEEP	POR	BOR
上电复位	0x000000	0	0	0	0	0	0	0	1	1
欠压复位	0x000000	u	u	u	u	u	u	u	0	1
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	0	0	0	0	u	u
正常工作期间的软件复位	0x000000	u	u	0	1	0	0	0	u	u
休眠期间的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	u	0	0	1	u	u
空闲期间的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	u	0	1	0	u	u
WDT 超时复位	0x000000	u	u	0	0	1	0	0	u	u
WDT 唤醒	PC + 2	u	u	u	u	1	u	1	u	u
通过中断从休眠唤醒	PC + 2 ⁽¹⁾	u	u	u	u	u	u	1	u	u
时钟故障陷阱	0x000004	u	u	u	u	u	u	u	u	u
陷阱复位	0x000000	1	u	u	u	u	u	u	u	u
非法操作陷阱	0x000000	u	1	u	u	u	u	u	u	u

图注: u = 不变, x = 未知, — = 未实现位, 读为 0

注 1: 当由于允许的中断而唤醒时, PC 将被装入相应的中断向量。

20.5 看门狗定时器 (WDT)

20.5.1 看门狗定时器的工作原理

看门狗定时器 (WDT) 的首要功能就是在发生软件故障时复位处理器。WDT 是一个可自由运行的定时器，它不依赖片上 RC 振荡器，也无需外部元件。因此，即使主处理器时钟 (例如，晶振) 发生故障，WDT 定时器也能继续运行。

20.5.2 使能和禁止 WDT

只能通过配置寄存器 **FWDTC** 中的一个配置位 (**FWDTCEN**) 来使能或禁止看门狗定时器。

设置 **FWDTCEN = 1** 将使能看门狗定时器。使能定时器的操作可在编程器件时完成。默认情况下，芯片擦除后，**FWDTCEN** 位 = 1。任何能够编程 dsPIC30F 器件的编程器都能对此位和其他配置位编程。

如果使能，WDT 将不断递增直到溢出或超时为止。WDT 超时将强制器件复位 (休眠期间除外)。为了防止 WDT 超时，用户必须使用 **CLRWDTC** 指令清零看门狗定时器。

如果 WDT 在休眠期间超时，则器件将唤醒。**RCON** 寄存器中的 **WDTCO** 位清零表示由于 WDT 超时而导致器件唤醒。

设置 **FWDTCEN = 0** 允许用户软件通过 **SWDTCEN** (**RCON<5>**) 控制位使能 / 禁止看门狗定时器。

20.6 低电压检测

低电压检测 (Low-Voltage Detect, LVD) 模块用于检测器件的 **VDD** 何时降低到门限值 **VLVD** 以下，**VLVD** 可由 **LVDL<3:0>** 位 (**RCON<11:8>**) 确定，因此可由用户编程。内部参考电压电路需要经过一段标称时间来稳定，**BGST** 位 (**RCON<13>**) 指示参考电压何时稳定。

在一些器件中，可通过 **LVDIN** 引脚外部施加 LVD 门限电压。

将 **LVDEN** 位 (**RCON<12>**) 置 1 可使能 LVD 模块。

20.7 省电模式

有两种可通过执行特殊指令 **PWRSVAV** 进入的省电状态：休眠和空闲。

PWRSVAV 指令的格式如下：

PWRSVAV <parameter>，其中 “parameter” 定义空闲或休眠模式。

20.7.1 休眠模式

在休眠模式中，到 **CPU** 和外设的时钟被关闭。如果正在使用片上振荡器，也会将其关闭。

在休眠期间，因为没有要监视的时钟，所以故障保护时钟监视器不可用。但是，如果 WDT 在休眠期间运行，则 **LPRC** 时钟保持活动。

欠压保护电路和低电压检测 (LVD) 电路在使能时可在休眠期间继续运行。

当发生以下任何一种情况时，处理器将从休眠唤醒：

- 任何单独允许且达到所需优先级的中断
- 任何复位 (**POR**、**BOR** 和 **MCLR**)
- **WDT** 超时

从休眠模式唤醒时，处理器将重新启动进入休眠模式前有效的时钟。如果使能了时钟切换，则由 **COSC<2:0>** 位确定唤醒时要使用的振荡器源。如果禁止时钟切换，则只有一个系统时钟。

注： 如果发生 **POR** 或 **BOR**，则根据 **FOS<2:0>** 和 **FPR<4:0>** 配置位选择振荡器。

如果时钟源为某一振荡器，则器件时钟将暂停，直到 **OST** 超时为止 (表示振荡器稳定)。如果使用了 **PLL**，则系统时钟暂停，直到 **LOCK = 1** (表示 **PLL** 已稳定) 为止。在两种情形下，都应用 **TPOR**、**TLOCK** 和 **TPWRT** 延时。

如果使用 **EC**、**FRC**、**LPRC** 或 **ERC** 振荡器，则应用 **TPOR** (约 10 μ s) 的延时。这可能是从休眠唤醒时最短的延时。

此外，如果 **LP** 振荡器在休眠时有效且 **LP** 是唤醒时使用的振荡器，则起振延时等于 **TPOR**。不应用 **PWRT** 延时和 **OST** 定时器延时。为了使从休眠唤醒时的起振延时最短，在进入休眠前应选择较快的唤醒选项之一。

任何（使用相应的 IE 位）单独允许并满足优先级占先的中断均可唤醒处理器。处理器处理中断并跳转到 ISR。RCON 寄存器中的 SLEEP 状态位在唤醒时置 1。

注： 尽管应用了各种延时（TPOR、TLOCK 和 TPWRT），晶振（和 PLL）仍可能在延时结束时处于非活动状态（例如，低频晶振）。在这种情形下，如果使能 FSCM，器件会将此情况检测为一个时钟故障并处理时钟故障陷阱，FRC 振荡器将使能且用户需要重新使能晶振。如果未使能 FSCM，器件将直接暂停代码执行，直到时钟稳定为止，并保持休眠状态直到振荡器时钟启动完毕。

所有的复位都会将处理器从休眠模式唤醒。除 POR 外的任何复位都会将 SLEEP 状态位置 1。POR 时，SLEEP 位被清零。

如果使能看门狗定时器，处理器将在 WDT 超时从休眠模式唤醒。SLEEP 和 WDTO 状态位将同时置 1。

20.7.2 空闲模式

在空闲模式中，到 CPU 的时钟被关闭，而到外设的时钟保持运行。与休眠模式不同，时钟源将保持活动。

有些外设在其模块中具有控制位，可允许其在空闲模式中运行。

如果使能时钟故障检测，LPRC 故障保护时钟保持活动。

当发生以下任何一种情况时，处理器将从空闲模式中唤醒：

- 任何单独允许（即中断允许位为 1）且满足所需要的优先级的中断
- 任何复位（POR、BOR 和 MCLR）
- WDT 超时

从空闲模式中唤醒时，时钟重新应用到 CPU，指令从 PWRSAV 指令后的指令立即开始执行。

任何（使用 IE 位）单独允许并满足优先级占先的中断均可唤醒处理器。处理器处理中断并跳转到 ISR。RCON 寄存器中的 IDLE 状态位在唤醒时置 1。

除 POR 外的任何复位会将 IDLE 状态位置 1。POR 时，空闲位被清零。

如果使能看门狗定时器，处理器将在 WDT 超时从 IDLE 模式唤醒。IDLE 和 WDTO 状态位将同时置 1。

与从休眠唤醒不同，从空闲唤醒时无延时。

20.8 器件配置寄存器

各器件配置寄存器中的配置位用来指定器件模式，并可由器件编程器或通过使用器件的在线串行编程（ICSP™）功能进行编程。各器件配置寄存器均为 24 位寄存器，但仅使用各寄存器的低 16 位保存配置数据。有 5 个器件配置寄存器可供用户使用：

1. FOSC (0xF80000)：振荡器配置寄存器
2. FWDT (0xF80002)：看门狗定时器配置寄存器
3. FBORPOR (0xF80004)：BOR 和 POR 配置寄存器
4. FGS (0xF8000A)：通用代码段配置寄存器
5. FICD (0xF8000C)：调试配置寄存器

在器件编程器中选择器件时，配置位的位置是自动处理的。所需的配置位状态必须在源代码中指定（取决于所使用的语言工具），或通过编程接口指定。器件编程后，应用软件可能会通过表读指令读取配置位的值。更多信息，请参见器件的编程规范。

注： 如果已编程了代码保护配置熔丝位（FGS<GCP> 和 FGS<GWRP>），则只有在电压 $V_{DD} \geq 4.5V$ 时才能擦除整个代码保护器件。

20.9 外设模块禁止 (PMD) 寄存器

外设模块禁止 (PMD) 寄存器提供了一种通过停止提供给该模块的所有时钟源而禁止外设模块的方法。当通过相应的 PMD 控制位禁止外设时, 外设处于功耗最小的状态。与外设相关的控制和状态寄存器也被禁止, 因此向这些寄存器执行写操作不起作用且读取值无效。

只有当 PMD 寄存器中的相关位清零且特定 dsPIC DSC 器件支持该外设时才可使能外设模块。如果器件上已有该外设, 则默认在 PMD 寄存器中将其使能。

- 注 1:** 如果 PMD 位置 1, 则 1 个指令周期的延时后, 相应的模块会被禁止。与之相似, 如果 PMD 位被清零, 则 1 个指令周期的延时后相应的模块使能 (假设模块控制寄存器已经配置为使能模块的工作)。
- 2:** 在 dsPIC30F3014 器件中, T4MD、T5MD、IC7MD、IC8MD、OC3MD、OC4MD 和 DCIMD 是可读写的, 且置 1 时它们读为 1。

20.10 在线调试器

如果选择 MPLAB[®] ICD 2 作为调试器, 将使能在线调试功能。这一功能使器件在与 MPLAB IDE 配合使用时能轻松进行调试。器件使能此功能时, 会有一些资源不能通用。这些资源包括数据 RAM 的前 80 个字节和两个 I/O 引脚。

用户可使用 MPLAB IDE 中的配置选项选择四对调试 I/O 引脚中的一对。这几对引脚命名为 EMUD/EMUC、EMUD1/EMUC1、EMUD2/EMUC2 和 EMUD3/EMUC3。

在各种情形下, 选定的 EMUD 引脚为仿真 / 调试数据线, EMUC 引脚为仿真 / 调试时钟线。这些引脚与 MPLAB ICD 2 模块接口, 该模块可从 Microchip 获取。MPLAB ICD 2 使用所选的一对调试 I/O 引脚来发送命令和接收响应, 以及发送和接收数据。要使用器件的在线调试功能, 在设计中必须实现到 MCLR、VDD、VSS、PGC、PGD 和选定的一对 EMUDx/EMUCx 引脚的 ICSP 连接。

这样就产生了两种可能性:

1. 如果 EMUD/EMUC 被选为调试 I/O 引脚对, 则仅需要一个 5 引脚的接口, 因为在所有的 dsPIC30F 器件中, EMUD 和 EMUC 引脚的功能是与 PGD 和 PGC 引脚的功能复用的。
2. 如果 EMUD1/EMUC1、EMUD2/EMUC2 或 EMUD3/EMUC3 被选为调试 I/O 引脚, 则需要一个 7 引脚的接口, 因为 EMUDx/EMUCx 引脚的功能 (x=1、2 或 3) 不与 PGD 和 PGC 引脚的功能复用。

表 20-7: 系统集成寄存器映射 (1)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位状态
RCON	0740	TRAPR	IOPUWR	BGST	LVDEN	LVDL<3:0>			EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	(注 2)	
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>		POST<1:0>	LOCK	—	CF	—	LPOSCEN	OSWEN	(注 3)		
OSCTUN	0744	—	—	—	—	—	—	—	—	—	—	—	—	TUN3	TUN2	TUN1	TUN0	0000 0000 0000 0000
PMD1	0770	T5MD ⁽⁴⁾	T4MD ⁽⁴⁾	T3MD	T2MD	T1MD	—	—	DCIMD ⁽⁴⁾	I2CMD	U2MD	U1MD	—	SPI1MD	—	C1MD	ADCMD	0000 0000 0000 0000
PMD2	0772	IC8MD ⁽⁴⁾	IC7MD ⁽⁴⁾	—	—	—	—	IC2MD	IC1MD	—	—	—	—	OC4MD ⁽⁴⁾	OC3MD ⁽⁴⁾	OC2MD	OC1MD	0000 0000 0000 0000

图注: — = 未实现位, 读为 0

- 注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。
 2: 复位状态取决于复位类型。
 3: 复位状态取决于配置位。
 4: 这些位在 dsPIC30F3014 器件中不可用。

表 20-8: 器件配置寄存器映射 (1)

名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	
FOSC	F80000	FCKSM<1:0>			—	—	—	FOS<2:0>			—	—	—	FPR<4:0>				
FWDT	F80002	FWDTEN	—	—	—	—	—	—	—	—	—	FWPSA<1:0>		FWPSB<3:0>				
FBORPOR	F80004	MCLREN	—	—	—	—	PWMPIN ⁽²⁾	HPOL ⁽²⁾	LPOL ⁽²⁾	BOREN	—	BORV<1:0>		—	—	FPWRT<1:0>		
FBS	F80006	—	—	保留 ⁽³⁾		—	—	—	保留 ⁽³⁾	—	—	—	—	保留 ⁽³⁾				
FSS	F80008	—	—	保留 ⁽³⁾		—	—	保留 ⁽³⁾		—	—	—	—	保留 ⁽³⁾				
FGS	F8000A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	保留 ⁽⁴⁾	GCP	GWRP
FICD	F8000C	BKBUG	COE	—	—	—	—	—	—	—	—	—	—	—	—	—	ICS<1:0>	

图注: — = 未实现位, 读为 0

- 注 1: 有关寄存器位域的描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。
 2: 这些位是保留位 (读为 1 并且必须编程为 1)。
 3: 保留位读为 1 并且必须编程为 1。
 4: FGS<2> 位是 GCP 位 (FGS<1>) 的只读形式。

21.0 指令集综述

注 本数据手册总结了 dsPIC30F 系列器件的功能，但是不应把本手册当作无所不包的参考手册来使用。欲了解有关 CPU、外设、寄存器说明以及通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲了解有关器件指令集和编程的更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

dsPIC30F 指令集与以前的 PIC® MCU 指令集相比，增加了很多增强功能，并保持了易于从其他 PIC MCU 指令集移植的特点。

大多数指令的长度是一个程序存储字 (24 位)，只有 3 条指令需要占用两个程序存储单元。

每条单字指令都是一个 24 位字，由一个 8 位的操作码 (指定指令类型) 和一个或多个操作数 (进一步指定指令操作) 组成。

指令集是高度正交的，分为以下 5 种基本类型：

- 针对字或字节的操作类指令
- 针对位的操作类指令
- 立即数操作类指令
- DSP 操作类指令
- 控制操作类指令

表 21-1 显示了描述指令的通用符号。

表 21-2 中的 dsPIC30F 指令集汇总列出了所有指令及每条指令影响的状态标志位。

大部分针对字或字节的 W 寄存器指令 (包含桶形移位指令) 都有三个操作数：

- 第一个源操作数通常是不带地址修改符的 “Wb” 寄存器
- 第二个源操作数通常是带或不带地址修改符的 “Ws” 寄存器
- 结果的目标通常是带或不带地址修改符的 “Wd” 寄存器

而针对字或字节的文件寄存器指令有两个操作数：

- 文件寄存器 (由 “f” 指定)
- 目标寄存器 (可以是文件寄存器 “f” 也可以是表示为 “WREG” 的 W0 寄存器)

大部分针对位的操作类指令 (包括简单循环/移位指令) 都有两个操作数：

- W 寄存器 (带或不带地址修改符) 或文件寄存器 (由 “Ws” 或 “f” 的值指定)
- W 寄存器或文件寄存器中的位 (由立即数值指定，或由寄存器 “Wb” 中的内容间接指定)

涉及数据传送的立即数指令可使用以下操作数：

- 要被装载到 W 寄存器或文件寄存器的立即数值 (由 “k” 值指定)
- 要装载立即数值的 W 寄存器或文件寄存器 (由 “Wb” 或 “f” 指定)

而涉及算术或逻辑运算的立即数指令使用以下操作数：

- 第一个源操作数是不带任何地址修改符的 “Wb” 寄存器
- 第二个源操作数是立即数值
- 结果的目标地址 (仅在与第一个源操作数不同时) 通常为带或不带地址修改符的 “Wd” 寄存器

DSP 指令的 MAC 类指令可使用以下操作数：

- 要使用的累加器 (A 或 B) (必需的操作数)
- 要用作两个操作数的 W 寄存器
- X 和 Y 地址空间预取操作
- X 和 Y 地址空间预取目标寄存器
- 累加器回写目标寄存器

其他 DSP 指令不涉及任何乘法，可包括：

- 要使用的累加器 (必需)
- 带或不带地址修改符的源或目标操作数 (分别用 Wso 或 Wdo 指定)
- 由 W 寄存器 “Wn” 或立即数值指定的移位数量

控制指令可使用以下操作数：

- 程序存储器地址
- 表读和表写指令的模式

除某些双字指令外，所有指令都是单字指令。双字指令中所有必需的信息都在此 48 位中，第二个字的高 8 位都是 0。如果第二个字作为一条指令来执行的话，它将作为一条 NOP 指令来执行。

dsPIC30F3014/4013

除非条件测试为真或者指令执行后改变了程序计数器的值，否则执行大多数单字指令执行都只需一个指令周期。对于上述两种特殊情况，执行指令需要两个指令周期，第二个指令周期相当于执行了一条 NOP 指令。值得注意的特殊指令有：BRA（无条件/计算转移指令）、间接 CALL/GOTO 指令、所有表读和表写指令以及 RETURN/RETFIE 指令，这些指令都是单字指令，但执行起来需要 2 或 3 个指令周期。某些涉及跳过下一条指令的指令，

执行跳过时需要两或三个指令周期，具体周期数取决于被跳过的指令是单字指令还是双字指令。此外，双字传送需要两个周期。双字指令的执行需要两个指令周期。

注 欲知有关指令集更多信息，请参见《16 位 MCU 和 DSC 程序员参考手册》(DS70157E_CN)。

表 21-1: 操作码描述中使用的符号

字段	说明
#text	表示由“text”指定的立即数
(text)	表示“text 的内容”
[text]	表示“由 text 指定的地址单元”
{ }	可选字段或操作
<n:m>	寄存器位域
.b	字节模式选择
.d	双字模式选择
.S	影子寄存器选择
.w	字模式选择（默认）
Acc	两个累加器 {A, B} 之一
AWB	累加器回写目标地址寄存器 $\in \{W13, [W13]+2\}$
bit4	4 位位选择字段（用于字寻址指令） $\in \{0...15\}$
C, DC, N, OV, Z	MCU 状态位：进位、半进位、负数、溢出标志和全零标志
Expr	绝对地址、标号或表达式（由链接器解析）
f	文件寄存器地址 $\in \{0x0000...0x1FFF\}$
lit1	1 位无符号立即数 $\in \{0,1\}$
lit4	4 位无符号立即数 $\in \{0...15\}$
lit5	5 位无符号立即数 $\in \{0...31\}$
lit8	8 位无符号立即数 $\in \{0...255\}$
lit10	10 位无符号立即数，字节模式下 $\in \{0...255\}$ ，字模式下 $\in \{0:1023\}$
lit14	14 位无符号立即数 $\in \{0...16384\}$
lit16	16 位无符号立即数 $\in \{0...65535\}$
lit23	23 位无符号立即数 $\in \{0...8388608\}$ ，LSb 必须为 0
None	该字段不必有输入项，可以为空白
OA, OB, SA, SB	DSP 状态位：AccA 溢出，AccB 溢出，AccA 饱和和 AccB 饱和
PC	程序计数器
Slit10	10 位有符号立即数 $\in \{-512...511\}$
Slit16	16 位有符号立即数 $\in \{-32768...32767\}$
Slit6	6 位有符号立即数 $\in \{-16...16\}$

表 21-1: 操作码描述中使用的符号 (续)

字段	说明
Wb	基本 W 寄存器 $\in \{W0..W15\}$
Wd	目标 W 寄存器 $\in \{Wd, [Wd], [Wd++] , [Wd--], [++Wd], [--Wd]\}$
Wdo	目标 W 寄存器 $\in \{Wnd, [Wnd], [Wnd++] , [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb]\}$
Wm,Wn	被除数和除数工作寄存器对 (直接寻址)
Wm*Wm	用于平方指令的被乘数和乘数工作寄存器对 $\in \{W4*W4, W5*W5, W6*W6, W7*W7\}$
Wm*Wn	用于 DSP 指令的被乘数和乘数工作寄存器对 $\in \{W4*W5, W4*W6, W4*W7, W5*W6, W5*W7, W6*W7\}$
Wn	16 个工作寄存器之一 $\in \{W0..W15\}$
Wnd	16 个目标工作寄存器之一 $\in \{W0..W15\}$
Wns	16 个源工作寄存器之一 $\in \{W0..W15\}$
WREG	W0 (文件寄存器指令中的工作寄存器)
Ws	源 W 寄存器 $\in \{Ws, [Ws], [Ws++] , [Ws--], [++Ws], [--Ws]\}$
Wso	源 W 寄存器 $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb]\}$
Wx	用于 DSP 指令的 X 数据空间预取地址寄存器 $\in \{[W8] += 6, [W8] += 4, [W8] += 2, [W8], [W8] -= 6, [W8] -= 4, [W8] -= 2, [W9] += 6, [W9] += 4, [W9] += 2, [W9], [W9] -= 6, [W9] -= 4, [W9] -= 2, [W9+W12], \text{空}\}$
Wxd	用于 DSP 指令的 X 数据空间预取目标寄存器 $\in \{W4..W7\}$
Wy	用于 DSP 指令的 Y 数据空间预取地址寄存器 $\in \{[W10] += 6, [W10] += 4, [W10] += 2, [W10], [W10] -= 6, [W10] -= 4, [W10] -= 2, [W11] += 6, [W11] += 4, [W11] += 2, [W11], [W11] -= 6, [W11] -= 4, [W11] -= 2, [W11+W12], \text{空}\}$
Wyd	用于 DSP 指令的 Y 数据空间预取目标寄存器 $\in \{W4..W7\}$

dsPIC30F3014/4013

表 21-2: 指令集汇总

基本指令 #	汇编助记符	汇编语法	说明	字数	周期数	受影响的标志位
1	ADD	ADD Acc	累加器相加	1	1	OA,OB,SA,SB
		ADD f	$f = f + WREG$	1	1	C,DC,N,OV,Z
		ADD f, WREG	$WREG = f + WREG$	1	1	C,DC,N,OV,Z
		ADD #lit10, Wn	$Wd = lit10 + Wd$	1	1	C,DC,N,OV,Z
		ADD Wb, Ws, Wd	$Wd = Wb + Ws$	1	1	C,DC,N,OV,Z
		ADD Wb, #lit5, Wd	$Wd = Wb + lit5$	1	1	C,DC,N,OV,Z
		ADD Wso, #Slit4, Acc	16 位有符号数加到累加器	1	1	OA,OB,SA,SB
2	ADDC	ADDC f	$f = f + WREG + (C)$	1	1	C,DC,N,OV,Z
		ADDC f, WREG	$WREG = f + WREG + (C)$	1	1	C,DC,N,OV,Z
		ADDC #lit10, Wn	$Wd = lit10 + Wd + (C)$	1	1	C,DC,N,OV,Z
		ADDC Wb, Ws, Wd	$Wd = Wb + Ws + (C)$	1	1	C,DC,N,OV,Z
		ADDC Wb, #lit5, Wd	$Wd = Wb + lit5 + (C)$	1	1	C,DC,N,OV,Z
3	AND	AND f	$f = f .AND. WREG$	1	1	N,Z
		AND f, WREG	$WREG = f .AND. WREG$	1	1	N,Z
		AND #lit10, Wn	$Wd = lit10 .AND. Wd$	1	1	N,Z
		AND Wb, Ws, Wd	$Wd = Wb .AND. Ws$	1	1	N,Z
		AND Wb, #lit5, Wd	$Wd = Wb .AND. lit5$	1	1	N,Z
4	ASR	ASR f	f = 算术右移 f	1	1	C,N,OV,Z
		ASR f, WREG	WREG = 算术右移 f	1	1	C,N,OV,Z
		ASR Ws, Wd	Wd = 算术右移 Ws	1	1	C,N,OV,Z
		ASR Wb, Wns, Wnd	Wnd = 将 Wb 算术右移 Wns 位	1	1	N,Z
		ASR Wb, #lit5, Wnd	Wnd = 将 Wb 算术右移 lit5 位	1	1	N,Z
5	BCLR	BCLR f, #bit4	将 f 寄存器中的某位清零	1	1	无
		BCLR Ws, #bit4	将 Ws 中的某位清零	1	1	无
6	BRA	BRA C, Expr	进位则跳转	1	1 (2)	无
		BRA GE, Expr	如果大于等于则跳转	1	1 (2)	无
		BRA GEU, Expr	如果无符号大于等于则跳转	1	1 (2)	无
		BRA GT, Expr	如果大于则跳转	1	1 (2)	无
		BRA GTU, Expr	如果无符号大于则跳转	1	1 (2)	无
		BRA LE, Expr	如果小于等于则跳转	1	1 (2)	无
		BRA LEU, Expr	如果无符号小于等于则跳转	1	1 (2)	无
		BRA LT, Expr	如果小于则跳转	1	1 (2)	无
		BRA LTU, Expr	如果无符号小于则跳转	1	1 (2)	无
		BRA N, Expr	为负则跳转	1	1 (2)	无
		BRA NC, Expr	无进位则跳转	1	1 (2)	无
		BRA NN, Expr	不为负则跳转	1	1 (2)	无
		BRA NOV, Expr	不溢出则跳转	1	1 (2)	无
		BRA NZ, Expr	不为零则跳转	1	1 (2)	无
		BRA OA, Expr	累加器 A 溢出则跳转	1	1 (2)	无
		BRA OB, Expr	累加器 B 溢出则跳转	1	1 (2)	无
		BRA OV, Expr	溢出则跳转	1	1 (2)	无
		BRA SA, Expr	累加器 A 饱和则跳转	1	1 (2)	无
		BRA SB, Expr	累加器 B 饱和则跳转	1	1 (2)	无
		BRA Expr	无条件跳转	1	2	无
BRA Z, Expr	为零则跳转	1	1 (2)	无		
		BRA Wn	计算转移	1	2	无
7	BSET	BSET f, #bit4	将 f 寄存器中的某位置 1	1	1	无
		BSET Ws, #bit4	将 Ws 中的某位置 1	1	1	无
8	BSW	BSW.C Ws, Wb	将 C 位写入 Ws<Wb>	1	1	无
		BSW.Z Ws, Wb	将 Z 位写入 Ws<Wb>	1	1	无

表 21-2: 指令集汇总 (续)

基本指令 #	汇编助记符	汇编语法	说明	字数	周期数	受影响的标志位
9	BTG	BTG $f, \#bit4$	将 f 中的某位取反	1	1	无
		BTG $Ws, \#bit4$	将 Ws 中的某位取反	1	1	无
10	BTSC	BTSC $f, \#bit4$	检测 f 中的某位, 为 0 则跳过	1	1 (2 或 3)	无
		BTSC $Ws, \#bit4$	测试 Ws 中的某位, 为 0 则跳过	1	1 (2 或 3)	无
11	BTSS	BTSS $f, \#bit4$	检测 f 中的某位, 为 1 则跳过	1	1 (2 或 3)	无
		BTSS $Ws, \#bit4$	检测 Ws 中的某位, 为 1 则跳过	1	1 (2 或 3)	无
12	BTST	BTST $f, \#bit4$	检测 f 中的某位	1	1	Z
		BTST.C $Ws, \#bit4$	检测 Ws 中的位并将结果存储到 C	1	1	C
		BTST.Z $Ws, \#bit4$	检测 Ws 中的位并将结果的反码存储到 Z	1	1	Z
		BTST.C Ws, Wb	检测 $Ws < Wb >$ 并将结果存储到 C	1	1	C
		BTST.Z Ws, Wb	检测 $Ws < Wb >$ 并将结果的反码存储到 Z	1	1	Z
13	BTSTS	BTSTS $f, \#bit4$	检测 f 寄存器中的位并将该位置 1	1	1	Z
		BTSTS.C $Ws, \#bit4$	检测 Ws 中的位并将结果存储到 C, 然后将将被检测位置 1	1	1	C
		BTSTS.Z $Ws, \#bit4$	检测 Ws 中的位并将结果的反码存储到 Z, 然后将将被检测位置 1	1	1	Z
14	CALL	CALL $lit23$	调用子程序	2	2	无
		CALL Wn	间接调用子程序	1	2	无
15	CLR	CLR f	$f = 0x0000$	1	1	无
		CLR WREG	WREG = $0x0000$	1	1	无
		CLR Ws	$Ws = 0x0000$	1	1	无
		CLR $Acc, Wx, Wxd, Wy, Wyd, AWB$	清零累加器	1	1	OA,OB,SA,SB
16	CLRWDI	CLRWDI	将看门狗定时器清零	1	1	WDTO, 休眠
17	COM	COM f	$f = \bar{f}$	1	1	N,Z
		COM $f, WREG$	WREG = \bar{f}	1	1	N,Z
		COM Ws, Wd	$Wd = \bar{Ws}$	1	1	N,Z
18	CP	CP f	将 f 寄存器与 WREG 作比较	1	1	C,DC,N,OV,Z
		CP $Wb, \#lit5$	将 Wb 与 $lit5$ 作比较	1	1	C,DC,N,OV,Z
		CP Wb, Ws	将 Wb 与 Ws 作比较 ($Wb - Ws$)	1	1	C,DC,N,OV,Z
19	CP0	CP0 f	将 f 寄存器与 $0x0000$ 作比较	1	1	C,DC,N,OV,Z
		CP0 Ws	将 Ws 寄存器与 $0x0000$ 作比较	1	1	C,DC,N,OV,Z
20	CPB	CPB f	将 f 寄存器与 WREG 作比较 (通过带借位减法实现)	1	1	C,DC,N,OV,Z
		CPB $Wb, \#lit5$	将 Wb 与 $lit5$ 作比较 (通过带借位减法实现)	1	1	C,DC,N,OV,Z
		CPB Wb, Ws	将 Wb 与 Ws 作比较 (通过带借位减法实现) ($Wb - Ws - C$)	1	1	C,DC,N,OV,Z
21	CPSEQ	CPSEQ Wb, Wn	将 Wb 与 Wn 作比较, 如果相等则跳过	1	1 (2 或 3)	无
22	CPSGT	CPSGT Wb, Wn	将 Wb 与 Wn 作比较, 如果大于则跳过	1	1 (2 或 3)	无
23	CPSLT	CPSLT Wb, Wn	将 Wb 与 Wn 作比较, 如果小于则跳过	1	1 (2 或 3)	无
24	CPSNE	CPSNE Wb, Wn	将 Wb 与 Wn 作比较, 如果不相等则跳过	1	1 (2 或 3)	无
25	DAW	DAW Wn	$Wn =$ 对 Wn 进行十进制调整	1	1	C
26	DEC	DEC f	$f = f - 1$	1	1	C,DC,N,OV,Z
		DEC $f, WREG$	WREG = $f - 1$	1	1	C,DC,N,OV,Z
		DEC Ws, Wd	$Wd = Ws - 1$	1	1	C,DC,N,OV,Z
27	DEC2	DEC2 f	$f = f - 2$	1	1	C,DC,N,OV,Z
		DEC2 $f, WREG$	WREG = $f - 2$	1	1	C,DC,N,OV,Z
		DEC2 Ws, Wd	$Wd = Ws - 2$	1	1	C,DC,N,OV,Z
28	DISI	DISI $\#lit14$	在 k 个指令周期内禁止中断	1	1	无

dsPIC30F3014/4013

表 21-2: 指令集汇总 (续)

基本指令 #	汇编助记符	汇编语法	说明	字数	周期数	受影响的标志位
29	DIV	DIV.S Wm, Wn	有符号 16/16 位整数除法	1	18	N,Z,C,OV
		DIV.SD Wm, Wn	有符号 32/16 位整数除法	1	18	N,Z,C,OV
		DIV.U Wm, Wn	无符号 16/16 位整数除法	1	18	N,Z,C,OV
		DIV.UD Wm, Wn	无符号 32/16 位整数除法	1	18	N,Z,C,OV
30	DIVF	DIVF Wm, Wn	有符号 16/16 位小数除法	1	18	N,Z,C,OV
31	DO	DO #lit14, Expr	循环执行自下一条指令起到 PC+Expr 之间的代码 lit14+1 次	2	2	无
		DO Wn, Expr	循环执行自下一条指令起到 PC+Expr 之间的代码 (Wn)+1 次	2	2	无
32	ED	ED Wm*Wm, Acc, Wx, Wy, Wxd	欧几里德距离 (无累加)	1	1	OA,OB,OAB,SA,SB,SAB
33	EDAC	EDAC Wm*Wm, Acc, Wx, Wy, Wxd	欧几里德距离	1	1	OA,OB,OAB,SA,SB,SAB
34	EXCH	EXCH Wns, Wnd	将 Wns 和 Wnd 交换	1	1	无
35	FBCL	FBCL Ws, Wnd	从左边第一位 (MSb) 开始查找位变化	1	1	C
36	FF1L	FF1L Ws, Wnd	从左边第一位 (MSb) 开始查找第一个 1	1	1	C
37	FF1R	FF1R Ws, Wnd	从右边第一位 (LSb) 开始查找第一个 1	1	1	C
38	GOTO	GOTO Expr	跳转到指定地址	2	2	无
		GOTO Wn	跳转到间接地址	1	2	无
39	INC	INC f	f = f + 1	1	1	C,DC,N,OV,Z
		INC f, WREG	WREG = f + 1	1	1	C,DC,N,OV,Z
		INC Ws, Wd	Wd = Ws + 1	1	1	C,DC,N,OV,Z
40	INC2	INC2 f	f = f + 2	1	1	C,DC,N,OV,Z
		INC2 f, WREG	WREG = f + 2	1	1	C,DC,N,OV,Z
		INC2 Ws, Wd	Wd = Ws + 2	1	1	C,DC,N,OV,Z
41	IOR	IOR f	f = f .IOR.WREG	1	1	N,Z
		IOR f, WREG	WREG = f .IOR.WREG	1	1	N,Z
		IOR #lit10, Wn	Wd = lit10 .IOR.Wd	1	1	N,Z
		IOR Wb, Ws, Wd	Wd = Wb .IOR.Ws	1	1	N,Z
		IOR Wb, #lit5, Wd	Wd = Wb .IOR. lit5	1	1	N,Z
42	LAC	LAC Wso, #Slit4, Acc	加载累加器	1	1	OA,OB,OAB,SA,SB,SAB
43	LNK	LNK #lit14	分配堆栈帧	1	1	无
44	LSR	LSR f	f = 逻辑右移 f	1	1	C,N,OV,Z
		LSR f, WREG	WREG = 逻辑右移 f	1	1	C,N,OV,Z
		LSR Ws, Wd	Wd = 逻辑右移 Ws	1	1	C,N,OV,Z
		LSR Wb, Wns, Wnd	Wnd = 将 Wb 逻辑右移 Wns 位	1	1	N,Z
		LSR Wb, #lit5, Wnd	Wnd = 将 Wb 逻辑右移 lit5 位	1	1	N,Z
45	MAC	MAC Wm*Wn, Acc, Wx, Wxd, Wy, W yd, AWB	相乘并累加	1	1	OA,OB,OAB,SA,SB,SAB
		MAC Wm*Wm, Acc, Wx, Wxd, Wy, W yd	平方并累加	1	1	OA,OB,OAB,SA,SB,SAB
46	MOV	MOV f, Wn	将 f 寄存器的内容传送给 Wn	1	1	无
		MOV f	将源寄存器的内容送入目标寄存器	1	1	N,Z
		MOV f, WREG	将 f 寄存器的内容传送给 WREG	1	1	N,Z
		MOV #lit16, Wn	将 16 位立即数传送给 Wn	1	1	无
		MOV.b #lit8, Wn	将 8 位立即数传送给 Wn	1	1	无
		MOV Wn, f	将 Wn 的内容传送给 f 寄存器	1	1	无
		MOV Wso, Wdo	将 Ws 的内容送至 Wd	1	1	无
		MOV WREG, f	将 WREG 传送给 f	1	1	N,Z
		MOV.D Wns, Wd	从 W(ns):W(ns+1) 传送双字给 Wd	1	2	无
		MOV.D Ws, Wnd	从 Ws 传送双字给 W(nd+1):W(nd)	1	2	无
47	MOVSAC	MOVSAC Acc, Wx, Wxd, Wy, W yd, AWB	预取并保存累加器	1	1	无

表 21-2: 指令集汇总 (续)

基本指令 #	汇编助记符	汇编语法	说明	字数	周期数	受影响的标志位
48	MPY	MPY $Wm * Wn, Acc, Wx, Wxd, Wy, Wyd$	用 Wm 乘 Wn , 结果保存到累加器	1	1	OA,OB,OAB,SA,SB,SAB
		MPY $Wm * Wm, Acc, Wx, Wxd, Wy, Wyd$	Wm 平方, 结果保存到累加器	1	1	OA,OB,OAB,SA,SB,SAB
49	MPY.N	MPY.N $Wm * Wn, Acc, Wx, Wxd, Wy, Wyd$	- (用 Wm 乘 Wn) 结果保存到累加器	1	1	无
50	MSC	MSC $Wm * Wm, Acc, Wx, Wxd, Wy, W yd, AWB$	相乘, 并将结果从累加器中减去	1	1	OA,OB,OAB,SA,SB,SAB
51	MUL	MUL.SS Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{有符号}(Wb) * \text{有符号}(Ws)$	1	1	无
		MUL.SU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{有符号}(Wb) * \text{无符号}(Ws)$	1	1	无
		MUL.US Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{无符号}(Wb) * \text{有符号}(Ws)$	1	1	无
		MUL.UU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} = \text{无符号}(Wb) * \text{无符号}(Ws)$	1	1	无
		MUL.SU $Wb, \#lit5, Wnd$	$\{Wnd+1, Wnd\} = \text{有符号}(Wb) * \text{无符号}(lit5)$	1	1	无
		MUL.UU $Wb, \#lit5, Wnd$	$\{Wnd+1, Wnd\} = \text{无符号}(Wb) * \text{无符号}(lit5)$	1	1	无
		MUL f	$W3:W2 = f * WREG$	1	1	无
52	NEG	NEG Acc	清零累加器	1	1	OA,OB,OAB,SA,SB,SAB
		NEG f	$f = \bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG $f, WREG$	$WREG = \bar{f} + 1$	1	1	C,DC,N,OV,Z
		NEG Ws, Wd	$Wd = \bar{Ws} + 1$	1	1	C,DC,N,OV,Z
53	NOP	NOP	空操作	1	1	无
		NOPR	空操作	1	1	无
54	POP	POP f	将 f 寄存器的内容从栈顶 (TOS) 弹出	1	1	无
		POP Wdo	将栈顶 (TOS) 的内容弹出到 Wdo 中	1	1	无
		POP.D Wnd	将栈顶 (TOS) 的内容弹出到 $W(nd):W(nd+1)$ 中	1	2	无
		POP.S	将影子寄存器中的内容弹出	1	1	全部
55	PUSH	PUSH f	将 f 寄存器的内容压入栈顶 (TOS)	1	1	无
		PUSH Wso	将 Wso 的内容压入栈顶 (TOS)	1	1	无
		PUSH.D Wns	将 $W(ns):W(ns+1)$ 压入栈顶 (TOS)	1	2	无
		PUSH.S	压入影子寄存器	1	1	无
56	PWRSVAV	PWRSVAV $\#lit1$	进入休眠或空闲模式	1	1	WDTO, 休眠
57	RCALL	RCALL $Expr$	相对调用	1	2	无
		RCALL Wn	计算调用	1	2	无
58	REPEAT	REPEAT $\#lit14$	将下一条指令重复 $lit14 + 1$ 次	1	1	无
		REPEAT Wn	将下一条指令重复 $(Wn) + 1$ 次	1	1	无
59	RESET	RESET	用软件使器件复位	1	1	无
60	RETFIE	RETFIE	从中断返回	1	3 (2)	无
61	RETLW	RETLW $\#lit10, Wn$	返回并将立即数存入 Wn	1	3 (2)	无
62	RETURN	RETURN	从子程序返回	1	3 (2)	无
63	RLC	RLC f	$f =$ 对 f 执行带进位循环左移	1	1	C,N,Z
		RLC $f, WREG$	$WREG =$ 对 f 执行带进位循环左移	1	1	C,N,Z
		RLC Ws, Wd	$Wd =$ 对 Ws 执行带进位循环左移	1	1	C,N,Z
64	RLNC	RLNC f	$f =$ 将 f 循环左移 (不带进位)	1	1	N,Z
		RLNC $f, WREG$	$WREG =$ 将 f 循环左移 (不带进位)	1	1	N,Z
		RLNC Ws, Wd	$Wd =$ 将 Ws 循环左移 (不带进位)	1	1	N,Z
65	RRC	RRC f	$f =$ 对 f 执行带进位循环右移	1	1	C,N,Z
		RRC $f, WREG$	$WREG =$ 对 f 执行带进位循环右移	1	1	C,N,Z
		RRC Ws, Wd	$Wd =$ 对 Ws 执行带进位循环右移	1	1	C,N,Z

dsPIC30F3014/4013

表 21-2: 指令集汇总 (续)

基本指令 #	汇编助记符	汇编语法	说明	字数	周期数	受影响的标志位
66	RRNC	RRNC <i>f</i>	<i>f</i> = 将 <i>f</i> 循环右移 (不带进位)	1	1	N,Z
		RRNC <i>f</i> , WREG	WREG = 将 <i>f</i> 循环右移 (不带进位)	1	1	N,Z
		RRNC <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = 将 <i>Ws</i> 循环右移 (不带进位)	1	1	N,Z
67	SAC	SAC <i>Acc</i> , # <i>Slit4</i> , <i>Wdo</i>	保存累加器	1	1	无
		SAC.R <i>Acc</i> , # <i>Slit4</i> , <i>Wdo</i>	保存舍入后的累加器	1	1	无
68	SE	SE <i>Ws</i> , <i>Wnd</i>	<i>Wnd</i> = 对 <i>Ws</i> 进行符号扩展	1	1	C,N,Z
69	SETM	SETM <i>f</i>	<i>f</i> = 0xFFFF	1	1	无
		SETM WREG	WREG = 0xFFFF	1	1	无
		SETM <i>Ws</i>	<i>Ws</i> = 0xFFFF	1	1	无
70	SFTAC	SFTAC <i>Acc</i> , <i>Wn</i>	算术移位累加器 <i>Wn</i> 次	1	1	OA,OB,OAB,SA,SB,SAB
		SFTAC <i>Acc</i> , # <i>Slit6</i>	算术移位累加器 <i>Slit6</i> 次	1	1	OA,OB,OAB,SA,SB,SAB
71	SL	SL <i>f</i>	<i>f</i> = 左移 <i>f</i>	1	1	C,N,OV,Z
		SL <i>f</i> , WREG	WREG = 左移 <i>f</i>	1	1	C,N,OV,Z
		SL <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = 左移 <i>Ws</i>	1	1	C,N,OV,Z
		SL <i>Wb</i> , <i>Wns</i> , <i>Wnd</i>	<i>Wnd</i> = 将 <i>Wb</i> 左移 <i>Wns</i> 位	1	1	N,Z
		SL <i>Wb</i> , # <i>lit5</i> , <i>Wnd</i>	<i>Wnd</i> = 将 <i>Wb</i> 左移 <i>lit5</i> 位	1	1	N,Z
72	SUB	SUB <i>Acc</i>	累加器相减	1	1	OA,OB,OAB,SA,SB,SAB
		SUB <i>f</i>	<i>f</i> = <i>f</i> - WREG	1	1	C,DC,N,OV,Z
		SUB <i>f</i> , WREG	WREG = <i>f</i> - WREG	1	1	C,DC,N,OV,Z
		SUB # <i>lit10</i> , <i>Wn</i>	<i>Wn</i> = <i>Wn</i> - <i>lit10</i>	1	1	C,DC,N,OV,Z
		SUB <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> - <i>Ws</i>	1	1	C,DC,N,OV,Z
		SUB <i>Wb</i> , # <i>lit5</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> - <i>lit5</i>	1	1	C,DC,N,OV,Z
73	SUBB	SUBB <i>f</i>	<i>f</i> = <i>f</i> - WREG - (\bar{C})	1	1	C,DC,N,OV,Z
		SUBB <i>f</i> , WREG	WREG = <i>f</i> - WREG - (\bar{C})	1	1	C,DC,N,OV,Z
		SUBB # <i>lit10</i> , <i>Wn</i>	<i>Wn</i> = <i>Wn</i> - <i>lit10</i> - (\bar{C})	1	1	C,DC,N,OV,Z
		SUBB <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> - <i>Ws</i> - (\bar{C})	1	1	C,DC,N,OV,Z
		SUBB <i>Wb</i> , # <i>lit5</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> - <i>lit5</i> - (\bar{C})	1	1	C,DC,N,OV,Z
74	SUBR	SUBR <i>f</i>	<i>f</i> = WREG - <i>f</i>	1	1	C,DC,N,OV,Z
		SUBR <i>f</i> , WREG	WREG = WREG - <i>f</i>	1	1	C,DC,N,OV,Z
		SUBR <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Ws</i> - <i>Wb</i>	1	1	C,DC,N,OV,Z
		SUBR <i>Wb</i> , # <i>lit5</i> , <i>Wd</i>	<i>Wd</i> = <i>lit5</i> - <i>Wb</i>	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR <i>f</i>	<i>f</i> = WREG - <i>f</i> - (\bar{C})	1	1	C,DC,N,OV,Z
		SUBBR <i>f</i> , WREG	WREG = WREG - <i>f</i> - (\bar{C})	1	1	C,DC,N,OV,Z
		SUBBR <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Ws</i> - <i>Wb</i> - (\bar{C})	1	1	C,DC,N,OV,Z
		SUBBR <i>Wb</i> , # <i>lit5</i> , <i>Wd</i>	<i>Wd</i> = <i>lit5</i> - <i>Wb</i> - (\bar{C})	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b <i>Wn</i>	<i>Wn</i> = 将 <i>Wn</i> 的两个半字节相交换	1	1	无
		SWAP <i>Wn</i>	<i>Wn</i> = 将 <i>Wn</i> 的两个字节相交换	1	1	无
77	TBLRDH	TBLRDH <i>Ws</i> , <i>Wd</i>	将程序存储单元的 bit<23:16> 读出到 <i>Wd</i> <7:0> 中	1	2	无
78	TBLRDL	TBLRDL <i>Ws</i> , <i>Wd</i>	将程序存储单元的 bit<15:0> 读出到 <i>Wd</i> 中	1	2	无
79	TBLWTH	TBLWTH <i>Ws</i> , <i>Wd</i>	将 <i>Ws</i> <7:0> 写入程序存储单元的 bit<23:16>	1	2	无
80	TBLWTL	TBLWTL <i>Ws</i> , <i>Wd</i>	将 <i>Ws</i> 写入程序存储单元的 bit<15:0>	1	2	无
81	ULNK	ULNK	释放堆栈帧	1	1	无
82	XOR	XOR <i>f</i>	<i>f</i> = <i>f</i> .XOR.WREG	1	1	N,Z
		XOR <i>f</i> , WREG	WREG = <i>f</i> .XOR.WREG	1	1	N,Z
		XOR # <i>lit10</i> , <i>Wn</i>	<i>Wd</i> = <i>lit10</i> .XOR. <i>Wd</i>	1	1	N,Z
		XOR <i>Wb</i> , <i>Ws</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> .XOR. <i>Ws</i>	1	1	N,Z
		XOR <i>Wb</i> , # <i>lit5</i> , <i>Wd</i>	<i>Wd</i> = <i>Wb</i> .XOR. <i>lit5</i>	1	1	N,Z
83	ZE	ZE <i>Ws</i> , <i>Wnd</i>	<i>Wnd</i> = 对 <i>Ws</i> 进行零扩展	1	1	C,Z,N

22.0 开发支持

一系列软件及硬件开发工具对 PIC® 单片机和 dsPIC® 数字信号控制器提供支持：

- 集成开发环境
 - MPLAB® IDE 软件
- 编译器 / 汇编器 / 链接器
 - 适用于各种器件系列的 MPLAB C 编译器
 - 适用于各种器件系列的 HI-TECH C 编译器
 - MPASM™ 汇编器
 - MPLINK™ 目标链接器 / MPLIB™ 目标库管理器
 - 适用于各种器件系列的 MPLAB 汇编器 / 链接器 / 库管理器
- 模拟器
 - MPLAB SIM 软件模拟器
- 仿真器
 - MPLAB REAL ICE™ 在线仿真器
- 在线调试器
 - MPLAB ICD 3
 - PICkit™ 3 Debug Express
- 器件编程器
 - PICkit™ 2 编程器
 - MPLAB PM3 器件编程器
- 低成本演示 / 开发板、评估工具包及入门工具包

22.1 MPLAB 集成开发环境软件

MPLAB IDE 软件为 8/16/32 位单片机市场提供了前所未有的易于使用的软件开发平台。MPLAB IDE 是基于 Windows® 操作系统的应用软件，包括：

- 一个包含所有调试工具的图形界面
 - 模拟器
 - 编程器（单独销售）
 - 在线仿真器（单独销售）
 - 在线调试器（单独销售）
- 具有彩色上下文代码显示的全功能编辑器
- 多项目管理器
- 内容可直接编辑的可定制式数据窗口
- 高级源代码调试
- 鼠标停留在变量上进行查看的功能
- 将变量从源代码窗口拖放到 Watch（观察）窗口
- 丰富的在线帮助
- 集成了可选的第三方工具，如 IAR C 编译器

MPLAB IDE 可以让您：

- 编辑源文件（C 语言或汇编语言）
- 点击一次即可完成编译或汇编，并将代码下载到仿真器和模拟器工具中（自动更新所有项目信息）
- 可使用如下各项进行调试：
 - 源文件（C 语言或汇编语言）
 - 混合 C 语言和汇编语言
 - 机器码

MPLAB IDE 在单个开发范例中支持使用多种调试工具，包括从成本效益高的模拟器到低成本的在线调试器，再到全功能的仿真器。这样缩短了用户升级到更加灵活而功能强大的工具时的学习时间。

22.2 适用于各种器件系列的 MPLAB C 编译器

MPLAB C 编译器代码开发系统是完整的 ANSI C 编译器，适用于 Microchip 的 PIC18、PIC24 和 PIC32 系列单片机及 dsPIC30 和 dsPIC33 系列数字信号控制器。这些编译器提供强大的集成功能和出众的代码优化能力，且使用方便。

为便于源代码调试，编译器提供针对 MPLAB IDE 调试器优化的符号信息。

22.3 适用于各种器件系列的 HI-TECH C 编译器

HI-TECH C 编译器代码开发系统是完整的 ANSI C 编译器，适用于 Microchip 的 PIC 系列单片机及 dsPIC 系列数字信号控制器。这些编译器提供强大的集成功能和全知代码生成能力，且使用方便。

为便于源代码调试，编译器提供针对 MPLAB IDE 调试器优化的符号信息。

编译器包括一个宏汇编器、链接器、预处理程序和单步驱动程序，可以在多种平台上运行。

22.4 MPASM 汇编器

MPASM 汇编器是全功能通用宏汇编器，适用于 PIC10/12/16/18 MCU。

MPASM 汇编器可生成用于 MPLINK 目标链接器的可重定位目标文件、Intel® 标准 HEX 文件、详细描述存储器使用状况和符号参考的 MAP 文件、包含源代码行及生成机器码的绝对 LST 文件以及用于调试的 COFF 文件。

MPASM 汇编器具有如下特性：

- 集成在 MPLAB IDE 项目中
- 用户定义的宏可简化汇编代码
- 对多用途源文件进行条件汇编
- 允许完全控制汇编过程的指令

22.5 MPLINK 目标链接器 / MPLIB 目标库管理器

MPLINK 目标链接器包含了由 MPASM 汇编器、MPLAB C18 C 编译器产生的可重定位目标。通过使用链接器脚本中的指令，它还可链接预编译库中的可重定位目标。

MPLIB 目标库管理器管理预编译代码库文件的创建和修改。当从源文件调用库中的一段子程序时，只有包含此子程序的模块被链接到应用程序。这样可使大型库在许多不同应用中被高效地利用。

目标链接器 / 库管理器具有如下特性：

- 高效地连接单个的库而不是许多小文件
- 通过将相关的模块组合在一起增强代码的可维护性
- 只要列出、替换、删除和抽取模块，便可灵活地创建库

22.6 适用于各种器件系列的 MPLAB 汇编器、链接器和库管理器

MPLAB 汇编器为 PIC24、PIC32 和 dsPIC 器件从符号汇编语言生成可重定位机器码。MPLAB C 编译器使用该汇编器生成目标文件。汇编器产生可重定位目标文件之后，可将这些目标文件存档，或其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特性：

- 支持整个器件指令集
- 支持定点数据和浮点数据
- 命令行界面
- 丰富的指令集
- 灵活的宏语言
- MPLAB IDE 兼容性

22.7 MPLAB SIM 软件模拟器

MPLAB SIM 软件模拟器通过在指令级对 PIC MCU 和 dsPIC[®] DSC 进行模拟，可在 PC 主机环境下进行代码开发。对于任何给定的指令，都可以对数据区进行检查或修改，并通过一个全面的激励控制器来施加激励。可以将各寄存器记录在文件中，以便进行进一步的运行时分析。跟踪缓冲区和逻辑分析器的显示使软件模拟器还能记录和跟踪程序的执行、I/O 的动作、大部分的外设及内部寄存器。

MPLAB SIM 软件模拟器完全支持使用 MPLAB C 编译器以及 MPASM 和 MPLAB 汇编器的符号调试。该软件模拟器可用于在硬件实验室环境外灵活地开发和调试代码，是一款完美且经济的软件开发工具。

22.8 MPLAB REAL ICE 在线仿真器系统

MPLAB REAL ICE 在线仿真器系统是 Microchip 针对其闪存 DSC 和 MCU 器件而推出的新一代高速仿真器。结合 MPLAB 集成开发环境 (IDE) 所具有的易于使用且功能强大的图形用户界面，该仿真器可对 PIC[®] 闪存 MCU 和 dsPIC[®] 闪存 DSC 进行调试和编程。IDE 是随每个工具包一起提供的。

该仿真器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与在线调试器系统兼容的连接器和 (RJ11) 或新型抗噪声、高速低压差分信号 (LVDS) 互连电缆 (CAT5) 与目标板相连。

可通过 MPLAB IDE 下载将来版本的固件，对该仿真器进行现场升级。在即将推出的 MPLAB IDE 版本中，会支持许多新器件，还将增加一些新特性。在同类仿真器中，MPLAB REAL ICE 的优势十分明显：低成本、全速仿真、运行时变量查看、跟踪分析、复杂断点、耐用的探针接口及较长（长达 3 米）的互连电缆。

22.9 MPLAB ICD 3 在线调试器系统

MPLAB ICD 3 在线调试器系统是 Microchip 成本效益最高的高速硬件调试器 / 编程器，适用于 Microchip 闪存数字信号控制器 (DSC) 和单片机 (MCU) 器件。结合 MPLAB 集成开发环境 (IDE) 所具有的功能强大但易于使用的图形用户界面，该调试器可对 PIC[®] 闪存单片机和 dsPIC[®] DSC 进行调试和编程。

MPLAB ICD 3 在线调试器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与 MPLAB ICD 2 或 MPLAB REAL ICE 系统兼容的连接器和 (RJ-11) 与目标板相连。MPLAB ICD 3 支持所有 MPLAB ICD 2 转接器。

22.10 PICkit 3 在线调试器 / 编程器及 PICkit 3 Debug Express

结合 MPLAB 集成开发环境 (IDE) 所具有的功能强大的图形用户界面，MPLAB PICkit 3 可对 PIC[®] 闪存单片机和 dsPIC[®] 数字信号控制器进行调试和编程，且价位较低。MPLAB PICkit 3 通过全速 USB 接口与设计工程师的 PC 相连，并利用 Microchip 调试 (RJ-11) 连接器 (与 MPLAB ICD 3 和 MPLAB REAL ICE 兼容) 与目标板相连。连接器使用两个器件 I/O 引脚和复位线来实现在线调试和在线串行编程。

PICkit 3 Debug Express 包括 PICkit 3、演示板和单片机、连接电缆和光盘 (内含用户指南、课程、教程、编译器 and MPLAB IDE 软件)。

22.11 PICkit 2 开发编程器 / 调试器及 PICkit 2 Debug Express

PICkit™ 2 开发编程器 / 调试器是一款低成本开发工具，具有易于使用的界面，适用于对 Microchip 的闪存系列单片机进行编程和调试。这一全功能的 Windows® 编程界面支持低档（PIC10F、PIC12F5xx 和 PIC16F5xx）、中档（PIC12F6xx 和 PIC16F）、PIC18F、PIC24、dsPIC30、dsPIC33 和 PIC32 系列的 8 位、16 位及 32 位单片机，以及许多 Microchip 串行 EEPROM 产品。结合 Microchip 功能强大的 MPLAB 集成开发环境（IDE），PICkit 2 可对大多数 PIC® 单片机进行在线调试。即使 PIC 单片机已嵌入应用，在线调试功能仍可以运行、暂停和单步执行程序。在断点处暂停时，可以检查和修改文件寄存器。

PICkit 2 Debug Express 包括 PICkit 2、演示板和单片机、连接电缆和光盘（内含用户指南、课程、教程、编译器及 MPLAB IDE 软件）。

22.12 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款符合 CE 规范的通用器件编程器，在 VDDMIN 和 VDDMAX 点对其可编程电压进行校验以确保可靠性最高。它有一个用来显示菜单和错误消息的大 LCD 显示器（128 x 64），以及一个支持各种封装类型的可拆卸模块化插槽装置。编程器标准配置中带有一根 ICSP™ 电缆。在单机模式下，MPLAB PM3 器件编程器不必与 PC 相连即可对 PIC 器件进行读取、校验和编程。在该模式下它还可设置代码保护。MPLAB PM3 通过 RS-232 或 USB 电缆连接到 PC 主机上。MPLAB PM3 具备高速通信能力以及优化算法，可对具有大存储器的器件进行快速编程。它还包含了 MMC 卡，用于文件存储及数据应用。

22.13 演示 / 开发板、评估工具包及入门工具包

有许多演示、开发和评估板可用于各种 PIC MCU 和 dsPIC DSC，实现对全功能系统的快速应用开发。大多数的演示、开发和评估板都有实验布线区，供用户添加定制电路；还有应用固件和源代码，用于检查和修改。

这些板支持多种功能部件，包括 LED、温度传感器、开关、扬声器、RS-232 接口、LCD 显示器、电位计和附加 EEPROM 存储器。

演示和开发板可用于教学环境，在实验布线区设计定制电路，从而掌握各种单片机应用。

除了 PICDEM™ 和 dsPICDEM™ 演示 / 开发板系列电路外，Microchip 还有一系列评估工具包和演示软件，适用于模拟滤波器设计、KEELOQ® 数据安全产品 IC、CAN、IrDA®、PowerSmart 电池管理、SEEVAL® 评估系统、Σ-Δ ADC、流速传感器，等等。

同时还提供入门工具包，其中包含体验指定器件功能所需的所有软硬件。通常提供单个应用以及调试功能，都包含在一块电路板上。

有关演示、开发和评估工具包的完整列表，请访问 Microchip 网站（www.microchip.com）。

23.0 电气特性

本节概述了 dsPIC30F 器件的电气特性。额外的信息将会在此文档的后续版本中给出。

有关 dsPIC30F 架构和内核的详细信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F 系列器件的绝对最大值如下所列。器件长时间工作在最大值条件下，其稳定性会受到影响。我们建议不要使器件在该规范规定的范围外工作。

绝对最大值 (†)

环境温度.....	-40°C 至 +125°C
储存温度.....	-65°C 至 +150°C
任一引脚（除 VDD 和 $\overline{\text{MCLR}}$ 外）相对于 VSS 的电压（注 1）.....	-0.3V 至 (VDD + 0.3V)
VDD 相对于 VSS 的电压	-0.3 至 +5.5V
$\overline{\text{MCLR}}$ 相对于 VSS 的电压	0 至 +13.25V
VSS 引脚的最大输出电流.....	300 mA
VDD 引脚的最大输入电流（注 2）.....	250 mA
输入钳位电流 I _{IK} (V _I < 0 或 V _I > VDD)	±20 mA
输入钳位电流 I _{OK} (V _O < 0 或 V _O > VDD)	±20 mA
任一 I/O 引脚的最大灌电流	25 mA
任一 I/O 引脚的最大拉电流	25 mA
所有端口的最大灌电流总和.....	200 mA
所有端口的最大拉电流总和（注 2）.....	200 mA

注 1: 如果 $\overline{\text{MCLR}}/\text{VPP}$ 引脚上的尖峰电压低于 VSS，感应电流大于 80mA，可能会引起器件锁死。因此当 $\overline{\text{MCLR}}/\text{VPP}/\text{RE3}$ 引脚驱动为低电平时，应该串联一个 50-100Ω 的电阻，而不是直接把该引脚连接到 VSS。

2: 最大容许电流为器件最大功耗的函数。请参见表 23-4。

†注意：如果器件工作条件超过上述“绝对最大值”，可能会对器件造成永久性损坏。上述值仅为运行条件极大值，我们建议不要使器件在该规范规定的范围以外运行。器件长时间工作在最大值条件下，其稳定性会受到影响。

注： 所有外设的电气特性均已指定。关于特定器件上的可用外设，请参见 dsPIC30F3014/4013 控制器系列表。

dsPIC30F3014/4013

23.1 直流特性

表 23-1: 工作速度（单位 MIPS）与电压的关系

VDD 范围	温度范围	MIPS 最大值	
		dsPIC30FXXX-30I	dsPIC30FXXX-20E
4.5 至 5.5V	-40°C 至 +85°C	30	—
4.5 至 5.5V	-40°C 至 125°C	—	20
3.0 至 3.6V	-40°C 至 +85°C	15	—
3.0 至 3.6V	-40°C 至 125°C	—	10
2.5 至 3.0V	-40°C 至 +85°C	10	—

表 23-2: 工作温度条件

额定指标	符号	最小值	典型值	最大值	单位
dsPIC30F3014-30I dsPIC30F4013-30I					
工作结点温度范围	T _J	-40	—	+125	°C
工作环境温度范围	T _A	-40	—	+85	°C
dsPIC30F3014-20E dsPIC30F4013-20E					
工作结点温度范围	T _J	-40	—	+150	°C
工作环境温度范围	T _A	-40	—	+125	°C
功耗: 内部芯片功耗: $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$	PD	P _{INT} + P _{I/O}			W
I/O 引脚功耗: $P_{I/O} = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$					
最大允许功耗	PD _{MAX}	(T _J - T _A) / θ _{JA}			W

表 23-3: 封装热阻特性

特性	符号	典型值	最大值	单位	注
封装热阻, 40 引脚 DIP (P)	θ _{JA}	—	47	°C/W	1
封装热阻, 44 引脚 TQFP (10x10x1mm)	θ _{JA}	—	39.3	°C/W	1
封装热阻, 44 引脚 QFN	θ _{JA}	—	27.8	°C/W	1

注 1: 通过封装模拟获取结点到环境的热阻参数 Theta-JA (θ_{JA})。

表 23-4: 直流温度和电压规范

直流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
工作电压 ⁽²⁾							
DC10	VDD	供电电压	2.5	—	5.5	V	工业级温度
DC11	VDD	供电电压	3.0	—	5.5	V	扩展级温度
DC12	VDR	RAM 数据保持电压 ⁽³⁾	1.75	—	—	V	
DC16	VPOR	VDD 启动电压 确保能够产生内部上电复位信号	—	—	VSS	V	
DC17	SVDD	VDD 上升速率 确保能够产生内部上电复位信号	0.05	—	—	V/ms	在 0.1 秒内上升幅度 为 0 至 5V 在 60 ms 内上升幅度 为 0 至 3V

注 1: 除非另外说明, “典型值” 栏中的数据均为 5V, 25°C 下的值。这些参数仅供设计参考, 未经测试。

2: 这些参数仅为特征值, 未经生产测试。

3: 这是在不丢失 RAM 数据的前提下, VDD 所能降到的最小电压值。

dsPIC30F3014/4013

表 23-5: 直流特性: 工作电流 (IDD)

直流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)		
			工作温度		
			-40°C ≤ TA ≤ +85°C (工业级)		
			-40°C ≤ TA ≤ +125°C (扩展级)		
参数编号	典型值	最大值	单位	条件	
工作电流 (IDD) (1)					
DC31a	2	4	mA	25°C	3.3V 0.128 MIPS LPRC (512 kHz)
DC31b	2	4	mA	85°C	
DC31c	2	4	mA	125°C	
DC31e	4	6	mA	25°C	
DC31f	4	6	mA	85°C	
DC31g	4	6	mA	125°C	
DC30a	6	11	mA	25°C	3.3V 1.8 MIPS FRC (7.37 MHz)
DC30b	6	11	mA	85°C	
DC30c	7	11	mA	125°C	
DC30e	11	16	mA	25°C	
DC30f	11	16	mA	85°C	
DC30g	11	16	mA	125°C	
DC23a	13	20	mA	25°C	3.3V 4 MIPS
DC23b	13	20	mA	85°C	
DC23c	14	20	mA	125°C	
DC23e	22	31	mA	25°C	
DC23f	22	31	mA	85°C	
DC23g	22	31	mA	125°C	
DC24a	27	39	mA	25°C	3.3V 10 MIPS
DC24b	28	39	mA	85°C	
DC24c	28	39	mA	125°C	
DC24e	46	64	mA	25°C	
DC24f	46	64	mA	85°C	
DC24g	46	64	mA	125°C	
DC27c	86	120	mA	25°C	5V 20 MIPS
DC27e	85	120	mA	85°C	
DC27f	85	120	mA	125°C	
DC29a	123	170	mA	25°C	5V 30 MIPS
DC29b	122	170	mA	85°C	

注 1: 供电电流主要是由工作电压和频率决定的。其他因素, 如 I/O 引脚负载和开关频率、振荡器类型、内部代码执行模式和温度等, 也会影响电流消耗。所有 IDD 测量的测试条件如下: OSC1 由满幅外部方波驱动; 所有 I/O 引脚配置为输入, 并拉至 VDD; MCLR = VDD; 禁止 WDT、FSCM、LVD 和 BOR; CPU、SRAM、程序存储器和数据存储器正常工作; 所有外设模块不工作。

表 23-6: 直流特性: 空闲电流 (I_{IDLE})

直流特性			标准工作条件: 2.5 至 5.5V (除非另外说明)		
			工作温度		
			-40°C ≤ T _A ≤ +85°C (工业级)		
			-40°C ≤ T _A ≤ +125°C (扩展级)		
参数编号	典型值	最大值	单位	条件	
工作电流 (I _{IDLE}) ⁽¹⁾					
DC51a	1.4	3	mA	25°C	0.128 MIPS LPRC (512 kHz)
DC51b	1.5	3	mA	85°C	
DC51c	1.5	3	mA	125°C	
DC51e	3	5	mA	25°C	
DC51f	3	5	mA	85°C	
DC51g	3	5	mA	125°C	
DC50a	4	6	mA	25°C	1.8 MIPS FRC (7.37 MHz)
DC50b	4	6	mA	85°C	
DC50c	4	6	mA	125°C	
DC50e	8	11	mA	25°C	
DC50f	8	11	mA	85°C	
DC50g	8	11	mA	125°C	
DC43a	7	11	mA	25°C	4 MIPS
DC43b	7	11	mA	85°C	
DC43c	8	11	mA	125°C	
DC43e	13	17	mA	25°C	
DC43f	13	17	mA	85°C	
DC43g	13	17	mA	125°C	
DC44a	16	22	mA	25°C	10 MIPS
DC44b	16	22	mA	85°C	
DC44c	17	22	mA	125°C	
DC44e	27	36	mA	25°C	
DC44f	27	36	mA	85°C	
DC44g	28	36	mA	125°C	
DC47d	50	65	mA	25°C	20 MIPS
DC47e	51	65	mA	85°C	
DC47f	52	65	mA	125°C	
DC49a	74	95	mA	25°C	30 MIPS
DC49b	75	95	mA	85°C	

注 1: 基本 I_{IDLE} 电流是在关闭内核和所有模块, 但保持时钟工作的条件下测得的。

dsPIC30F3014/4013

表 23-7: 直流特性: 掉电电流 (IPD)

DC 特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	典型值	最大值	单位	条件			
掉电电流 (IPD) (1)							
DC60a	1	—	μA	25°C	3.3V	基本掉电电流 (2)	
DC60b	3	30	μA	85°C			
DC60c	30	60	μA	125°C			
DC60e	2	—	μA	25°C	5V		
DC60f	6	45	μA	85°C			
DC60g	55	90	μA	125°C			
DC61a	7	11	μA	25°C	3.3V		看门狗定时器电流: ΔI _{WDT} (2)
DC61b	7	11	μA	85°C			
DC61c	7	11	μA	125°C			
DC61e	14	21	μA	25°C	5V		
DC61f	14	21	μA	85°C			
DC61g	14	21	μA	125°C			
DC62a	—	—	μA	25°C	3.3V	带有 32 kHz 晶振的 Timer1: ΔI _{T132} (2)	
DC62b	—	—	μA	85°C			
DC62c	—	—	μA	125°C			
DC62e	—	—	μA	25°C	5V		
DC62f	—	—	μA	85°C			
DC62g	30	45	μA	125°C			
DC63a	30	45	μA	25°C	3.3V		欠压复位: ΔI _{BOR} (2)
DC63b	33	50	μA	85°C			
DC63c	34	51	μA	125°C			
DC63e	34	51	μA	25°C	5V		
DC63f	37	56	μA	85°C			
DC63g	37	56	μA	125°C			
DC66a	18	27	μA	25°C	3.3V	低电压检测: ΔI _{LVD} (2)	
DC66b	20	30	μA	85°C			
DC66c	21	32	μA	125°C			
DC66e	22	33	μA	25°C	5V		
DC66f	23	35	μA	85°C			
DC66g	24	36	μA	125°C			

注 1: 基本 IPD 是在关闭所有外设和时钟的条件下测得的。所有 I/O 配置为输入, 并拉为高电平。关闭 LVD、BOR 和 WDT 等所有模块。

注 2: Δ 电流是当使能模块时额外消耗的电流。该电流应与基本 IPD 电流相加。

表 23-8: 直流特性: I/O 引脚输入规范

直流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件	
DI10	V _{IL}	输入低电压⁽²⁾ I/O 引脚: 带施密特触发缓冲器	V _{SS}	—	0.2 V _{DD}	V		
DI15		MCLR	V _{SS}	—	0.2 V _{DD}	V		
DI16		OSC1 (XT、HS 和 LP 模式)	V _{SS}	—	0.2 V _{DD}	V		
DI17		OSC1 (RC 模式) ⁽³⁾	V _{SS}	—	0.3 V _{DD}	V		
DI18		SDA, SCL	V _{SS}	—	0.3 V _{DD}	V		禁止 SM 总线
DI19		SDA, SCL	V _{SS}	—	0.8	V		使能 SM 总线
DI20	V _{IH}	输入高电压⁽²⁾ I/O 引脚: 带施密特触发缓冲器	0.8 V _{DD}	—	V _{DD}	V		
DI25		MCLR	0.8 V _{DD}	—	V _{DD}	V		
DI26		OSC1 (XT、HS 和 LP 模式)	0.7 V _{DD}	—	V _{DD}	V		
DI27		OSC1 (RC 模式) ⁽³⁾	0.9 V _{DD}	—	V _{DD}	V		
DI28		SDA, SCL	0.7 V _{DD}	—	V _{DD}	V		禁止 SM 总线
DI29		SDA, SCL	2.1	—	V _{DD}	V		使能 SM 总线
DI30	ICNPU	CNxx 上拉电流⁽²⁾	50	250	400	μA	V _{DD} = 5V, V _{PIN} = V _{SS}	
DI50	I _{IL}	输入泄漏电流^(2,4,5) I/O 端口	—	0.01	±1	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态	
DI51		模拟输入引脚	—	0.50	—	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态	
DI55		MCLR	—	0.05	±5	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD}	
DI56		OSC1	—	0.05	±5	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , XT、HS 和 LP 振荡器 模式	

- 注**
- 1: 除非另外说明,“典型值”栏中的数据均为 5V、25°C 下的值。这些参数仅供设计参考,未经测试。
 - 2: 这些参数仅为特征值,未经生产测试。
 - 3: 在 RC 振荡器配置中, OSC1/CLKI 引脚被配置为施密特触发器输入。在 RC 模式下,建议不要使用外部时钟驱动 dsPIC30F 器件。
 - 4: MCLR 引脚上的泄漏电流主要由施加在该引脚上的电平决定。规定电平为正常工作条件下的电平。在不同的输入电压下可能会测得更高的泄漏电流。
 - 5: 负电流定义为自引脚流出的电流。

dsPIC30F3014/4013

表 23-9: 直流特性: I/O 引脚输出规范

直流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ Ta ≤ +85°C (工业级)				
			-40°C ≤ Ta ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
DO10	VOL	输出低电压 ⁽²⁾ I/O 端口	—	—	0.6	V	IOL = 8.5 mA, VDD = 5V
			—	—	0.15	V	IOL = 2.0 mA, VDD = 3V
DO16		OSC2/CLKO (RC 或 EC 振荡器模式)	—	—	0.6	V	IOL = 1.6 mA, VDD = 5V
			—	—	0.72	V	IOL = 2.0 mA, VDD = 3V
DO20	VOH	输出高电压 ⁽²⁾ I/O 端口	VDD - 0.7	—	—	V	IOH = -3.0 mA, VDD = 5V
			VDD - 0.2	—	—	V	IOH = -2.0 mA, VDD = 3V
DO26		OSC2/CLKO (RC 或 EC 振荡器模式)	VDD - 0.7	—	—	V	IOH = -1.3 mA, VDD = 5V
			VDD - 0.1	—	—	V	IOH = -2.0 mA, VDD = 3V
DO50	Cosc2	输出引脚上的容性负载 ⁽²⁾ OSC2/SOSC2 引脚	—	—	15	pF	使用外部时钟驱动 OSC1 时, 处于 XTL、XT、HS 和 LP 模式下。
DO56	Cio	所有 I/O 引脚和 OSC2	—	—	50	pF	RC 或 EC 振荡器模式
DO58	CB	SCL 和 SDA	—	—	400	pF	I ² C™ 模式

注 1: 除非另外说明, “典型值” 栏中的数据均为 5V、25°C 下的值。这些参数仅供设计参考, 未经测试。

注 2: 这些参数仅为特征值, 未经生产测试。

图 23-1: 低电压检测特性

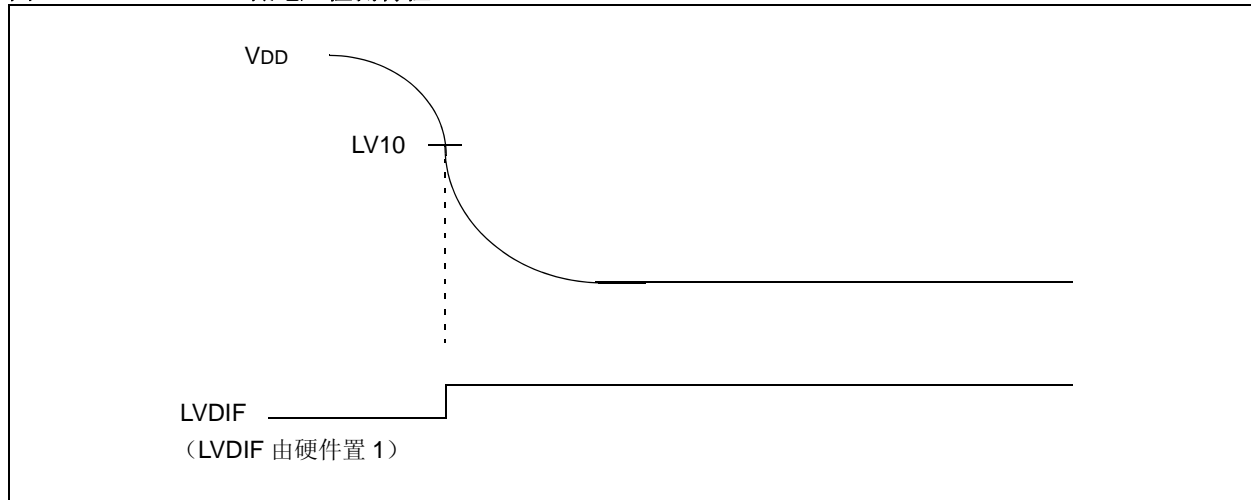
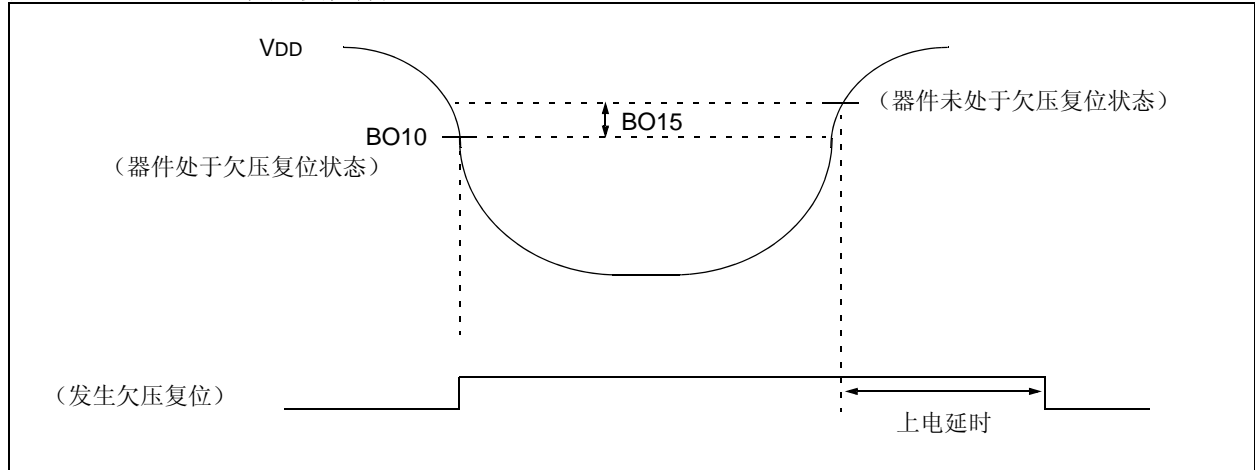


表 23-10: 电气特性: LVDL

直流特性		标准工作条件: 2.5V 至 5.5V (除非另外指明)						
		工作温度						
		-40°C ≤ TA ≤ +85°C (工业级)						
		-40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	符号	特性 (1)	最小值	典型值	最大值	单位	条件	
LV10	VPLVD	VDD 由高电平跳变为低电平时的 LVDL 电压	LVDL = 0000 ⁽²⁾	—	—	—	V	
			LVDL = 0001 ⁽²⁾	—	—	—	V	
			LVDL = 0010 ⁽²⁾	—	—	—	V	
			LVDL = 0011 ⁽²⁾	—	—	—	V	
			LVDL = 0100	2.50	—	2.65	V	
			LVDL = 0101	2.70	—	2.86	V	
			LVDL = 0110	2.80	—	2.97	V	
			LVDL = 0111	3.00	—	3.18	V	
			LVDL = 1000	3.30	—	3.50	V	
			LVDL = 1001	3.50	—	3.71	V	
			LVDL = 1010	3.60	—	3.82	V	
			LVDL = 1011	3.80	—	4.03	V	
			LVDL = 1100	4.00	—	4.24	V	
			LVDL = 1101	4.20	—	4.45	V	
LVDL = 1110	4.50	—	4.77	V				
LV15	VLVDIN	外部 LVD 输入引脚门限电压	LVDL = 1111	—	—	—	V	

注 1: 这些参数仅为特征值, 未经生产测试。
 注 2: 这些值不在可用的工作范围内。

图 23-2: 欠压复位特性



dsPIC30F3014/4013

表 23-11: 电气特性: BOR

直流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件	
BO10	VBOR	VDD 由高电平跳转到低电平时的 BOR 电压 ⁽²⁾	BORV = 11 ⁽³⁾	—	—	—	V	不在工作范围内
			BORV = 10	2.6	—	2.71	V	
			BORV = 01	4.1	—	4.4	V	
			BORV = 00	4.58	—	4.73	V	
BO15	VBHYS		—	5	—	mV		

- 注 1: 除非另外说明, “典型值” 栏中的数据均为 5V、25°C 下的值。这些参数仅供设计参考, 未经测试。
 注 2: 参数仅为特征值, 未经生产测试。
 注 3: 11 值不在可用的工作范围内。

表 23-12: 直流特性: 程序存储器和 EEPROM 存储器

直流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
		数据 EEPROM 存储器⁽²⁾					
D120	ED	字节的耐擦写能力	100K	1M	—	E/W	-40°C ≤ TA ≤ +85°C 使用 EECON 读 / 写 V _{MIN} = 最小工作电压 RTSP
D121	VDRW	用于读写操作的 VDD	V _{MIN}	—	5.5	V	
D122	TDEW	擦 / 写周期	0.8	2	2.6	ms	
D123	TRETD	保存时间	40	100	—	年	
D124	IDEW	编程时的 IDD	—	10	30	mA	
		闪存程序存储器⁽²⁾					
D130	EP	耐擦写能力	10K	100K	—	E/W	-40°C ≤ TA ≤ +85°C V _{MIN} = 最小工作电压 RTSP
D131	VPR	读取时的 VDD	V _{MIN}	—	5.5	V	
D132	VEB	块擦除时的 VDD	4.5	—	5.5	V	
D133	VPEW	擦 / 写操作时的 VDD	3.0	—	5.5	V	
D134	TPEW	擦 / 写周期	0.8	2	2.6	ms	
D135	TRETD	特性保持时间	40	100	—	年	
D137	IPEW	编程时的 IDD	—	10	30	mA	
D138	IEB	编程时的 IDD	—	10	30	mA	

- 注 1: 除非另外说明, “典型值” 栏中的数据均为 5V、25°C 下的值。
 注 2: 这些参数仅为特征值, 未经生产测试。

23.2 交流特性和时序参数

本小节包含的信息定义了 dsPIC30F 系列的交流特性和时序参数。

表 23-13: 温度和电压规范 —— AC

交流特性	标准工作条件: 2.5V 至 5.5V (除非另外说明)
	工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级)
	$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)
	工作电压 V_{DD} 的范围如直流规范所述, 见表 23-1。

图 23-3: 器件时序规范的负载条件

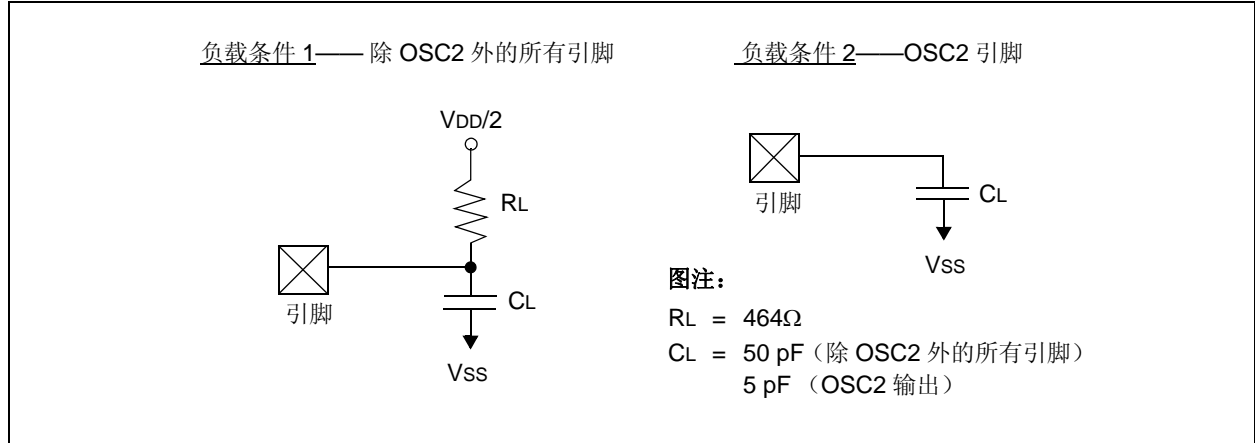
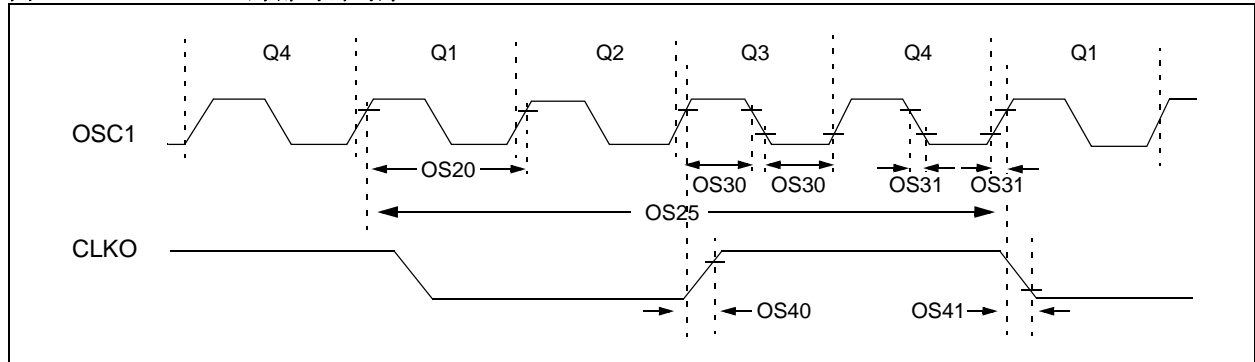


图 23-4: 外部时钟时序



dsPIC30F3014/4013

表 23-14: 外部时钟时序要求

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ Ta ≤ +85°C (工业级)				
			-40°C ≤ Ta ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
OS10	Fosc	外部 CLKI 频率 ⁽²⁾ (外部时钟只能用在 EC 模式)	DC	—	40	MHz	EC
			4	—	10	MHz	带 4xPLL 的 EC
			4	—	10	MHz	带 8xPLL 的 EC
			4	—	7.5 ⁽³⁾	MHz	带 16xPLL 的 EC
	振荡器频率 ⁽²⁾	DC	—	4	MHz	RC	
		0.4	—	4	MHz	XTL	
		4	—	10	MHz	XT	
		4	—	10	MHz	带 4xPLL 的 XT	
		4	—	10	MHz	带 8xPLL 的 XT	
		4	—	7.5 ⁽³⁾	MHz	带 16xPLL 的 XT	
		10	—	25	MHz	HS	
		10	—	20 ⁽⁴⁾	MHz	带 4xPLL 的 HS/2	
		10	—	20 ⁽⁴⁾	MHz	带 8xPLL 的 HS/2	
		10	—	15 ⁽³⁾	MHz	带 16xPLL 的 HS/2	
12 ⁽⁴⁾	—	25	MHz	带 4xPLL 的 HS/3			
12 ⁽⁴⁾	—	25	MHz	带 8xPLL 的 HS/3			
12 ⁽⁴⁾	—	22.5 ⁽³⁾	MHz	带 16xPLL 的 HS/3			
—	32.768	—	—	kHz	LP		
OS20	Tosc	Tosc = 1/Fosc	—	—	—	—	Fosc 值见参数 OS10
OS25	Tcy	指令周期 ^(2,5)	33	—	DC	ns	见表 23-16
OS30	TosL, TosH	外部时钟输入 (OSC1) 高电平或低电平时间 ⁽²⁾	.45 xTosc	—	—	ns	EC
OS31	TosR, TosF	外部时钟输入 (OSC1) 上升或下降时间 ⁽²⁾	—	—	20	ns	EC
OS40	TckR	CLKO 上升时间 ^(2,6)	—	—	—	ns	见参数 DO31
OS41	TckF	CLKO 下降时间 ^(2,6)	—	—	—	ns	见参数 DO32

- 注 1: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数仅供设计参考, 未经测试。
- 2: 这些参数仅为特征值, 未经生产测试。
- 3: 受到 PLL 输出频率范围限制。
- 4: 受到 PLL 输入频率范围限制。
- 5: 指令周期 (Tcy) 等于输入振荡器时钟周期的 2 倍。所有规范值均为器件在标准工作条件下执行代码时对应特定振荡器类型的特性数据。超过规范值可能导致振荡器运行不稳定和 / 或电流消耗超出预期值。所有器件在测试“最小值”时, 都在 OSC1/CLKI 引脚连接了外部时钟。当使用了外部时钟输入时, 所有器件的“最大”周期时间限制为“DC”(无时钟)。
- 6: 数据是在 EC 或 ERC 模式下测得的。CLKO 信号是在 OSC2 引脚上测得的。CLKO 在 Q1-Q2 周期 (1/2 Tcy) 为低电平, Q3-Q4 周期 (1/2 Tcy) 为高电平。

表 23-15: PLL 抖动

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件	
OS61	x4 PLL	—	0.251	0.413	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0 至 3.6V
		—	0.251	0.413	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0 至 3.6V
		—	0.256	0.47	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5 至 5.5V
		—	0.256	0.47	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5 至 5.5V
	x8 PLL	—	0.355	0.584	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0 至 3.6V
		—	0.355	0.584	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0 至 3.6V
		—	0.362	0.664	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5 至 5.5V
		—	0.362	0.664	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5 至 5.5V
	x16 PLL	—	0.67	0.92	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0 至 3.6V
		—	0.632	0.956	%	-40°C ≤ TA ≤ +85°C	VDD = 4.5 至 5.5V
		—	0.632	0.956	%	-40°C ≤ TA ≤ +125°C	VDD = 4.5 至 5.5V

注 1: 这些参数仅为特征值, 未经生产测试。

表 23-16: 内部时钟时序示例

时钟振荡器模式	Fosc (MHz) (1)	Tcy (μs) (2)	MIPS ⁽³⁾ 不带 PLL	MIPS ⁽³⁾ 带有 PLL x4	MIPS ⁽³⁾ 带有 PLL x8	MIPS ⁽³⁾ 带有 PLL x16
EC	0.200	20.0	0.05	—	—	—
	4	1.0	1.0	4.0	8.0	16.0
	10	0.4	2.5	10.0	20.0	—
	25	0.16	6.25	—	—	—
XT	4	1.0	1.0	4.0	8.0	16.0
	10	0.4	2.5	10.0	20.0	—

注 1: 假设: 振荡器后分频器的分频比为 1。

2: 指令执行周期: Tcy = 1 / MIPS。

3: 指令执行频率: MIPS = (Fosc * PLLx)/4 [因为每个指令周期都有 4 个 Q 时钟]。

dsPIC30F3014/4013

表 23-17: 交流特性: 内部 FRC 精度

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)				
		工作温度				
		-40°C ≤ TA ≤ +85°C (工业级)				
		-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	特性	最小值	典型值	最大值	单位	条件
内部 FRC 精度 (FRC 频率 = 7.37 MHz ⁽¹⁾)						
OS63	FRC	—	—	±2.00	%	-40°C ≤ TA ≤ +85°C VDD = 3.0 至 5.5V
		—	—	±5.00	%	-40°C ≤ TA ≤ +125°C VDD = 3.0 至 5.5V

注 1: 在 25°C、5V 时将频率校准为 7.372 MHz ±2%。TUN (OSCCON<3:0>) 位可用于补偿温度漂移。

表 23-18: 交流特性: 内部 LPRC 精度

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)				
		工作温度				
		-40°C ≤ TA ≤ +85°C (工业级)				
		-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	特性	最小值	典型值	最大值	单位	条件
LPRC (频率 = 512 kHz ⁽¹⁾)						
OS65A		-50	—	+50	%	VDD = 5.0V, ±10%
OS65B		-60	—	+60	%	VDD = 3.3V, ±10%
OS65C		-70	—	+70	%	VDD = 2.5V

注 1: LPRC 频率随 VDD 变化而变化。

图 23-5: CLKO 和 I/O 时序特性

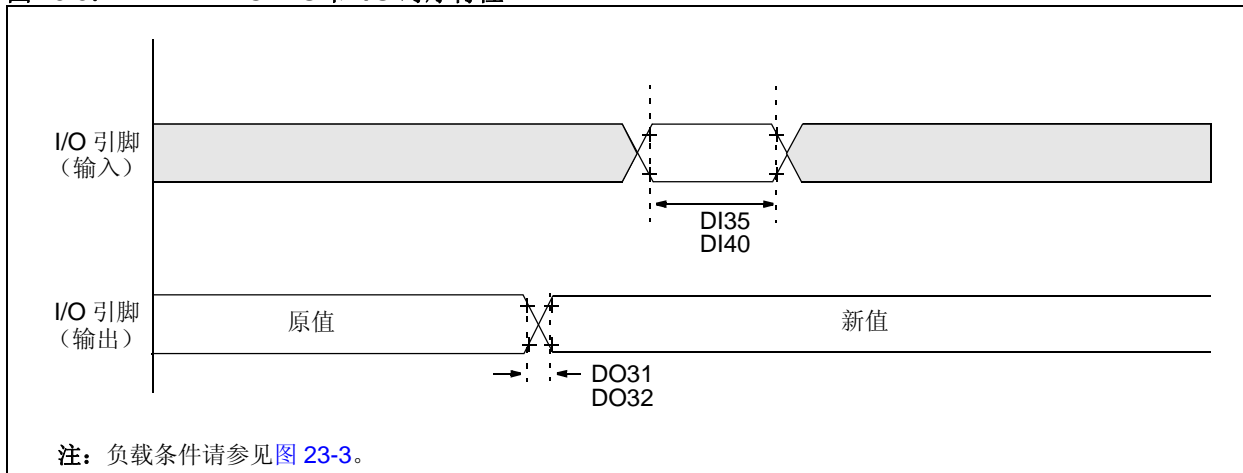


表 23-19: CLKO 和 I/O 时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性 (1,2,3)	最小值	典型值 (4)	最大值	单位	条件
DO31	TioR	端口输出上升时间	—	7	20	ns	
DO32	TioF	端口输出下降时间	—	7	20	ns	
DI35	TINP	INTx 引脚高电平或低电平时间 (输出)	20	—	—	ns	
DI40	TRBP	CNx 引脚高电平或低电平时间 (输入)	2 Tcy	—	—	ns	

- 注 1: 这些参数为异步事件, 与任何内部时钟边沿无关。
 2: 测量是在 RC 和 EC 模式下进行的, 其中 CLKO 输出等于 4 x T_{osc}。
 3: 这些参数仅为特征值, 未经生产测试。
 4: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。

dsPIC30F3014/4013

图 23-6: 复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序特性

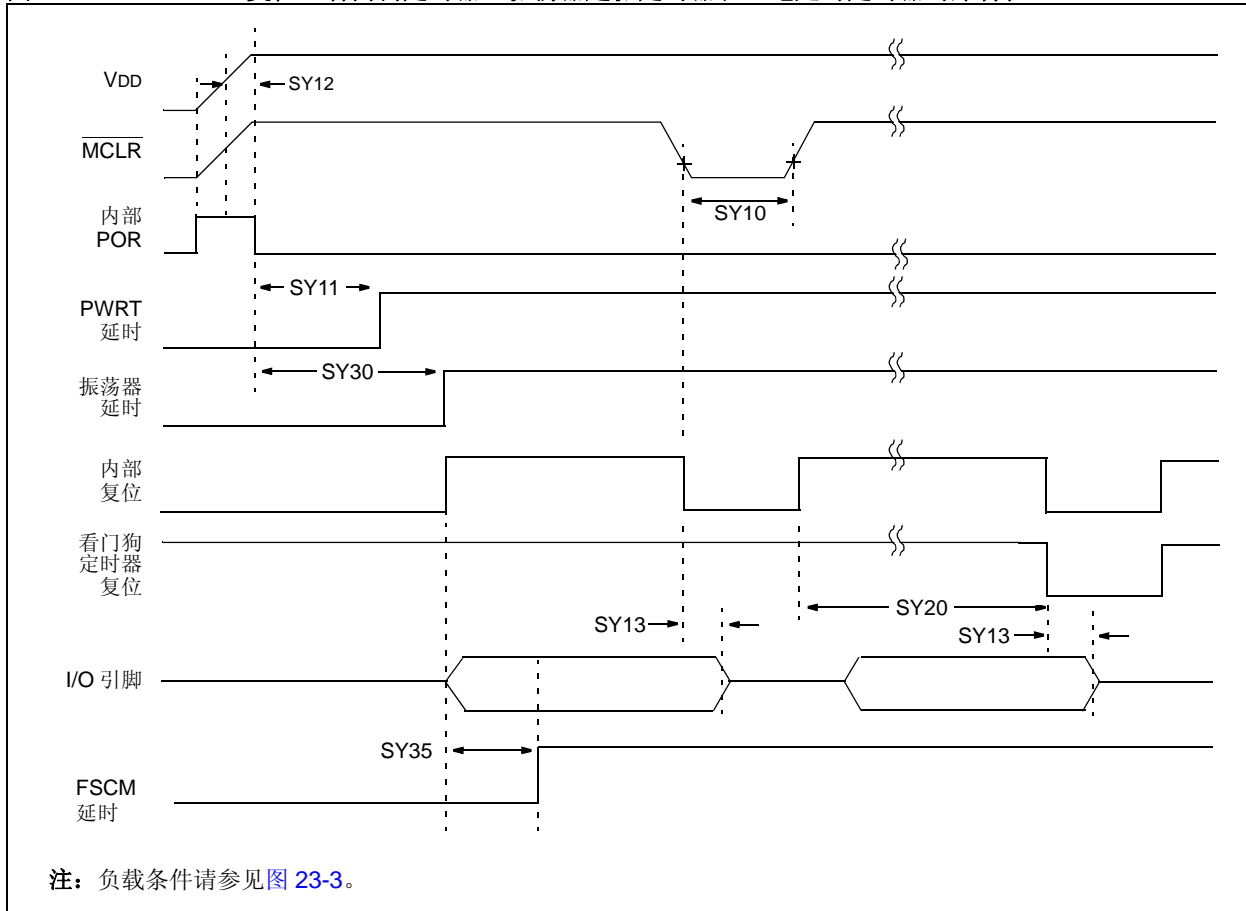


表 23-20: 复位、看门狗定时器、振荡器起振定时器、上电延时定时器和欠压复位时序要求

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SY10	Tmcl	MCLR 脉冲宽度 (低电平)	2	—	—	μs	-40°C 至 +85°C
SY11	TPWRT	上电延时定时器周期	2 10 43	4 16 64	8 32 128	ms	-40°C 至 +85°C VDD = 5V 可由用户编程
SY12	TPOR	上电复位延时	3	10	30	μs	-40°C 至 +85°C
SY13	TIOZ	从 MCLR 低电平或看门狗定时器复位开始 I/O 处于高阻态的时间	—	0.8	1.0	μs	
SY20	TWDT1 TWDT2 TWDT3	看门狗定时器超时周期 (无预分频器)	1.1 1.2 1.3	2.0 2.0 2.0	6.6 5.0 4.0	ms ms ms	VDD = 5V, VDD = 3.3V, ±10% -40°C 至 +85°C
SY25	TBOR	欠压复位脉冲宽度 (3)	100	—	—	μs	VDD ≤ VBOR (D034)
SY30	TOST	振荡器起振定时器周期	—	1024 TOSC	—	—	TOSC = OSC1 周期
SY35	TFSCM	故障保护时钟监视器延时	—	500	900	μs	-40°C 至 +85°C

- 注 1: 这些参数仅为特征值, 未经生产测试。
 2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。
 3: 有关 BOR 的情况, 请参见图 23-2 和表 23-11。

图 23-7: 带隙起振时序特性

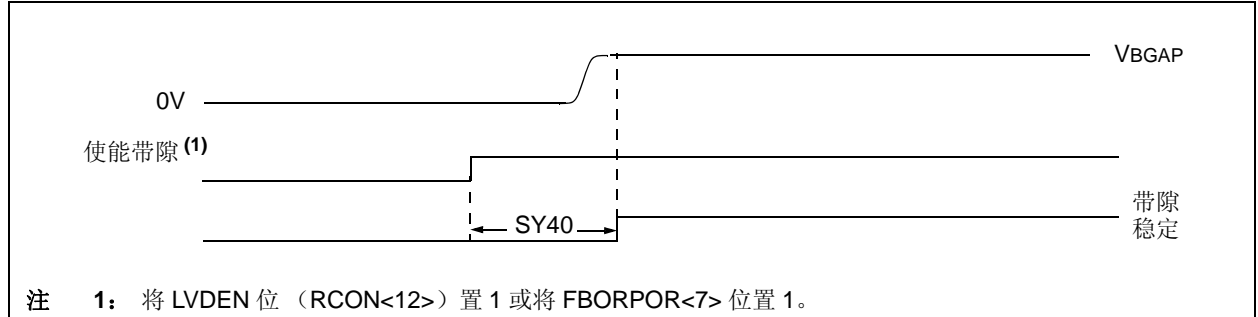


表 23-21: 带隙起振时序要求

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SY40	TBGAP	带隙起振时间	—	40	65	μs	定义为使能带隙的瞬间与带隙参考电压稳定 (RCON<13> 状态位) 时之间的时间。

- 注 1: 这些参数仅为特征值, 未经生产测试。
 2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。

dsPIC30F3014/4013

图 23-8: A、B 和 C 类定时器的外部时钟时序特性

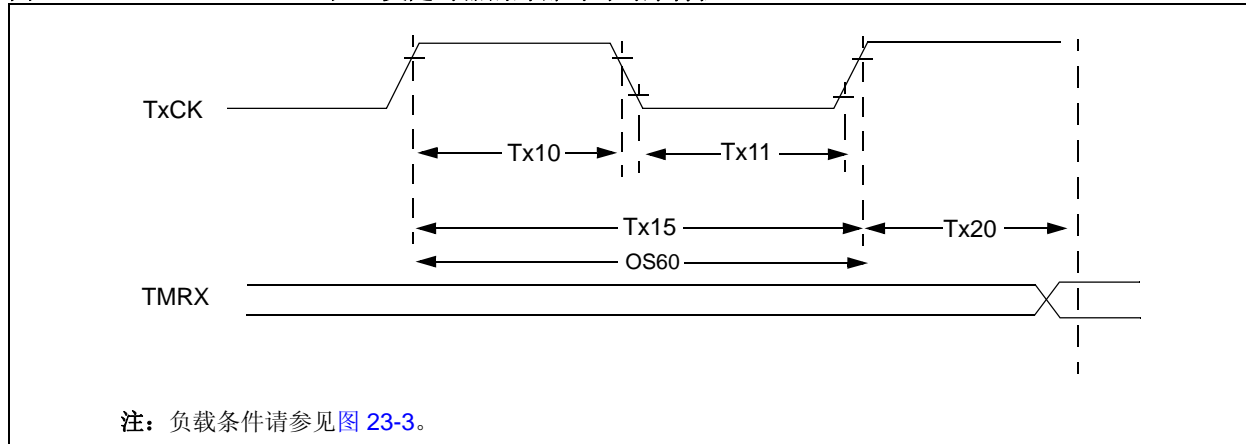


表 23-22: A 类定时器 (TIMER1) 外部时钟时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)						
		工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	符号	特性	最小值	典型值	最大值	单位	条件	
TA10	T _{TxH}	TxCK 高电平时间	同步, 无预分频器	0.5 T _{CY} + 20	—	—	ns	还必须满足参数 TA15
			同步, 带预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA11	T _{TxL}	TxCK 低电平时间	同步, 无预分频器	0.5 T _{CY} + 20	—	—	ns	还必须满足参数 TA15
			同步, 带预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA15	T _{TxP}	TxCK 输入周期	同步, 无预分频器	T _{CY} + 10	—	—	ns	N = 预分频值 (1, 8, 64, 256)
			同步, 带预分频器	取两者中的较大值: 20 ns 或 (T _{CY} + 40)/N	—	—	—	
			异步	20	—	—	ns	
OS60	Ft1	SOSC1/T1CK 振荡器输入频率范围 (通过将 TCS (T1CON 的 bit 1) 位置 1 使能振荡器)	DC	—	50	kHz		
TA20	T _{CKEXTMRL}	从外部 TxCK 时钟边沿到定时器递增的延时	0.5 T _{CY}	—	1.5 T _{CY}	—		

注 1: Timer1 为 A 类定时器。

表 23-23: B 类定时器 (TIMER2 和 TIMER4) 外部时钟时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)						
		工作温度						
		-40°C ≤ TA ≤ +85°C (工业级)						
		-40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TB10	TtxH	TxCK 高电平时间	同步, 无预分频器	0.5 Tcy + 20	—	—	ns	还必须满足参数 TB15
			同步, 带预分频器	10	—	—	ns	
TB11	TtxL	TxCK 低电平时间	同步, 无预分频器	0.5 Tcy + 20	—	—	ns	还必须满足参数 TB15
			同步, 带预分频器	10	—	—	ns	
TB15	TtxP	TxCK 输入周期	同步, 无预分频器	Tcy + 10	—	—	ns	N = 预分频值 (1, 8, 64, 256)
			同步, 带预分频器	取两者中的较大值: 20 ns 或 (Tcy + 40)/N	—	—		
TB20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增的延时		0.5 Tcy	—	1.5 Tcy	—	

注 1: Timer2 和 Timer4 为 B 类定时器。

表 23-24: C 类定时器 (TIMER3 和 TIMER5) 外部时钟时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)						
		工作温度						
		-40°C ≤ TA ≤ +85°C (工业级)						
		-40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TC10	TtxH	TxCK 高电平时间	同步	0.5 Tcy + 20	—	—	ns	还必须满足参数 TC15
TC11	TtxL	TxCK 低电平时间	同步	0.5 Tcy + 20	—	—	ns	还必须满足参数 TC15
TC15	TtxP	TxCK 输入周期	同步, 无预分频器	Tcy + 10	—	—	ns	N = 预分频值 (1, 8, 64, 256)
			同步, 带预分频器	取两者中的较大值: 20 ns 或 (Tcy + 40)/N	—	—		
TC20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增的延时		0.5 Tcy	—	1.5 Tcy	—	

注 1: Timer3 和 Timer5 为 C 类定时器。

dsPIC30F3014/4013

图 23-9: 输入捕捉 (CAPx) 时序特性

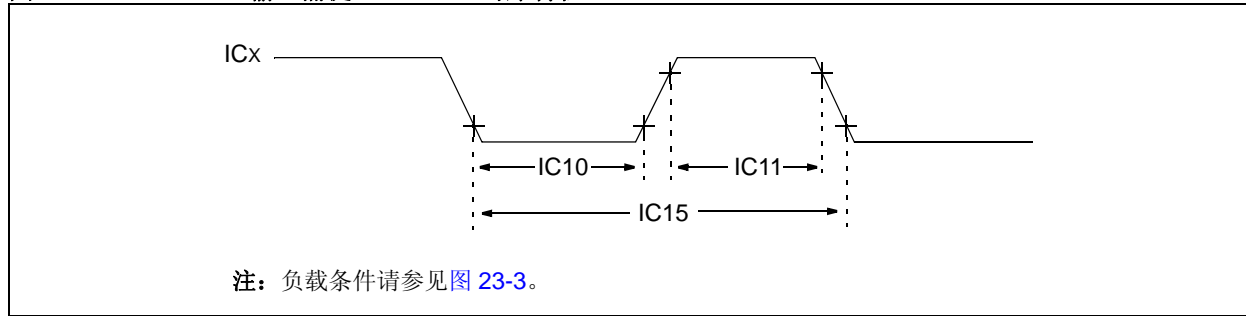


表 23-25: 输入捕捉时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)				
		工作温度				
		-40°C ≤ TA ≤ +85°C (工业级)				
		-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	最大值	单位	条件
IC10	TccL	ICx 输入低电平时间	无预分频器	0.5 Tcy + 20	—	ns
		带预分频器	10	—	ns	
IC11	TccH	ICx 输入高电平时间	无预分频器	0.5 Tcy + 20	—	ns
			带预分频器	10	—	ns
IC15	TccP	ICx 输入周期	(2 Tcy + 40)/N	—	ns	N = 预分频值 (1, 4, 16)

注 1: 这些参数仅为特征值, 未经生产测试。

图 23-10: 输出比较模块 (OCx) 时序特性

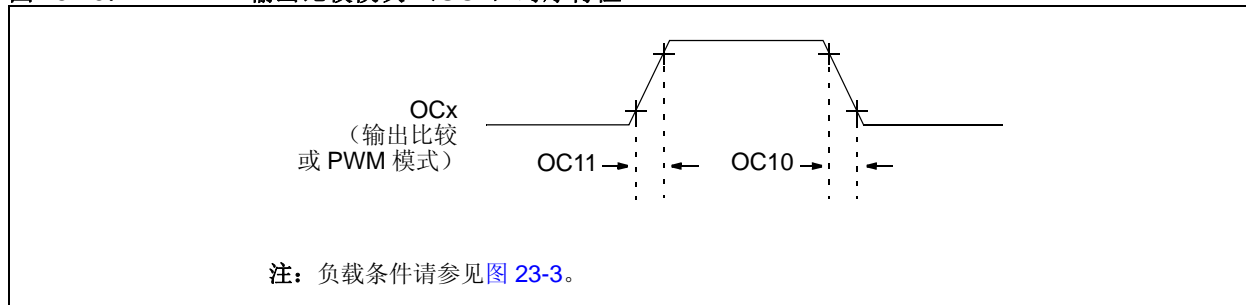


表 23-26: 输出比较模块时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)					
		工作温度					
		-40°C ≤ TA ≤ +85°C (工业级)					
		-40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
OC10	TccF	OCx 输出下降时间	—	—	—	ns	参见参数 DO32
OC11	TccR	OCx 输出上升时间	—	—	—	ns	参见参数 DO31

注 1: 这些参数仅为特征值, 未经生产测试。

注 2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。

图 23-11: OCx/PWM 模块时序特性

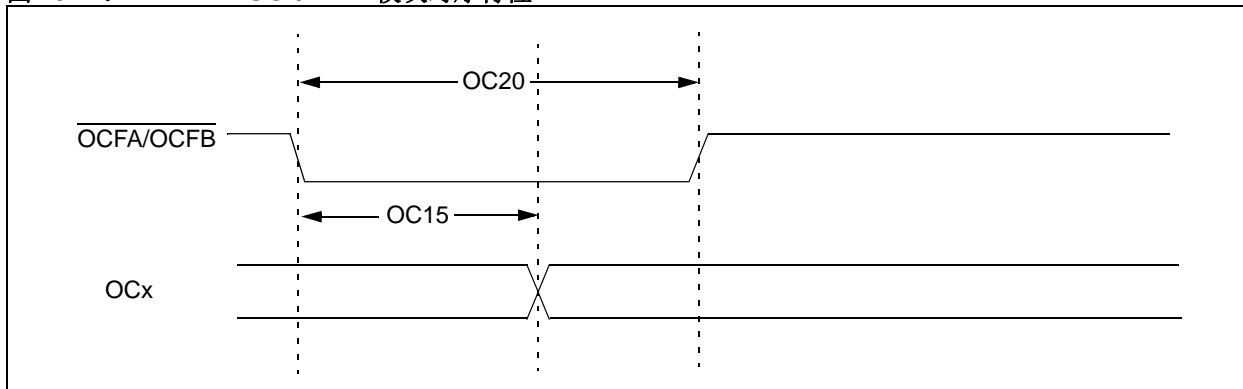


表 23-27: 简单 OCx/PWM 模式时序要求

交流特性				标准工作条件: 2.5V 到 5.5V (除非另外说明)			
				工作温度			
				-40°C ≤ TA ≤ +85°C (工业级)			
				-40°C ≤ TA ≤ +125°C (扩展级)			
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
OC15	TFD	故障输入到 PWM I/O 改变	—	—	50	ns	
OC20	TFLT	故障输入脉冲宽度	50	—	—	ns	

注 1: 这些参数仅为特征值, 未经生产测试。

2: 除非另外说明, “典型值” 栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。

dsPIC30F3014/4013

图 23-12: DCI 模块 (多通道, I²S 模式) 时序特性

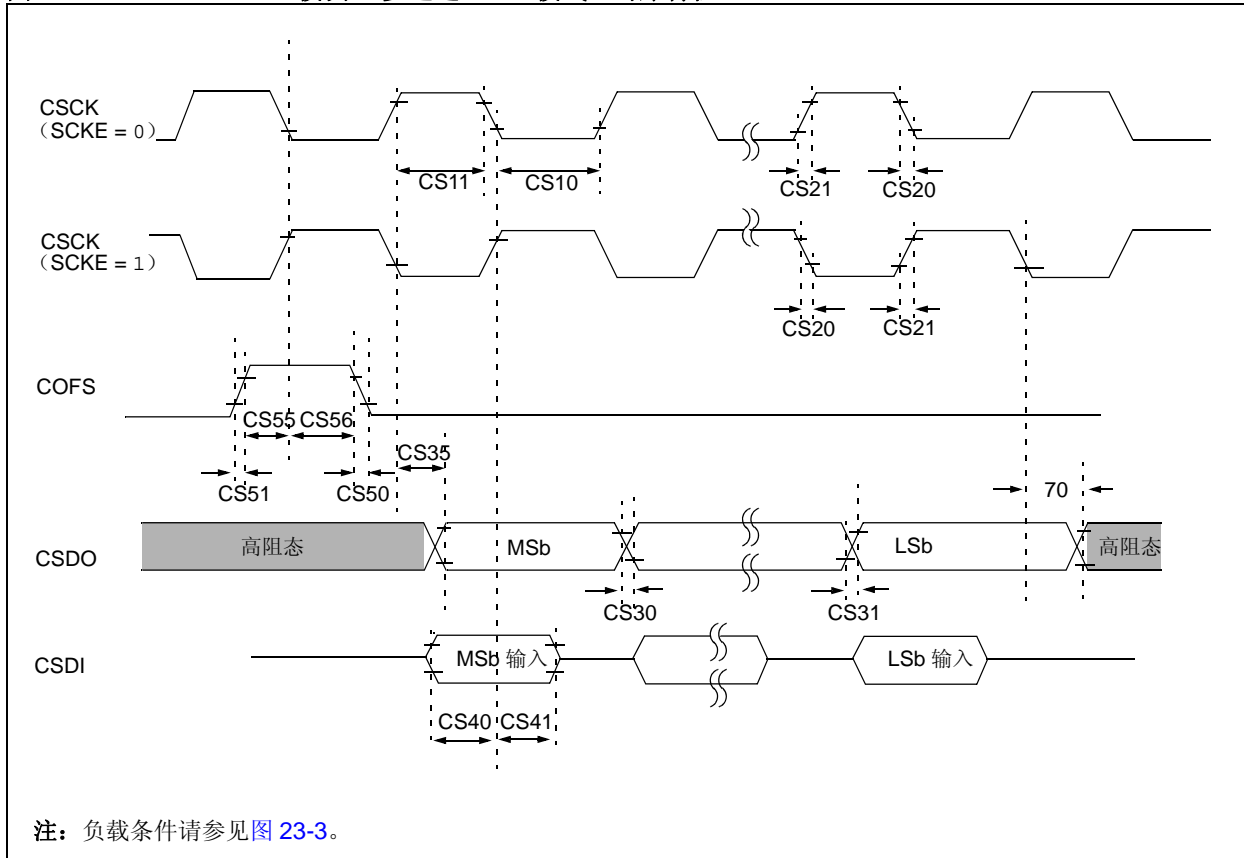


表 23-28: DCI 模块（多通道，I²S 模式）时序要求

交流特性			标准工作条件: 2.5V 到 5.5V (除非另外说明) 工作温度 -40°C ≤ T _A ≤ +85°C (工业级) -40°C ≤ T _A ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
CS10	TcSCKL	C _{SCK} 输入低电平时间 (C _{SCK} 引脚为输入)	T _{cy} /2 + 20	—	—	ns	
		C _{SCK} 输出低电平时间 (3) (C _{SCK} 引脚为输出)	30	—	—	ns	
CS11	TcSCKH	C _{SCK} 输入高电平时间 (C _{SCK} 引脚为输入)	T _{cy} /2 + 20	—	—	ns	
		C _{SCK} 输出高电平时间 (3) (C _{SCK} 引脚为输出)	30	—	—	ns	
CS20	TcSCKF	C _{SCK} 输出下降时间 (4) (C _{SCK} 引脚为输出)	—	10	25	ns	
CS21	TcSCKR	C _{SCK} 输出上升时间 (4) (C _{SCK} 引脚为输出)	—	10	25	ns	
CS30	TcSDOF	C _{SDO} 数据输出下降时间 (4)	—	10	25	ns	
CS31	TcSDOR	C _{SDO} 数据输出上升时间 (4)	—	10	25	ns	
CS35	TDV	时钟边沿到 C _{SDO} 数据有效的 时间	—	—	10	ns	
CS36	TDIV	时钟边沿到 C _{SDO} 呈现三态的 时间	10	—	20	ns	
CS40	TcSDI	C _{SDI} 数据输入到 C _{SCK} 边沿的 建立时间 (C _{SCK} 引脚为输入或 输出)	20	—	—	ns	
CS41	THCSDI	C _{SDI} 数据输入到 C _{SCK} 边沿的 保持时间 (C _{SCK} 引脚为输入或 输出)	20	—	—	ns	
CS50	TcoFSF	COFS 下降时间 (COFS 引脚为输出)	—	10	25	ns	注 1
CS51	TcoFSR	COFS 上升时间 (COFS 引脚为输出)	—	10	25	ns	注 1
CS55	TscOFS	COFS 数据输入到 C _{SCK} 边沿的 建立时间 (COFS 引脚为输入或 输出)	20	—	—	ns	
CS56	THCOFS	COFS 数据输入到 C _{SCK} 边沿的 保持时间 (COFS 引脚为输入或 输出)	20	—	—	ns	
CS57	TPCCK	C _{SCK} 时钟周期	100	—	—	ns	

注 1: 这些参数仅为特征值, 未经生产测试。

2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。

3: C_{SCK} 的最小时钟周期是 100 ns。所以, 在主模式下产生的时钟必须遵循该规范。

4: 假设所有 DCI 引脚上的负载都为 50 pF。

dsPIC30F3014/4013

图 23-13: DCI 模块 (AC-LINK 模式) 时序特性

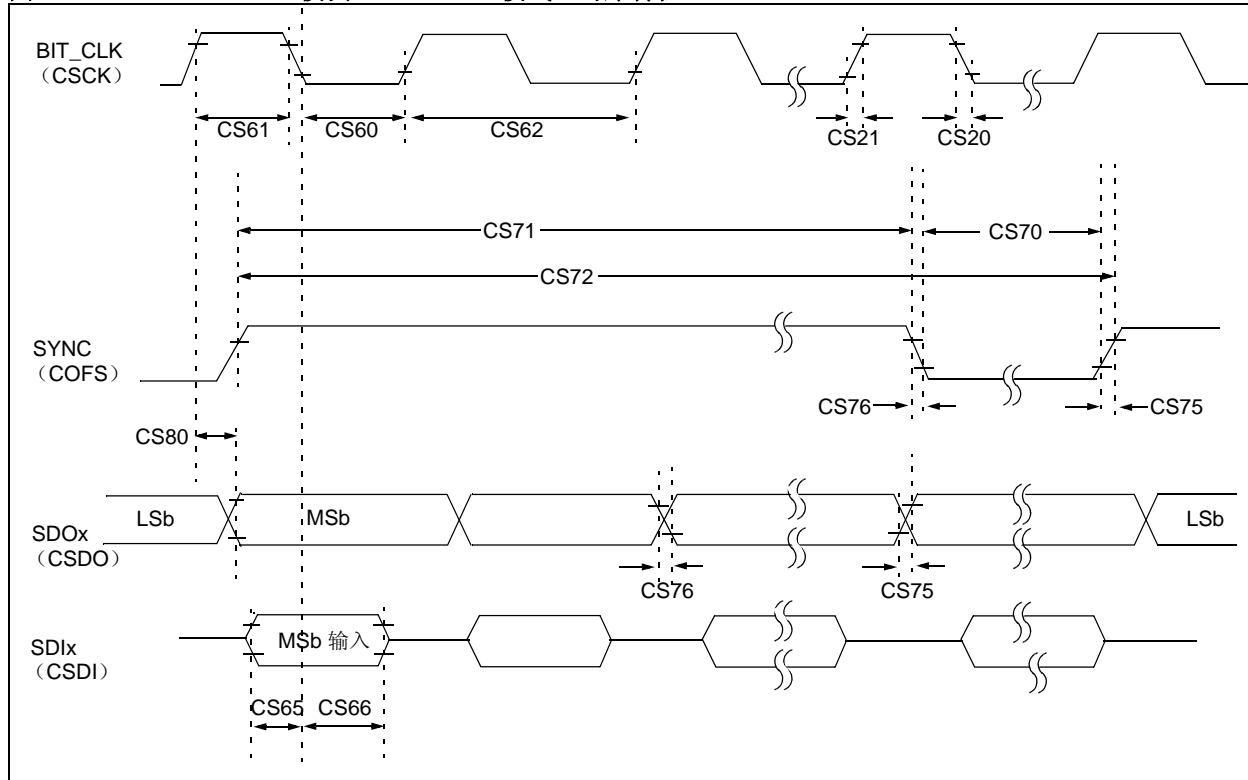
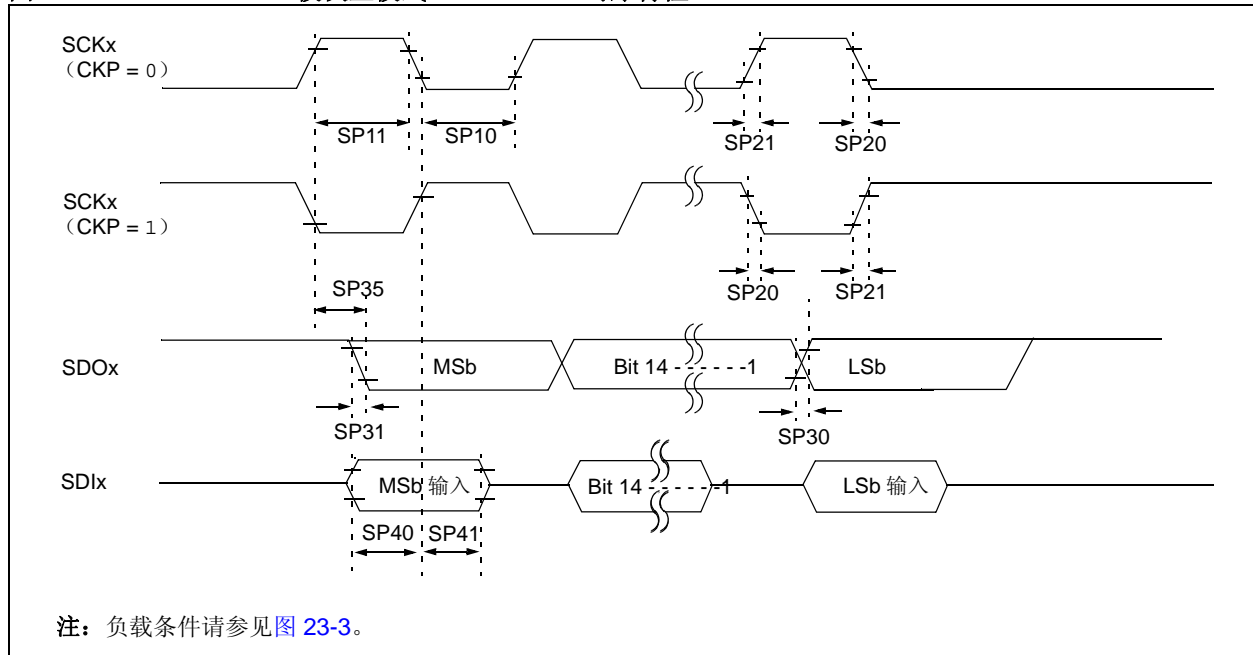


表 23-29: DCI 模块 (AC-LINK 模式) 时序要求

交流特性		标准工作条件: 2.5V 到 5.5V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性 (1,2)	最小值	典型值 (3)	最大值	单位	条件
CS60	TBCLKL	BIT_CLK 低电平时间	36	40.7	45	ns	
CS61	TBCLKH	BIT_CLK 高电平时间	36	40.7	45	ns	
CS62	TBCLK	BIT_CLK 周期	—	81.4	—	ns	位时钟为输入
CS65	TSACL	到 BIT_CLK 的下降沿的输入建立时间	—	—	10	ns	
CS66	THACL	到 BIT_CLK 的下降沿的输入保持时间	—	—	10	ns	
CS70	TSYNCL	同步数据输出低电平时间	—	19.5	—	μs	注 1
CS71	TSYNCH	同步数据输出高电平时间	—	1.3	—	μs	注 1
CS72	TSYNC	同步数据输出周期	—	20.8	—	μs	注 1
CS75	TRACL	上升时间, 同步, SDATA_OUT	—	10	25	ns	CLOAD = 50 pF, VDD = 5V
CS76	TFACL	下降时间, 同步, SDATA_OUT	—	10	25	ns	CLOAD = 50 pF, VDD = 5V
CS80	TOVDA	从 BIT_CLK 的上升沿到输出有效的延时	—	—	15	ns	

- 注 1: 这些参数仅为特征值, 未经生产测试。
 注 2: 这些值假设 BIT_CLK 频率为 12.288 MHz。
 注 3: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。

图 23-14: SPI 模块主模式 (CKE = 0) 时序特性



dsPIC30F3014/4013

表 23-30: SPI 模块主模式 (CKE = 0) 时序要求

交流特性			标准工作条件: 2.5V 到 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间 (3)	Tcy/2	—	—	ns	
SP11	TscH	SCKx 输出高电平时间 (3)	Tcy/2	—	—	ns	
SP20	TscF	SCKx 输出下降时间 (4)	—	—	—	ns	见参数 DO32
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 DO31
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 DO32
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 DO31
SP35	Tsch2doV, TscL2doV	SCKx 沿之后的 SDOx 数据输出有效时间	—	—	30	ns	
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	

- 注 1: 这些参数仅为特征值, 未经生产测试。
 2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。
 3: SCKx 的最小时钟周期是 100 ns。所以, 在主模式下生成的时钟必须遵循该规范。
 4: 假设所有 SPI 引脚上的负载都为 50 pF。

图 23-15: SPI 模块主模式 (CKE = 1) 时序特性

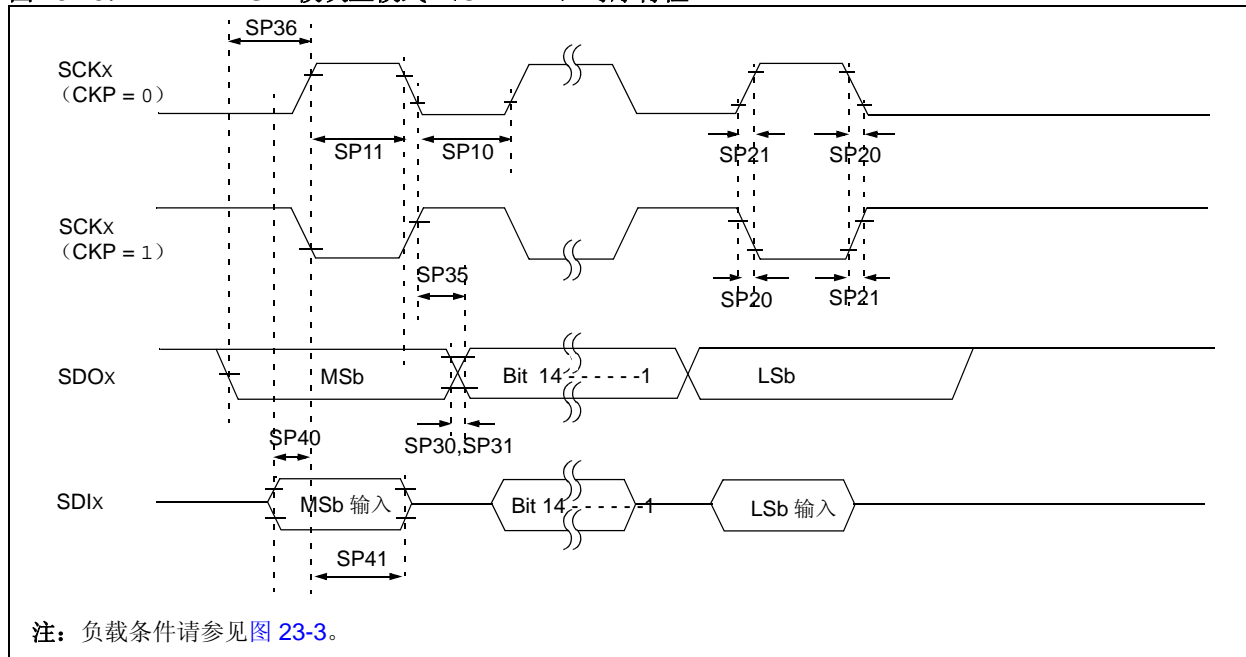
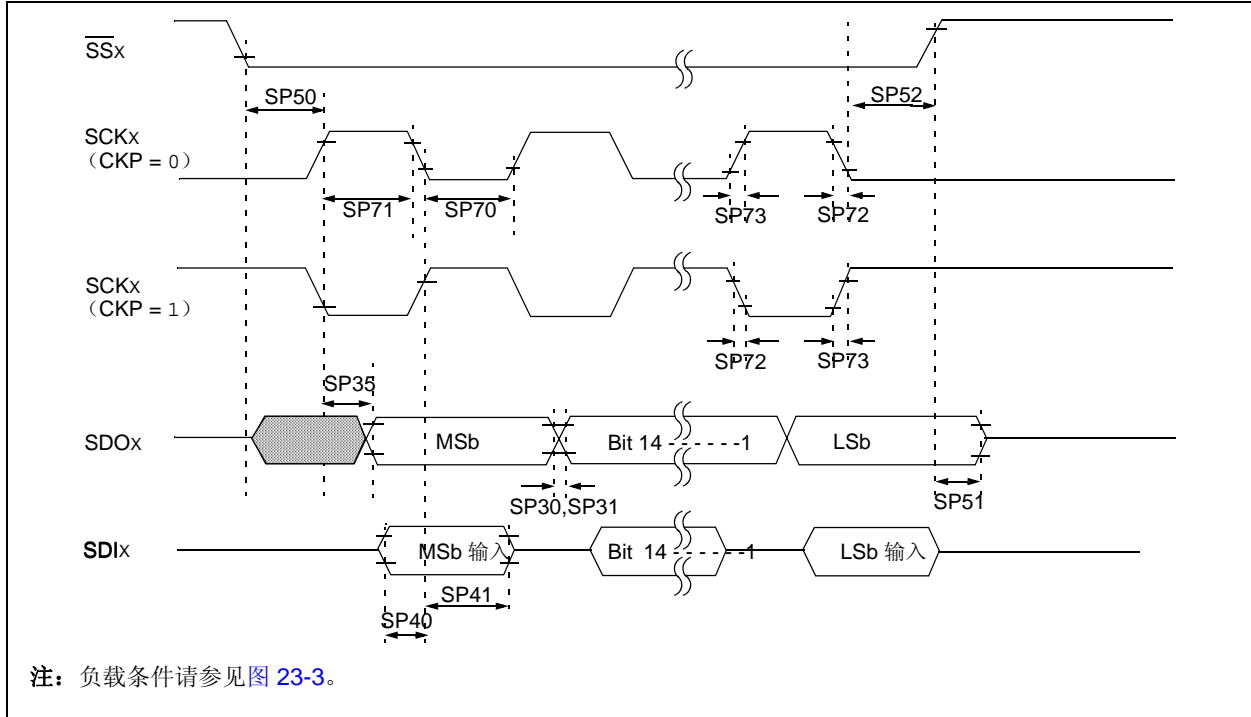


表 23-31: SPI 模块主模式 (CKE = 1) 时序要求

交流特性			标准工作条件: 2.5V 到 5.5V (除非另外指明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间 (3)	Tcy/2	—	—	ns	
SP11	TscH	SCKx 输出高电平时间 (3)	Tcy/2	—	—	ns	
SP20	TscF	SCKx 输出下降时间 (4)	—	—	—	ns	见参数 DO32
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 DO31
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 DO32
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 DO31
SP35	Tsch2do, TscL2doV	SCKx 边沿后 SDOx 数据输出有效的时间	—	—	30	ns	
SP36	TdoV2sc, TdoV2scL	SDOx 数据输出到第一个 SCKx 边沿的建立时间	30	—	—	ns	
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	

- 注 1: 这些参数仅为特征值, 未经生产测试。
 2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。
 3: SCKx 的最大时钟周期是 100 ns。所以, 在主模式下生成的时钟必须遵循该规范。
 4: 假设所有 SPI 引脚上的负载都为 50 pF。

图 23-16: SPI 模块从模式 (CKE = 0) 时序特性



dsPIC30F3014/4013

表 23-32: SPI 模块从模式 (CKE = 0) 时序要求

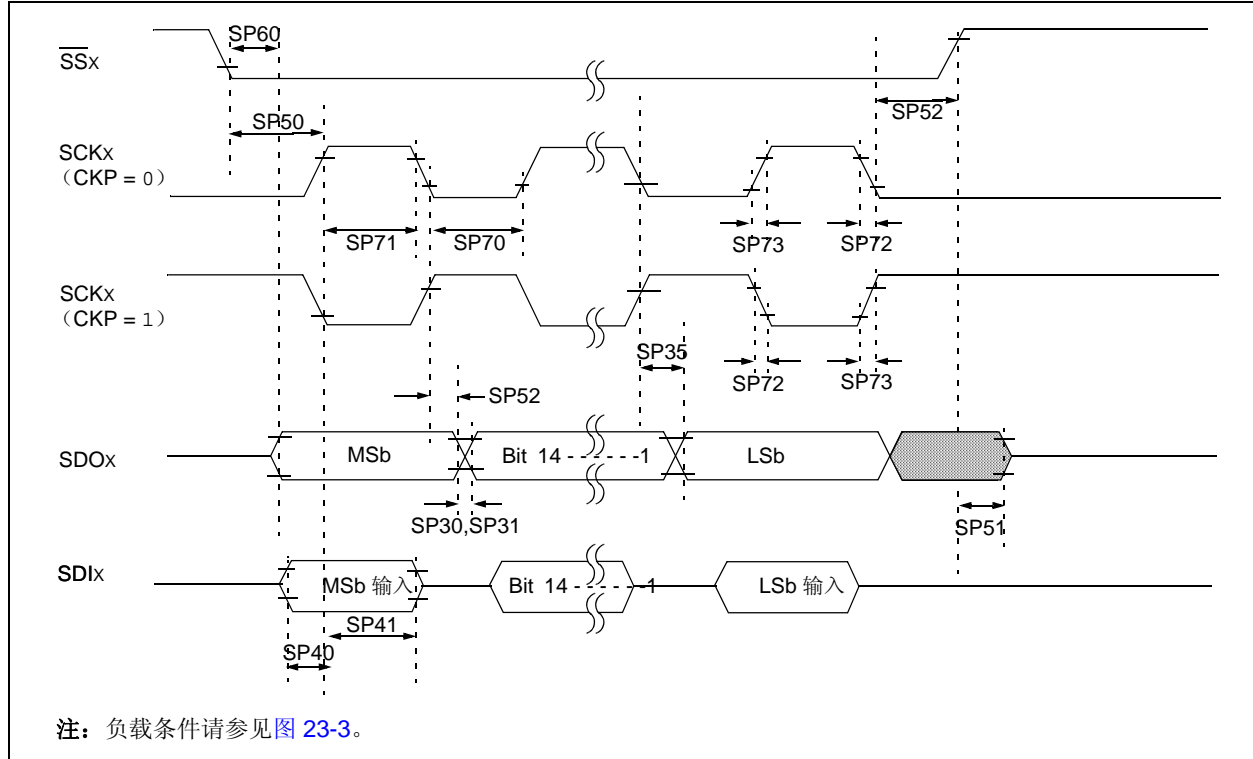
交流特性			标准工作条件: 2.5V 到 5.5V (除非另外指明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	
SP72	TscF	SCKx 输入下降时间 (3)	—	—	25	ns	
SP73	TscR	SCKx 输入上升时间 (3)	—	—	25	ns	
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 DO32
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 DO31
SP35	Tsch2do, TscL2doV	SCKx 边沿后 SDOx 数据输出有效的 时间	—	—	30	ns	
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的 建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的 保持时间	20	—	—	ns	
SP50	Tssl2scH, Tssl2scL	SSx↓ 到 SCKx↑ 或 SCKx↓ 输入的 时间	120	—	—	ns	
SP51	TssH2doZ	SSx↑ 到 SDOx 输出呈现高阻的 时间 (3)	10	—	50	ns	
SP52	Tsch2ssH TscL2ssH	SCKx 边沿后 SSx 有效的 时间	1.5 Tcy +40	—	—	ns	

注 1: 这些参数仅为特征值, 未经生产测试。

2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。

3: 假设所有 SPI 引脚上的负载都为 50 pF。

图 23-17: SPI 模块从模式 (CKE = 1) 时序特性



dsPIC30F3014/4013

表 23-33: SPI 模块从模式 (CKE = 1) 时序要求

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度		-40°C ≤ TA ≤ +85°C (工业级)		
					-40°C ≤ TA ≤ +125°C (扩展级)		
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	
SP72	TscF	SCKx 输入下降时间 (3)	—	—	25	ns	
SP73	TscR	SCKx 输入上升时间 (3)	—	—	25	ns	
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 DO32
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 DO31
SP35	Tsch2do, TscL2doV	SCKx 边沿后 SDOx 数据输出有效的 时间	—	—	30	ns	
SP40	TdiV2sch, TdiV2scL	SDIx 数据输入到 SCKx 边沿的 建立时间	20	—	—	ns	
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的 保持时间	20	—	—	ns	
SP50	TssL2sch, TssL2scL	\overline{SSx} ↓ 到 SCKx↓ 或 SCKx↑ 输入的 时间	120	—	—	ns	
SP51	TssH2doZ	\overline{SS} ↑ 到 SDOx 输出呈现高阻的 时间 (4)	10	—	50	ns	
SP52	Tsch2ssH TscL2ssH	SCKx 边沿后 \overline{SSx} ↑ 的时间	1.5 Tcy +40	—	—	ns	
SP60	TssL2doV	SCKx 边沿后 SDOx 数据输出有效的 时间	—	—	50	ns	

- 注 1: 这些参数仅为特征值, 未经生产测试。
- 2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。
- 3: SCKx 的最大时钟周期是 100 ns。所以, 在主模式下生成的时钟必须遵循该规范。
- 4: 假设所有 SPI 引脚上的负载都为 50 pF。

图 23-18: I²C™ 总线启动/停止位时序特性 (主模式)

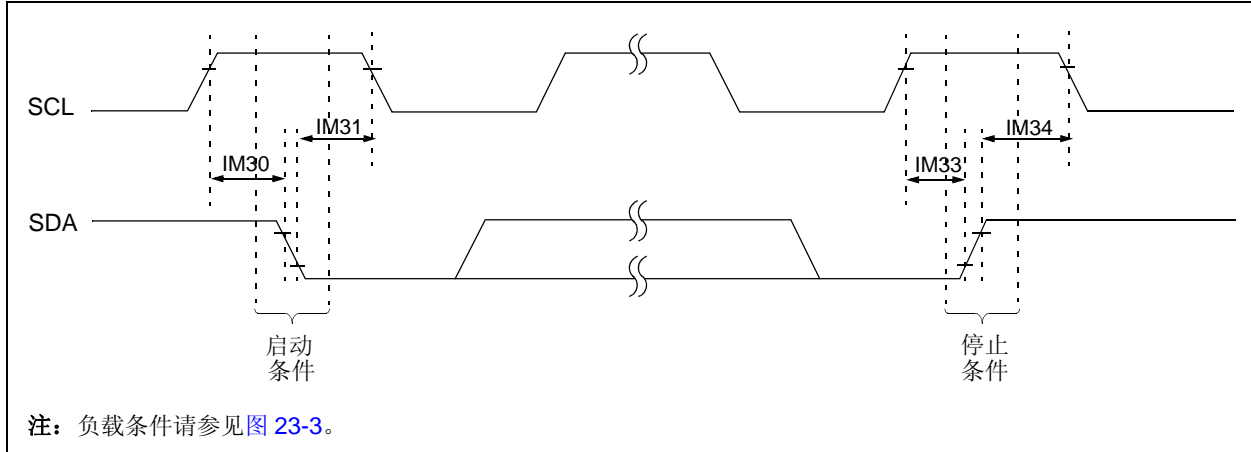


图 23-19: I²C™ 总线数据时序特性 (主模式)

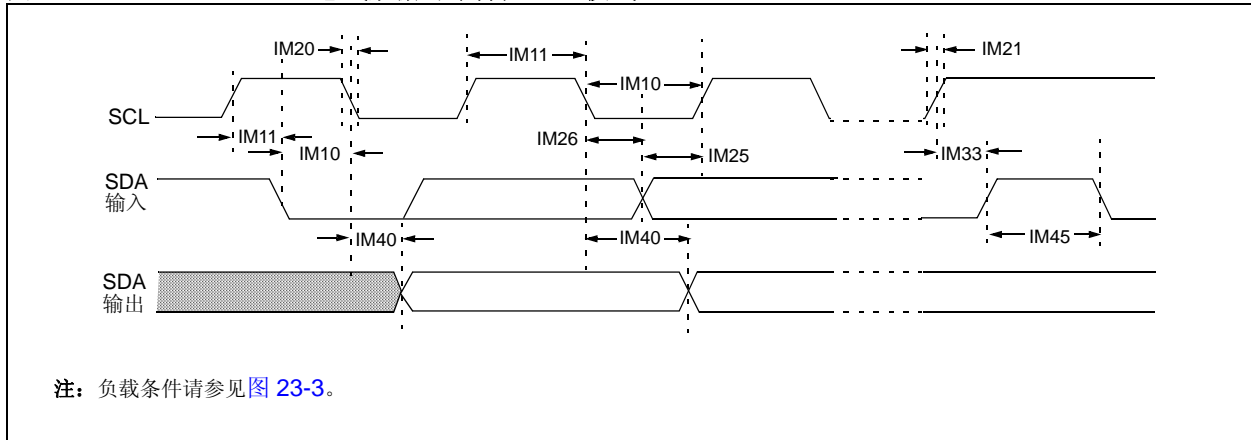


表 23-34: I²C™ 总线数据时序要求 (主模式)

交流特性			标准工作条件: 2.5V 到 5.5V (除非另外说明)			
			工作温度			
			-40°C ≤ Ta ≤ +85°C (工业级)			
			-40°C ≤ Ta ≤ +125°C (扩展级)			
参数编号	符号	特性	最小值 ⁽¹⁾	最大值	单位	条件
IM10	TLO:SCL	时钟低电平时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs
			1 MHz 模式 ⁽²⁾	Tcy/2 (BRG + 1)	—	μs
IM11	THI:SCL	时钟高电平时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs
			1 MHz 模式 ⁽²⁾	Tcy/2 (BRG + 1)	—	μs
IM20	TF:SCL	SDA 和 SCL 下降时间	100 kHz 模式	—	300	ns
			400 kHz 模式	20 + 0.1 Cb	300	ns
			1 MHz 模式 ⁽²⁾	—	100	ns

注 1: BRG 是 I²C 波特率发生器的值。请参见《dsPIC30F 系列参考手册》(DS70046E_CN) 中的第 21 章“I²C™”(DS70068)。

注 2: 对所有 I²C 引脚, 最大引脚电容 = 10 pF (仅限 1 MHz 模式)。

dsPIC30F3014/4013

表 23-34: I²C™ 总线数据时序要求 (主模式) (续)

交流特性			标准工作条件: 2.5V 到 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ T _A ≤ +85°C (工业级)				
			-40°C ≤ T _A ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值 ⁽¹⁾	最大值	单位	条件	
IM21	TR:SCL	SDA 和 SCL 上升时间	100 kHz 模式	—	1000	ns	规定 C _B 在 10 pF 到 400 pF 之间
			400 kHz 模式	20 + 0.1 C _B	300	ns	
			1 MHz 模式 ⁽²⁾	—	300	ns	
IM25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	
			400 kHz 模式	100	—	ns	
			1 MHz 模式 ⁽²⁾	—	—	ns	
IM26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	ns	
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 ⁽²⁾	—	—	ns	
IM30	TSU:STA	启动条件建立时间	100 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	只与重复启动条件有关
			400 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	T _{cy} /2 (BRG + 1)	—	μs	
IM31	THD:STA	启动条件保持时间	100 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	该周期之后产生第一个时钟脉冲
			400 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	T _{cy} /2 (BRG + 1)	—	μs	
IM33	TSU:STO	停止条件建立时间	100 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			400 kHz 模式	T _{cy} /2 (BRG + 1)	—	μs	
			1 MHz 模式 ⁽²⁾	T _{cy} /2 (BRG + 1)	—	μs	
IM34	THD:STO	停止条件保持时间	100 kHz 模式	T _{cy} /2 (BRG + 1)	—	ns	
			400 kHz 模式	T _{cy} /2 (BRG + 1)	—	ns	
			1 MHz 模式 ⁽²⁾	T _{cy} /2 (BRG + 1)	—	ns	
IM40	TAA:SCL	时钟输出有效时间	100 kHz 模式	—	3500	ns	
			400 kHz 模式	—	1000	ns	
			1 MHz 模式 ⁽²⁾	—	—	ns	
IM45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	μs	启动一个新的发送之前总线必须处于空闲状态的时间
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 ⁽²⁾	—	—	μs	
IM50	C _B	总线容性负载	—	400	pF		

注 1: BRG 是 I²C 波特率发生器的值。请参见《dsPIC30F 系列参考手册》(DS70046E_CN) 中的第 21 章 “I²C™” (DS70068)。

注 2: 对所有 I²C 引脚, 最大引脚电容 = 10 pF (仅限 1 MHz 模式)。

图 23-20: I²C™ 总线启动/停止位时序特性 (从模式)

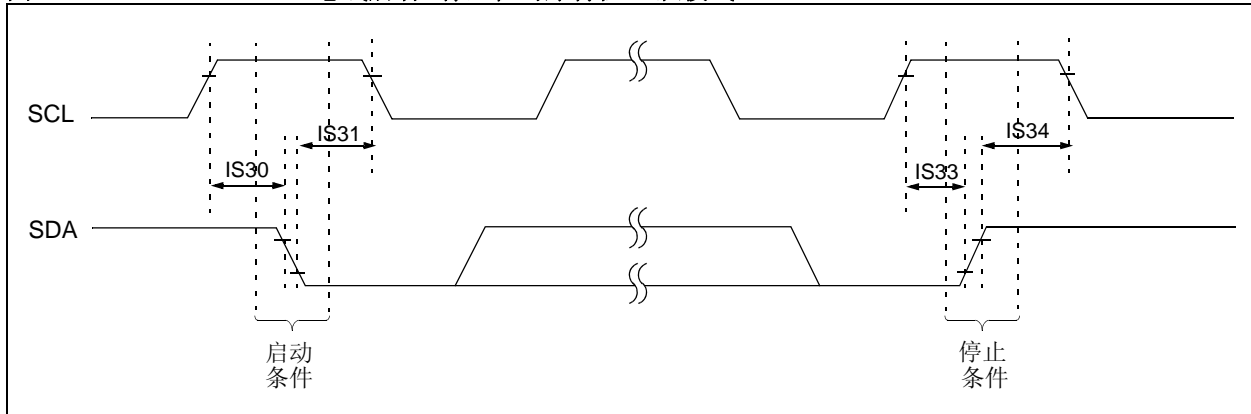
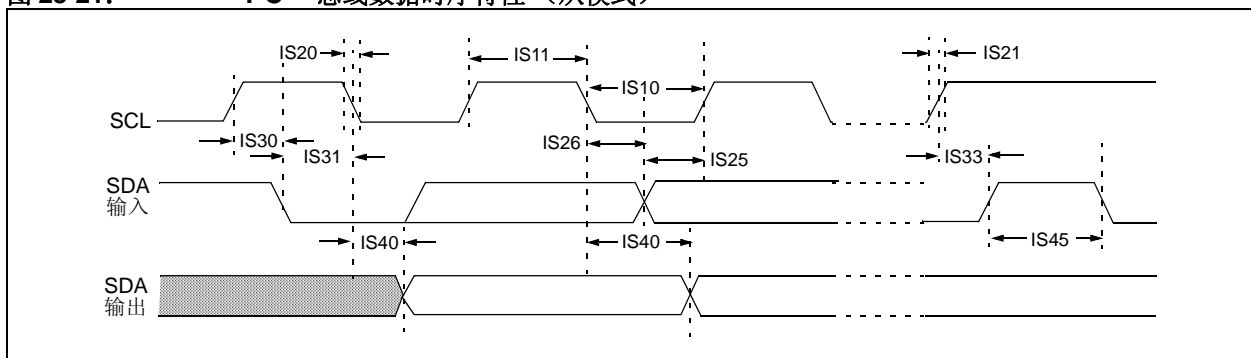


图 23-21: I²C™ 总线数据时序特性 (从模式)



dsPIC30F3014/4013

表 23-35: I²C™ 总线数据时序要求 (从模式)

交流特性		标准工作条件: 2.5V 到 5.5V (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)						
参数编号	符号	特性	最小值	最大值	单位	条件		
IS10	TLO:SCL	时钟低电平时间	100 kHz 模式	4.7	—	μs	器件工作频率最小不能低于 1.5 MHz	
			400 kHz 模式	1.3	—	μs		器件工作频率最小不能低于 10 MHz
			1 MHz 模式 (1)	0.5	—	μs		
IS11	THI:SCL	时钟高电平时间	100 kHz 模式	4.0	—	μs	器件工作频率最小不能低于 1.5 MHz	
			400 kHz 模式	0.6	—	μs		器件工作频率最小不能低于 10 MHz
			1 MHz 模式 (1)	0.5	—	μs		
IS20	TF:SCL	SDA 和 SCL 下降时间	100 kHz 模式	—	300	ns	规定 CB 在 10 pF 到 400 pF 之间	
			400 kHz 模式	20 + 0.1 CB	300	ns		
			1 MHz 模式 (1)	—	100	ns		
IS21	TR:SCL	SDA 和 SCL 上升时间	100 kHz 模式	—	1000	ns	规定 CB 在 10 pF 到 400 pF 之间	
			400 kHz 模式	20 + 0.1 CB	300	ns		
			1 MHz 模式 (1)	—	300	ns		
IS25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns		
			400 kHz 模式	100	—	ns		
			1 MHz 模式 (1)	100	—	ns		
IS26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	ns		
			400 kHz 模式	0	0.9	μs		
			1 MHz 模式 (1)	0	0.3	μs		
IS30	TSU:STA	启动条件建立时间	100 kHz 模式	4.7	—	μs	只与重复启动条件有关	
			400 kHz 模式	0.6	—	μs		
			1 MHz 模式 (1)	0.25	—	μs		
IS31	THD:STA	启动条件保持时间	100 kHz 模式	4.0	—	μs	该周期之后, 产生第一个时钟脉冲	
			400 kHz 模式	0.6	—	μs		
			1 MHz 模式 (1)	0.25	—	μs		
IS33	TSU:STO	停止条件建立时间	100 kHz 模式	4.7	—	μs		
			400 kHz 模式	0.6	—	μs		
			1 MHz 模式 (1)	0.6	—	μs		
IS34	THD:STO	停止条件保持时间	100 kHz 模式	4000	—	ns		
			400 kHz 模式	600	—	ns		
			1 MHz 模式 (1)	250	—	ns		
IS40	TAA:SCL	时钟输出有效时间	100 kHz 模式	0	3500	ns		
			400 kHz 模式	0	1000	ns		
			1 MHz 模式 (1)	0	350	ns		
IS45	TBF:SDA	总线空闲时间	100 kHz 模式	4.7	—	μs	启动一个新的发送之前总线必须处于空闲状态的时间	
			400 kHz 模式	1.3	—	μs		
			1 MHz 模式 (1)	0.5	—	μs		
IS50	CB	总线容性负载	—	400	pF			

注 1: 对于所有的 I²C 引脚, 最大引脚电容 = 10 pF (仅限 1 MHz 模式)。

图 23-22: CAN 模块 I/O 时序特性

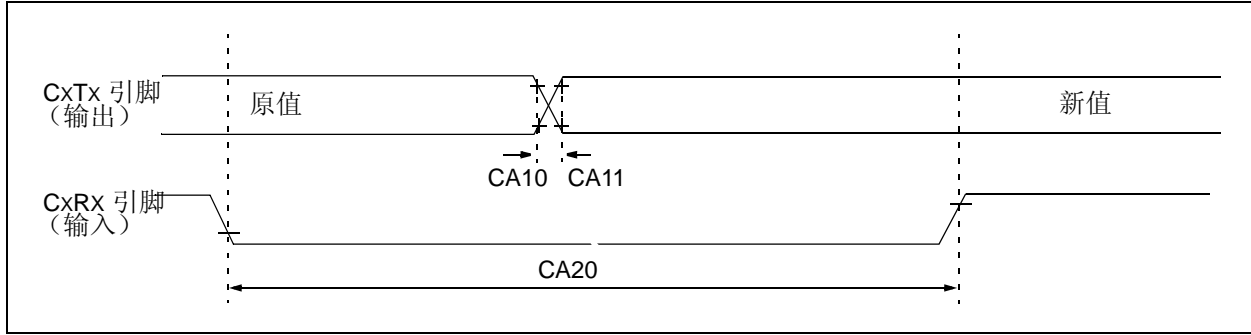


表 23-36: CAN 模块 I/O 时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外说明)					
		工作温度					
		-40°C ≤ TA ≤ +85°C (工业级)					
		-40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
CA10	TioF	端口输出下降时间	—	10	25	ns	
CA11	TioR	端口输出上升时间	—	10	25	ns	
CA20	Tcwf	触发 CAN 唤醒滤波器的脉冲宽度	500	—	—	ns	

注 1: 这些参数仅为特征值, 未经生产测试。

2: 除非另外说明, 否则“典型值”栏中的数据均为 5V、25°C 条件下的值。这些参数未经测试, 仅供设计参考。

dsPIC30F3014/4013

表 23-37: 12 位 A/D 模块规范

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数	符号	特性	最小值	典型值	最大值	单位	条件
器件电源							
AD01	AVDD	模块电源 VDD	取两者中的较大值: VDD - 0.3 或 2.7	—	取两者中的较小值: VDD + 0.3 或 5.5	V	
AD02	AVSS	模块电源 VSS	VSS - 0.3	—	VSS + 0.3	V	
参考输入							
AD05	VREFH	参考高电压	AVSS + 2.7	—	AVDD	V	
AD06	VREFL	参考低电压	AVSS	—	AVDD - 2.7	V	
AD07	VREF	绝对参考电压	AVSS - 0.3	—	AVDD + 0.3	V	
AD08	IREF	电流消耗	—	200 .001	300 2	μA μA	A/D 工作 A/D 关闭
模拟输入							
AD10	VINH-VINL	满量程输入范围	VREFL	—	VREFH	V	注 1
AD11	VIN	绝对输入电压	AVSS - 0.3	—	AVDD + 0.3	V	
AD12	—	泄漏电流	—	±0.001	±0.610	μA	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V 源阻抗 = 2.5 kΩ
AD13	—	泄漏电流	—	±0.001	±0.610	μA	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V 源阻抗 = 2.5 kΩ
AD15	RSS	开关阻抗	—	3.2K	—	Ω	
AD16	CSAMPLE	采样电容	—	18	—	pF	
AD17	RIN	模拟电压源的推荐阻抗	—	—	2.5K	Ω	
ADC 精度							
AD20	Nr	分辨率	12 个数据位			位	
AD21	INL	积分非线性	—	—	<±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD21A	INL	积分非线性	—	—	<±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD22	DNL	微分非线性	—	—	<±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD22A	DNL	微分非线性	—	—	<±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD23	GERR	增益误差	+1.25	+1.5	+3	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD23A	GERR	增益误差	+1.25	+1.5	+3	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V

注 1: A/D 转换结果不会因输入电压的上升而减小, 并且不会丢失编码。

表 23-37: 12 位 A/D 模块规范 (续)

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ T _A ≤ +85°C (工业级)				
			-40°C ≤ T _A ≤ +125°C (扩展级)				
参数	符号	特性	最小值	典型值	最大值	单位	条件
AD24	E _{OFF}	失调误差	-2	-1.5	-1.25	LSb	V _{INL} = AV _{SS} = V _{REFL} = 0V, AV _{DD} = V _{REFH} = 5V
AD24A	E _{OFF}	失调误差	-2	-1.5	-1.25	LSb	V _{INL} = AV _{SS} = V _{REFL} = 0V, AV _{DD} = V _{REFH} = 3V
AD25	—	单调性 ⁽¹⁾	—	—	—	—	保证
动态性能							
AD30	THD	总谐波失真	—	-71	—	dB	
AD31	SINAD	信噪比和失真	—	68	—	dB	
AD32	SFDR	无杂散动态范围	—	83	—	dB	
AD33	F _{NYQ}	输入信号带宽	—	—	100	kHz	
AD34	ENOB	有效位数	10.95	11.1	—	位	

注 1: A/D 转换结果不会因输入电压的上升而减小, 并且不会丢失编码。

dsPIC30F3014/4013

图 23-23: 12 位 A/D 转换时序特性 (ASAM = 0 且 SSRC = 000)

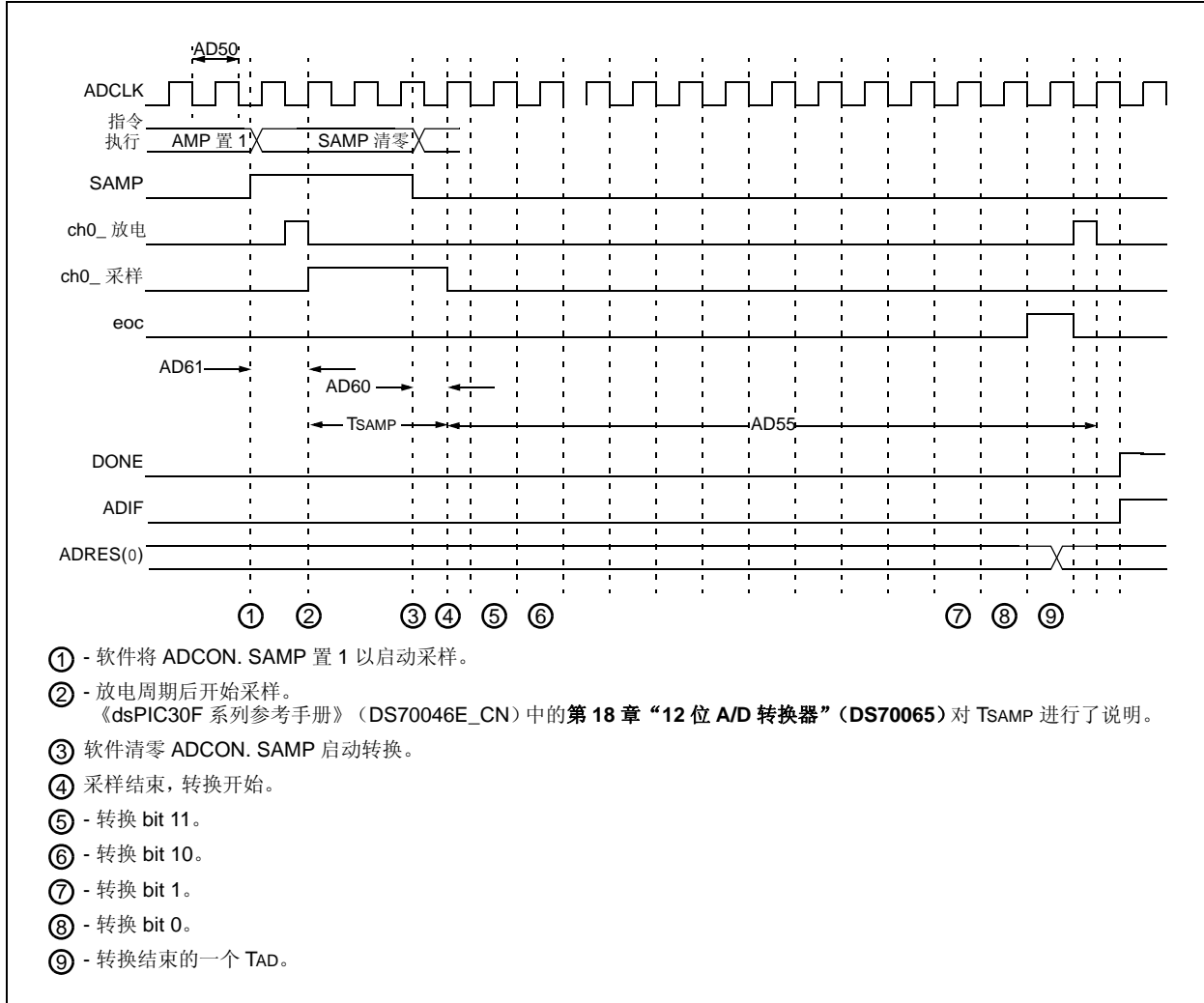


表 23-38: 12 位 A/D 转换时序要求

交流特性			标准工作条件: 2.7V 至 5.5V (除非另外说明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
时钟参数							
AD50	TAD	A/D 时钟周期	334	—	—	ns	VDD = 3-5.5V (注 1)
AD51	tRC	A/D 内部 RC 振荡器周期	1.2	1.5	1.8	μs	
转换速率							
AD55	tCONV	转换时间	—	14 TAD	—	ns	
AD56	FCNV	吞吐率	—	200	—	ksps	VDD = VREF = 5V
AD57	TSAMP	采样时间	1 TAD	—	—	ns	VDD = 3-5.5V 源阻抗 RS = 0-2.5 kΩ
时序参数							
AD60	tPCS	从采样触发到转换启动的时间	—	1 TAD	—	ns	
AD61	tPSS	将采样位 (SAMP) 置 1 开始采样	0.5 TAD	—	1.5 TAD	ns	
AD62	tCSS	转换完成到采样开始 (ASAM = 1) 的时间	—	0.5 TAD	—	ns	
AD63	tDPU ⁽²⁾	从 A/D 关闭到 A/D 打开模拟级的稳定时间	—	—	20	μs	

注 1: 由于采样电容最终将放电, 小于 10 kHz 的时钟速率会影响线性性能 (尤其在温度上升的情况下)。

注 2: tDPU 是开启 ADC 模块时 (ADCON1<ADON> = 1), 其到达稳定所需的时间。在此期间, ADC 结果是不确定的。

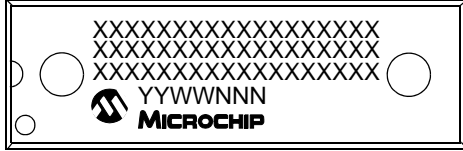
dsPIC30F3014/4013

注:

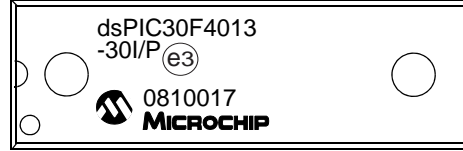
24.0 封装信息

24.1 封装标识信息

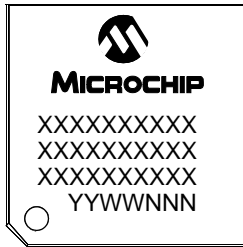
40 引脚 PDIP



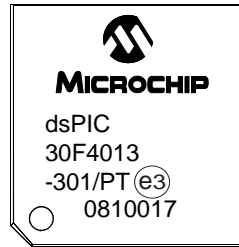
示例



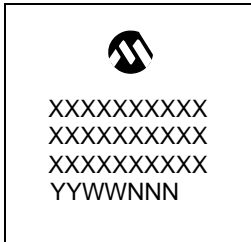
44 引脚 TQFP



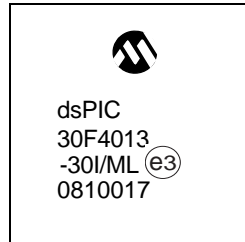
示例



44 引脚 QFN



示例

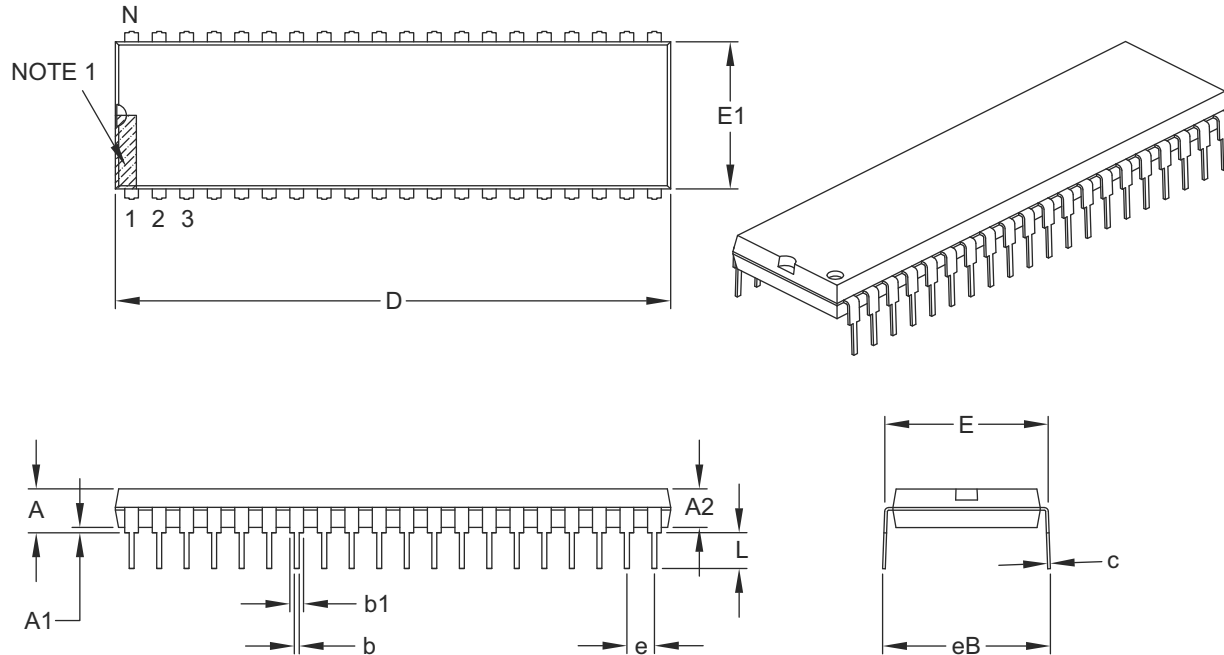


图注:	XX...X	客户信息
	Y	年份代码 (日历年的最后一位数字)
	YY	年份代码 (日历年的最后两位数字)
	WW	星期代码 (1月1日的星期代码为“01”)
	NNN	以字母数字排序的追踪代码
	(e3)	雾锡 (Matte Tin, Sn) 的 JEDEC 无铅标志
	*	表示无铅封装。JEDEC 无铅标志 (e3) 标示于此种封装的外包装上。
注:	Microchip 元器件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户信息的字符数。	

dsPIC30F3014/4013

40 引脚塑封双列直插式封装 (P) —— 主体 600 mil [PDIP]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	INCHES		
		MIN	NOM	MAX
Number of Pins	N	40		
Pitch	e	.100 BSC		
Top to Seating Plane	A	–	–	.250
Molded Package Thickness	A2	.125	–	.195
Base to Seating Plane	A1	.015	–	–
Shoulder to Shoulder Width	E	.590	–	.625
Molded Package Width	E1	.485	–	.580
Overall Length	D	1.980	–	2.095
Tip to Seating Plane	L	.115	–	.200
Lead Thickness	c	.008	–	.015
Upper Lead Width	b1	.030	–	.070
Lower Lead Width	b	.014	–	.023
Overall Row Spacing §	eB	–	–	.700

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.

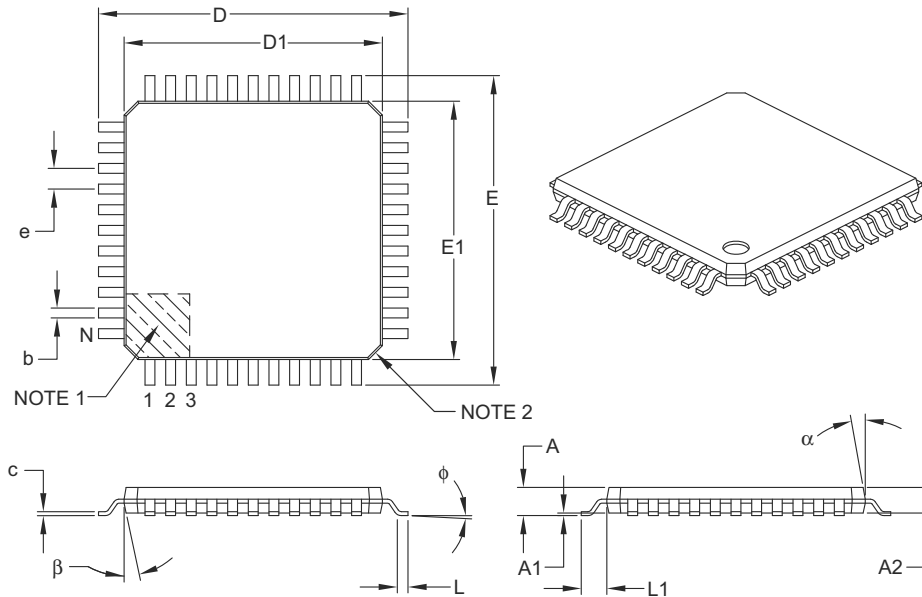
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-016B

dsPIC30F3014/4013

44 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 引脚长度 2.00mm [TQFP]

注: 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Leads	N	44		
Lead Pitch	e	0.80 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	ϕ	0°	3.5°	7°
Overall Width	E	12.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.30	0.37	0.45
Mold Draft Angle Top	α	11°	12°	13°
Mold Draft Angle Bottom	β	11°	12°	13°

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

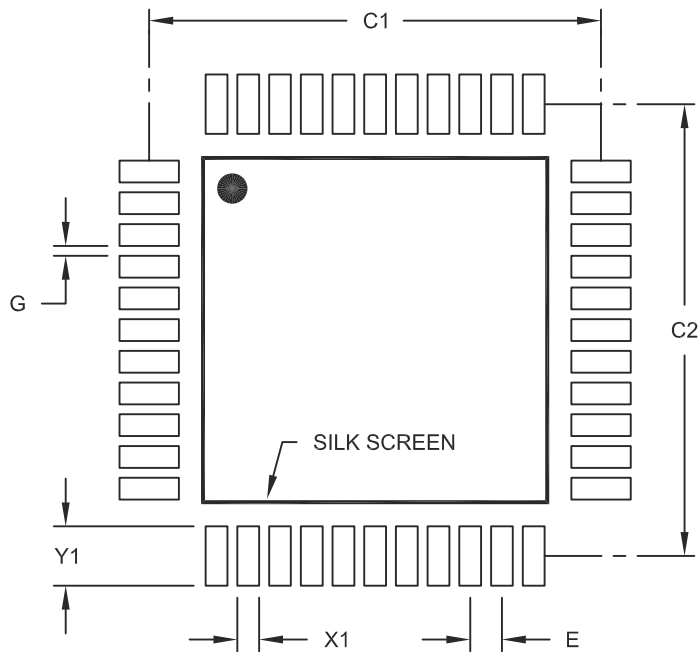
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-076B

dsPIC30F3014/4013

44 引脚塑封薄型正方扁平封装（PT）—— 主体 10x10x1 mm，引脚长度 2.00mm [TQFP]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.80 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X44)	X1			0.55
Contact Pad Length (X44)	Y1			1.50
Distance Between Pads	G	0.25		

Notes:

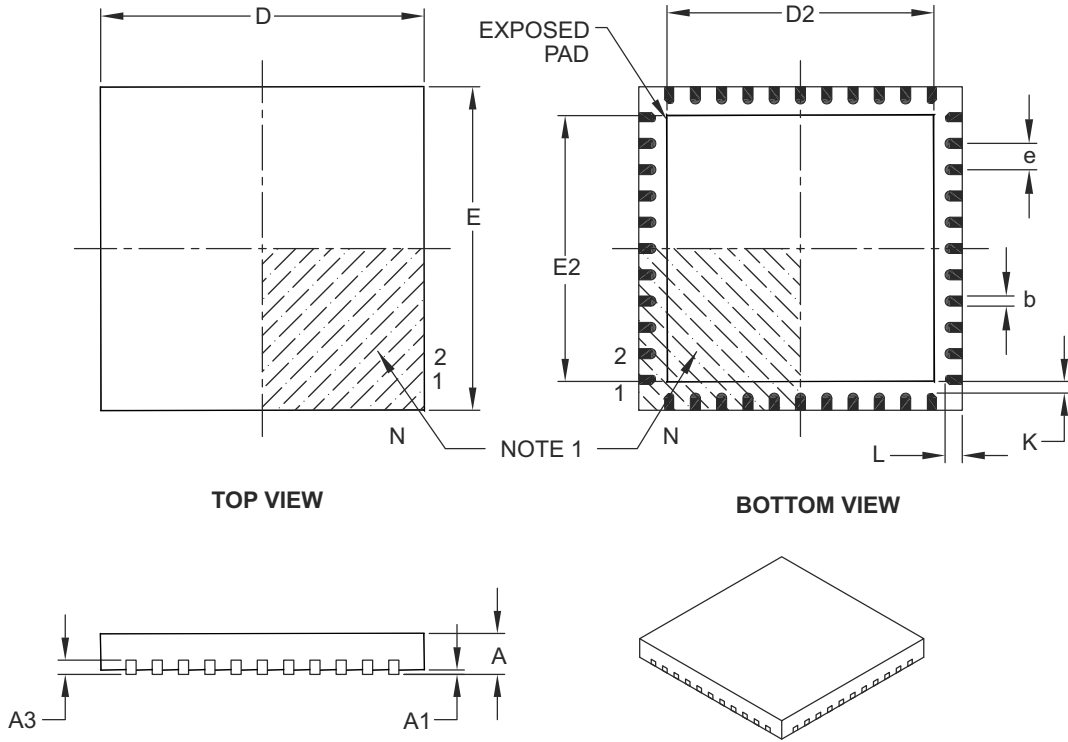
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2076A

44 引脚塑封正方扁平无脚封装 (ML) —— 主体 8x8 mm [QFN]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	44		
Pitch	e	0.65 BSC		
Overall Height	A	0.80	0.90	1.00
Standoff	A1	0.00	0.02	0.05
Contact Thickness	A3	0.20 REF		
Overall Width	E	8.00 BSC		
Exposed Pad Width	E2	6.30	6.45	6.80
Overall Length	D	8.00 BSC		
Exposed Pad Length	D2	6.30	6.45	6.80
Contact Width	b	0.25	0.30	0.38
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

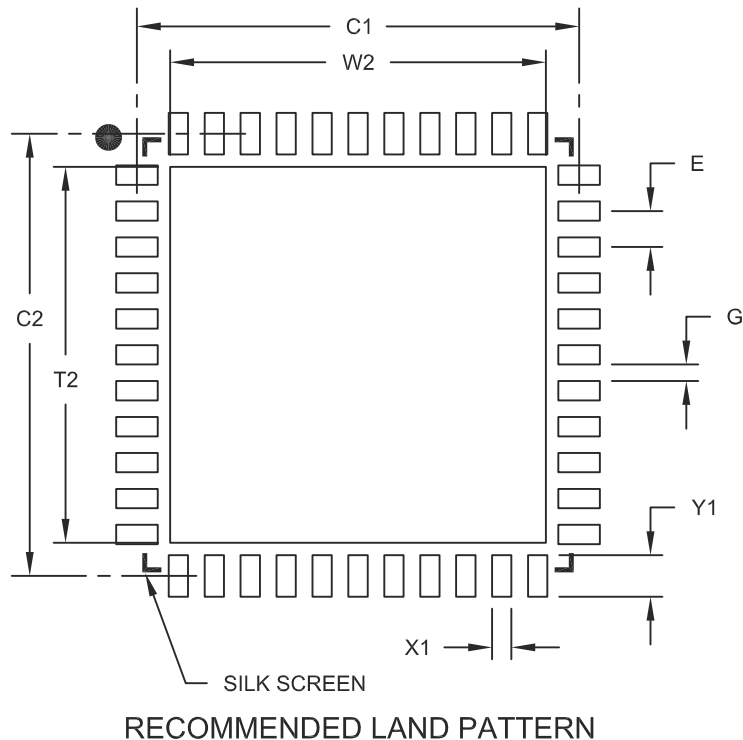
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-103B

dsPIC30F3014/4013

44 引脚塑封正方扁平无脚封装 (ML) —— 主体 8x8 mm [QFN]

注：最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	W2			6.80
Optional Center Pad Length	T2			6.80
Contact Pad Spacing	C1		8.00	
Contact Pad Spacing	C2		8.00	
Contact Pad Width (X44)	X1			0.35
Contact Pad Length (X44)	Y1			0.80
Distance Between Pads	G	0.25		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2103A

附录 A: 版本历史

版本 D (2006 年 6 月)

该数据手册的早期版本包含超前信息或初稿信息。这些版本中的特性数据不完整。

该版本做了以下更改:

- 修改了 I²C 从地址 (见表 14-1)
更新了 ADC 转换时钟选择的示例 (见第 19.0 节 “12 位模数转换器 (ADC) 模块”)
- 从指令集中删除了基本指令 CP1 (见表 21-2)
- 修改了电气特性:
 - 工作电流 (I_{DD}) 规范 (见表 23-5)
 - 空闲电流 (I_{IDLE}) 规范 (见表 23-6)
 - 掉电电流 (I_{PD}) 规范 (见表 23-7)
 - I/O 引脚输入规范 (见表 23-8)
 - 欠压复位 (BOR) 规范 (见表 23-11)
 - 看门狗定时器超时限制 (见表 23-20)

版本 E (2007 年 1 月)

该版本对封装图做了更新。

版本 F (2008 年 4 月)

该版本做了以下更新:

- 增加了 FUSE 配置寄存器 (FICD) 的详细信息 (见第 20.8 节 “器件配置寄存器” 和表 20-8)
- 在器件配置寄存器表中增加了注 2 (表 20-8)
- 删除了有关 CAN 接收错误发生的错误说明 (见第 17.4.5 节 “接收错误”)
- 更新了 ADC 转换时钟和采样速率的计算 (见例 19-1)。最小 TAD 为 334 ns。
- 更新了与输入电平变化通知模块相关的详细信息:
 - 更新了第 7.3 节 “输入电平变化通知模块” 的第一段的最后一句话。
 - 更新了表 7-2
 - 删除了表 7-3、表 7-4 和表 7-5。

- 电气特性:
 - 给出了参数 DO10、DO16、DO20 和 DO26 的值代替 TBD 值 (见表 23-9)
 - 更新了 10 位高速 ADC t_{PDU} 时序参数 (到达稳定的时间): 从典型值 20 μs 更新为最大值 20 μs (见表 23-38)
 - 扩展了参数 OS65 (内部 RC 精度), 增加对于不同温度的多个最小值和最大值 (见表 23-18)
 - 更新了参数 DC12 (RAM 数据保持电压) 以包含最小值 (见表 23-4)
 - 更新了参数 D134 (擦 / 写周期) 以包含最小值和最大值, 并且删除了其典型值 (见表 23-12)
 - 删除了参数 OS62 (内部 FRC 抖动) 和 OS64 (内部 FRC 漂移), 以及 AC 特性中的注 2 (见表 23-17)
 - 扩展了参数 OS63 (内部 FRC 精度), 增加对于不同温度的多个最小值和最大值 (见表 23-17)
 - 删除了表 23-5 和表 23-6 中的参数 DC27a、DC27b、DC47a 和 DC47b (速度为 20 MIPS、电压为 3.3V 条件下的 I_{DD})
 - 删除了表 23-29 中参数 CS77 和 CS78 (工作电压为 3.3V 条件下的 TRACL 和 TFACL)
 - 更新了参数 SY11 的最小值、最大值和条件, 并更新了 SY20 的最小值、典型值、最大值和条件 (见表 23-20)
- 对全文做了其他较少修改

dsPIC30F3014/4013

版本 G（2010 年 11 月）

该版本对数据手册中的文字和格式做了较小修改。

表 A-1 列出了各章节涉及的其他主要修改。

表 A-1: 主要章节更新

章节名称	更新说明
“高性能 16 位数字信号控制器”	对所有 QFN 引脚图增加了注 1（见“引脚图”）。
第 1.0 节 “器件概述”	删除了 dsPIC30F3014 框图中的“DCI”外设框（见图 1-1）。 更新了 AVDD 和 AVSS 的 I/O 引脚配置说明（见表 1-1）。
第 20.0 节 “系统集成”	在第 20.2.5 节 “快速 RC 振荡器（FRC）”中增加了 OSCTUN 功能的注。 更新了以下振荡器工作模式的工作频率（见表 20-1）： <ul style="list-style-type: none">• XTL• 带 16 倍频 PLL 的 XT• 带 4 倍频、8 倍频和 16 倍频 PLL 的 HS/2• 带 4 倍频、8 倍频和 16 倍频 PLL 的 HS/3• 带 4 倍频、8 倍频和 16 倍频 PLL 的 EC
第 23.0 节 “电气特性”	更新了 I/O 引脚输入规范中的参数 DI19 的最大值和参数 DI29 的最小值（见表 23-8）。 在程序存储器和 EEPROM 存储器规范中删除参数 D136 并且更新参数 D122 和 D134 的最小值、典型值、最大值和条件（见表 23-12）。

索引

数字

12 位模数转换器 (A/D) 模块 131

A

A/D 131

ADCHS 寄存器 131

ADCON1 寄存器 131

ADCON2 寄存器 131

ADCON3 寄存器 131

ADCSSL 寄存器 131

ADPCFG 寄存器 131

编程采样触发器 133

CPU 空闲模式下的操作 137

CPU 休眠模式下的操作 137

采样要求 136

掉电模式 137

复位影响 137

寄存器映射 139

结果缓冲区 132

连接注意事项 138

配置模拟端口引脚 54, 138

输出格式 137

中止转换 133

转换操作 132

AC-Link 模式操作 128

16 位模式 128

20 位模式 129

AC 特性 180

负载条件 181

AC 温度和电压规范 181

ADC 133

选择转换时钟 133

B

BOR。参见欠压复位 180

BOR 特性 180

备用向量表 64

编程操作

擦除程序存储器的一行 45

对闪存程序存储器编程的算法 45

启动编程序列 46

装载写锁存器 46

编程模型 16

变更通知客户服务 225

表

DCI 模块

常见编解码 CSMC 频率的器件频率 125

数据地址空间

无效存储器访问的结果 32

表指令操作汇总 43

C

CAN 模块 111

报文发送 115

报文接收 114

波特率设置 116

CAN1 寄存器映射 118

操作模式 113

概述 111

I/O 时序特性 205

I/O 时序要求 205

帧类型 111

C 编译器

MPLAB C18 168

CLKO 和 I/O 时序

特性 185

要求 185

CPU 架构概述 15

CPU 空闲模式下的输出比较操作 88

CPU 休眠和空闲模式下的 I²C 模块工作原理 96

操作模式

初始化 113

环回 113

监听 113

监听所有报文 113

禁止 113

正常操作 113

程序存储器和 EEPROM 存储器特性 180

程序地址空间

表指令

TBLRDH 27

TBLRDL 27

TBLWTH 27

TBLWTL 27

存储器映射 25

访问程序空间中的数据, 地址生成 26

构成 26

将数据空间映射到程序空间的操作 29

使用表指令访问程序存储器中的数据 27

使用程序空间可视性访问程序存储器中的数据 28

数据表访问 (lsw) 27

数据表访问 (MSB) 28

程序计数器 16

除法支持 18

指令 (表) 18

从休眠和空闲模式中唤醒 64

从休眠模式中唤醒 141

存储器构成 25

内核寄存器映射 33

D

DCI 模块

COFS 引脚 121

CPU 空闲模式下的操作 128

CPU 休眠模式下的操作 128

CSCK 引脚 121

CSDI 引脚 121

CSDO 模式位 128

CSDO 引脚 121

采样时钟边沿控制位 126

操作 123

从器件帧同步操作 124

发送 / 接收移位寄存器 121

发送时隙使能位 126

发送状态位 127

缓冲长度控制 127

缓冲区控制 121

缓冲区数据对齐 121

I/O 引脚 121

寄存器映射 130

介绍 121

接收时隙使能位 126

接收状态位 127

使能 123

时隙状态位 128

时序特性

AC-Link 模式 194

多路通道, I²S 模式 192

dsPIC30F3014/4013

时序要求			
AC-Link 模式	195	复位时序特性	186
多路通道, I ² S 模式	193	复位时序要求	187
数据对齐控制位	126	复位顺序	61
数据帧与缓冲区的对齐	127	复位源	61
数字环回模式	128	复位源	
同步数据传输	126	非法指令陷阱	61
位时钟发生器	125	看门狗定时器延时	61
下溢模式控制位	128	欠压复位 (BOR)	61
帧同步的间隙使能位操作	126	未初始化的 W 寄存器陷阱	61
帧同步发生器	123	陷阱锁定	61
帧同步模式控制位	123	负载条件	181
中断	128	G	
主器件帧同步操作	123	公式	
字长选择位	123	ADC 转换时钟	133
DC 特性		波特率	107
BOR	180	COFSG 周期	123
程序存储器和 EEPROM 存储器	180	串行时钟速率	96
低电压检测	178	发生时钟的时间份额	117
掉电电流 (IPD)	176	位时钟频率	125
工作电流 (IDD)	174	工作电流 (IDD)	174
I/O 引脚输出规范	178	H	
I/O 引脚输入规范	178	汇编器	
空闲电流 (I _{IDLE})	175	MPASM 汇编器	168
LVDL	179	I	
欠压复位	179	I/O 端口	53
dsPIC30F3014/4013 的端口寄存器映射	55	并行 (PIO)	53
DSP 引擎	19	I/O 引脚规范	
乘法器	21	输出	178
代码保护	141	输入	178
代码示例		I ² C 10 位从模式工作原理	93
擦除程序存储器的一行	45	发送	93
擦除数据 EEPROM 块	50	接收	94
擦除数据 EEPROM 字	50	I ² C 7 位从模式工作原理	93
读数据 EEPROM	49	发送	93
启动编程序列	46	接收	93
数据 EEPROM 块写操作	52	I ² C 模块	91
数据 EEPROM 字写操作	51	编程模型	91
装载写锁存器	46	CPU 休眠和空闲模式下的工作原理	96
带隙起振时间		地址	93
时序特性	187	多种模式	91
要求	187	功能描述	91
低电压检测特性	178	寄存器	91
低电压检测 (LVD)	155	寄存器映射	97
地址发生器单元	37	IPMI 支持	95
电气特性	171	全局呼叫地址支持	95
AC	180	软件控制的时钟延时 (STREN = 1)	94
掉电电流 (IPD)	176	斜率控制	95
读者反馈表	226	引脚配置	91
端口写 / 读示例	54	中断	95
F		总线启动 / 停止位时序特性	
防止意外写入 OSCCON	146	从模式	203
封装	211	主模式	201
标识	211	总线数据时序特性	
封装信息	211	从模式	203
复位	141, 151	主模式	201
BOR, 可编程	153	总线数据时序要求	
POR		从模式	204
具有长晶振起振时间	153	主模式	201
在无 FSCM 和 PWRT 的条件下工作	153	I ² C 主模式工作原理	
POR (上电复位)	151	波特率发生器	96
欠压复位 (BOR)	141	多主器件通信、总线冲突和总线仲裁	96
上电复位 (POR)	141	发送	95
上电延时定时器 (PWRT)	141	接收	96
振荡器起振定时器 (OST)	141		

时钟仲裁.....	96
I ² C 主器件支持.....	95
I ² S 模式操作.....	129
帧和数据字长选择.....	129
I ² S 模式工作原理	
数据对齐.....	129
J	
简单捕捉事件模式.....	81
缓冲区操作.....	82
霍尔传感器模式.....	82
Timer2 和 Timer3 选择模式.....	82
预分频器.....	81
简单 OC/PWM 模式时序要求.....	191
简单 PWM 模块.....	86
输入引脚默认保护.....	86
周期.....	87
简单输出比较匹配模式.....	86
节能模式.....	155
空闲.....	156
休眠.....	155
禁止 UART.....	105
K	
开发支持.....	167
看门狗定时器	
时序特性.....	186
时序要求.....	187
看门狗定时器 (WDT).....	141, 155
工作原理.....	155
使能和禁止.....	155
勘误表.....	9
可编程.....	141
客户通知服务.....	225
客户支持.....	225
空闲电流 (I _{IDLE}).....	175
控制寄存器.....	44
NVMADR.....	44
NVMADRU.....	44
NVMCON.....	44
NVMKEY.....	44
快速现场保存.....	64
框图	
12 位 A/D 功能.....	131
16 位 Timer1.....	67
16 位 Timer2.....	73
16 位 Timer3.....	73
16 位 Timer4.....	78
16 位 Timer5.....	78
32 位 Timer2/3.....	72
32 位 Timer4/5.....	77
CAN 缓冲区和协议引擎.....	112
DCI 模块.....	122
dsPIC30F3014.....	11
复位系统.....	151
共用端口结构.....	54
I ² C.....	92
SPI.....	100
SPI 主 / 从器件连接.....	100
输出比较模块.....	85
输入捕捉模块.....	81
UART 发送器.....	103
外部上电复位电路.....	153
振荡器系统.....	143
专用端口结构.....	53

L	
LVDL 特性.....	179
M	
Microchip 因特网网站.....	225
MPLAB ASM30 汇编器、链接器和库管理器.....	168
MPLAB PM3 器件编程器.....	170
MPLAB REAL ICE 在线仿真器系统.....	169
MPLAB 集成开发环境软件.....	167
MPLINK 目标链接器 /MPLIB 目标库管理器.....	168
模数转换器。参见 A/D	
模寻址.....	38
递增缓冲区操作示例.....	39
起始和结束地址.....	39
W 寻址寄存器选择.....	39
应用.....	40
N	
NVM	
寄存器映射.....	47
内部时钟时序示例.....	183
内核架构	
概述.....	15
O	
OC/PWM 模块时序特性.....	191
P	
POR。参见上电复位。	
Q	
器件概述.....	11
器件 ID 存储单元.....	141
器件配置	
寄存器映射.....	158
器件配置寄存器	
FBORPOR.....	156
FGS.....	156
FICD.....	156
FOSC.....	156
FWDT.....	156
欠压复位	
时序要求.....	187
特性.....	179
R	
RCON 寄存器状态位含义及其初始化状态, 示例 1.....	154
RCON 寄存器状态位含义及其初始化状态, 示例 2.....	154
软件堆栈指针、帧指针.....	16
软件模拟器 (MPLAB SIM).....	169
S	
SPI 模块.....	99
CPU 空闲模式下工作.....	101
CPU 休眠模式下工作.....	101
从选择同步.....	101
功能描述.....	99
禁止 SDOx.....	99
SPI1 寄存器映射.....	102
时序特性	
从模式 (CKE = 1).....	197, 199
主模式 (CKE = 0).....	195
主模式 (CKE = 1).....	196
时序要求	
从模式 (CKE = 0).....	198
从模式 (CKE = 1).....	200

dsPIC30F3014/4013

主模式 (CKE = 0)	196	多路通道, I ² S 模式	193
主模式 (CKE = 1)	197	带隙起振时间	187
帧 SPI 支持	100	复位	187
字和字节通信	99	I ² C 总线数据 (从模式)	204
闪存程序存储器	43	I ² C 总线数据 (主模式)	201
上电延时定时器		简单 OC/PWM 模式	191
时序特性	186	看门狗定时器	187
时序要求	187	欠压复位	187
时间框图		SPI 模块	
PWM 输出	87	从模式 (CKE = 0)	198
使能并配置 UART		从模式 (CKE = 1)	200
备用 I/O	105	主模式 (CKE = 0)	196
设置数据、奇偶校验以及停止位选项	105	主模式 (CKE = 1)	197
使能 UART	105	上电延时定时器	187
时序框图。参见时序特性。		输出比较模块	190
时序特性		输入捕捉	190
A、B 和 C 类定时器的外部时钟	188	外部时钟	182
CAN 模块 I/O	205	振荡器起振定时器	187
CLKO 和 I/O 时序特性	185	输出比较模块	85
DCI 模块		寄存器映射 dsPIC30F3014	89
AC-Link 模式	194	寄存器映射 dsPIC30F4013	89
多路通道, I ² S 模式	192	时序特性	190
带隙起振时间	187	时序要求	190
复位	186	输出比较休眠模式操作	88
I ² C 总线启动 / 停止位		输出比较中断	88
从模式	203	数据地址空间	30
主模式	201	存储器映射	30
I ² C 总线数据		对齐	32
从模式	203	空间	32
主模式	201	宽度	32
看门狗定时器	186	MCU 和 DSP (MAC 类) 指令示例	31
OC/PWM 模块	191	Near 数据空间	33
SPI 模块		软件堆栈	33
从模式 (CKE = 0)	197	数据 EEPROM 存储器	49
从模式 (CKE = 1)	199	擦除	50
主模式 (CKE = 0)	195	擦除块	50
主模式 (CKE = 1)	196	擦除字	50
上电延时定时器	186	读	49
输出比较模块	190	防止误写的保护措施	52
输入捕捉 (CAPX)	190	写	51
外部时钟	181	写块	51
振荡器起振定时器	186	写校验	52
时序图		数据累加器和加法器 / 减法器	21
CAN 位	116	回写	22
I ² S 接口帧同步	124	舍入逻辑	22
上电延时的超时序 (MCLR 不与 VDD 连接), 示例 1.		数据空间写饱和	23
152		溢出和饱和	21
上电延时的超时序 (MCLR 不与 VDD 连接), 示例 2.		数据转换器接口 (DCI) 模块	121
152		输入变化通知模块	56
上电延时的超时序 (MCLR 连接到 VDD)	152	dsPIC30F3014 寄存器映射 (bit 15-0)	57
帧同步, AC-Link 帧起始	124	输入捕捉 (CAPX) 定时器特性	190
帧同步, 多通道模式	124	输入捕捉定时器要求	190
时序图和规范		输入捕捉模块	81
DC 规范		寄存器映射	83
内部 RC 精度	183	中断	82
时序要求		双重输出比较匹配模式	86
A/D 转换		单脉冲模式	86
低速率	209	T	
A 类定时器外部时钟	188	Timer2/3 模块	71
B 类定时器外部时钟	189	32 位定时器模式	71
CAN 模块 I/O	205	ADC 事件触发器	74
CLKO 和 I/O	185	定时器预分频器	74
C 类定时器外部时钟	189	寄存器映射	75
DCI 模块		门控工作	74
AC-Link 模式	195	休眠模式下工作	74

中断.....	74	X	
Timer4/5 模块.....	77	系统集成.....	141
寄存器映射.....	79	寄存器映射.....	158
Timer2 和 Timer3 选择模式.....	85	休眠和空闲模式下的输入捕捉操作.....	82
Timer1 模块.....		CPU 空闲模式.....	82
中断.....	68	CPU 休眠模式.....	82
Timer1 模块.....	67	Y	
16 位定时器模式.....	67	异常顺序.....	
16 位同步计数器模式.....	67	陷阱源.....	62
寄存器映射.....	69	引脚配置说明.....	13
门控工作.....	68	因特网地址.....	225
实时时钟.....	68	运行时自编程 (RTSP).....	43
振荡器工作.....	68	Z	
中断.....	68	在线串行编程 (ICSP).....	43, 141
休眠模式下的工作.....	68	振荡器.....	
预分频器.....	68	工作模式 (表).....	142
桶形移位器.....	23	控制寄存器.....	147
通用异步收发器 (UART) 模块.....	103	配置.....	144
图.....		初始时钟源选择.....	144
数据对齐.....	32	低功耗 RC 振荡器 (LPRC).....	145
U		故障保护时钟监视器.....	146
UART 工作原理.....		快速 RC 振荡器 (FRC).....	145
空闲模式.....	108	LP 振荡器控制.....	145
休眠模式.....	108	起振定时器 (OST).....	144
UART 模块.....		锁相环 (PLL).....	145
8 位或 9 位数据模式下的接收.....	106	系统概述.....	141
8 位数据模式下的发送.....	105	振荡器起振定时器.....	
9 位数据模式下的发送.....	105	时序特性.....	186
波特率发生器.....	107	时序要求.....	187
地址检测模式.....	107	振荡器选择.....	141
发送缓冲区 (UxTXB).....	105	指令集.....	
发送数据.....	105	综述.....	159
发送中断.....	106	指令寻址模式.....	37
概述.....	103	MAC 指令.....	38
环回模式.....	107	MCU 指令.....	37
接收错误处理.....	106	其他指令.....	38
接收缓冲区 (UxRXB).....	106	文件寄存器指令.....	37
接收缓冲溢出错误 (OERR 位).....	106	支持的基本寻址模式.....	37
接收数据.....	106	传送和累加指令.....	38
接收中断.....	106, 107	中断.....	59
空闲状态.....	107	中断控制器.....	
奇偶校验错误 (PERR).....	107	寄存器映射.....	66
使能并配置.....	105	中断序列.....	63
UART1 寄存器映射.....	109	中断堆栈帧.....	63
UART2 寄存器映射.....	109	中断优先级.....	60
在 CPU 休眠和空闲模式下工作.....	108	陷阱.....	62
帧错误 (FERR).....	107	状态寄存器.....	16
自动波特率支持.....	108	自动时钟延时.....	94
W		10 位寻址时 (STREN = 1).....	94
WWW 地址.....	225	7 位寻址时 (STREN = 1).....	94
外部时钟时序特性.....		发送模式.....	94
A、B 和 C 类型定时器.....	188	接收模式.....	94
外部时钟时序要求.....	182		
A 类定时器.....	188		
B 类定时器.....	189		
C 类定时器.....	189		
外部中断请求.....	64		
外设模块禁止 (PMD) 寄存器.....	157		
位反转寻址.....	40		
示例.....	40		
实现.....	40		
序列表 (16 个实例).....	41		
温度和电压规范.....			
AC.....	181		

dsPIC30F3014/4013

注:

MICROCHIP 网站

Microchip 网站 (www.microchip.com) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的因特网浏览器即可访问。网站提供以下信息：

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及存档软件
- **一般技术支持**——常见问题 (FAQ)、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

变更通知客户服务

Microchip 的变更通知客户服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请登录 Microchip 网站 www.microchip.com。在“支持” (Support) 下，点击“变更通知客户 (Customer Change Notification)”服务后按照注册说明完成注册。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过 <http://microchip.com/support> 获得网上技术支持。

dsPIC30F3014/4013

读者反馈表

我们努力为您提供最佳文档，以确保您能够成功使用 Microchip 产品。如果您对文档的组织、条理性、主题及其他有助于提高文档质量的方面有任何意见或建议，请填写本反馈表并传真给我公司 TRC 经理，传真号码为 86-21-5407-5066。请填写以下信息，并从下面各方面提出您对本文档的意见。

致: TRC 经理 总页数 _____
关于: 读者反馈
发自: 姓名 _____
公司 _____
地址 _____
国家 / 省份 / 城市 / 邮编 _____
电话 (_____) _____ 传真 (_____) _____

应用 (选填):

您希望收到回复吗? 是____ 否____

器件: dsPIC30F3014/4013 文献编号: DS70138G_CN

问题

1. 本文档中哪些部分最有特色?

2. 本文档是否满足了您的软硬件开发要求? 如何满足的?

3. 您认为本文档的组织结构便于理解吗? 如果不便于理解, 那么问题何在?

4. 您认为本文档应该添加哪些内容以改善其结构和主题?

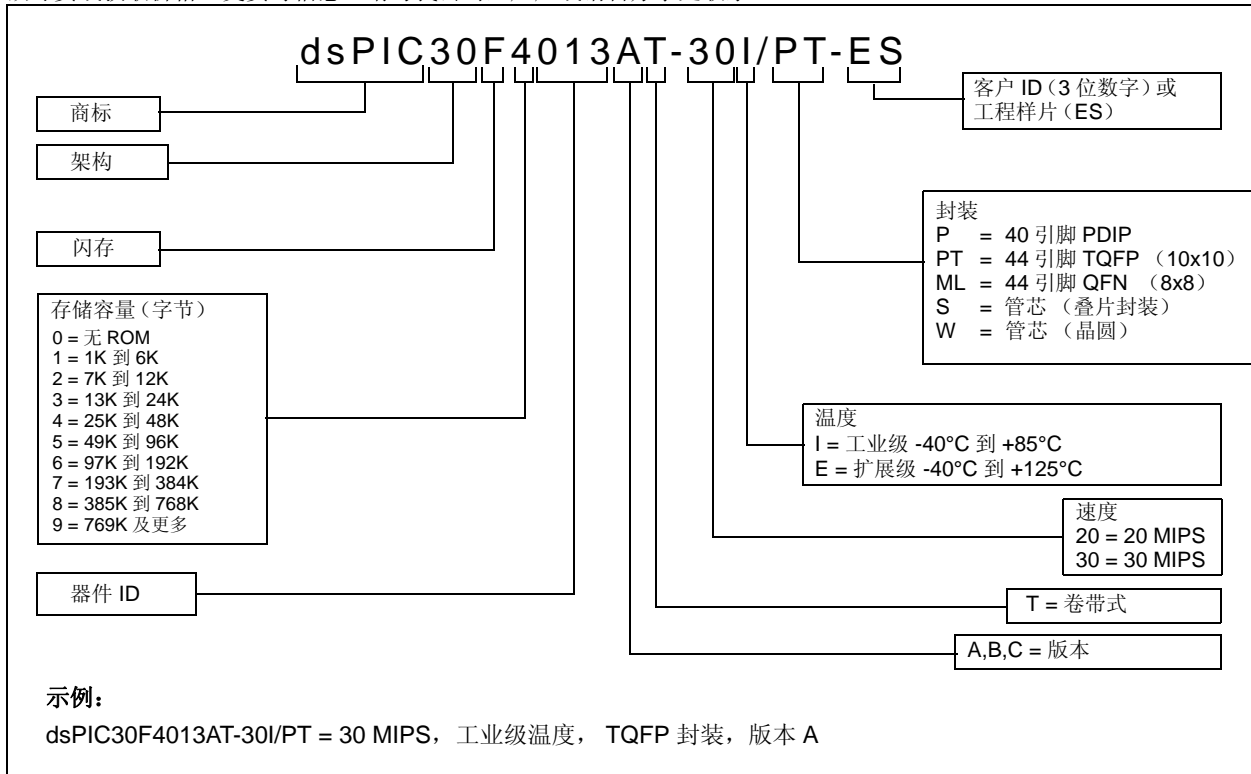
5. 您认为本文档中可以删减哪些内容, 而又不会影响整体使用效果?

6. 本文档中是否存在错误或误导信息? 如果存在, 请指出是什么信息及其具体页数。

7. 您认为本文档还有哪些方面有待改进?

产品标识体系

欲订货或获取价格、交货等信息，请与我公司生产厂或销售办事处联系。



全球销售及及服务网点

美洲

公司总部 **Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277

技术支持:

<http://www.microchip.com/support>

网址: www.microchip.com

亚特兰大 Atlanta

Duluth, GA
Tel: 1-678-957-9614
Fax: 1-678-957-1455

波士顿 Boston

Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago

Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

克里夫兰 Cleveland

Independence, OH
Tel: 1-216-447-0464
Fax: 1-216-447-0643

达拉斯 Dallas

Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit

Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

印第安纳波利斯 Indianapolis

Noblesville, IN
Tel: 1-317-773-8323
Fax: 1-317-773-5453

洛杉矶 Los Angeles

Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara

Santa Clara, CA
Tel: 1-408-961-6444
Fax: 1-408-961-6445

加拿大多伦多 Toronto

Mississauga, Ontario,
Canada
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 Asia Pacific Office

Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 北京

Tel: 86-10-8528-2100
Fax: 86-10-8528-2104

中国 - 成都

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 重庆

Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

中国 - 香港特别行政区

Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 南京

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳

Tel: 86-755-8203-2660
Fax: 86-755-8203-1760

中国 - 武汉

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 西安

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 厦门

Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 - 珠海

Tel: 86-756-321-0040
Fax: 86-756-321-0049

台湾地区 - 高雄

Tel: 886-7-213-7830
Fax: 886-7-330-9305

台湾地区 - 台北

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

亚太地区

台湾地区 - 新竹

Tel: 886-3-6578-300
Fax: 886-3-6578-370

澳大利亚 Australia - Sydney

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore

Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

印度 India - New Delhi

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune

Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama

Tel: 81-45-471-6166
Fax: 81-45-471-6122

韩国 Korea - Daegu

Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul

Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang

Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 Philippines - Manila

Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore

Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok

Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels

Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark - Copenhagen

Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris

Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Munich

Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan

Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Druenen

Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid

Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham

Tel: 44-118-921-5869
Fax: 44-118-921-5820