

深圳市粤原点科技有限公司

(Microchip Authorized Design Partner)指定授权

总部地址：深圳市福田区福虹路世贸广场C座1103座

Add：Room 1103,Block C,World Trade Plaza,

9Fuhong Road,Futian District Shen Zhen City

电话(tel)：86-755-83666321,83666320,83666325

传真(fax)：86-755-83666329

Web: WWW.ORIGIN-GD.COM

E-mail：01@LZmcu.com abc85185@163.com

联系人：马先生,王小姐,汤小姐

在线咨询：QQ:42513912 MSN:action_tech@hotmail.com

7x24小时在线产品咨询:13509674380 13798484366



MICROCHIP

dsPIC30F6010A/6015

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最安全的产品之一。
- 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之中。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、适销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、Accuron、dsPIC、KEELOQ、KEELOQ 徽标、microID、MPLAB、PIC、PICmicro、PICSTART、PRO MATE、rPIC 和 SmartShunt 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

AmpLab、FilterLab、Linear Active Thermistor、Migratable Memory、MXDEV、MXLAB、SEEVAL、SmartSensor 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、ECAN、ECONOMONITOR、FanSense、FlexROM、fuzzyLAB、In-Circuit Serial Programming、ICSP、ICEPIC、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、PICkit、PICDEM、PICDEM.net、PICLAB、PICtail、PowerCal、PowerInfo、PowerMate、PowerTool、REAL ICE、rFLAB、Select Mode、Smart Serial、SmartTel、Total Endurance、UNI/O、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。

© 2007, Microchip Technology Inc. 版权所有。

**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。

dsPIC30F6010A/6015 增强型闪存 16 位数字信号控制器 (DSC)

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

高性能改进型 RISC CPU：

- 改进的哈佛架构
- 带有灵活寻址模式的优化的 C 语言编译器指令集架构
- 83 条基本指令
- 24 位宽指令，16 位宽数据路径
- 144 KB 片上闪存程序空间（指令字）
- 8 KB 片上数据 RAM
- 4 KB 非易失性数据 EEPROM
- 工作速度最高可达 30 MIPS：
 - DC 至 40 MHz 外部时钟输入
 - 4 MHz-10 MHz 振荡器输入，带 PLL（4 倍频、8 倍频和 16 倍频）
 - 7.37 MHz 内部 RC，带有效的 PLL（4 倍频、8 倍频和 16 倍频）
- 44 个中断源：
 - 5 个外部中断源
 - 每一个中断源具有 8 个用户可选择的中断优先级
 - 4 个处理器陷阱源
- 16 x 16 位工作寄存器阵列

DSP 引擎特性：

- 双数据取操作
- DSP 运算的累加器回写操作
- 模寻址和位反转寻址模式
- 两个具备可选饱和逻辑的 40 位宽累加器
- 17 位 x 17 位单周期硬件小数 / 整数乘法器
- 所有 DSP 指令均为单周期指令
- 在一个周期内可将数据左右移动 16 位

外设特性：

- 高灌 / 拉电流 I/O 引脚：25 mA/25 mA
- 带可编程预分频器的定时器模块：
 - 5 个 16 位定时器 / 计数器；可选择将 16 位定时器配对组成 32 位定时器模块
- 16 位捕捉输入功能
- 16 位比较 / PWM 输出功能
- 3 线 SPI 模块（支持 4 种帧模式）
- I²C™ 模块支持多主器件 / 从模式和 7 位 / 10 位寻址
- 2 个带有 FIFO 缓冲区的 UART 模块
- 2 个符合 2.0B 的 CAN 模块（dsPIC306010A）
- 1 个符合 2.0B 的 CAN 模块（dsPIC306015）

电机控制 PWM 模块特性：

- 8 个 PWM 输出通道：
 - 互补或独立输出模式
 - 边沿对齐模式和中心对齐模式
- 4 个占空比发生器
- 专用时基
- 可编程输出极性
- 互补模式的死区时间控制
- 手动输出控制
- A/D 转换触发器

正交编码器接口模块特性：

- A 相、B 相和索引脉冲输入
- 16 位递增 / 递减位置计数器
- 计数方向状态
- 位置测量（x2 和 x4）模式
- 输入端上的可编程数字噪声滤波器
- 备用 16 位定时器 / 计数器模式
- 位置计数器计满回零 / 下溢时产生中断

dsPIC30F6010A/6015

模拟特性:

- 10 位模数转换器 (Analog-to-Digital Converter, ADC) 具有 4 个采样 / 保持输入:
 - 转换速率为 1 Msps
 - 16 个输入通道
 - 在休眠和空闲模式下可以进行转换
- 可编程欠压复位

单片机的特殊性能:

- 增强型闪存程序存储器:
 - 对于工业级温度范围, 最少擦写次数 1 万次, 典型擦写次数 10 万次。
- 数据 EEPROM 存储器:
 - 对于工业级温度范围, 最少擦写次数 10 万次, 典型擦写次数 100 万次。
- 软件控制下, 可以自行再编程

- 上电复位 (Power-on Reset, POR)、上电延时定时器 (Power-up Timer, PWRT) 以及振荡器起振定时器 (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, 能保证可靠的运行
- 故障保护时钟监视器操作检测时钟故障, 并切换到片上低功耗 RC 振荡器
- 可编程代码保护
- 在线串行编程 (In-Circuit Serial Programming™, ICSP™) 功能
- 可选的功耗管理模式
 - 休眠、空闲和备用时钟模式

CMOS 技术:

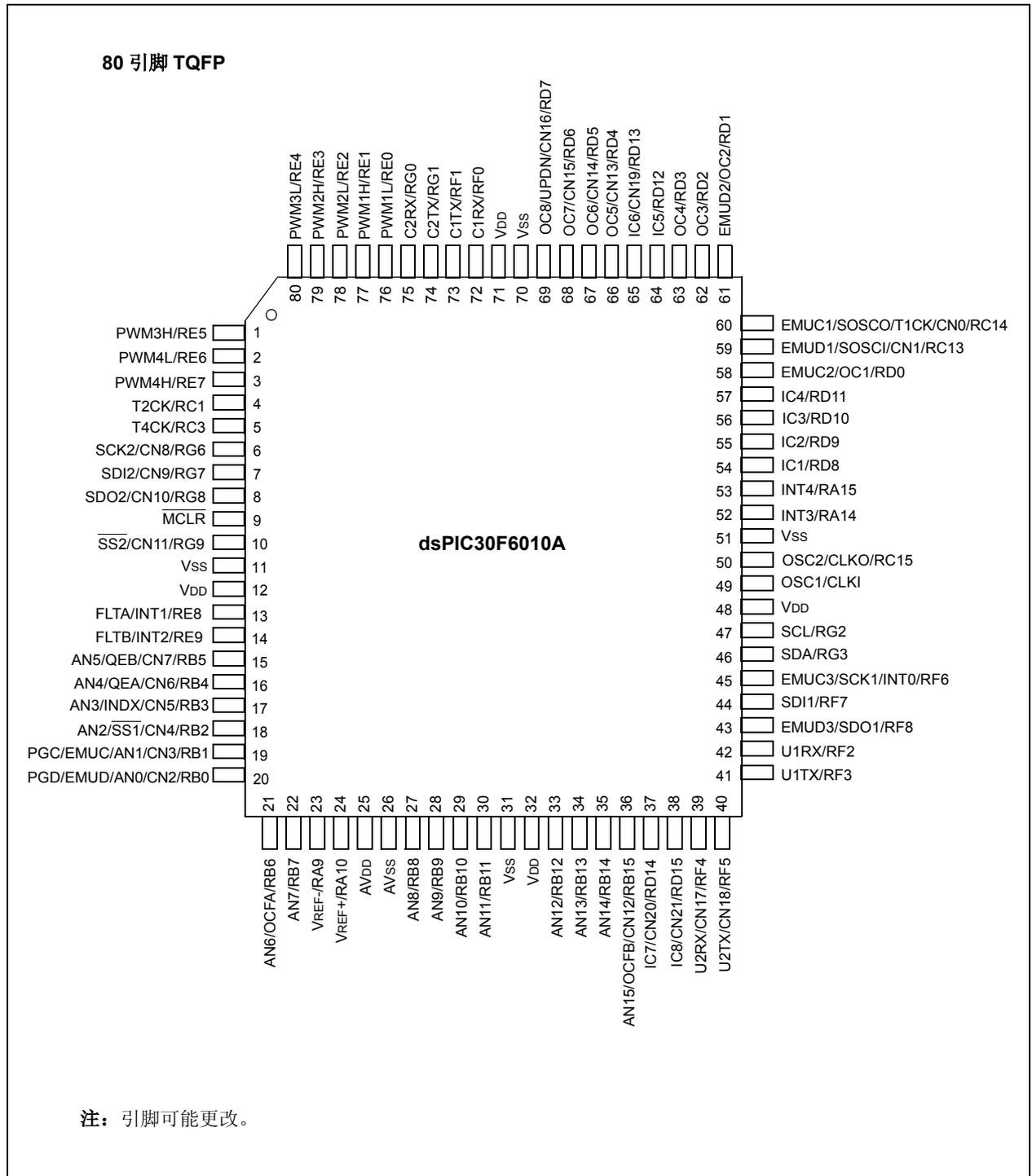
- 低功耗高速度闪存技术
- 宽工作电压范围为 2.5V 到 5.5V
- 工业级温度范围和扩展级温度范围
- 低功耗

dsPIC30F 电机控制和功耗转换系列 *

器件	引脚	程序存储器字节 / 指令	SRAM 字节	EEPROM 字节	16 位定时器	输入捕捉	输出比较 / 标准 PWM	电机控制 PWM	10 位 A/D 转换速率为 1 Msps	正交编码器	UART	SPI	I ² C™	CAN
dsPIC30F2010	28	12K/4K	512	1024	3	4	2	6 通道	6 通道	有	1	1	1	—
dsPIC30F3010	28	24K/8K	1024	1024	5	4	2	6 通道	6 通道	有	1	1	1	—
dsPIC30F4012	28	48K/16K	2048	1024	5	4	2	6 通道	6 通道	有	1	1	1	1
dsPIC30F3011	40/ 44	24K/8K	1024	1024	5	4	4	6 通道	9 通道	有	2	1	1	—
dsPIC30F4011	40/ 44	48K/16K	2048	1024	5	4	4	6 通道	9 通道	有	2	1	1	1
dsPIC30F5015	64	66K/22K	2048	1024	5	4	4	8 通道	16 通道	有	1	2	1	1
dsPIC30F5016	80	66K/22K	2048	1024	5	4	4	8 通道	16 通道	有	1	2	1	1
dsPIC30F6010A	80	144K/48K	8192	4096	5	8	8	8 通道	16 通道	有	2	2	1	2
dsPIC30F6015	64	144K/48K	8192	4096	5	8	8	8 通道	16 通道	有	2	2	1	1

* 该表给出了 dsPIC30F 外设特性的汇总。上表所列的其他 dsPIC30F 电机控制和功耗转换系列器件供特性比较。

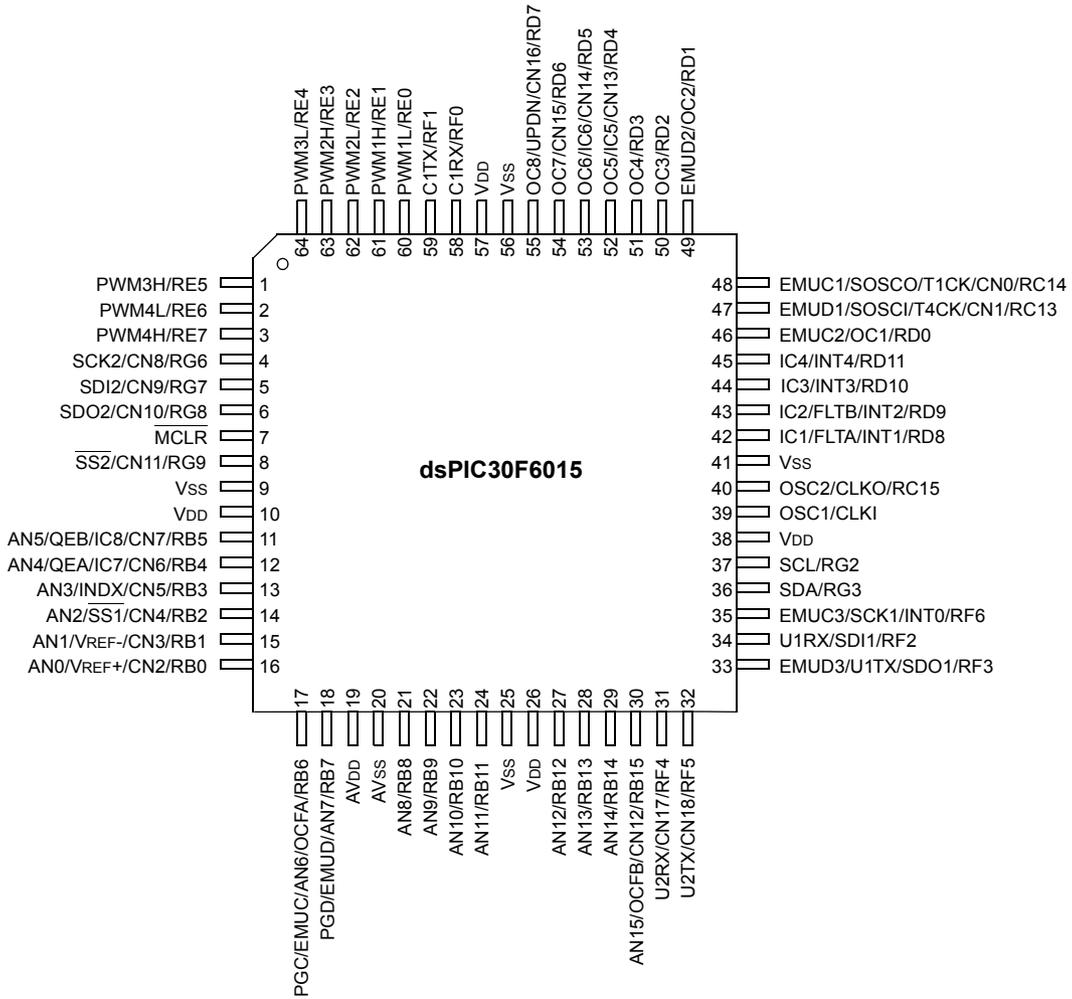
引脚图



dsPIC30F6010A/6015

引脚图

64 引脚 TQFP



注：引脚可能更改。

目录

1.0 器件概述	7
2.0 CPU 架构概述	13
3.0 存储器构成	21
4.0 地址发生器单元	33
5.0 中断	39
6.0 闪存程序存储器	47
7.0 数据 EEPROM 存储器	53
8.0 I/O 端口	57
9.0 Timer1 模块	63
10.0 Timer2/3 模块	67
11.0 Timer4/5 模块	75
12.0 输入捕捉模块	79
13.0 输出比较模块	83
14.0 正交编码器接口 (QE1) 模块	87
15.0 电机控制 PWM 模块	93
16.0 SPI 模块	103
17.0 I2C™ 模块	107
18.0 通用异步收发器 (UART) 模块	115
19.0 CAN 模块	123
20.0 10 位高速模数转换器 (ADC) 模块	135
21.0 系统集成	147
22.0 指令集汇总	163
23.0 开发支持	171
24.0 电气特性	175
25.0 封装信息	215
附录 A: 版本历史	219
附录 B: 器件比较	221
附录 C: 从 dsPIC30F6010 移植到 dsPIC30F6010A	223
索引	225
Microchip 网站	231
变更通知客户服务	231
客户支持	231
读者反馈表	232
产品标识体系	233

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 的网站: <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。

dsPIC30F6010A/6015

注:

1.0 器件概述

注 本数据手册总结了 dsPIC30F 系列器件的功能,但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息,请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息,请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

该数据手册包含 dsPIC30F6010A 和 dsPIC30F6015 器件的特定信息: dsPIC30F 系列器件在高性能 16 位单片机 (MCU) 架构中融合了大量数字信号处理器 (Digital Signal Processor, DSP) 功能。图 1-1 给出了 dsPIC30F6010A 器件的框图。图 1-2 给出了 dsPIC30F6015 器件的框图。

dsPIC30F6010A/6015

图 1-1: dsPIC30F6010A 框图

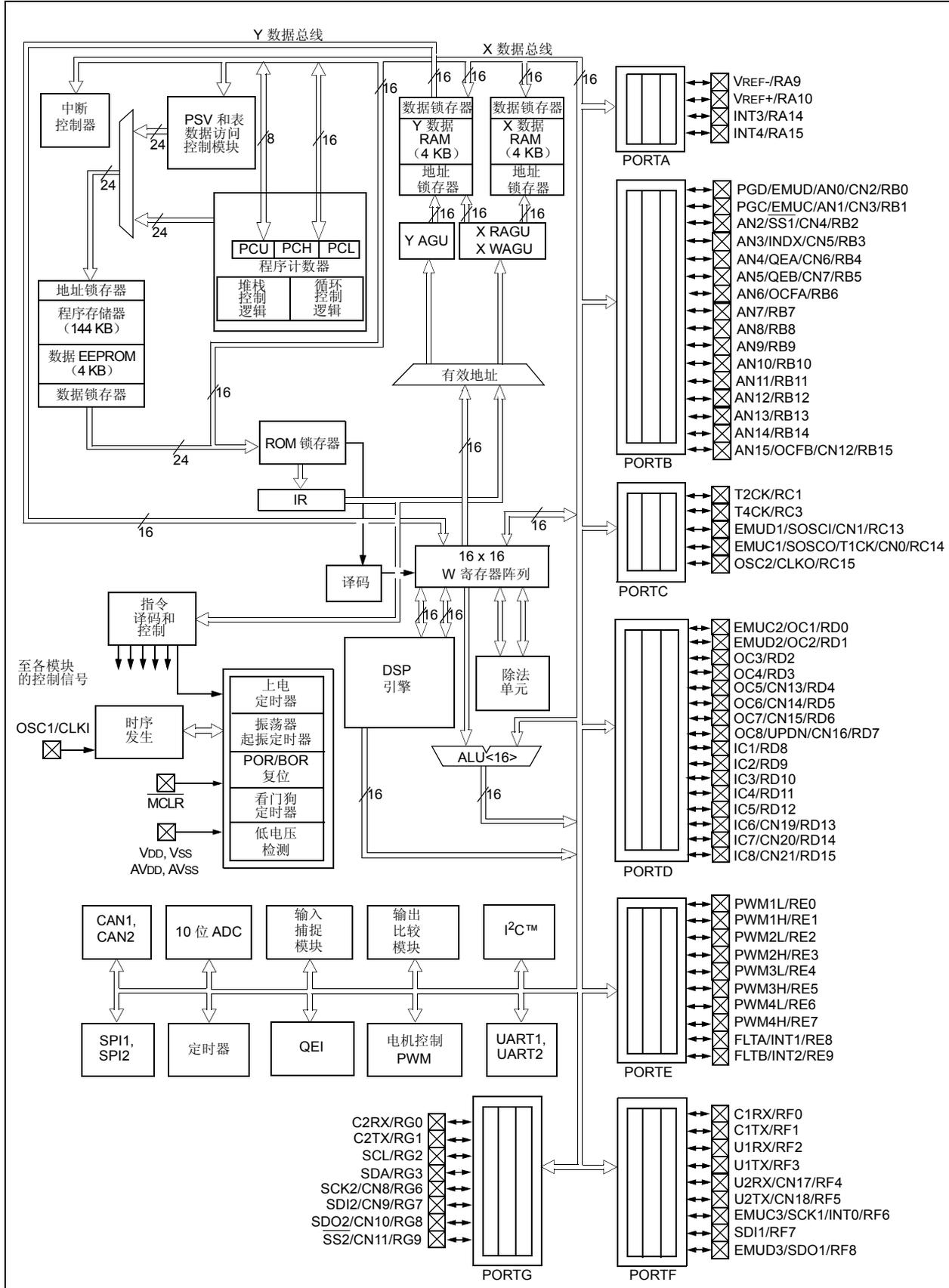
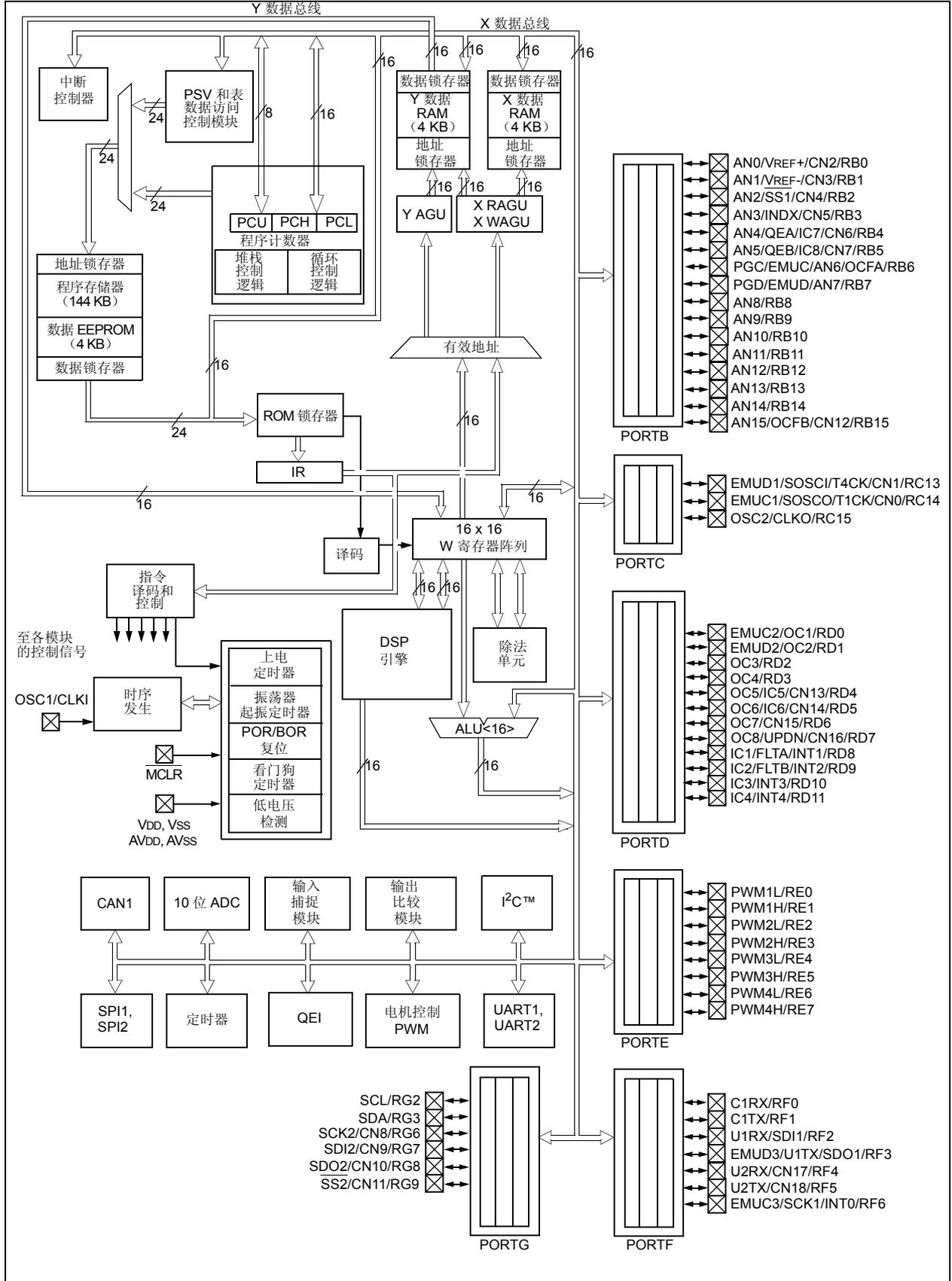


图 1-2: dsPIC30F6015 框图



dsPIC30F6010A/6015

表 1-1 对器件的 I/O 引脚配置和端口引脚的复用功能进行了简要描述。端口引脚可具有多种功能。当发生复用时，外设模块的功能要求可强制改写端口引脚的数据方向。

表 1-1: dsPIC30F6010A/6015 I/O 引脚说明

引脚名称	引脚类型	缓冲器类型	说明
AN0-AN15	I	模拟	模拟输入通道。ANO 和 AN1 还分别用于器件编程数据和时钟输入。
AVDD	P	P	模拟模块正电源。
AVss	P	P	模拟模块的参考地。
CLKI CLKO	I O	ST/CMOS —	外部时钟源输入。总是与 OSC1 引脚功能相关联。 振荡器晶振输出。在晶振模式下，连接到晶振或谐振器。在 RC 和 EC 模式下，可选择作为 CLKO。总是与 OSC2 引脚功能相关联。
CN0-CN23	I	ST	输入状态变化通知输入。可通过软件编程设定所有输入为内部弱上拉。
C1RX C1TX C2RX C2TX	I O I O	ST — ST —	CAN1 总线接收引脚。 CAN1 总线发送引脚。 CAN2 总线接收引脚。 CAN2 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 主通信通道数据输入 / 输出引脚。 ICD 主通信通道时钟输入 / 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-IC8	I	ST	捕捉输入 1 到 8。
INDX QE A QEB UPDN	I I I O	ST ST ST CMOS	正交编码器索引脉冲输入。 在 QE1 模式下为正交编码器 A 相输入。 在定时器模式下为辅助定时器外部时钟 / 门控输入。 在 QE1 模式下为正交编码器 A 相输入。 在定时器模式下为辅助定时器外部时钟 / 门控输入。 递增 / 递减位置计数器方向状态。
INT0 INT1 INT2 INT3 INT4	I I I I I	ST ST ST ST ST	外部中断 0。 外部中断 1。 外部中断 2。 外部中断 3。 外部中断 4。

图注: CMOS = CMOS 兼容输入或输出 模拟 = 模拟输入
 ST = CMOS 电平的施密特触发器输入 O = 输出
 I = 输入 P = 电源

dsPIC30F6010A/6015

表 1-1: dsPIC30F6010A/6015 I/O 引脚说明 (续)

引脚名称	引脚类型	缓冲器类型	说明
U1RX	I	ST	UART1 接收。
U1TX	O	—	UART1 发送。
U1ARX	I	ST	UART1 备用接收。
U1ATX	O	—	UART1 备用发送。
U2RX	I	ST	UART2 接收。
U2TX	O	—	UART2 发送。
VDD	P	—	逻辑电路和 I/O 引脚的正电源。
VSS	P	—	逻辑电路和 I/O 引脚的参考地。
VREF+	I	模拟	模拟参考电压 (高电压) 输入。
VREF-	I	模拟	模拟参考电压 (低电压) 输入。

图注: CMOS = CMOS 兼容输入或输出 模拟 = 模拟输入
 ST = CMOS 电平的施密特触发器输入 O = 输出
 I = 输入 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，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

本章概述了 dsPIC30F6010A/6015 的 CPU 和外设功能。欲知功能的完整描述请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

2.1 内核概述

该系列器件的内核具有 24 位指令字。程序计数器 (PC) 为 23 位宽，且最低有效位 (Least Significant bit, LSb) 总是处于清零状态 (见第 3.1 节“程序地址空间”)。除某些专用指令外，在正常程序执行期间，忽略最高有效位 (Most Significant bit, MSb)。鉴于此，PC 最多可寻址 4M 指令字的用户程序空间。它使用指令预取机制来帮助维持吞吐量。使用 DO 和 REPEAT 指令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 位寄存器组成，每个工作寄存器都可以充当数据、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 (32K 字)，它被分成两块，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单元 (Address Generation Unit, AGU)。大部分指令只通过 X 存储区 AGU 进行操作，这样对外界而言数据空间就是单独而统一的。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对数据进行寻址。

访问程序存储器中的数据的方法有两种：

- 可以选择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 (用户空间)。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访问。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 (Effective Address, EA) 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 操作数的指令，使 $C = A + B$ 操作能在单周期内执行。

内核包含一个 DSP 引擎，从而能够显著增强内核的运算和吞吐能力。DSP 引擎具有一个高速 17 位 × 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，至多可将累加器或任何工作寄存器中的数据左右移动 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。对于 MAC 类指令，这是通过将某些工作寄存器专用于每个地址空间，以透明而灵活的方式实现的。

内核不支持多级指令流水线，它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取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执行完毕。

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。异常由最多 8 个陷阱 (其中 4 个保留) 和 54 个中断组成。根据用户指定的 1 到 7 之间的优先级 (1 为最低优先级，7 为最高优先级)，以及预定义的“自然顺序”，决定每个中断的优先级。陷阱的优先级是固定的，其优先级范围是从 8 到 15。

2.2 编程模型

图 2-1 为编程模型，它包括 16 个 16 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)。工作寄存器可充当数据、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。影子寄存器都是不可直接访问的。如下指令适用于寄存器与影子寄存器间的数据传输。

- PUSH.S 和 POP.S
W0、W1、W2、W3 和 SR (仅限 DC、N、OV、Z 和 C 位) 与对应的影子寄存器之间进行数据传输。
- DO 指令
循环开始时，DOSTART、DOEND 和 DCOUNT 寄存器的内容压入影子寄存器，在循环结束时其内容从各自的影子寄存器中弹出。

当对一个工作寄存器执行字节操作时，仅目的寄存器的低字节会受到影响。但对于存储器映射的工作寄存器来说，可以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，这一点是很有益的。

2.2.1 软件堆栈指针 / 帧指针

dsPIC® DSC 器件具有一个软件堆栈。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： 为了防止出现不对齐的堆栈访问，W15<0> 始终保持为零。

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内的任何单元。

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 (LSB) 被称为 SR 低字节 (SRL)，它的高字节 (MSB) 被称为 SR 高字节 (SRH)。有关 SR 的组成请参见图 2-1。

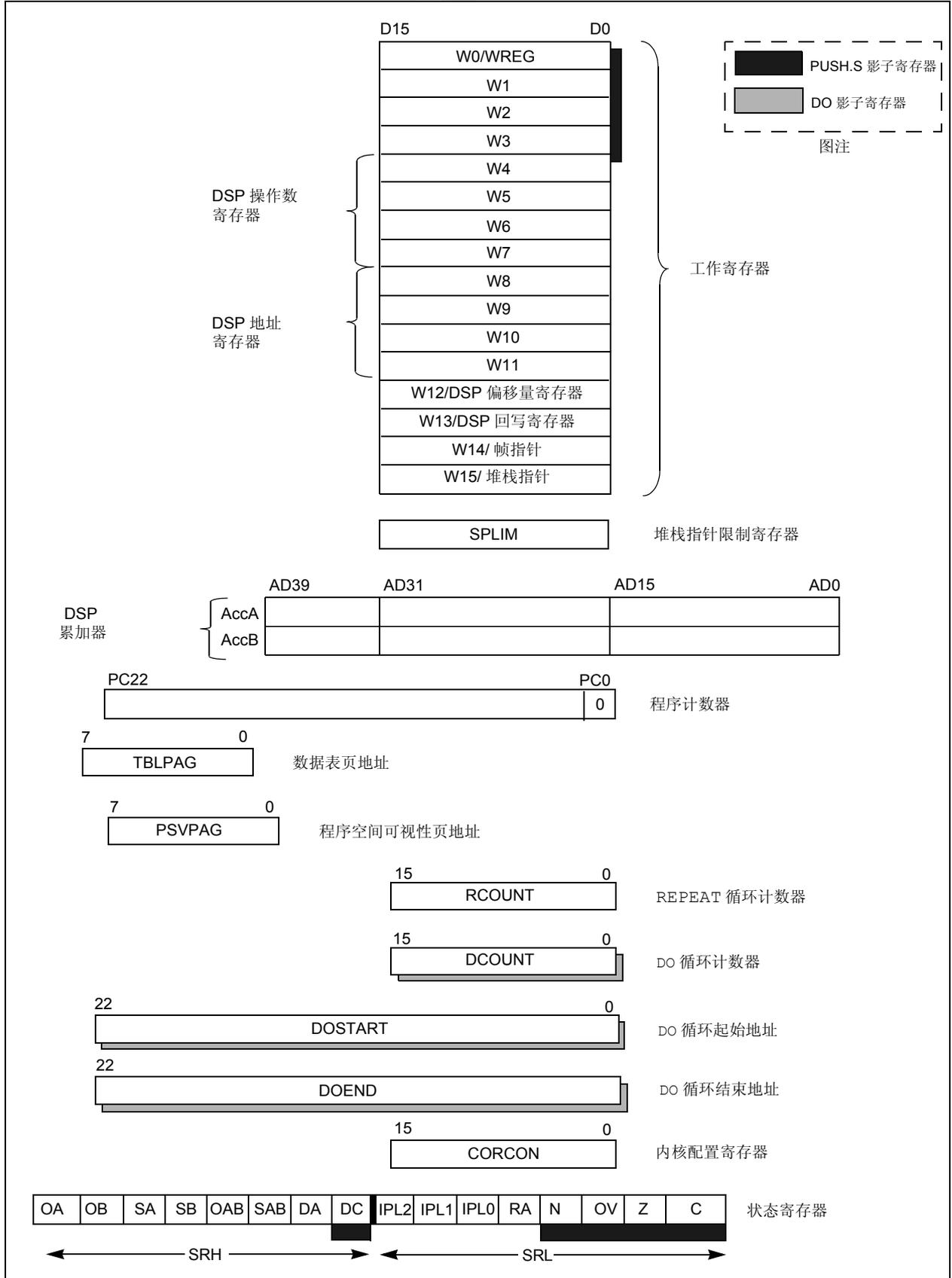
SRL 包含了所有的 MCU ALU 运算状态标志位 (包括 Z 位)，以及 CPU 中断优先级状态位 IPL<2:0> 和循环有效状态位 RA。在异常处理期间，SRL 与 PC 的 MSB 连在一起形成一个完整的字值，然后将该字的值压入堆栈。

SR 寄存器的高字节包含 DSP 加法 / 减法状态位、DO 循环有效位 (DA) 以及半进位 (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: dsPIC30F6010A/6015 编程模型



dsPIC30F6010A/6015

2.3 除法支持

dsPIC DSC 器件支持 16 位 /16 位有符号小数除法运算、32 位 /16 位和 16 位 /16 位有符号和无符号整数除法运算。所有除法运算都是单指令周期内的迭代操作。支持以下指令和数据长度：

1. DIVF —— 16/16 有符号小数除法
2. DIV.sd —— 32/16 有符号除法
3. DIV.ud —— 32/16 无符号除法
4. DIV.s —— 16/16 有符号除法
5. DIV.u —— 16/16 无符号除法

必须在一个 REPEAT 循环周期内执行除法指令。任何其他执行方式（例如一系列不连续的除法指令）都无法得到正确的结果，因为指令流取决于 RCOUNT 的值。除法指令本身无法自动设置 RCOUNT 值，因此必须在 REPEAT 指令中明确且正确给出该值，如表 2-1 所示（REPEAT 将执行目标指令 {操作数的值 + 1} 次）。DIV/DIVF 指令在 REPEAT 循环中的迭代次数应当被设置为 18 次。因此，一个完整的除法运算需要 19 个指令周期。

注： 除法流是可中断的。但用户需要正确地保护现场。

表 2-1: 除法指令

指令	功能
DIVF	有符号小数除法: $Wm/Wn \rightarrow W0$; $Rem \rightarrow W1$
DIV.sd	有符号除法: $(Wm+1:Wm)/Wn \rightarrow W0$; $Rem \rightarrow W1$
DIV.s	有符号除法: $Wm/Wn \rightarrow W0$; $Rem \rightarrow W1$
DIV.ud	无符号除法: $(Wm+1:Wm)/Wn \rightarrow W0$; $Rem \rightarrow W1$
DIV.u	无符号除法: $Wm/Wn \rightarrow W0$; $Rem \rightarrow W1$

2.4 DSP 引擎

DSP 引擎由一个高速 17 位 x 17 位乘法器、一个桶形移位寄存器和一个 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。

dsPIC30F 器件具有单周期指令流，可执行 DSP 或 MCU 指令。DSP 和 MCU 指令共享许多硬件资源。例如，指令集中有共用同一个硬件乘法器的 DSP 和 MCU 乘法指令。

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。

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. 常规或收敛舍入（RND）。
4. AccA 自动饱和和使能 / 禁止（SATA）。
5. AccB 自动饱和和使能 / 禁止（SATB）。
6. 用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）。

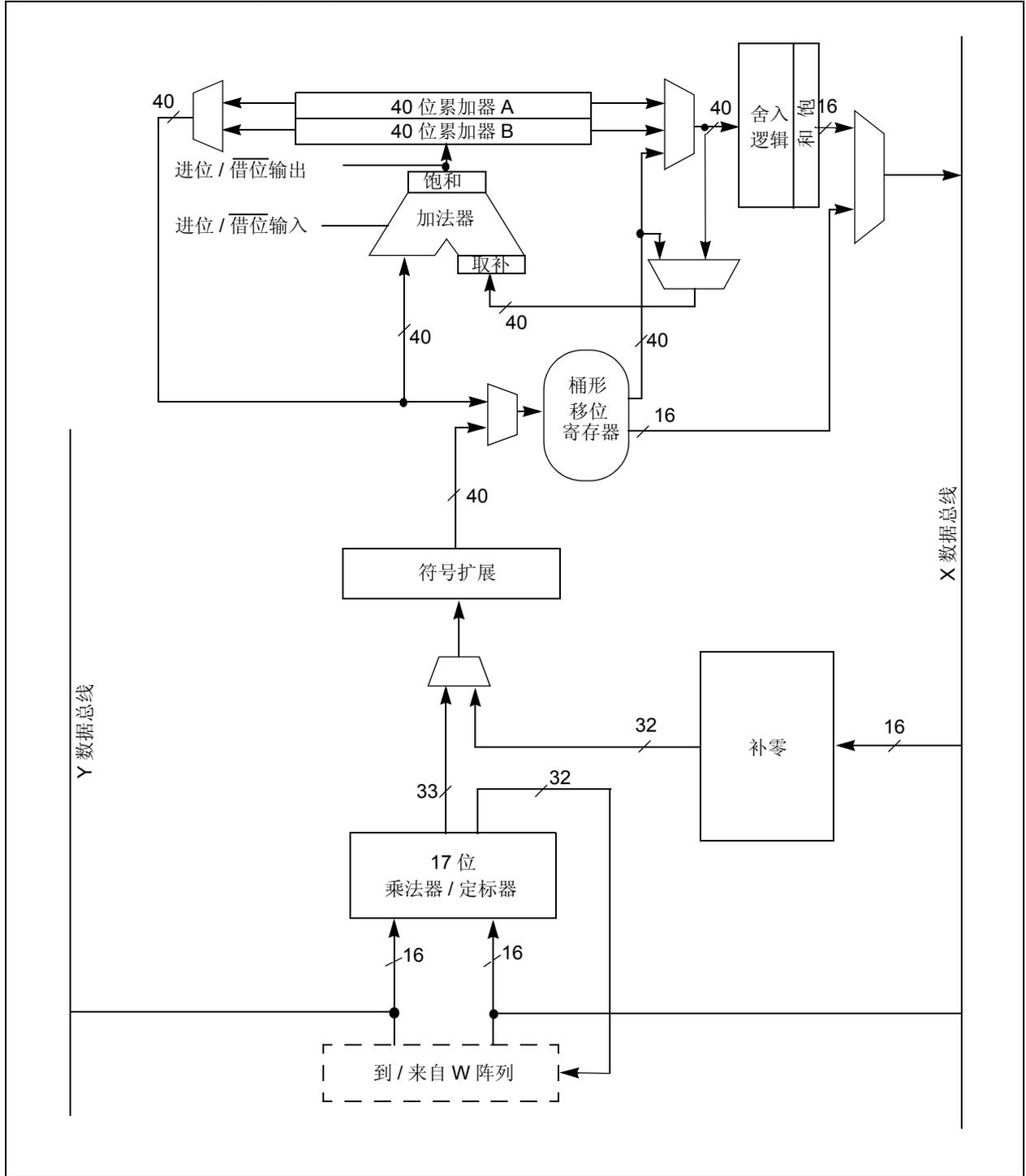
注： 请参见表 3-3 了解 CORCON 的组成。

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
CLR	$A = 0$
ED	$A = (x - y)^2$
EDAC	$A = A + (x - y)^2$
MAC	$A = A + (x * y)$
MOVSAC	A 的值不发生变化
MPY	$A = x * y$
MPY.N	$A = -x * y$
MSC	$A = A - x * y$

图 2-2: DSP 引擎框图



2.4.1 乘法器

17 位 x17 位的乘法器可以进行有符号或无符号的运算，其输出经过定标器进行换算后可支持 1.31 小数 (Q31) 或 32 位整数结果。无符号操作数经过零扩展后，送入乘法器输入值的第 17 位。有符号操作数经过符号扩展后，送入乘法器输入值的第 17 位。17 位 x17 位乘法器 / 定标器的输出是 33 位值，它将被符号扩展为 40 位。整型数据的固有表示形式为有符号的二进制补码值，其中，MSB 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} 至 $2^{N-1} - 1$ 。对于 16 位整数，数据范围是 -32768 (0x8000) 至 32767 (0x7FFF)，包括 0 在内。对于 32 位整数，数据范围是 -2,147,483,648 (0x8000 0000) 至 2,147,483,645 (0x7FFF FFFF)。

当乘法器配置为小数乘法时，数据表示为二进制补码小数，其中 MSB 定义为符号位，小数点暗含在符号位之后 (QX 格式)。暗含小数点的 N 位二进制补码小数的范围是 -1.0 至 $(1-2^{1-N})$ 。对于 16 位小数，Q15 数据范围是 -1.0 (0x8000) 至 0.999969482 (0x7FFF)，包括 0 在内，其精度为 3.01518×10^{-5} 。在小数方式下，16x16 乘法运算将产生 1.31 乘积，其精度为 4.65661×10^{-10} 。

同一个乘法器还用来支持 MCU 乘法指令，包括整数的 16 位有符号、无符号和混和符号乘法。

MUL 指令可以使用字节或字长度的操作数。字节操作数将产生 16 位结果，而字操作数将产生 32 位结果，结果存放在 W 寄存器阵列的指定寄存器中。

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。它可以选择两个累加器 (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换算。

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，一侧输入可以选择为零，而另一侧的输入可以是数据的原码或补码。对于加法，进位 / 借位输入是高有效的，另一侧输入是数据的原码 (没有求补的)；对于减法，进位 / 借位输入是低有效的，另一侧输入是数据的补码。由状态寄存器中的 SA/SB 和 OA/OB 提供加法器 / 减法器的溢出状态：

- 从 bit 39 溢出：这是一种灾难性的溢出，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 5.0 节“中断”)。这使得用户能够立即采取措施，如修正系统增益。

每次数据通过加法器 / 减法器时，SA 和 SB 位就会被修改，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，当饱和被禁止时，SA 和 SB 位将产生算术警告陷阱。

可选择将状态寄存器（SR）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某个累加器是否已溢出或饱和。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 9.31 值（0x7FFFFFFF）或最小的负 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。前提是当前指令不对该累加器进行操作。通过 X 总线将数据写入组合的 X 和 Y 地址空间。支持下列寻址模式：

- 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 小数形式写入 W13 指向的地址。W13 的值递增 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）过程中可以执行常规的（有偏）或收敛的（无偏）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 数据值而只丢弃低位字

常规舍入取累加器 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。此算法的结果经过一系列随机舍入操作，值将稍稍偏大（正偏）。

收敛（或无偏）舍入操作与常规舍入操作相同，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 的值增 1。如果该位为 0，ACCxH 的值不变。假设 bit 16 本身就是一个随机数，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）存储到数据存储空间。（有关数据饱和的信息，请参见第 2.4.2.4 节“数据空间写饱和”）。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受一个来自舍入逻辑模块的 16 位 1.15 小数值作为其输入，还接受来自源（累加器）和 16 位舍入加法器的溢出状态。这些经过组合，用来选择恰当的 1.15 小数值作为输出，写入至数据存储空间中。

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。源累加器的最高位（bit 39）用来决定被检测的操作数的符号。

如果 CORCON 寄存器中的 SATDW 位没有置 1，则输入数据都将通过，在任何情况下都不会被修改。

2.4.3 桶形移位器

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（位的数量）。正数将操作数右移。负数将操作数左移。0 值不改变操作数。

桶形移位寄存器是 40 位宽的，于是，它为 DSP 移位操作提供了 40 位的结果，而为 MCU 移位操作提供 16 位的结果。来自 X 总线的数据在桶形移位寄存器中的存放方式是：右移则数据存放在 bit 16 至 bit 31，左移则存放在 bit 0 至 bit 15。

3.0 存储器构成

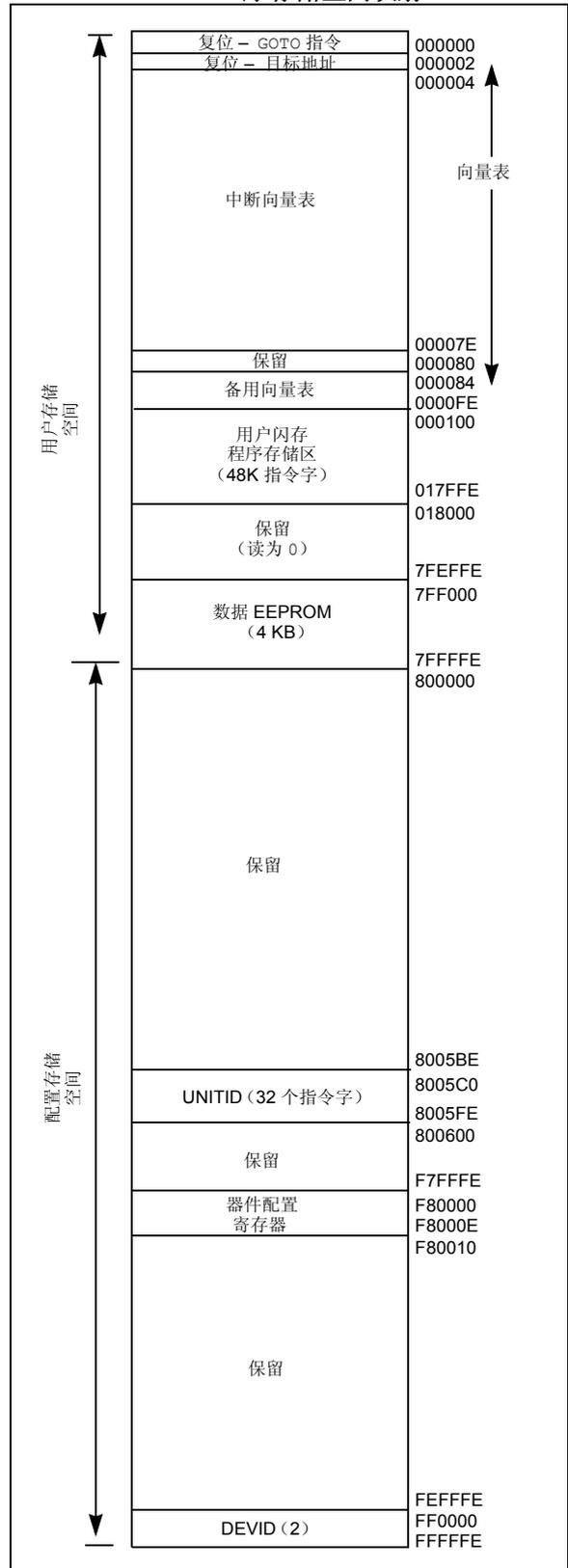
注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

3.1 程序地址空间

程序地址空间为 4M 指令字。如表 3-1 所述，当程序空间映射到数据空间时，可通过 23 位 PC、表指令或数据空间有效地址 (EA) 的 24 位值寻址程序空间。注意，为了提供与数据空间寻址的兼容性，在两个连续的程序字之间，程序空间地址递增量为 2。

对于除 TBLRD/TBLWT 外的所有访问，用户程序空间访问被限制在低 4M 指令字 (地址范围为 0x000000 至 0x7FFFFFFE)；TBLRD/TBLWT 使用 TBLPAG<7>来决定访问用户空间还是配置空间。在表 3-1 读/写指令中，bit 23 置 1 时允许访问器件 ID、用户 ID 和配置位。对于其他情形，bit 23 始终清零。

图 3-1: dsPIC30F6010A/60150 的程序存储空间映射

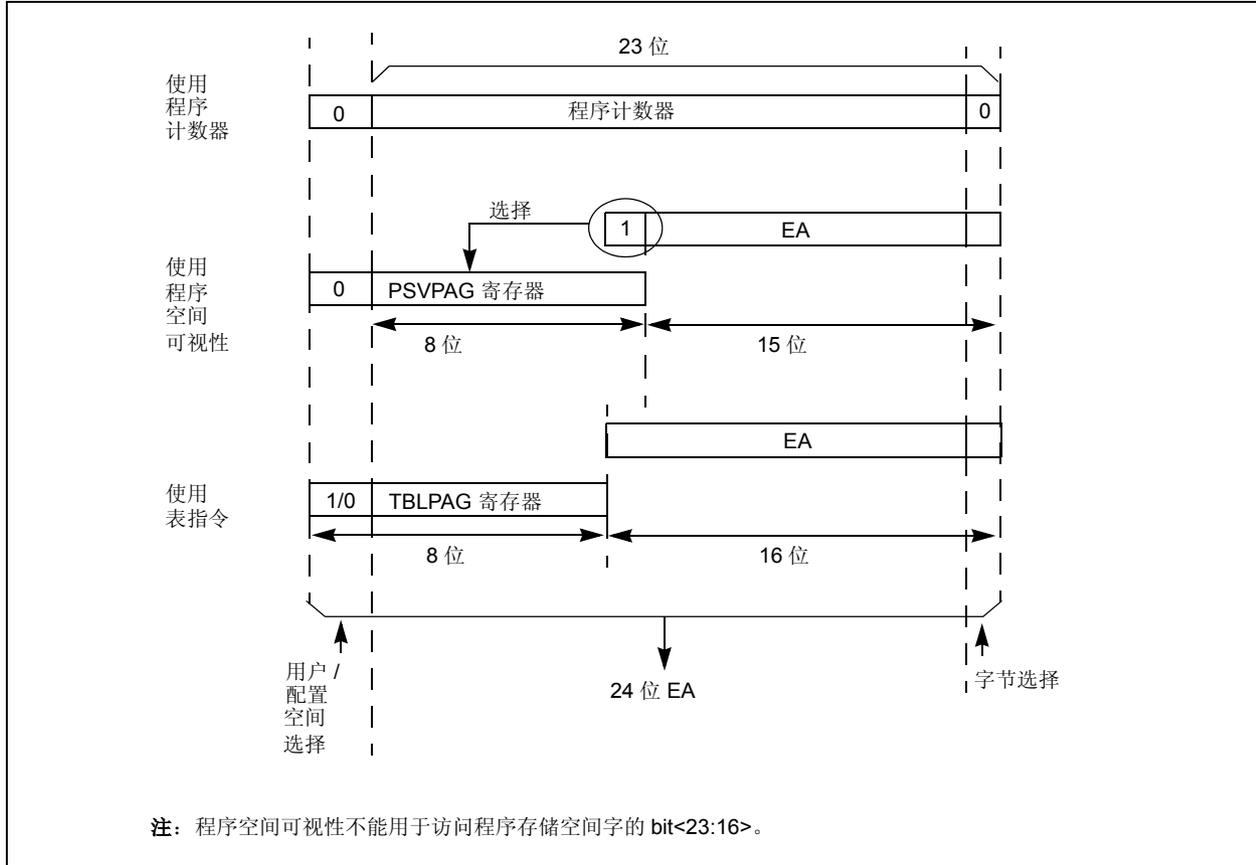


dsPIC30F6010A/6015

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>
使用指令访问	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	用户空间	0	PSVPAG<7:0>	数据 EA<14:0>	

图 3-2: 访问程序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

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

访问程序空间的方法有两种：通过特殊表指令，或通过 16K 字大小的程序空间页重新映射到数据空间的上半部分（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 指令提供了无需通过数据空间，直接读写程序空间任何地址的最低字（lsw）的方法，TBLRDH 和 TBLWTH 指令是可以把一个程序空间字的高 8 位作为数据存取的惟一方法。

对于每个连续的 24 位程序字，PC 的递增量为 2。这使得程序存储区地址能够直接映射到数据空间地址。从而可将程序存储区视作两个 16 位字宽的地址空间，它们并排放置，具有相同的地址范围。TBLRDL 和 TBLWTL 访问包含低位字的空间，TBLRDH 和 TBLWTH 访问包含高位字的空间。

图 3-2 说明了为表操作和数据空间访问创建有效地址（EA）的方法（PSV = 1）。这里 P<23:0> 指的是程序空间字，而 D<15:0> 指的是数据空间字。

提供了一组表指令，可使用它们与程序空间交换字或字节大小的数据。

1. **TBLRDL**: 对低地址部分执行表读操作
字: 读程序地址的低字部分; P<15:0> 映射到 D<15:0>。
字节: 读程序地址的某个较低的字节; 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**: 对低地址部分执行表写操作（有关闪存编程的详细信息见第 6.0 节“闪存程序存储器”）。
3. **TBLRDH**: 对高地址部分执行表读操作
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: 读程序地址的某个较高的字节; 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**: 对高地址部分执行表写操作（有关闪存编程的详细信息见第 6.0 节“闪存程序存储器”）。

图 3-3: 程序数据表访问（最低有效字）

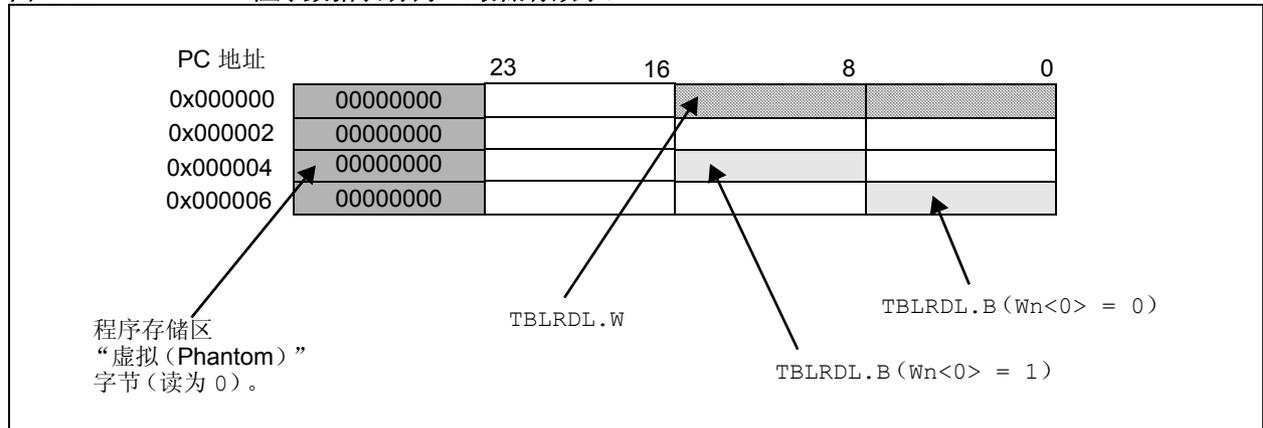
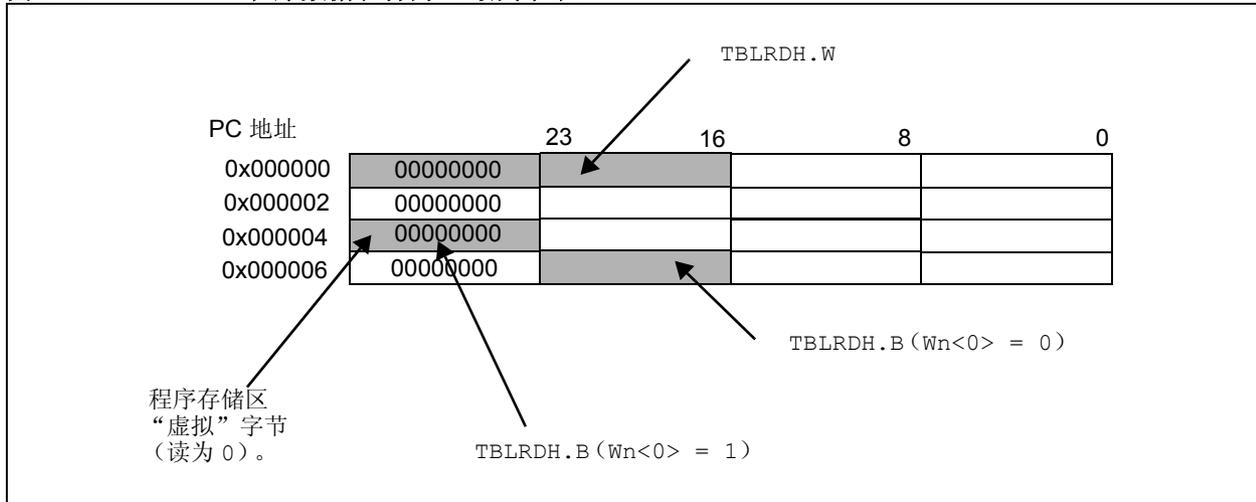


图 3-4: 程序数据表访问 (最高字节)



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 字程序空间页。这提供了通过 X 数据空间对存储的常量数据的透明访问，而无需使用特殊指令（即 TBLRDH/H 和 TBLWTL/H 指令）。

如果数据空间 EA 的最高位置 1，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，就能通过数据空间访问程序空间。第 2.4 节“DSP 引擎”中讨论了 CORCON 的功能。

访问此区域的数据会增加一个额外的指令周期，因为需要进行取两次程序存储字。

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这个存储区域时，Y 数据空间通常应该存放 DSP 操作的状态（变量）数据，而 X 数据空间通常应该存放系数（常量）数据。

尽管每个数据空间地址大于等于 0x8000 的部分直接映射到对应的程序存储器地址单元中（见图 3-6），但只使用 24 位程序字的低 16 位来存储数据。应强制将高 8 位编程为一条无效指令以维持器件的可靠性。有关指令编码的细节，请参见《dsPIC30F/33F 程序员参考手册》（DS70157B_CN）。

注意，每访问一个程序字，PC 就会增 2，数据空间地址的低 15 位直接被映射到相应的程序空间地址的低 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： 在表读 / 写操作时暂时禁止 PSV 访问。

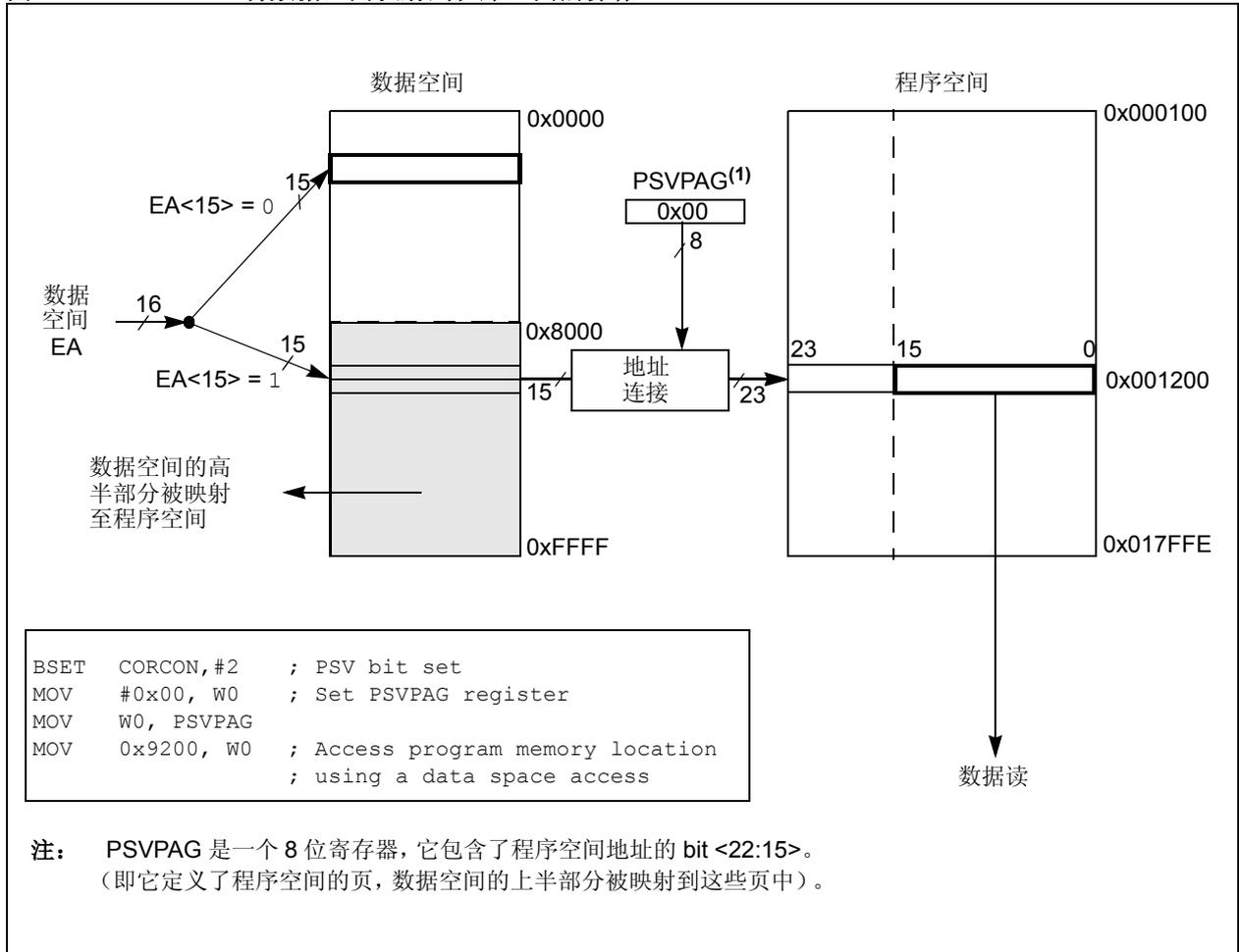
对于在 REPEAT 循环外执行的使用 PSV 的指令：

- 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 3-5: 将数据空间映射到程序空间的操作



3.2 数据地址空间

内核具有两个数据空间。它们可被视作是独立的数据空间（对于某些 DSP 指令），或被视作一个统一的线性地址范围（对于某些 MCU 指令）。可通过两个地址发生单元（Address Generation Units, AGU）和独立的数据路径访问数据空间。

3.2.1 数据存储空间映射

数据存储空间被分成 X 和 Y 数据空间两大块。此架构的关键之处在于，Y 空间是 X 空间的子集且它被完全的包含在 X 空间中。为了提供明显的线性寻址空间，X 和 Y 空间要有连续的地址。

执行 MAC 类指令以外的任何指令时，X 块由 64 KB 数据地址空间（包括了全部的 Y 空间地址）组成。在执行 MAC 类指令时，X 块由不包括 Y 地址（只用于数据读操作）的 64 KB 数据地址空间组成。也就是说，所有其他指令将整个数据存储空间视作一个组合的地址空间。MAC 类指令把 Y 地址空间从数据空间中分离出来，使用来自 W10 和 W11 的 EA 对 Y 空间寻址。使用 W8 和 W9 对剩余的 X 数据空间寻址。只有 MAC 类指令才能同时访问两个地址空间。

数据存储空间映射如图 3-6 所示。

图 3-7 中给出了使用 MCU 和 DSP 指令访问 X 和 Y 数据空间的图形概述。

dsPIC30F6010A/6015

图 3-6: dsPIC30F6010A/6015 数据空间存储器映射

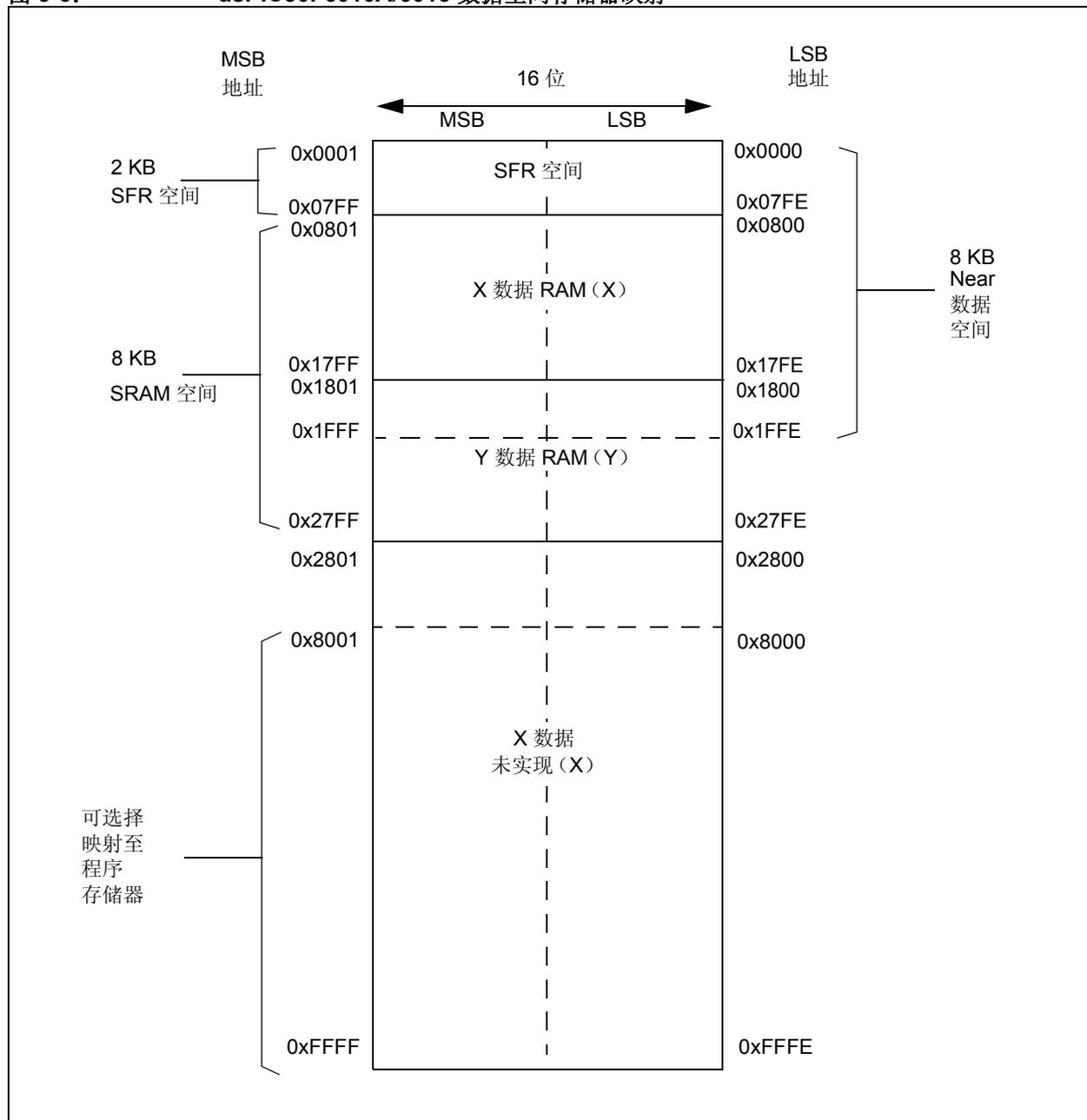
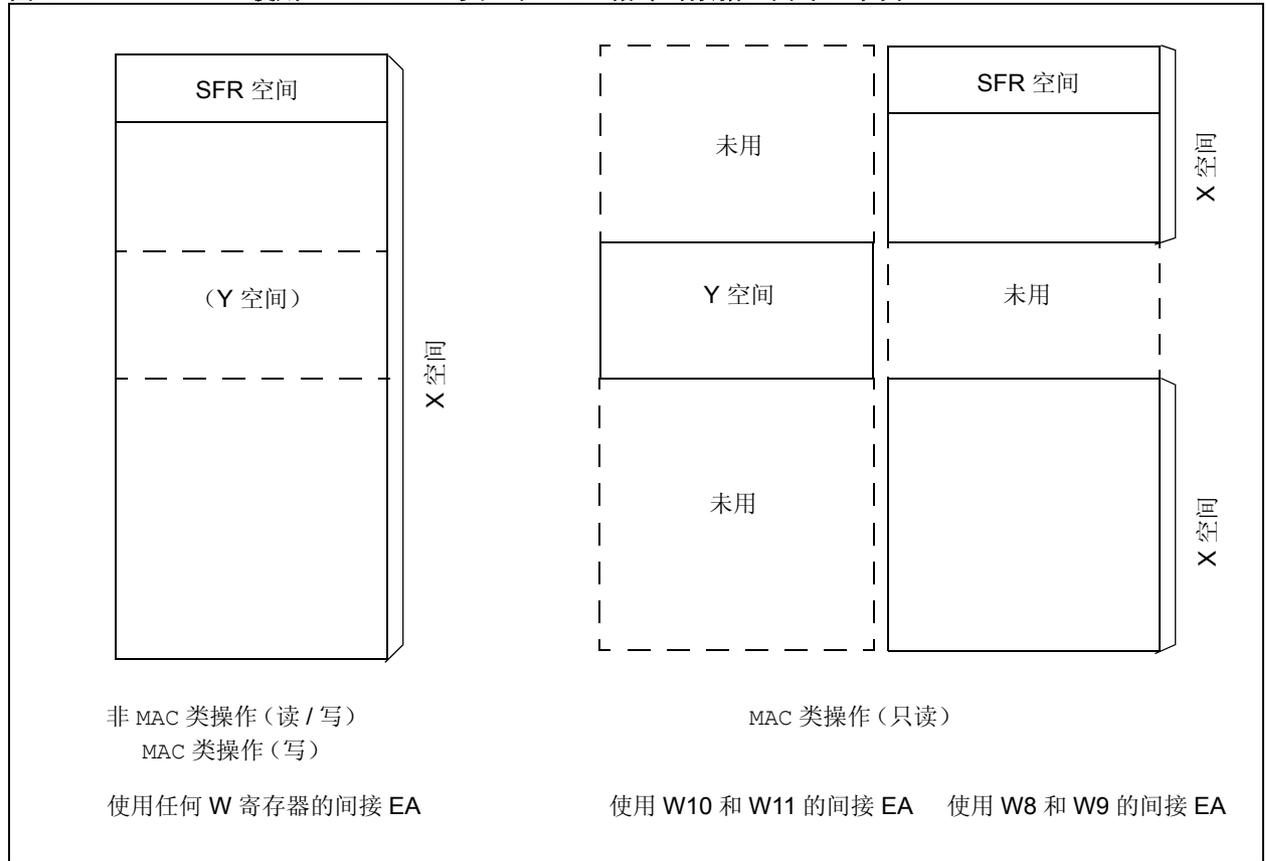


图 3-7: 使用 MCU (MAC 类) 和 DSP 指令对数据空间寻址示例



3.2.2 数据空间

X 数据空间可用于全部指令并支持所有寻址模式。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惟一路径。

X 数据空间还支持所有指令的模寻址，但是会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。

MAC 类指令（CLR、ED、EDAC、MAC、MOVSAC、MPY、MPY.N 和 MSC）使用 Y 数据空间与 X 数据空间配合以提供两条可以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 数据空间仅用于与 MAC 类指令相关的数据预取操作。它也支持用于自动循环缓冲的模寻址。当然，所有其他指令可通过将 Y 数据地址空间视作组合线性空间的一部分，通过 X 数据总线对其进行访问。

图 3-6 给出了 X 和 Y 数据空间之间的边界定义，用户不能通过编程更改 X 和 Y 的数据空间边界。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
在 MAC 类指令中使用 W8 或 W9 访问 Y 数据空间	0x0000
在 MAC 类指令中使用 W10 或 W11 访问 X 数据空间	0x0000。

所有有效地址都是 16 位宽的，且指向数据空间内的字节单元。所以，数据空间地址范围是 64 KB 或 32K 字。

3.2.3 数据空间宽度

内核数据宽度为 16 位。所有内部寄存器都是以 16 位宽的字构成的。数据存储空间是由 16 位宽的字节可寻址块构成的。

3.2.4 数据对齐

为了保持与 PIC[®] 器件的向后兼容性，并提高数据存储空间的使用效率，dsPIC30F 指令集既支持字操作，也支持字节操作。在数据存储器和寄存器中，数据是按字对齐的，但所有数据空间有效地址都被解析为字节。读数据字节的操作将读取包含此字节的整个字，使用有效地址的最低位（LSb）决定要选择的字节，选中的字节将被放在 X 数据总线的 LSB 中（因为 MAC 类指令只能进行取字操作，因此不能通过 Y 数据总线进行字节访问）。也就是说，数据存储器和寄存器是由两个共享（字）地址译码，而写入线相互独立的字节宽度的并行实体构成的。数据字节写操作仅写入存储阵列和寄存器中与字节地址匹配的相应部分。

这类字节访问要求所有的有效地址计算（包含那些只能处理字大小数据的 DSP 操作生成的地址）在内部进行调整以逐次访问字对齐的存储器。例如，对于后修改寄存器间接寻址模式 [Ws++]，字节操作时，内核将其识别为值 Ws + 1，而字操作时，内核将其识别为 Ws + 2。

所有字访问必须按偶地址对齐。不支持取非对齐的字数据的操作，因此在混合使用字节和字的操作或转换 8 位 MCU 代码时必须要小心。任何尝试进行非对齐读或写操作都将产生一个地址错误陷阱。如果在读操作时产生错误，正在执行的指令将完成；如果在写操作时产生错误，则执行指令，但不会执行写操作。无论上述 2 种情况的哪一种，都会随之执行陷阱操作。从而允许系统和 / 或用户检查地址错误发生之前的机器状态。

图 3-8: 数据对齐



所有载入W寄存器的字节都将载入W寄存器的低字节，W寄存器的高字节不变。

提供符号扩展指令（SE）以允许用户将8位有符号数据转换成16位有符号值。或者，对于16位无符号数据，通过在相应的地址执行零扩展指令（ZE），用户可清零W寄存器的高字节。

尽管大多数指令能够对字或字节数据进行操作，但必须注意的是，某些指令（包括DSP指令）只能对字进行操作。

3.2.5 NEAR 数据空间

在X地址存储空间中的0x0000至0x1FFF保留了一个8KB的near数据空间；所有存储器直接寻址指令都可以通过一个13位的绝对地址来直接访问这个数据空间。其余的X地址空间和全部的Y地址空间都是可间接寻址的。此外，使用MOV指令可以寻址整个X数据空间，它支持通过16位字段进行存储器直接寻址。

3.2.6 软件堆栈

dsPIC DSC 器件具备一个软件堆栈。W15被用作堆栈指针。

堆栈指针总是指向第一个可用的空字，并从低地址到高地址方向递增。如图3-9所示，出栈操作时该指针预减，压栈操作时该指针预加。注意，对于任何CALL指令时的PC压栈操作，PC的高字节部分在压栈前要进行零扩展以确保高字节总是处于清零状态

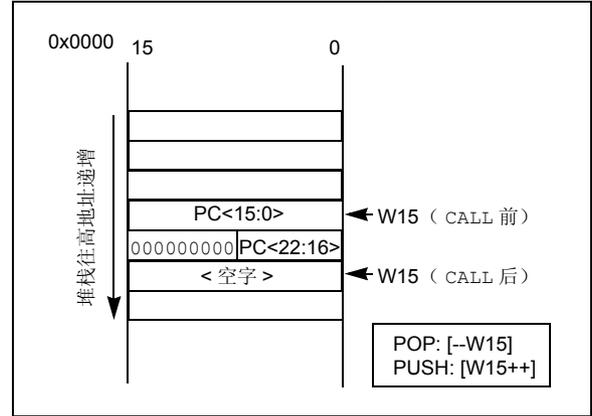
注： 在异常事件处理期间的压栈操作，首先要先将PC的高字节与SRL寄存器组合在一起。

与堆栈指针相关联的寄存器是堆栈指针限制寄存器（SPLIM）。复位时不会初始化SPLIM。与堆栈指针的情况一样，因为所有的堆栈操作必须是字对齐的，SPLIM<0>被强制置0。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）与SPLIM寄存器的内容相等，则执行压栈操作，此时不会产生堆栈错误陷阱。但在随后的压栈操作中会产生堆栈错误陷阱。例如，如果需要在堆栈递增至超过RAM的0x2000地址时产生堆栈错误，应使用0x1FFE初始化SPLIM。

类似地，当堆栈指针地址小于0x0800时，就会产生堆栈指针下溢（堆栈错误）陷阱，这避免了堆栈与特殊功能寄存器（SFR）空间重叠。

在对SPLIM寄存器执行写操作之后，不要立即使用W15对该寄存器进行间接读操作。

图 3-9: CALL 堆栈帧



3.2.7 数据 RAM 保护功能

dsPIC30F6010A/6015 支持数据 RAM 保护功能，从而当 RAM 与引导和安全代码段一起使用时，能够保证 RAM 段的安全。使能 RAM 保护功能时，只能通过引导段闪存代码访问 BSRAM（BS 的安全 RAM 段），只能通过安全段闪存代码访问 SSRAM（RAM 的安全 RAM 段）。

表 3-3 为 BSRAM 和 SSRAM 的 SFR 概述。

表 3-3: 内核寄存器映射

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 uuuu0
DOSTARTH	003C	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOSTARTH	0000 0000 0uuu uuuu
DOENDL	003E	DOENDL																0	uuuu uuuu uuuu uuuu0
DOENDH	0040	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DOENDH	0000 0000 0uuu uuuu

图注: u = 未初始化的位

注: 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 (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
SR	0042	OA	OB	SA	SB	OAB	SAB	DA	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000 0000 0000 0000
CORCON	0044	—	—	—	US	EDT	DL2	DL1	DL0	SATA	SATB	SATDW	ACCSAT	IPL3	PSV	RND	IF	0000 0000 0010 0000
MODCON	0046	XMODEN	YMODE N	—	—	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
BSRAM	0750	—	—	—	—	—	—	—	—	—	—	—	—	—	IW_BSR	IR_BSR	RL_BSR	0000 0000 0000 0000
SSRAM	0752	—	—	—	—	—	—	—	—	—	—	—	—	—	IW_SSR	IR_SSR	RL_SSR	0000 0000 0000 0000

图注: u = 未初始化的位

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

dsPIC30F6010A/6015

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，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

dsPIC DSC 内核包含两个独立的地址发生器单元 (AGU)：分别是 X AGU 和 Y AGU。Y AGU 仅支持 DSP MAC 类指令的字大小数据读取。dsPIC DSC AGU 支持以下三种数据寻址类型：

- 线性寻址
- 模（循环）寻址
- 位反转寻址

线性 and 模数据寻址模式可应用于数据空间或程序空间。位反转寻址模式只能用于数据空间地址。

4.1 指令寻址模式

寻址模式经过优化可以支持各指令的具体功能，具体的寻址模式在表 4-1 中给出。MAC 类指令中提供的寻址模式，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

大多数文件寄存器指令使用一个 13 位地址字段 (f) 来直接寻址数据存储器中的前 8192 个字节 (Near 数据空间)。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者 WREG (MUL 指令除外)，将结果写入寄存器或寄存器对。使用 MOV 指令能够获得更大的灵活性，在文件寄存器工作期间可以访问整个数据空间。

4.1.2 MCU 指令

三操作数 MCU 指令的形式是：

操作数 3 = 操作数 1 < 功能 > 操作数 2

其中，操作数 1 始终是称为 Wb 的工作寄存器（即，寻址模式只能是寄存器直接寻址）。操作数 2 可以是一个 W 寄存器，取自数据存储器或为一个 5 位立即数。结果位置可以是 W 寄存器或地址单元。MCU 指令支持下列寻址模式：

- 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面给出的全部寻址模式。某些指令可能只支持这些寻址模式中的某些模式，指令不同支持的寻址模式可能不同。

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	直接访问寄存器的内容。
寄存器间接寻址	Wn 的内容形成 EA。
执行后修改的寄存器间接寻址	Wn 的内容形成 EA，然后用一个常量值来修改 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 形成 EA。
带寄存器偏移量的寄存器间接寻址	Wn 和 Wb 的和形成 EA。
带立即数偏移量的寄存器间接寻址	Wn 和立即数的和形成 EA。

dsPIC30F6010A/6015

4.1.3 传送指令和累加器指令

与其他指令相比，传送指令和 DSP 累加器类指令提供了更为灵活的寻址模式。除了大多数 MCU 指令支持的寻址模式以外，传送和累加器指令还支持寄存器偏移量的寄存器间接寻址模式，这也叫做寄存器变址寻址模式。

注： 对于 MOV 指令，指令中指定的源寄存器和目的寄存器有效地址的寻址模式可以不同。然而，4 位 Wb（寄存器偏移量）字段供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，传送指令和累加器指令支持下列寻址模式：

- 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
- 带寄存器偏移量的寄存器间接寻址（变址寻址）
- 带立即数偏移量的寄存器间接寻址
- 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面给出的全部寻址模式。某些指令可能只支持这些寻址模式中的某些模式，指令不同支持的寻址模式可能不同。

4.1.4 MAC 指令

双源操作数 DSP 指令（CLR, ED, EDAC, MAC, MPY, MPY.N, MOV SAC 和 MSC），也称 MAC 指令，它们使用一组简化的寻址模式，允许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集合 {W8, W9, W10, W11} 的成员。对于数据读取，W8 和 W9 总是分配给 X RAGU，而 W10 和 W11 则始终用于 Y AGU。从而，产生的有效地址（无论是在修改之前还是之后），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，仅可用于 W9（在 X 空间中）和 W11（在 Y 空间中）。

概括地说，MAC 类指令支持下列寻址模式：

- 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
- 带寄存器偏移量的间接寻址（变址寻址）

4.1.5 其他指令

除了上述的各种寻址模式之外，一些指令使用各种大小的立即数。例如，BRA（转移）指令使用 16 位有符号立即数来直接指定转移的目标，而 DISI 指令使用一个 14 位无符号立即数字段。在一些指令中，比如 ADD Acc，操作数源和运算结果已经暗含在操作码中。某些操作，比如 NOP，没有任何操作数。

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。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（因为这两种空间的数据指针机制本质上是相同的）。每个 X（也提供指向程序空间的指针）和 Y 数据空间中都可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。

总的来说，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地址和结束地址判据，它们可以双向工作（即，在低地址边界和高地址边界上都可以进行地址边界检查）。

4.2.1 起始地址和结束地址

模寻址机制要求指定起始地址和结束地址，并把它们载入 16 位模缓冲区地址寄存器：XMODSRT, XMODEND, YMODSRT 和 YMODEND（见表 3-3）。

注： Y 空间模寻址的 EA 计算使用字大小数据（每个 EA 的 LSb 始终为零）。

循环缓冲区的长度没有直接指定，由相应的起始地址、结束地址之差决定其长度。循环缓冲区最大长度为 32K 字（64 KB）。

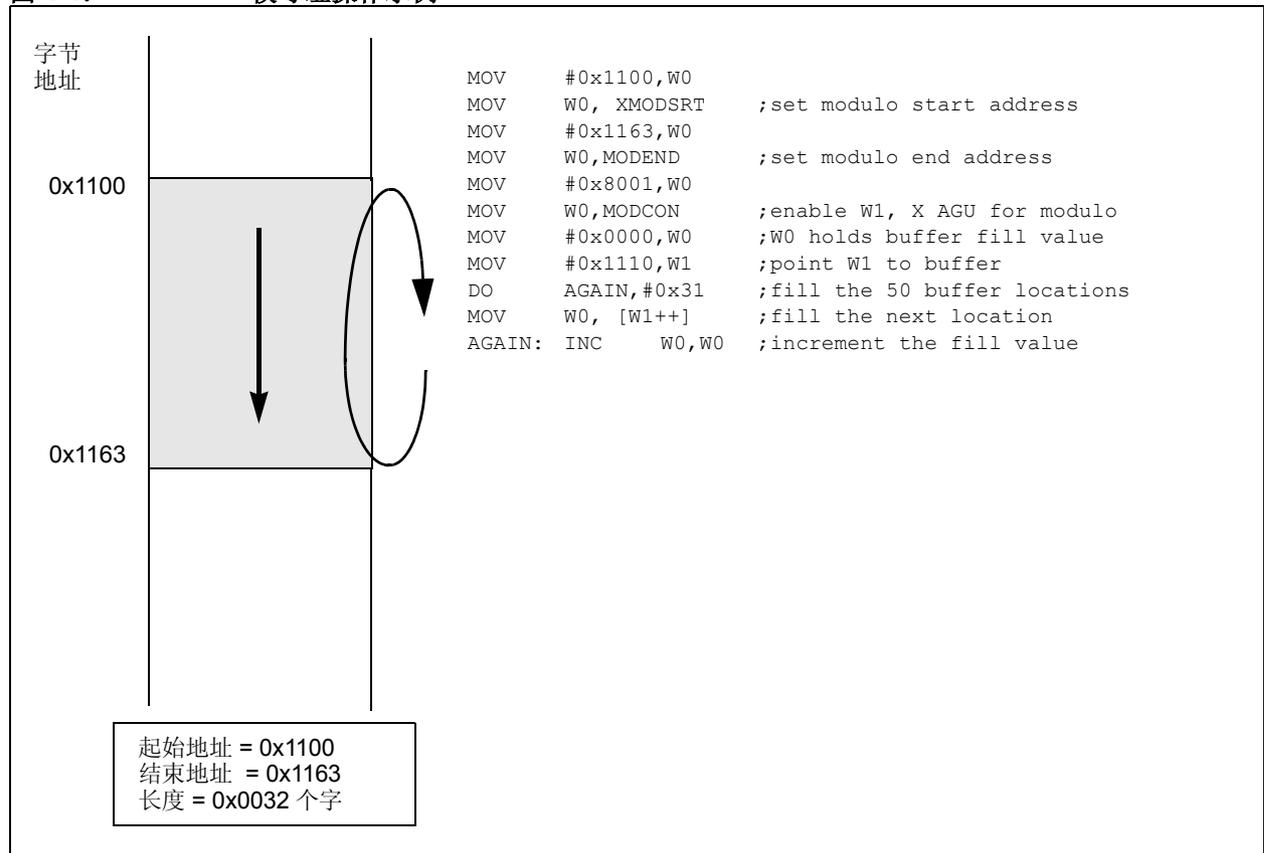
4.2.2 W 地址寄存器选择

模寻址和位反转寻址控制寄存器，MODCON<15:0> 中包含使能标志以及指定 W 地址寄存器的 W 寄存器字段。XWM 和 YWM 字段选择对哪些寄存器进行模寻址。如果 XWM = 15，则禁止 X RAGU 和 X WAGU 模寻址。类似地，如果 YWM = 15，则禁止 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，X 数据空间的模寻址被使能。

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 YMODEN 位（MODCON<14>）置 1 时，Y 数据空间的模寻址被使能。

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与 W 寄存器相关的有效地址 (EA) 计算中。重要的是要意识到, 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) 及是否低于下限 (对于递减缓冲区)。因此, 地址变化可能会越过边界, 但仍然可以正确调整。

注: 只有在使用执行前修改或执行后修改寻址模式来计算有效地址时, 模修正有效地址才被写回寄存器。如果使用了地址偏移量 (例如, $[W7 + W2]$), 会进行地址修改, 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。位反转寻址为 $XWAGU$ 所支持, 仅限于数据写入。

地址修改量, 可以是常数或寄存器的内容, 可视为将其位顺序反转。地址源和目的仍然是正常的顺序。于是, 唯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

位反转寻址在下列情况下被使能:

1. $MODCON$ 寄存器中 BWM (W 寄存器选择) 的值, 是除 15 以外的任何值 (不能使用位反转寻址访问堆栈), 且
2. $XBREV$ 寄存器中 $BREN$ 位置 1, 且
3. 使用的寻址模式是预递增或后递增的寄存器间接寻址模式。

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

$XB<14:0>$ 是位反转地址修改量或 “中心点” (pivot point), 通常是一个常数。对于 FFT 计算, 其值等于 FFT 数据缓冲区长度的一半。

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

使能位反转寻址时, 仅对预递增或后递增的寄存器间接寻址、且仅对字大小数据写入, 才会进行位反转寻址。对于任何其他寻址模式或对于字节大小数据, 不会进行位反转寻址, 而是生成正常的地址。在进行位反转寻址时, W 地址指针的增量将始终加上地址修改量 (XB), 与寄存器间接寻址模式相关的偏移量将被忽略。此外, 由于要求是字大小数据, EA 的 LSb 被忽略 (且始终为零)。

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 $XWAGU$, 位反转寻址将优先, $XWAGU$ 模寻址将被禁止。然而, 在 $XRAGU$ 中, 模寻址继续起作用。

如果通过置 1 $BREN$ ($XBREV<15>$) 位使能了位反转寻址, 那么, 在写 $XBREV$ 寄存器之后, 不应马上进行要使用被指定为位反转指针的 W 寄存器的间接读操作。

图 4-2: 位反转地址示例

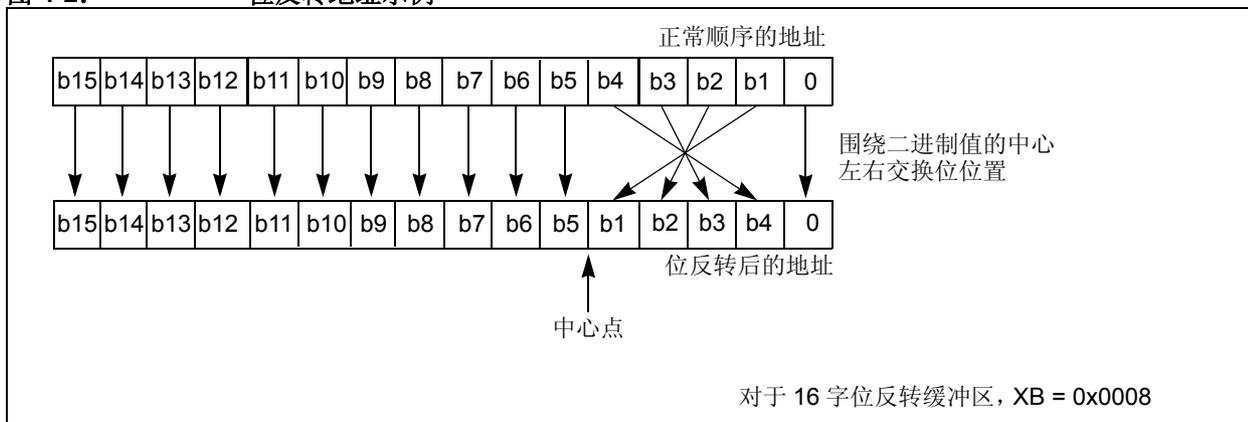


表 4-2: 位反转地址序列 (16 项)

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

缓冲区大小 (字)	XB<14:0> 位反转地址修改量
4096	0x0800
2048	0x0400
1024	0x0200
512	0x0100
256	0x0080
128	0x0040
64	0x0020
32	0x0010
16	0x0008
8	0x0004
4	0x0002
2	0x0001

dsPIC30F6010A/6015

注:

5.0 中断

注：本数据手册总结了 dsPIC30 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲知有关指令集和编程的更多详细信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

dsPIC30F6010A/6015 具有 44 个中断源和 4 个处理器异常（陷阱），必须根据优先级机制对它们进行仲裁。

CPU 负责读取中断向量表（Interrupt Vector Table, IVT）并将包含在中断向量中的地址传输给程序计数器。中断向量通过程序计数器输入端的 24 位宽的多路开关，从程序数据总线传输到程序计数器。

中断向量表（IVT）和备用中断向量表（Alternate Interrupt Vector Table, AIVT）位于程序存储器起始地址附近（0x000004）。IVT 和 AIVT 如图 5-1 所示。

中断和处理器异常传送到处理器内核前，中断控制器负责对它们进行预处理。使用集中化的特殊功能寄存器允许、控制外设中断并划分外设中断的优先级。

- IFS0<15:0>、IFS1<15:0> 和 IFS2<15:0>
这三个寄存器包含所有中断请求标志位。各中断请求标志位通过相应外设或外部信号置 1，由软件清零。
- IEC0<15:0>、IEC1<15:0> 和 IEC2<15:0>
这三个寄存器包含所有中断允许控制位。这些控制位用于单独允许外设或外部信号中断。
- IPC0<15:0> 至 IPC11<7:0>
与这 44 个中断相关的用户可分配优先级集中保存在这 12 个寄存器中。
- IPL<3:0>
当前 CPU 优先级明确地储存在 IPL 位。IPL<3> 位于 CORCON 寄存器中，而 IPL<2:0> 位于处理器内核中的状态寄存器（SR）中。
- INTCON1<15:0> 和 INTCON2<15:0>
全局中断控制功能通过这两个寄存器完成。INTCON1 包含处理器异常的控制和状态标志位。INTCON2 寄存器控制外部中断请求信号行为和备用中断向量表的使用。

• INTTREG<15:0>

INTTREG 寄存器包含相关的中断向量编号和新的 CPU 中断优先级，分别锁存在 INTTREG 寄存器中的向量编号（VECNUM<5:0>）和中断优先级（ILR<3:0>）位域中。新中断优先级是等待处理的中断的优先级。

注：发生中断时，无论相应的中断允许位的状态如何，中断标志位都将置 1。用户软件应确保在允许一个中断前清零相应的中断标志位。

通过 IPCx 寄存器，用户可为所有中断源分配 1 至 7 中的某一个优先级。各中断源与一个中断向量相关，如表 5-1 所示。优先级 7 和 1 分别表示最高和最低的可屏蔽优先级。

注：为中断源分配优先级 0，等效于禁止该中断。

如果 NSTDIS 控制位（INTCON1<15>）置 1，将防止中断嵌套。这样可防止在处理一个中断时处理另一个新中断，即使新中断的优先级高于正在处理的中断。

注：一旦 NSTDIS 位置 1，IPL 位就变为只读位。

一些中断具有边沿或电平触发中断及变化时中断等功能等特殊控制位。对这些功能的控制由产生中断的外设模块完成。

当 DISI 位（INTCON2<14>）保持置 1 时，可使用 DISI 指令禁止一定数量的指令对优先级为 6 或更低的中断进行处理。

处理中断时，PC 中载入存放在程序存储器中相应中断向量存储单元中的地址。IVT 中有 63 个不同的向量（见图 5-2）。这些向量保存在程序存储器的 0x000004 至 0x0000FE 存储单元中（见图 5-2）。这些存储单元包含 24 位地址，为保持鲁棒性，在正常执行期间当 PC 试图取这些字中的任意一个时，就会发生地址错误陷阱。这样便可防止由于以下情况而执行随机数据：PC 意外地递减到向量空间、意外地将数据空间地址映射到向量空间或 PC 在到达已实现的程序存储空间的末尾后重新回到 0x000000。执行 GOTO 指令跳转到此向量空间也会产生地址错误陷阱。

dsPIC30F6010A/6015

5.1 中断优先级

各中断源的用户可分配中断优先级（IP<2:0>）位于 IPCx 寄存器的各半字节的低三位中。各半字节的 bit 3 位未使用，读为 0。这些位定义了用户分配给某特定中断的优先级。

注： 用户可选择优先级范围为 0（最低优先级）至 7（最高优先级）。

因为可能会有一个以上的中断请求源被分配给某一特定的用户指定的优先级，所以提供了一种在给定的优先级内部分配优先级的方法。此方法称为“自然优先级”。

自然优先级取决于中断在向量表中的位置，仅当多个具有相同的用户分配的优先级的中断同时等待响应时才对中断操作产生影响。

表 5-1 列出了 dsPIC DSC 器件的中断编号和中断源及其相应的向量编号。

注 1： 自然优先级最高为 0，最低为 53。

注 2： 自然优先级编号与 INT 编号相同。

用户可为各中断分配七个优先级中的一个，这意味着用户可为一个具有较低自然优先级的中断分配一个非常高的总优先级。

表 5-1: 中断向量表

INT 编号	向量编号	中断源
最高自然优先级		
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 写完成
13	21	SI2C——I ² C™ 从中断
14	22	MI2C——I ² C 主中断
15	23	输入状态变化中断
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	SPI2
27	35	C1——CAN1 的组合 IRQ
28	36	IC3——输入捕捉 3
29	37	IC4——输入捕捉 4
30	38	IC5——输入捕捉 5
31	39	IC6——输入捕捉 6
32	40	OC5——输出比较 5
33	41	OC6——输出比较 6
34	42	OC7——输出比较 7
35	43	OC8——输出比较 8
36	44	INT3——外部中断 3
37	45	INT4——外部中断 4
38	46	C2——CAN2 的组合 IRQ
39	47	PWM——PWM 周期匹配
40	48	QE1——QE1 中断
41	49	保留
42	50	保留
43	51	FLTA——PWM 故障 A
44	52	FLTB——PWM 故障 B
45-53	53-61	保留
最低自然优先级		

5.2 复位过程

由于复位过程中不涉及到中断控制器，所以器件复位并不是真的异常。复位强制 PC 为零，处理器将初始化其寄存器以响应复位。处理器随后从存储单元 0x000000 处开始执行程序。GOTO 指令存储在第一个程序存储单元中，GOTO 指令的目标地址存储在紧随其后的单元中。处理器执行 GOTO，跳转到指定地址，然后开始指定目标（起始）地址处的操作。

5.2.1 复位源

有 6 个可导致器件复位的错误源。

- 看门狗定时器超时：
看门狗定时器已超时，表明处理器不再执行正确的代码流。
- 未初始化的 W 寄存器陷阱：
试图使用一个未初始化的 W 寄存器作为地址指针将引起复位。
- 非法指令陷阱：
试图执行未使用的操作码将引起非法指令陷阱。注意，如果由于流更改而使非法指令在执行前被移走，则取此非法指令不会引起非法指令陷阱。
- 欠压复位（BOR）：
检测到可能导致器件故障的电源电压短暂降低。
- 陷阱锁定：
多个陷阱条件同时发生会导致复位。

5.3 陷阱

可将陷阱看作一个表明软件或硬件错误的不可屏蔽的中断，它遵从如图 5-1 所示的预定义的优先级。陷阱旨在为用户提供一种方法，改正在调试和在应用中工作时的错误操作。

注： 如果用户不想在陷阱错误条件事件时采取校正措施，则必须将仅包含 RESET 指令的默认处理程序的地址装入向量。否则，如果调用一个包含无效地址的向量，会产生地址错误陷阱。

注意，很多陷阱条件只有在发生的时候才能被检测到。因此，在陷阱异常处理之前允许有问题的指令完成执行。如果用户选择从错误中恢复，则可能需要校正导致陷阱的错误操作的结果

陷阱有 8 个固定优先级：8 级至 15 级，这意味着处理陷阱时 IPL3 总是置 1。

如果用户当前不在执行陷阱，而将 IPL<3:0> 位设置为 0111（优先级 7），这样将禁止所有中断，但仍然能处理陷阱。

5.3.1 陷阱源

下列陷阱的优先级依次递增。然而，由于所有陷阱都可以嵌套，因此优先级的作用很小。

算术错误陷阱：

在以下四种情况下执行算术错误陷阱：

1. 如果试图进行以零作除数的除法运算，除法操作将在周期边界处中止，并产生陷阱。
2. 如果使能的话，当对累加器 A 或 B 进行的算术运算导致了 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. 如果使能的话，当对累加器 A 或 B 进行的算术运算导致了 bit 39 灾难性溢出，并且所有饱和和被禁止，将产生算术错误陷阱。
4. 如果在移位指令中指定的移位位数大于允许的最大移位位数，将产生陷阱。

dsPIC30F6010A/6015

地址错误陷阱:

当发生以下情况时, 将产生此陷阱:

1. 试图访问未对齐的数据字。
2. 试图从未实现的数据存储单元取数据。
3. 试图访问未实现的程序存储单元。
4. 试图从向量空间取指令。

注: 在 MAC 类指令中, 当数据空间分割为 X 和 Y 数据空间时, 未实现的 X 空间包括所有的 Y 空间, 未实现的 Y 空间包括所有的 X 空间。

5. 执行一条“BRA #literal”指令或一条“GOTO #literal”指令, 其中 literal 为一个未实现的程序存储器地址。
6. 修改 PC 使之指向未实现的程序存储器地址后执行指令。通过将值装入堆栈并执行 RETURN 指令可以修改 PC。

堆栈错误陷阱:

当发生以下情况时, 产生此陷阱:

1. 堆栈指针中载入的值大于写入到 SPLIM 寄存器的 (用户可编程) 极限值 (堆栈溢出)。
2. 堆栈指针中载入一个小于 0x0800 的值 (简单堆栈下溢)。

振荡器故障陷阱:

如果外部振荡器出现故障, 器件使用备用的内部 RC 振荡器工作, 就会产生振荡器故障陷阱。

5.3.2 硬陷阱和软陷阱

在同一周期内可能会产生多个陷阱 (例如, 把不对齐的字写入堆栈中的溢出地址)。此时, 图 5-1 所示的固定优先级就会起作用; 为了完全纠正错误, 这可能需要用户检查是否有其他等待处理的陷阱。

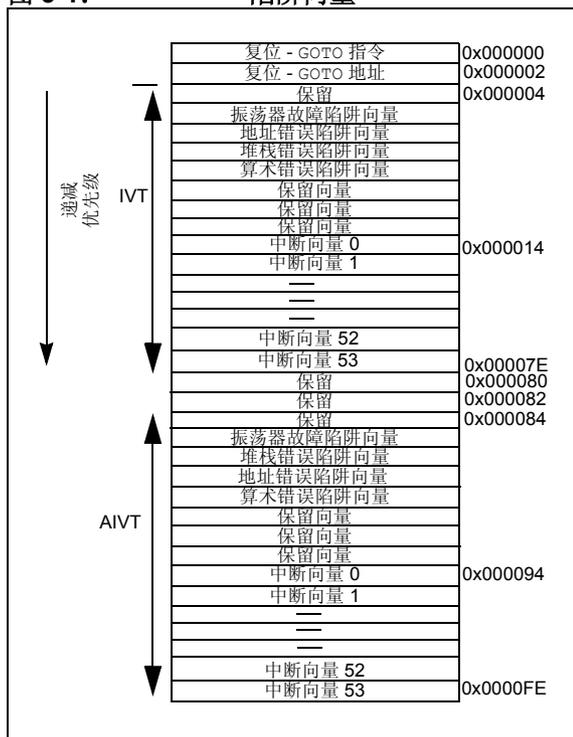
“软”陷阱包括优先级 8 到 11 的异常。算术错误陷阱 (优先级 11) 就属于这一类陷阱。

“硬”陷阱包括优先级 12 至 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) 陷阱就属于这一类。

每个硬陷阱产生时, 在执行任何代码之前, 必须先对它进行响应。在优先级较高的陷阱正在等待处理、被响应或正在处理过程中时, 如果产生了较低优先级的陷阱, 就会产生硬陷阱冲突。

在硬陷阱冲突条件下, 器件会自动复位。发生复位时, TRAPR 状态位 (RCON<15>) 置 1, 因此可在软件中检测该条件。

图 5-1: 陷阱向量



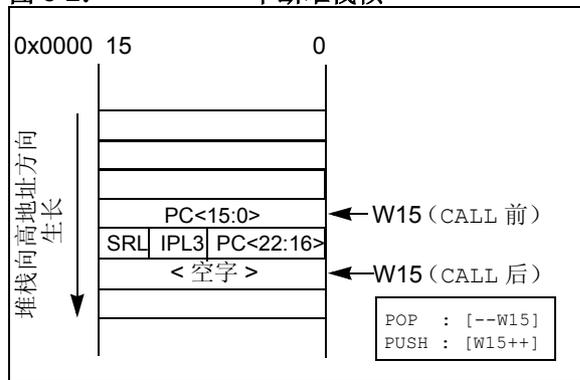
5.4 中断过程

由 IFSx 寄存器在各指令周期的开头采样所有中断事件标志位。IFSx 寄存器中等于 1 的标志位表示一个等待处理的中断请求 (IRQ)。如果中断允许 (IECx) 寄存器中的相应位置 1, 则 IRQ 会导致中断。在 IRQ 采样后余下的指令周期中, 将评估所有待处理中断请求的优先级。

如果有待处理的 IRQ, 它的优先级比 IPL 位中的当前处理器优先级高, 则处理器将被中断。

随后处理器将当前程序计数器的值和处理器状态寄存器的低字节 (SRL) 压入堆栈, 如图 5-2 所示。状态寄存器的低字节包含中断周期开始前的处理器优先级。处理器随后将此中断的优先级载入状态寄存器。完成中断服务程序前, 此操作禁止所有的低优先级中断。

图 5-2: 中断堆栈帧



注 1: 用户能始终通过向 SR 写入一个新值以降低优先级。必须在降低处理器中断优先级之前, 在中断服务程序中清除 IFSx 寄存器中的中断标志位以避免重复中断。

2: 处理中断时, IPL3 位 (CORCON<3>) 总是被清零。只有在进行陷阱处理时, 该位才会被置 1。

RETFIE (从中断返回) 指令将程序计数器和状态寄存器中的值弹出堆栈并使处理器返回到中断过程前的状态。

5.5 备用向量表

在程序存储器中, 中断向量表 (IVT) 之后紧跟备用中断向量表 (AIVT), 如图 5-1 所示。INTCON2 寄存器中的 ALTIVT 位控制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。备用向量与默认向量的构成相同。AIVT 提供了一种在应用和支持环境之间切换的方法, 从而支持仿真和调试功能, 而不需要对中断向量进行再编程。这个特性也支持运行时在不同应用之间切换, 以便评估各种软件算法。

如果不需要 AIVT, 则分配给 AIVT 的程序存储器可用于其他用途。AIVT 不是受保护的区域, 用户可对其自由编程。

5.6 快速现场保护

使用影子寄存器可保存现场。为 SR 中的 DC、N、OV、Z 和 C 位及寄存器 W0 至 W3 提供了影子寄存器。影子寄存器深度仅为一级。仅可使用 PUSH.S 和 POP.S 指令访问影子寄存器。

当处理器转移到中断向量开始处理中断时, 可以使用 PUSH.S 指令, 把上述寄存器的当前值保存到它们对应的影子寄存器中去。

如果一个具有特定优先级的 ISR 使用 PUSH.S 和 POP.S 指令进行快速现场保护, 那么优先级更高的 ISR 就不应再使用这两条指令。如果优先级较高的 ISR 使用了快速现场保护, 那么在优先级较低的中断处理期间, 用户必须保存关键寄存器。

5.7 外部中断请求

中断控制器支持五种外部中断请求信号: INT0-INT4。这些输入是边沿敏感的; 它们要求一个由低到高或由高到低的跳变以产生一个中断请求。INTCON2 寄存器具有 5 个位 (INT0EP-INT4EP), 用于选择边沿检测电路的极性。

5.8 从休眠和空闲模式唤醒

如果产生中断时处理器处于休眠或空闲模式, 则可使用中断控制器将处理器从休眠或空闲模式唤醒。

如果中断控制器接收到一个具有足够优先级的已允许的中断请求, 则标准中断请求会送至处理器。同时, 处理器会从休眠或空闲模式唤醒并开始执行处理中断请求所需的中断服务程序 (ISR)。

表 5-2: dsPIC30F6010A 中断控制器寄存器映射

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	OVBTE	COVTE	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000 0000 0000 0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	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	IC6IF	IC5IF	IC4IF	IC3IF	C1IF	SPI2IF	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	IC8IF	IC7IF	INT1IF	0000 0000 0000 0000
IFS2	0088	—	—	—	FLTBIF	FLTAIF	—	—	QE1IF	PWMIF	C2IF	INT4IF	INT3IF	OC8IF	OC7IF	OC6IF	OC5IF	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	IC6IE	IC5IE	IC4IE	IC3IE	C1IE	SPI2IE	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	IC8IE	IC7IE	INT1IE	0000 0000 0000 0000
IEC2	0090	—	—	—	FLTBIE	FLTAIE	—	—	QE1IE	PWMIE	C2IE	INT4IE	INT3IE	OC8IE	OC7IE	OC6IE	OC5IE	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	—	IC6IP<2:0>			—	IC5IP<2:0>			—	IC4IP<2:0>			—	IC3IP<2:0>			0100 0100 0100 0100
IPC8	00A4	—	OC8IP<2:0>			—	OC7IP<2:0>			—	OC6IP<2:0>			—	OC5IP<2:0>			0100 0100 0100 0100
IPC9	00A6	—	PWMIP<2:0>			—	C2IP<2:0>			—	INT41IP<2:0>			—	INT3IP<2:0>			0100 0100 0100 0100
IPC10	00A8	—	FLTAIP<2:0>			—	—	—	—	—	—	—	—	—	—	QE1IP<2:0>		0100 0000 0000 0000
IPC11	00AA	—	—	—	—	—	—	—	—	—	—	—	—	—	FLTBIP<2:0>		0000 0000 0000 0100	
INTTREG	00B0	—	—	—	—	ILR<3:0>					—	—	VECNUM<5:0>					0000 0000 0000 0000

图注: u = 未初始化的位

表 5-3: dsPIC30F6015 的中断控制器寄存器映射

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	OVBTE	COVTE	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000 0000 0000 0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	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	IC6IF	IC5IF	IC4IF	IC3IF	C1IF	SPI2IF	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	IC8IF	IC7IF	INT1IF	0000 0000 0000 0000
IFS2	0088	—	—	—	FLTBIF	FLTAIF	—	—	QEIIF	PWMIF	—	INT4IF	INT3IF	OC8IF	OC7IF	OC6IF	OC5IF	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	IC6IE	IC5IE	IC4IE	IC3IE	C1IE	SPI2IE	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	IC8IE	IC7IE	INT1IE	0000 0000 0000 0000
IEC2	0090	—	—	—	FLTBIE	FLTAIE	—	—	QEIIIE	PWMIE	—	INT4IE	INT3IE	OC8IE	OC7IE	OC6IE	OC5IE	0000 0000 0000 0000
IPC0	0094	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	INT0IP<2:0>			0100 0100 0100 0100
IPC1	0096	—	T3IP<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	—	IC6IP<2:0>			—	IC5IP<2:0>			—	IC4IP<2:0>			—	IC3IP<2:0>			0100 0100 0100 0100
IPC8	00A4	—	OC8IP<2:0>			—	OC7IP<2:0>			—	OC6IP<2:0>			—	OC5IP<2:0>			0100 0100 0100 0100
IPC9	00A6	—	PWMIP<2:0>			—	—	—	—	—	INT4IP<2:0>			—	INT3IP<2:0>			0100 0000 0100 0100
IPC10	00A8	—	FLTAIP<2:0>			—	—	—	—	—	—	—	—	—	QEIIIP<2:0>			0100 0000 0000 0000
IPC11	00AA	—	—	—	—	—	—	—	—	—	—	—	—	FLTBIP<2:0>			0000 0000 0000 0100	
INTTREG	00B0	—	—	—	—	ILR<3:0>			—	—	VECNUM<5:0>						0000 0000 0000 0000	

图注: u = 未初始化的位

注: 有关寄存器位域的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F6010A/6015

注:

6.0 闪存程序存储器

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_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。用户可使用以下两种方法对该存储器编程：

1. 在线串行编程 (ICSP™)
2. 运行时自编程 (RTSP)

6.1 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（分别称为 PGC 和 PGD）各一根，其余 3 根分别是电源线 (VDD)、接地线 (VSS) 和主复位线 (MCLR)。这允许用户在制造电路板时使用未编程器件，仅在产品交付前才对数字信号控制器进行编程。从而也可以将最新版本的固件或者定制固件烧写到器件中。

6.2 运行时自编程 (RTSP)

运行时自编程 (RTSP) 使用 TBLRD (表读) 和 TBLWT (表写) 指令实现。

使用 RTSP，用户可以一次擦除 32 个指令字 (96 字节) 的程序存储单元，一次可写入 32 个指令字 (96 字节) 的程序存储数据。

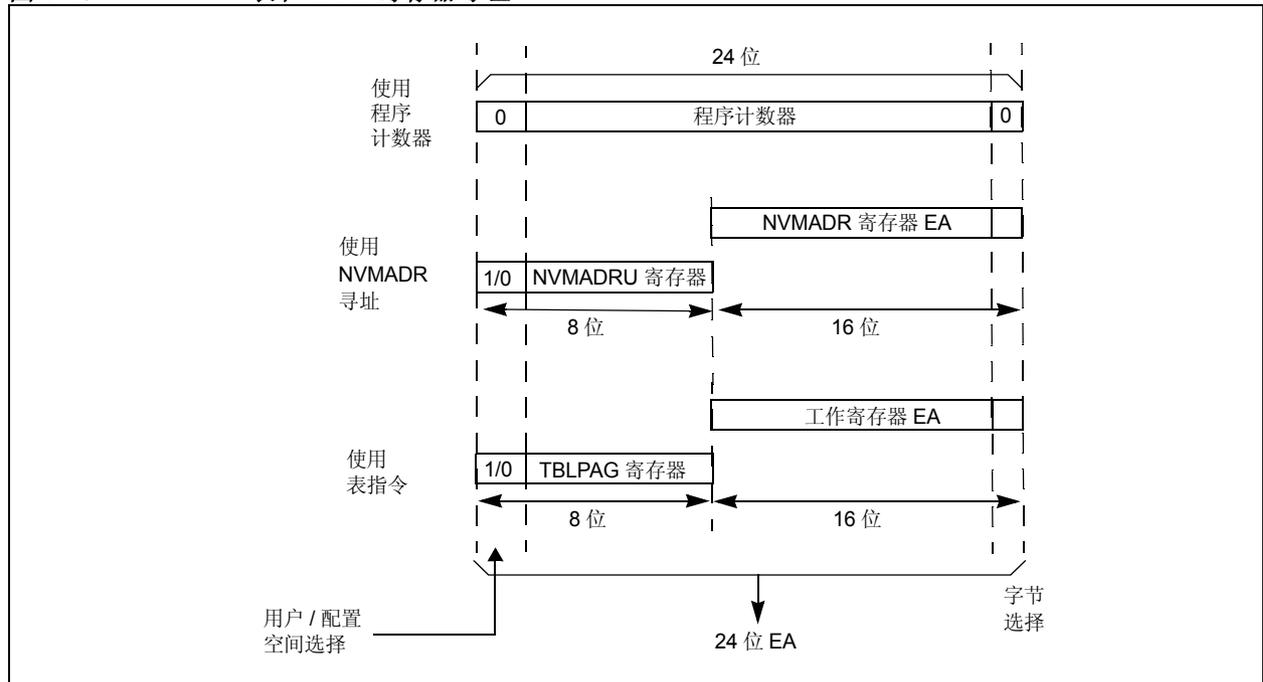
6.3 表指令操作综述

TBLRDL 和 TBLWTL 指令用来读或写程序存储器的 bit<15:0>。TBLRDL 和 TBLWTL 可以字或字节模式访问程序存储器。

TBLRDH 和 TBLWTH 指令用来读或写程序存储器的 bit<23:16>。TBLRDH 和 TBLWTH 可以字或字节模式访问程序存储器。

使用 TBLPAG 寄存器的 bits<7:0> 和来自表指令中指定的 W 寄存器的有效地址 (Effective Address, EA) 构成 24 位的程序存储器地址，如图 6-1 所示。

图 6-1: 表和 NVM 寄存器寻址



6.4 RTSP 操作

dsPIC30F 闪存程序存储器以行和板为单位构成。每行由 32 条指令或 96 字节组成。每板由 128 行或 4K x 24 个指令字组成。RTSP 允许用户每次擦除一行（32 条指令）、每次编程 32 条指令。

程序存储器的每板都包含写锁存器，它能够保存 32 个指令字的编程数据。在实际编程操作前，等待写入的数据必须先装入板的写锁存器。要写入板的数据按顺序装载到写锁存器中：指令 0，指令 1，依此类推。装载的指令字必须始终来自 32 个指令字的地址边界。

RTSP 编程的基本步骤是先建立一个表指针，然后执行一系列 TBLWT 指令，装载写锁存器。通过设置 NVMCON 寄存器中的特定位启动编程。装载 32 条指令需要 32 条 TBLWTL 和 32 条 TBLWTH 指令。

由于只需要写表锁存器，所以所有表写操作都是单字写入（2 个指令周期）。

写表锁存器之后，需要启动可编程操作以对数据进行编程。

在正常工作状态下的整个 VDD 范围内，闪存程序存储器都是可读 / 写和可擦除的。

6.5 RTSP 控制寄存器

有四个 SFR 用于读写闪存程序存储器，它们是：

- NVMCON
- NVMADR
- NVMADRU
- NVMKEY

6.5.1 NVMCON 寄存器

NVMCON 寄存器控制要擦除的块、要编程的存储器类型以及编程周期的启动。

6.5.2 NVMADR 寄存器

NVMADR 寄存器用于存放有效地址的两个低字节。它捕捉已被执行的上一条表指令的 EA<15:0>，选择要写入的行。

6.5.3 NVMADRU 寄存器

NVMADRU 寄存器用来保持有效地址的高字节。它捕捉已执行的上一条表指令的 EA<23:16>。

6.5.4 NVMKEY 寄存器

NVMKEY 是一个用于写保护的只写寄存器。要启动编程或擦除过程，用户必须把 0x55 和 0xAA 连续写入 NVMKEY 寄存器。更多详细信息请参见第 6.6 节“编程操作”。

注：	用户也可直接写 NVMADR 和 NVMADRU 寄存器，以指定待擦除或编程的程序存储器的地址。
-----------	--

6.6 编程操作

在 RTSP 模式下，对内部闪存进行编程或擦除需要完整的编程过程。编程操作持续时间的标称值为 2 ms，在此期间处理器暂停（等待）操作完成。将 WR 位（NVMCON<15>）置 1 启动编程操作，操作结束后将自动清零 WR 位。

6.6.1 闪存程序存储器的编程算法

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. 擦除闪存程序存储器行。
 - a) 设置 NVMCON 寄存器以允许对闪存程序存储器进行多字擦除，并将 WREN 位置 1。
 - b) 把要擦除的行地址写入寄存器 NVMADRU/NVMADR。
 - c) 把“55”写入 NVMKEY。
 - d) 把“AA”写入 NVMKEY。
 - e) 在擦除周期中 CPU 将暂停。
 - f) 将 WR 位置 1。这将开始擦除周期。
 - g) 将 WR 位置 1。这将开始擦除周期。

4. 从数据 RAM “映像”中把 32 个指令字的数据写入闪存程序存储器写锁存器。
5. 将 32 个指令字写入闪存程序存储器。
 - a) 设置 NVMCON 寄存器以允许对闪存程序存储器进行多字编程，并将 WREN 位置 1。
 - b) 把“55”写入 NVMKEY。
 - c) 把“AA”写入 NVMKEY。
 - d) 将 WR 位置 1。这将开始编程周期。
 - e) 在编程周期中 CPU 将暂停。
 - f) 当编程周期结束时，WR 位会被硬件清零。
6. 根据需要重复步骤 1 到 5，对所需的闪存程序存储器进行编程。

6.6.2 擦除程序存储器的一行

例 6-1 所示的代码序列可以用来擦除程序存储器的一行（32 个指令字）。

例 6-1: 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                ; Initialize PM Page Boundary SFR
MOV    #tbloffset(PROG_ADDR),W0 ; Intialize in-page EA[15:0] pointer
MOV    W0, NVMADR                ; Intialize NVMADR SFR
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

```

dsPIC30F6010A/6015

6.6.3 装载写锁存器

例 6-2 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。

例 6-2: 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 6-2 中，W3 中高字节的内容无效。

6.6.4 启动编程序列

出于保护的目，必须使用 NVMKEY 的写启动序列，以便允许进行任何擦除或编程操作。在执行编程命令后，用户必须等待一段时间（编程时间），直至编程完成。编程开始后紧跟的两条指令必须为 NOP 指令。

例 6-3: 启动编程序列

```
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             ; Start the erase sequence
NOP                                           ; Insert two NOPs after the erase
NOP                                           ; command is asserted
```

表 6-1: NVM 寄存器映射

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	复位状态
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

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

dsPIC30F6010A/6015

注:

7.0 数据 EEPROM 存储器

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

在整个 VDD 范围内的正常操作期间，数据 EEPROM 存储器是可读且可写的。数据 EEPROM 存储器直接映射到程序存储器地址空间。

用来读写闪存程序存储器的这四个特殊功能寄存器，也用来访问数据 EEPROM 存储器。如第 4.0 节“地址发生器单元”所述，这些寄存器是：

- NVMCON
- NVMADR
- NVMADRU
- NVMKEY

EEPROM 数据存储器允许读写单字和 16 字长的块。针对数据存储器时，NVMADR 与 NVMADRU 寄存器配合使用，用于寻址要访问的 EEPROM 单元。TBLRD 和 TBLWTL 指令用于读写数据 EEPROM。dsPIC30F6010 器件有最多 8 KB (4 KW) 的数据 EEPROM，地址范围从 0x7FF000 到 0x7FFFFE。

字写操作之前，应先擦除相应的存储单元。完成写操作通常需要 2 ms，但此时间将随电压和温度的变化而变化。

对数据 EEPROM 进行编程或擦除操作，不会停止指令流。在启动另一次数据 EEPROM 写/擦除操作之前，用户需要等待一段适当的时间。在编程或擦除操作进行过程中，若试图读数据 EEPROM，将返回不确定的数据。

控制位 WR 启动写操作，这与闪存程序存储器的写操作类似。可通过软件将 WR 位置 1，但不能将其清零。写操作完成时，硬件将清零 WR 位。不能用软件清零 WR 位，这样避免了意外或提早结束写操作。

WREN 位置 1 时将允许进行写操作。上电时，WREN 位清零。当写操作被正常工作期间的 MCLR 复位或 WDT 超时复位所中断时，WRERR 位置 1。在这种情况下，复位后，用户可以检查 WRERR，重写此存储单元。此地址寄存器 NVMADR 保持不变。

注：写操作完成后，IFS0 寄存器中的中断标志位 NVMIF 将置 1。NVMIF 位必须用软件清零。

7.1 读数据 EEPROM

TBLRD 指令读取位于当前程序字地址的字。下面的示例使用 W0 作为指向数据 EEPROM 的指针。结果存放在寄存器 W4 中，如例 7-1 所示。

例 7-1: 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
```

dsPIC30F6010A/6015

7.2 擦除数据 EEPROM

7.2.1 擦除数据 EEPROM 的一块

为了擦除数据 EEPROM 的一块，首先 NVMADRU 和 NVMADR 寄存器必须指向要擦除的存储块。将 NVMCON 配置为擦除数据 EEPROM 的块，并将 NVMCON 寄存器中的 WR 位和 WREN 位置 1。置 1 WR 位以启动擦除操作，如例 7-2 所示。

例 7-2: 数据 EEPROM 块擦除操作

```
; 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
```

7.2.2 擦除数据 EEPROM 的一个字

NVMADRU 和 NVMADR 寄存器必须指向要擦除的存储块。在 NVMCON 寄存器中，选择数据闪存的一个块，并将 WR 和 WREN 位置 1。将 WR 位置 1 启动擦除操作，如例 7-3 所示。

例 7-3: 数据 EEPROM 字擦除操作

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

7.3 写数据 EEPROM

要写 EEPROM 数据存储单元，必须按照下列顺序进行：

1. 擦除数据 EEPROM 的字。
 - a) 在 NVMCON 寄存器中，选择字擦除数据 EEPROM，并将 WREN 位置 1。
 - b) 把要擦除的字地址写入 NVMADRU/ NVMADR。
 - c) 允许 NVM 中断（可选）。
 - d) 把“55”写入 NVMKEY。
 - e) 把“AA”写入 NVMKEY。
 - f) 将 WR 位置 1。这将开始擦除周期。
 - g) 查询 NVMIF 位，或等待 NVMIF 中断。
 - h) 擦除周期结束后，WR 位会被清零。
2. 将数据字写入数据 EEPROM 写锁存器中。
3. 把 1 个数据字烧写到数据 EEPROM 中。
 - a) 在 NVMCON 寄存器中，选择字编程数据 EEPROM，并置 1 WREN 位。
 - b) 允许 NVM 写完成中断（可选）。
 - c) 把“55”写入 NVMKEY。
 - d) 把“AA”写入 NVMKEY。
 - e) 将 WR 位置 1。这将开始编程周期。
 - f) 查询 NVMIF 位，或等待 NVM 中断。
 - g) 写周期结束后，WR 位会被清零。

对于每个字，如果未严格遵守上述序列（把 0x55 写入 NVMKEY，把 0xAA 写入 NVMCON，然后将 WR 位置 1），写操作将不会启动。强烈建议在这段代码执行期间，禁止中断。

此外，必须将 NVMCON 中的 WREN 位置 1 来使能写操作。这样的机制防止了由于不可预料的代码-执行而意外地写数据 EEPROM。除非在更新 EEPROM 时，否则 WREN 位应该始终保持清零。硬件不能清零 WREN 位。

在写过程启动之后，清零 WREN 位将不会影响当前的写周期。禁止把 WR 位置 1，除非 WREN 位置 1。必须在前一条指令中将 WREN 位置 1。不能在同一条指令中将 WR 和 WREN 位置 1。

写周期结束时，硬件清零 WR 位，非易失性存储器写完成中断标志位（NVMIF）置 1。用户可以允许这个中断，或者查询此标志位。NVMIF 位必须由软件清零。

7.3.1 写数据 EEPROM 的一个字

一旦用户擦除了要编程的字，则表写指令用来写写锁存器，如例 7-4 所示。

例 7-4: 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
MOV          W0,NVMKEY                   ; Write the 0x55 key
MOV          #0xAA,W1
MOV          W1,NVMKEY                   ; Write the 0xAA key
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
    
```

dsPIC30F6010A/6015

7.3.2 写数据 EEPROM 的一块

要写入数据 EEPROM 的一个存储块，应首先写入所有 16 个锁存器，然后设置 NVMCON 寄存器，并对此存储块进行编程。

例 7-5: 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
MOV      W0,NVMKEY           ; Write the 0x55 key
MOV      #0xAA,W1
MOV      W1,NVMKEY           ; Write the 0xAA key
BSET     NVMCON,#WR          ; Start write cycle
NOP
NOP
```

7.4 写校验

根据不同的应用，可能要求把已写入存储器中的值与原始值进行校验，这是良好的编程习惯。大写操作次数过多且接近规范的极限值时，就应该对应用进行写校验。

7.5 防止误写入

某些情况下，并不打算写数据 EEPROM 存储器。器件内设置了各种机制来防止误写入 EEPROM。上电时，WREN 位清零；而且，上电延时定时器禁止 EEPROM 写操作。

写启动序列与 WREN 位配合使用，有助于防止欠压、电源干扰或软件故障期间的意外写入。

8.0 I/O 端口

注：本数据手册总结了 dsPIC30 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

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

所有 I/O 输入端口都为施密特触发器输入，以便增强抗干扰性。

8.1 并行 I/O (PIO) 端口

如果外设使能，并且外设正在使用相关引脚时，该引脚将不再作为通用 I/O 引脚使用。这时，可以读该 I/O 引脚，但对应并行端口引脚位的输出驱动器将被禁止。如果外设使能，但外设不在使用某引脚，则该引脚可以被端口驱动。

所有端口引脚都有三个寄存器，这些寄存器与端口引脚的工作直接相关。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)。但读取端口 (PORTx) 时，读到的是端口引脚的值；而写入端口引脚时，写入到锁存器 (LATx)。

任何位及其关联的数据和控制寄存器，如果其对于特定器件而言是无效的话，则将被禁止。这意味着相应的 LATx 和 TRISx 寄存器以及该端口引脚将读为 0。

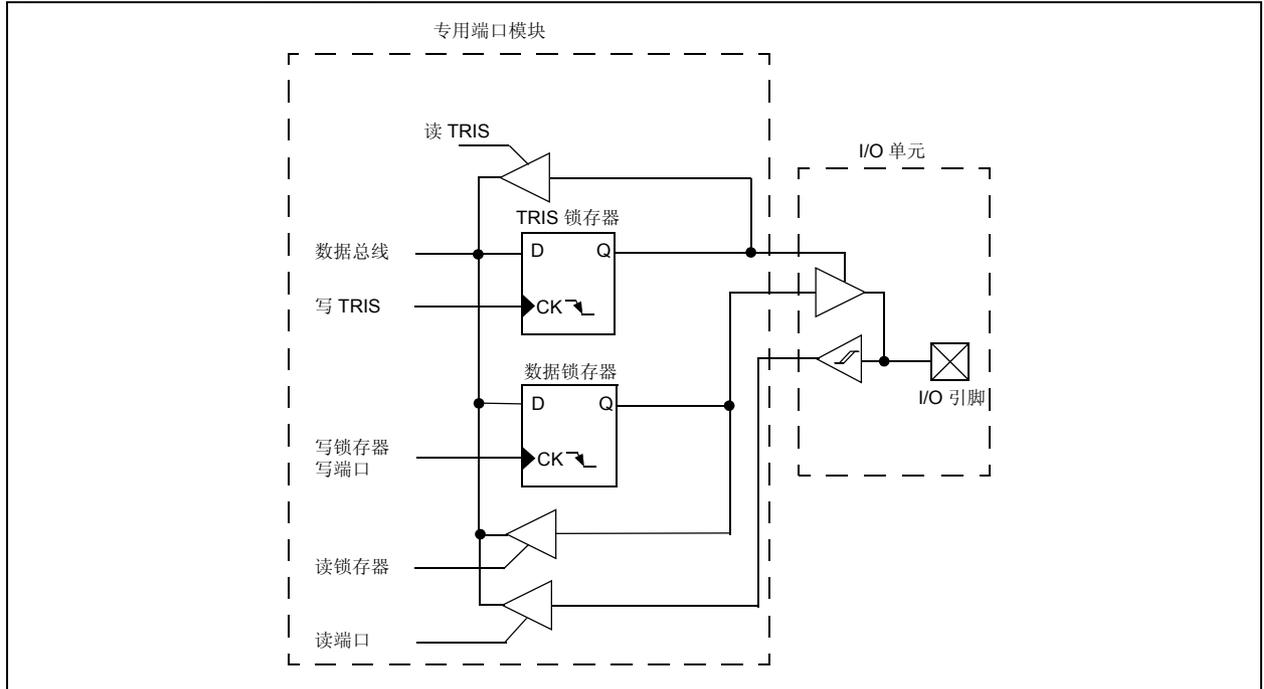
当一个定义为只用作输入的引脚与另一个外设或功能复用，由于没有其他竞争的输出源，它将被视为专用端口。INT4 引脚即为一个示例。图 8-1 显示了专用端口的结构。

PORTA 的寄存器格式如表 8-1 所示。

TRISA（数据方向控制）寄存器控制 RA<7:0> 引脚的方向，也控制 INTx 引脚和 VREF 引脚。LATA 寄存器提供输出数据，是可读写的。读 PORTA 寄存器将读取输入引脚的状态，而写 PORTA 寄存器将修改 LATA 寄存器的内容。

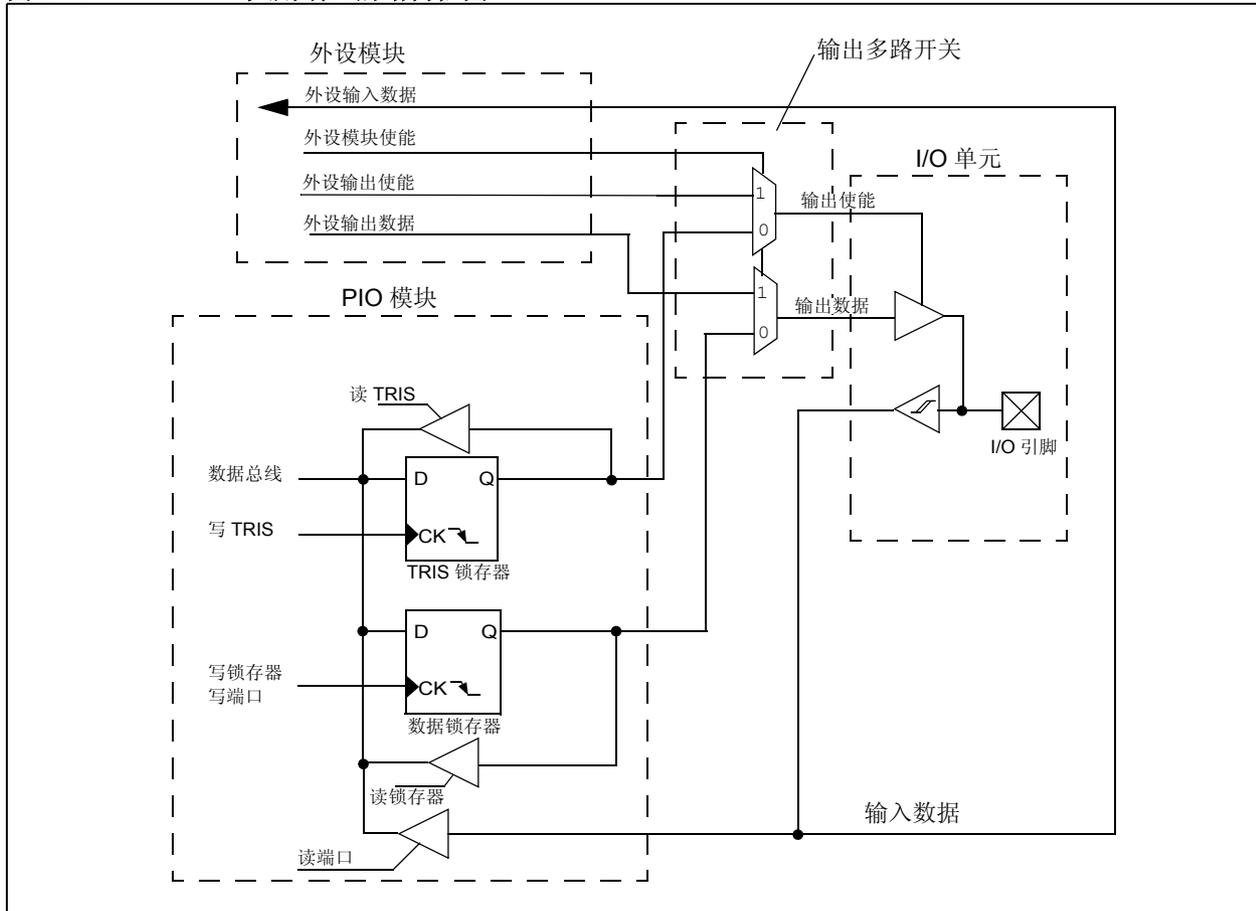
通常，与某个外设共用一个引脚的并行 I/O (PIO) 端口总是服从于该外设。外设的输出缓冲器数据和控制信号提供给一对多路开关。这对多路开关用于选择 I/O 引脚的输出数据和控制信号是用于外设还是相应的端口。图 8-2 中显示出端口与外设是如何复用的，以及对应的 I/O 引脚。表 8-1 显示了共用端口 PORTB 至 PORTG 的寄存器格式。

图 8-1: 专用端口的结构框图



dsPIC30F6010A/6015

图 8-2: 共用端口的结构框图



8.2 配置模拟端口引脚

ADPCFG 和 TRIS 寄存器用于控制 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），则该引脚的数字输出电平（VOH 或 VOL）将被转换。

读取端口寄存器时，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。

8.2.1 I/O 端口写 / 读时序

改变端口方向或对端口执行写操作，与对同一端口执行读操作之间需要间隔一个指令周期。通常在两者之间插入一条 NOP 指令。

例 8-1: 端口写 / 读示例

```
MOV 0xFF00, W0    ; Configure PORTB<15:8>
                  ; as inputs
MOV W0, TRISBB    ; and PORTB<7:0> as outputs
NOP               ; Delay 1 cycle
BTSS PORTB, #13  ; Next Instruction
```

表 8-1: dsPIC30F6010A 端口寄存器映射

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	TRISA15	TRISA14	—	—	—	TRISA10	TRISA9	—	—	—	—	—	—	—	—	—	1100 0110 0000 0000
PORTA	02C2	RA15	RA14	—	—	—	RA10	RA9	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
LATA	02C4	LATA15	LATA14	—	—	—	LATA10	LATA9	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
TRISB	02C6	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111 1111 1111
PORTB	02C8	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	0000 0000 0000 0000
LATB	02CB	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	0000 0000 0000 0000
TRISC	02CC	TRISC15	TRISC14	TRISC13	—	—	—	—	—	—	—	—	—	TRISC3	—	TRISC1	—	1110 0000 0000 1010
PORTC	02CE	RC15	RC14	RC13	—	—	—	—	—	—	—	—	—	RC3	—	RC1	—	0000 0000 0000 0000
LATC	02D0	LATC15	LATC14	LATC13	—	—	—	—	—	—	—	—	—	LATC3	—	LATC1	—	0000 0000 0000 0000
TRISD	02D2	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111 1111 1111
PORTD	02D4	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	0000 0000 0000 0000
LATD	02D6	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	0000 0000 0000 0000
TRISE	02D8	—	—	—	—	—	—	TRISE9	TRISE8	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	0000 0011 1111 1111
PORTE	02DA	—	—	—	—	—	—	RE9	RE8	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	0000 0000 0000 0000
LATE	02DC	—	—	—	—	—	—	LATE9	LATE8	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	0000 0000 0000 0000
TRISF	02EE	—	—	—	—	—	—	—	TRISF8	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	0000 0001 1111 1111
PORTF	02E0	—	—	—	—	—	—	—	RF8	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	0000 0000 0000 0000
LATF	02E2	—	—	—	—	—	—	—	LATF8	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	0000 0000 0000 0000
TRISG	02E4	—	—	—	—	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2	TRISG1	TRISG0	0000 0011 1100 1111
PORTG	02E6	—	—	—	—	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2	RG1	RG0	0000 0000 0000 0000
LATG	02E8	—	—	—	—	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2	LATG1	LATG0	0000 0000 0000 0000

图注: u = 未初始化的位

注: 有关寄存器位域的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 8-2: dsPIC30F6015 端口寄存器映射

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	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
PORTA	02C2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
LATA	02C4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000 0000 0000 0000
TRISB	02C6	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111 1111 1111
PORTB	02C8	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	0000 0000 0000 0000
LATB	02CB	LATB15	LATB14	LATB13	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	—	—	—	—	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	0000 1111 1111 1111
PORTD	02D4	—	—	—	—	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	0000 0000 0000 0000
LATD	02D6	—	—	—	—	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	0000 0000 0000 0000
TRISE	02D8	—	—	—	—	—	—	—	—	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	0000 0000 1111 1111
PORTE	02DA	—	—	—	—	—	—	—	—	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	0000 0000 0000 0000
LATE	02DC	—	—	—	—	—	—	—	—	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	0000 0000 0000 0000
TRISF	02EE	—	—	—	—	—	—	—	—	—	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
TRISG	02E4	—	—	—	—	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2	—	—	0000 0011 1100 1100
PORTG	02E6	—	—	—	—	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2	—	—	0000 0000 0000 0000
LATG	02E8	—	—	—	—	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2	—	—	0000 0000 0000 0000

图注: u = 未初始化的位

注: 有关寄存器位域的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

8.3 输入状态变化通知模块

I/O 端口的输入状态变化通知功能允许 dsPIC30F 器件在选定输入引脚的状态变化时，向处理器发出中断请求。当禁止时钟时，该特性还可在休眠模式下检测到输入状态改变。对于 dsPIC30F6010A，最多可以选择（允许）22 个外部信号（CN0 至 CN21）在输入状态发生变化时产生中断请求；对于 dsPIC30F6015，可选择（允许）19 个这样的外部信号（CN0 至 CN18）。

请参见引脚图获取 CN 引脚所在的位置。

表 8-3: 输入状态变化通知寄存器映射 (BIT 15-8)

SFR 名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	复位状态
CNEN1	00C0	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	0000 0000 0000 0000
CNEN2	00C2	—	—	—	—	—	—	—	—	0000 0000 0000 0000
CNPU1	00C4	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	0000 0000 0000 0000
CNPU2	00C6	—	—	—	—	—	—	—	—	0000 0000 0000 0000

图注: u = 未初始化的位

表 8-4: dsPIC30F6010A 的输入状态变化通知寄存器映射 (BIT 7-0)

SFR 名称	地址	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	—	—	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000 0000 0000 0000
CNPU1	00C4	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000 0000 0000 0000
CNPU2	00C6	—	—	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000 0000 0000 0000

图注: u = 未初始化的位

表 8-5: dsPIC30F6015 的输入状态变化通知寄存器映射 (BIT 7-0)

SFR 名称	地址	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	CN16IE	0000 0000 0000 0000
CNPU1	00C4	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000 0000 0000 0000
CNPU2	00C6	—	—	—	—	—	CN18PUE	CN17PUE	CN16PUE	0000 0000 0000 0000

图注: u = 未初始化的位

dsPIC30F6010A/6015

注:

9.0 TIMER1 模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本章介绍了 16 位通用 (General Purpose, GP) Timer1 模块以及相关的工作模式。

注： Timer1 是 A 类定时器。请参见本文档第 24.0 节“电气特性”中 A 类定时器的规范。

下面的各小节将详细说明该定时器各工作模式的设置和控制寄存器以及相应的框图。

Timer1 模块是一个 16 位的定时器，可作为实时时钟的时间计数器，或作为自由运行的时段定时器 / 计数器。

16 位定时器有下列模式：

- 16 位定时器
- 16 位同步计数器
- 16 位异步计数器

此外，支持如下工作特性：

- 定时器门控操作
- 可选的预分频比设置
- 在 CPU 空闲和休眠模式下的定时器操作
- 在 16 位周期寄存器匹配时或外部门控信号的下降沿产生中断

通过在 16 位 SFR, T1CON 中设置相应的位，选择上述工作模式。图 9-1 给出了 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，一直递增到与预先装入周期寄存器 PR1 中的值匹配，然后复位至 0 并继续计数。

CPU 进入空闲模式，定时器将停止递增，除非 TSIDL (T1CON<13>) 位 = 0。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，定时器将在外部时钟信号的上升沿递增，外部时钟信号与内部时钟同步。定时器计数，一直递增到与预先装入 PR1 中的值匹配，然后复位至 0，重新计数。

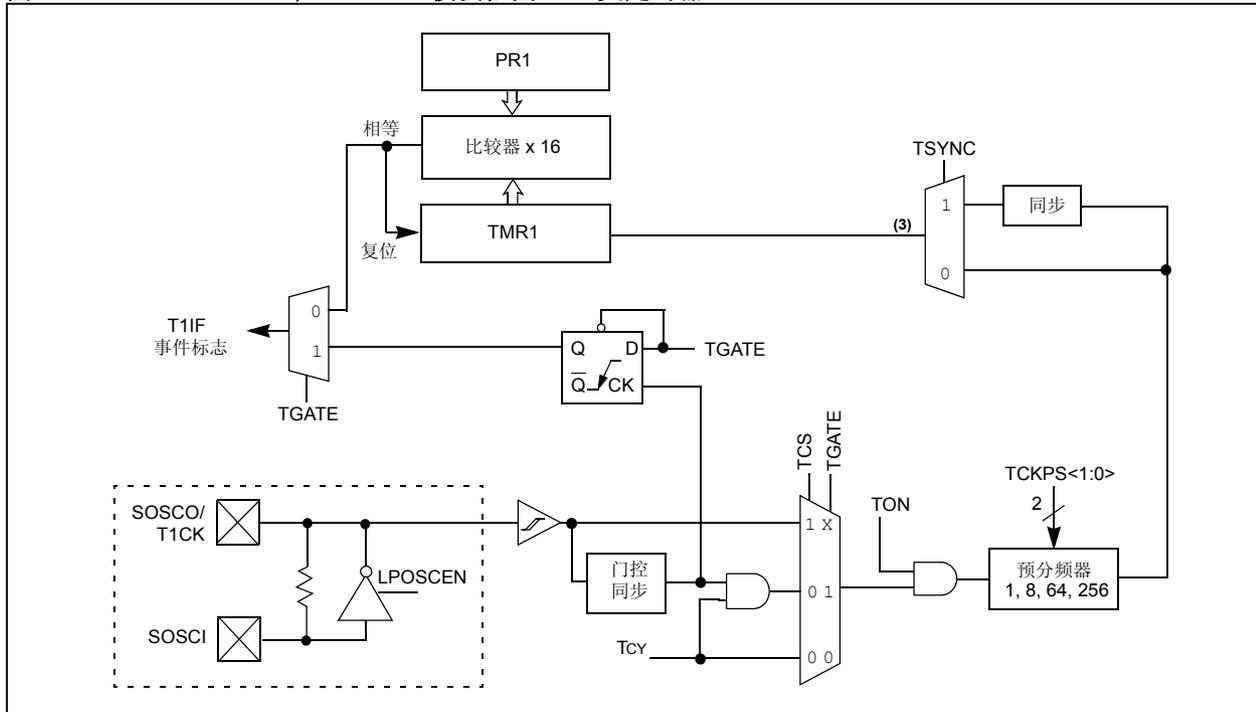
CPU 进入空闲模式，定时器将停止递增，除非 TSIDL 位 = 0。如果 TSIDL = 1，定时器模块逻辑在 CPU 空闲模式结束后恢复递增过程。

16 位异步计数器模式：在 16 位异步计数器模式下，定时器将在外部时钟信号的上升沿递增。定时器计数，一直递增到与预先装入 PR1 中的值匹配，然后复位至 0，重新计数。

当定时器配置为异步工作模式，且 CPU 进入空闲模式时，如果 TSIDL = 1，定时器将停止递增。

dsPIC30F6010A/6015

图 9-1: 16 位 TIMER1 模块框图 (A 类定时器)



9.1 定时器门控操作

16 位定时器可以设置为门控时间累加模式。该模式使得当门控输入信号 (T1CK 引脚) 为高电平时, 内部 Tcy 能够递增相应的定时器。要使得该模式, 必须置 1 控制位 TGATE (T1CON<6>)。必须使能定时器 (TON = 1), 且定时器时钟源必须被设置为内部时钟源 (TCS = 0)。

CPU 进入空闲模式时, 定时器将停止递增, 除非 TSIDL 位 = 0。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。

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 或 1:256 的预分频选择; 通过控制位 TCKPS<1:0> (T1CON<5:4>) 来选择预分频比。当出现下列事件之一时, 预分频器的计数器将清零:

- 写 TMR1 寄存器
- 清零 TON 位 (T1CON<15>)
- 器件复位, 如 POR 和 BOR

但是, 如果定时器被禁止 (TON = 0), 那么定时器预分频器将不会被复位, 因为预分频器的时钟停止了。

当写 T1CON 时, TMR1 不会被清零。通过写入 TMR1 寄存器将其清零。

9.3 休眠模式下的定时器操作

在 CPU 休眠模式下, 满足以下条件定时器将工作:

- 使能定时器模块 (TON = 1)
- 选择外部定时器时钟源 (TCS = 1)
- 设置 TSYNC 位 (T1CON<2>) 为逻辑 0, 将外部时钟源定义为异步时钟源。

发生上述三种情况时, 定时器将继续计数, 直到递增到与周期寄存器的内容匹配, 然后复位为 0x0000。

当定时器和周期寄存器之间发生匹配时, 如果相应的定时器中断允许位被置 1, 则产生中断。

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。当定时器的计数与周期寄存器相匹配时，T1IF位将置1，如果允许中断的话将产生中断。T1IF位必须在软件中清零。定时器中断标志T1IF位于中断控制器的IFS0控制寄存器中。

当使能门控时间累加模式时，在门控信号的下降沿（在累加周期结束时）将产生一个中断。

通过相应的定时器中断允许位T1IE来允许中断。定时器中断允许位，位于中断控制器的IEC0控制寄存器中。

9.5 实时时钟

当Timer1在实时时钟（Real-Time Clock, RTC）模式下工作时，具有当天时间和事件时间标记的功能。RTC的主要工作特性是：

- 在32 kHz LP振荡器下工作
- 8位预分频器
- 低功耗
- 实时时钟中断

通过设置T1CON控制寄存器的相应位，选择上述工作模式。

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，达到周期寄存器中指定的值，然后复位为0。

要进行正确操作，必须将TSYNC位置为逻辑0（异步模式）。

使能LPOSCEN（OSCCON<1>）位将禁止正常定时器和计数器模式并使能一个定时器超时唤醒事件。

当CPU进入休眠模式时，RTC继续工作，提供的32 kHz外部晶振仍然有效，且控制位没有改变。要使RTC继续在空闲模式下工作，TSIDL位必须清零0。

9.5.2 RTC 中断

中断事件发生时，如果允许，相应的中断标志T1IF将被置1并产生一个中断。T1IF位必须在软件中清零。相应的定时器中断标志T1IF，位于中断控制器的IFS0状态寄存器中。

通过相应定时器中断允许位（T1IE）来允许中断。定时器中断允许位，位于中断控制器的IEC0控制寄存器中。

图 9-2: 建议的TIMER1 LP振荡器 RTC 元件

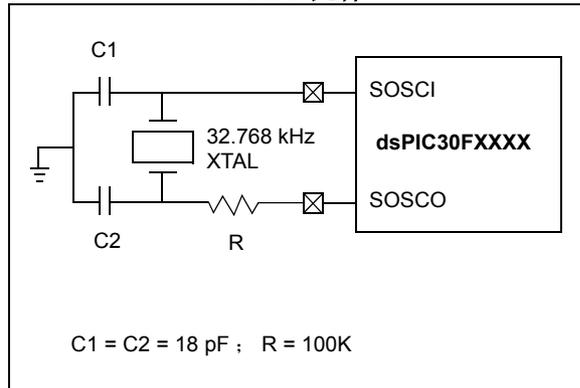


表 9-1: TIMER1 寄存器映射

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

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

10.0 TIMER2/3 模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本章介绍了 32 位通用 (GP) 定时器模块 (Timer2/3) 以及相关的工作模式。图 10-1 给出了 32 位 Timer2/3 模块的简化框图。图 10-3 和图 10-5 分别为将 Timer2/3 配置成的两个独立 16 位定时器 Timer2 和 Timer3 的框图。

注： Timer2 是 B 类定时器，而 Timer3 是 C 类定时器。请参见本文档第 24.0 节“电气特性”中相应的定时器类型。

Timer2/3 模块是 32 位定时器（可被配置为两个 16 位定时器），具有可选择的工作模式。这些定时器为其他外设模块所使用，例如：

- 输入捕捉
- 输出比较 / 简单 PWM

对于定时器的工作模式，下面的各小节将详细说明其设置和控制寄存器以及相应的框图。

32 位定时器有下列模式：

- 两个独立的 16 位定时器 (Timer2 和 Timer3)，支持所有 16 位工作模式 (异步计数器模式除外)
- 一个 32 位定时器
- 一个 32 位同步计数器

此外，支持如下工作特性：

- ADC 事件触发信号
- 定时器门控操作
- 可选择的预分频比设定
- 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

通过设定 16 位 T2CON 和 T3CON SFR 中的相应位来决定这些工作模式。

对于 32 位定时器 / 计数器操作，Timer2 是 32 位定时器的低位字，而 Timer3 是高位字。

注：对于 32 位定时器操作，T3CON 控制位将被忽略。其设置和控制只使用 T2CON 控制位。对于 32 位定时器模块，使用的是 Timer2 时钟和门控输入，但是，中断产生由 Timer3 中断标志位 (T3IF) 反映，且中断通过 Timer3 中断允许位 (T3IE) 来允许。

16 位模式：在 16 位模式下，Timer2 和 Timer3 可以配置为两个独立的 16 位定时器。每个定时器均可设置为 16 位定时器模式或 16 位同步计数器模式。这两种工作模式的细节，请参见第 9.0 节“Timer1 模块”。

Timer2 和 Timer3 之间惟一的功能差异是，Timer2 提供了时钟预分频器输出同步。这对于高频外部时钟输入很有用。

32 位定时器模式：在 32 位定时器模式下，定时器在每个指令周期递增，一直递增到与预先装入组合 32 位周期寄存器 PR3/PR2 中的值匹配，然后复位至 0，继续计数。

对于 Timer2/Timer3 对的同步 32 位读操作，读低位字 (TMR2 寄存器) 将导致高位字 (msw) 被读取并锁存到一个 16 位保持寄存器 (称作 TMR3HLD)。

对于同步 32 位写操作，必须首先写入保持寄存器 (TMR3HLD)。如果后续操作是写 TMR2 寄存器的话，则 TMR3HLD 的内容将传送并锁存到 32 位定时器 (TMR3) 的 MSB 中。

32 位同步计数器模式：在 32 位同步计数器模式下，定时器将在外部时钟信号的上升沿递增，外部时钟信号与内部时钟同步。定时器计数，一直递增到与预先装入组合 32 位周期寄存器 PR3/PR2 中的值匹配，然后复位至 0，继续计数。

当定时器配置为同步计数器工作模式且 CPU 进入空闲模式时，定时器将停止递增，除非 TSIDL (T2CON<13>) 位 = 0。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。

dsPIC30F6010A/6015

图 10-1: dsPIC30F6010A 的 32 位 TIMER2/3 框图

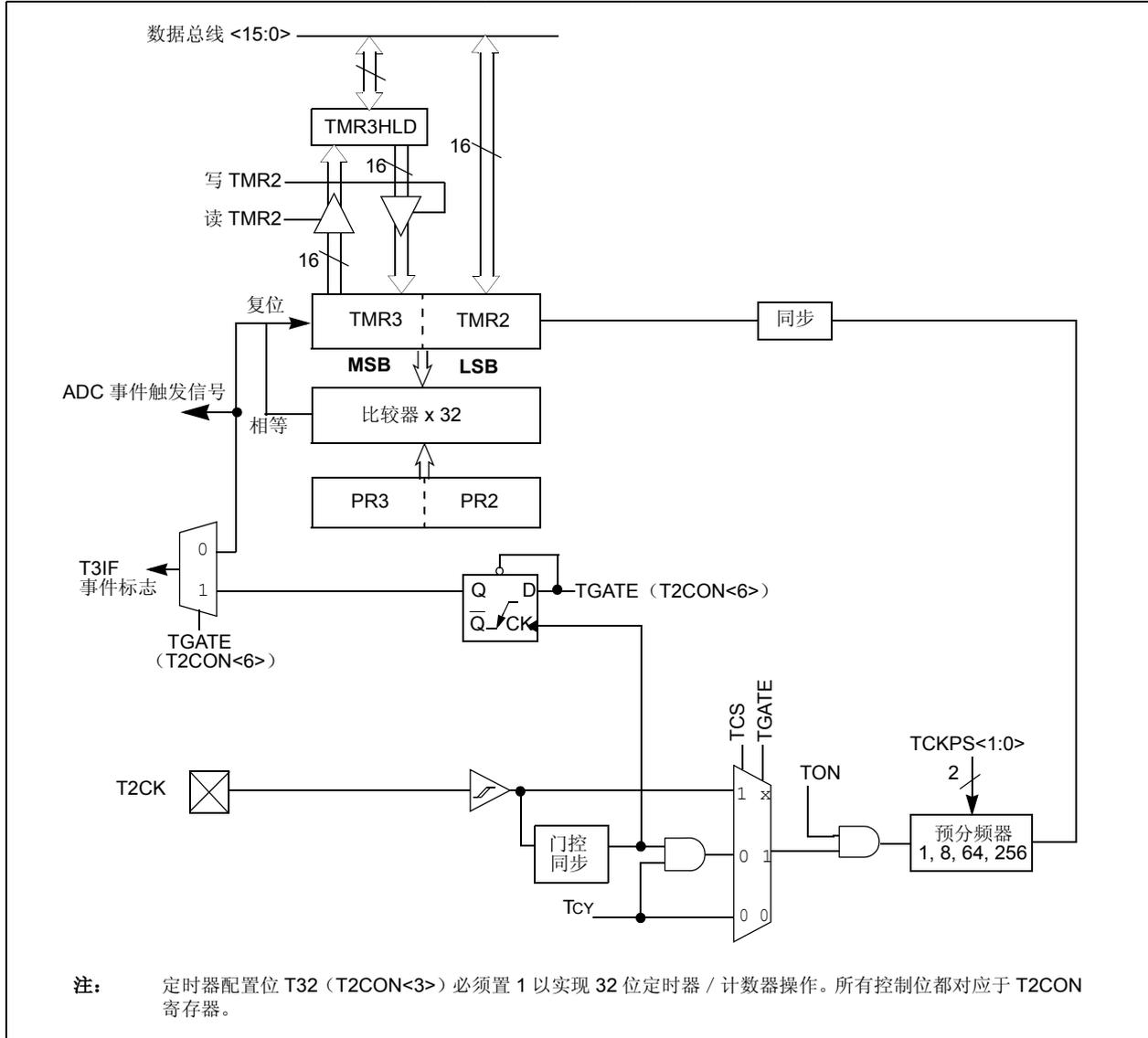
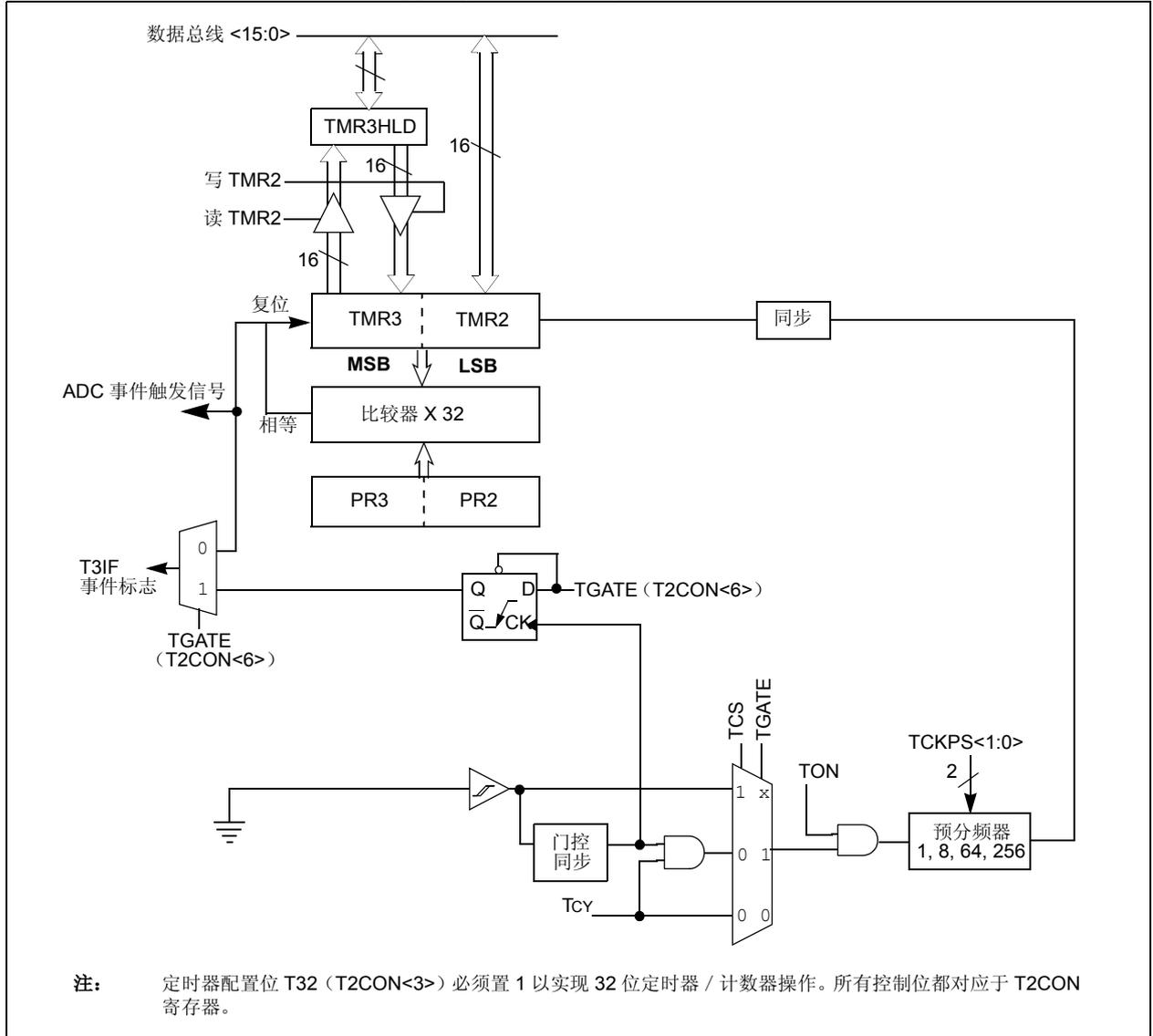


图 10-2: dsPIC30F6015 的 32 位 TIMER2/3 框图



dsPIC30F6010A/6015

图 10-3: dsPIC30F6010A 的 16 位 TIMER2 框图 (B 类定时器)

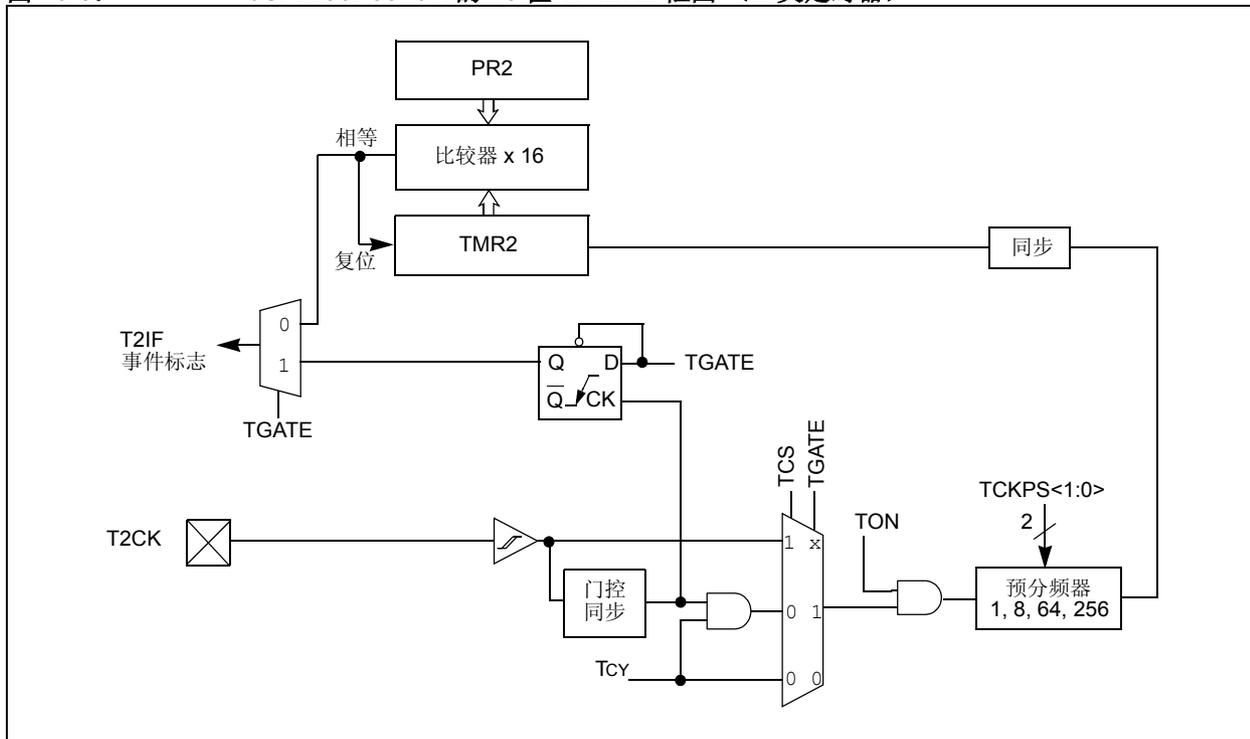


图 10-4: dsPIC30F6015 的 16 位 TIMER2 框图 (B 类定时器)

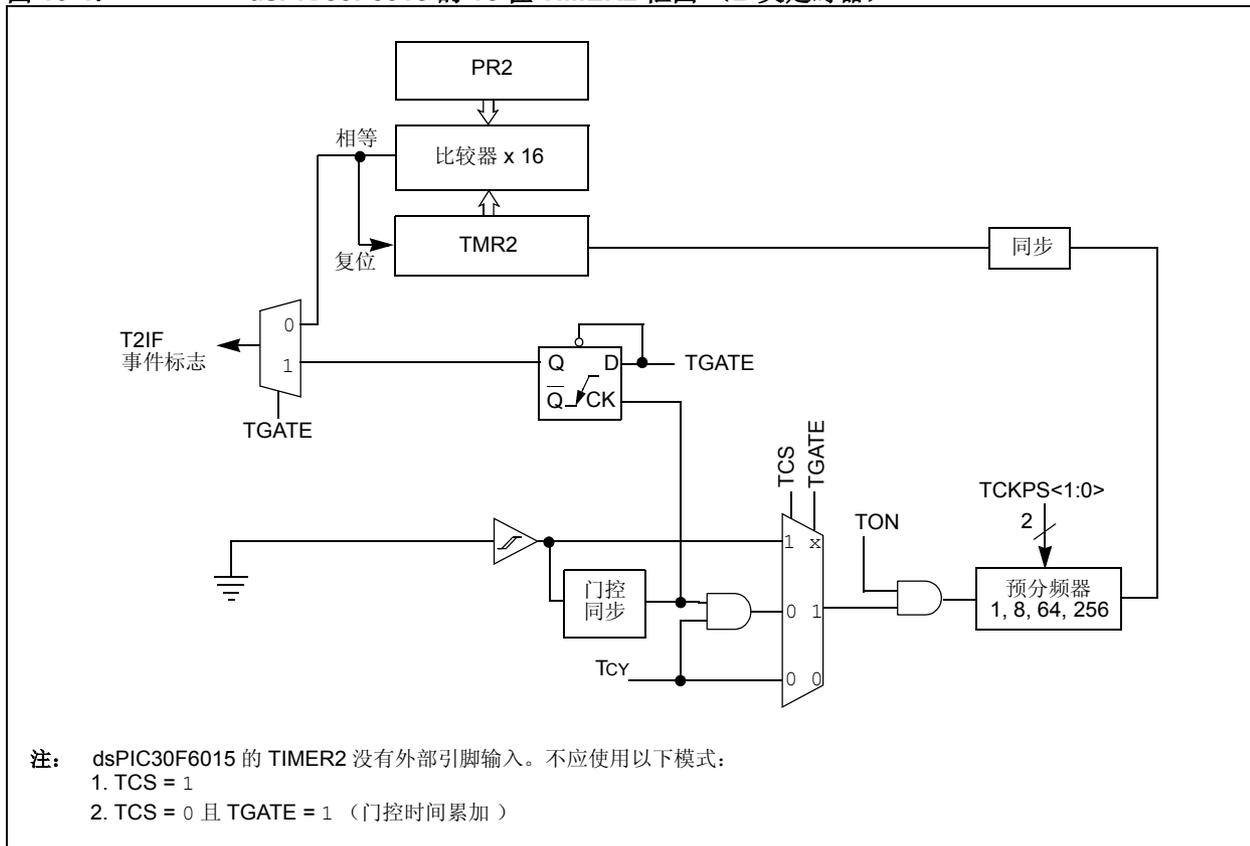
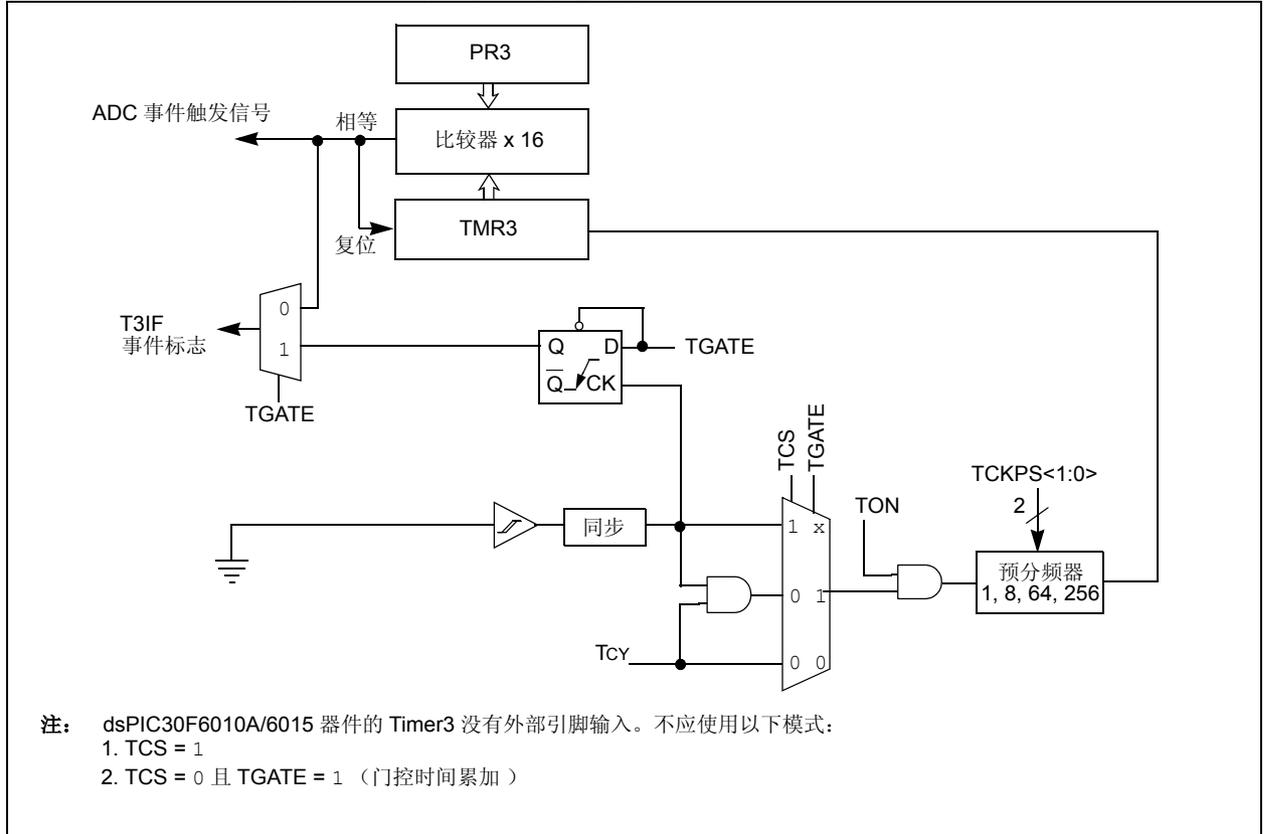


图 10-5: 16 位 TIMER3 框图 (C 类定时器)



10.1 定时器门控操作

可将 32 位定时器置于门控时间累加模式下。该模式使内部 TcY 能够在门控输入信号（T2CK 引脚）为高电平时，递增相应的定时器。要启用该模式，必须置 1 控制位 TGATE（T2CON<6>）。在此模式下，Timer2 是源时钟源。对于 Timer3，TGATE 设定被忽略。必须使能定时器（TON = 1），且定时器时钟源设置为内部时钟源（TCS = 0）。

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

10.2 ADC 事件触发信号

当 32 位定时器（TMR3/TMR2）与 32 位组合周期寄存器（PR3/PR2）匹配时，Timer3 将生成 ADC 特殊事件触发信号。

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 的预分频比选择；通过控制位 TCKPS<1:0>（T2CON<5:4> 和 T3CON<5:4>）来选择预分频比。对于 32 位定时器操作，发起时钟源是 Timer2。在此模式下不能进行 Timer3 的预分频器操作。当出现下列事件之一时，预分频器的计数器将清零：

- 写 TMR2/TMR3 寄存器
- 清零任一 TON 位（T2CON<15> 或 T3CON<15>）
- 器件复位，如 POR 和 BOR

但是，如果定时器被禁止（TON = 0），那么 Timer2 预分频器将不会被复位，因为预分频器的时钟停止了。

当写 T2CON/T3CON 时，TMR2/TMR3 不会被清零。

10.4 休眠模式下的定时器操作

在 CPU 休眠模式期间，定时器不工作，因为内部时钟被禁止。

10.5 定时器中断

32 位定时器模块具有在周期匹配时，或者在外部门控信号下降沿产生中断的能力。当 32 位定时器计数与相应的 32 位周期寄存器相匹配时，或者检测到外部“门控”信号的下降沿时，T3IF 位（IFS0<7>）将置 1，如果允许中断的话则将产生中断。在此模式下，T3IF 中断标志用作中断源。T3IF 位必须用软件清零。

通过相应的定时器中断允许位 T3IE（IEC0<7>）来允许中断。

表 10-1: TIMER2/3 寄存器映射

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 (仅适用于 32 位定时器操作)																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

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

dsPIC30F6010A/6015

注:

11.0 TIMER4/5 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本章介绍了辅助 32 位通用 (GP) 定时器模块 (Timer4/5) 及其相关的工作模式。图 11-1 给出了 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 配置成两个独立 16 位定时器 Timer4 和 Timer5 的框图。

注: Timer4 是 B 类定时器 Timer5 是 C 类定时器。请参见本文档第 24.0 节 "电气特性" 中的相应定时器类型。

Timer4/5 模块的工作原理与 Timer2/3 模块相似。但两者之间也有一些差别，如下所示：

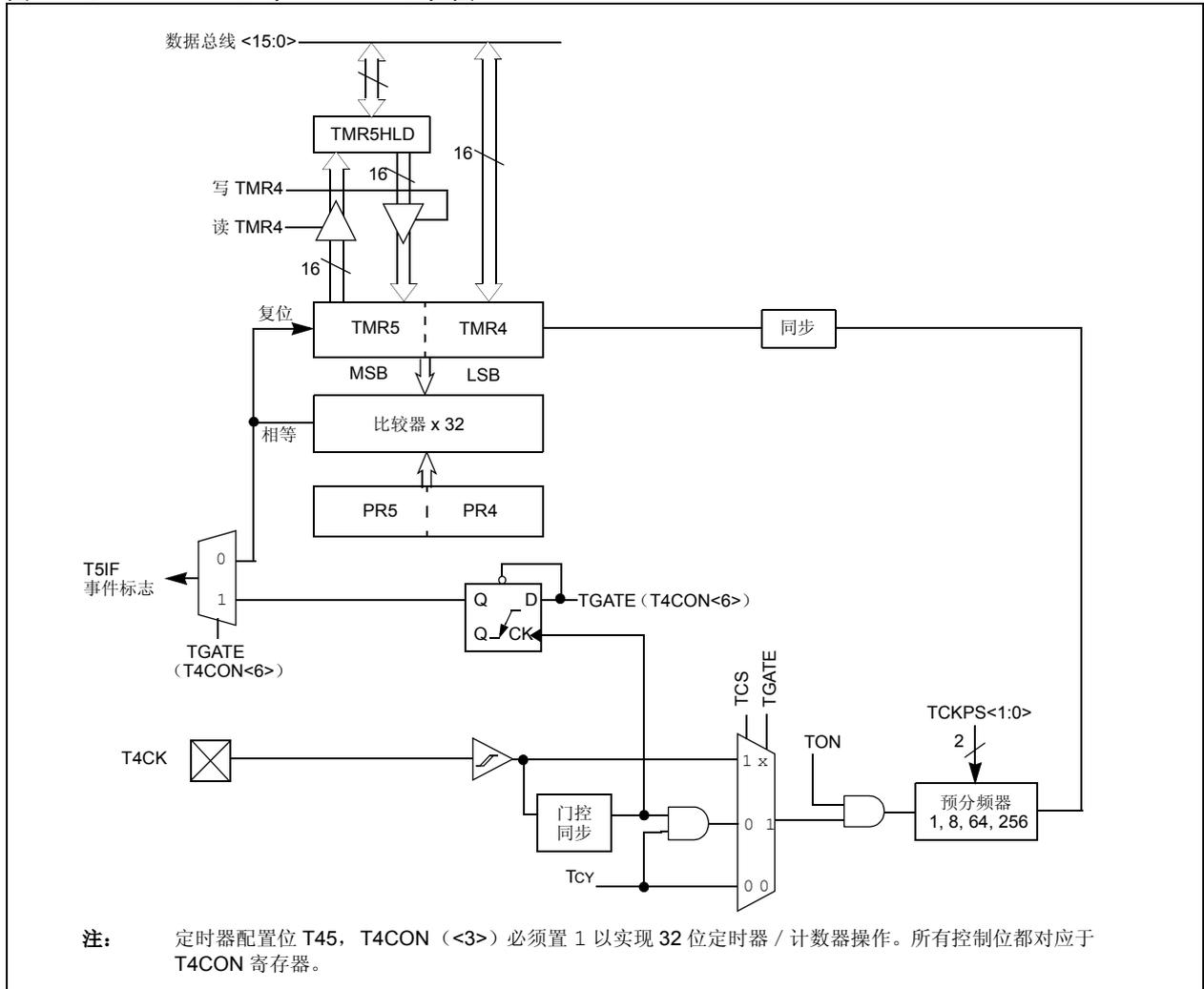
- Timer4/5 模块不支持 ADC 事件触发器特性
- Timer4/5 模块不能被其他的外设模块使用，例如输入捕捉模块和输出比较模块

通过设置 16 位 T4CON 和 T5CON SFR，选择 Timer4/5 模块的工作模式。

对于 32 位定时器 / 计数器操作，Timer4 是 32 位定时器的低位字 (lsw)，而 Timer5 是高位字 (msw)。

注: 对于 32 位定时器操作，T5CON 控制位将被忽略。设置和控制只使用 T4CON 控制位。对于 32 位定时器模块，使用的是 Timer4 时钟和门控输入，但是，中断产生由 Timer5 中断标志位 (T5IF) 反映，且中断通过 Timer5 中断允许位 (T5IE) 来允许。

图 11-1: 32 位 TIMER4/5 框图



dsPIC30F6010A/6015

图 11-2: 16 位 TIMER4 框图 (B 类定时器)

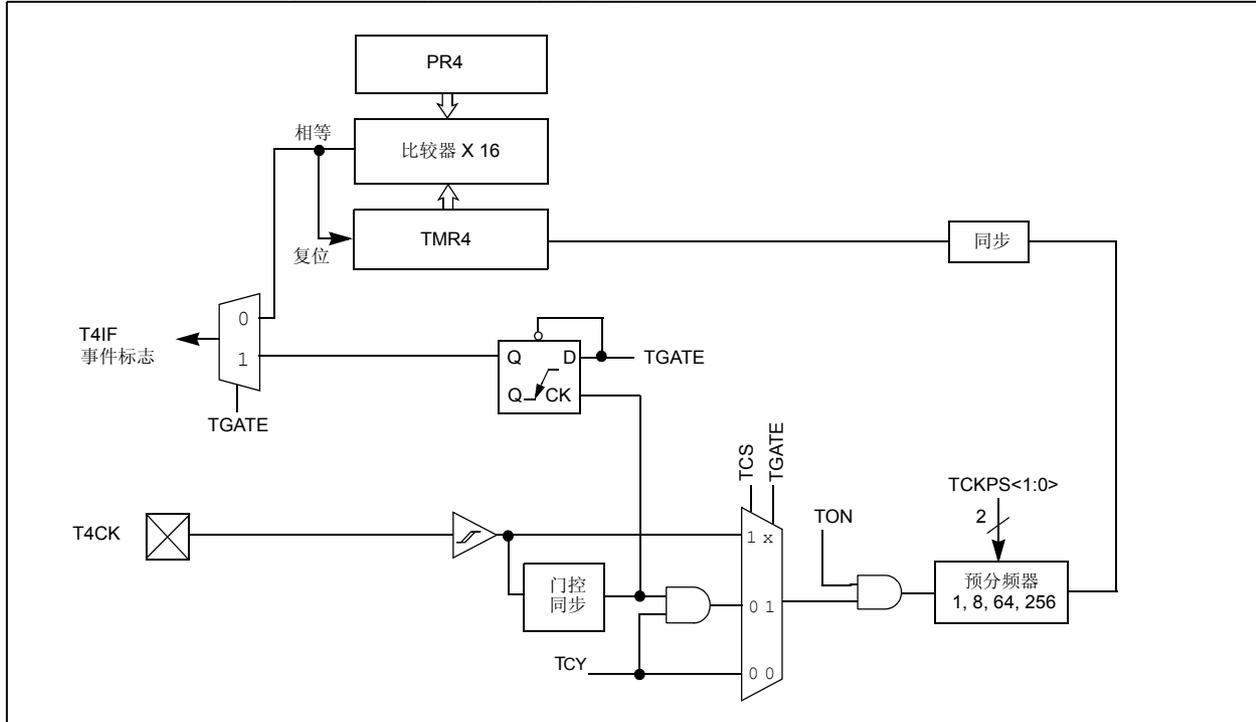


图 11-3: 16 位 TIMER5 框图 (C 类定时器)

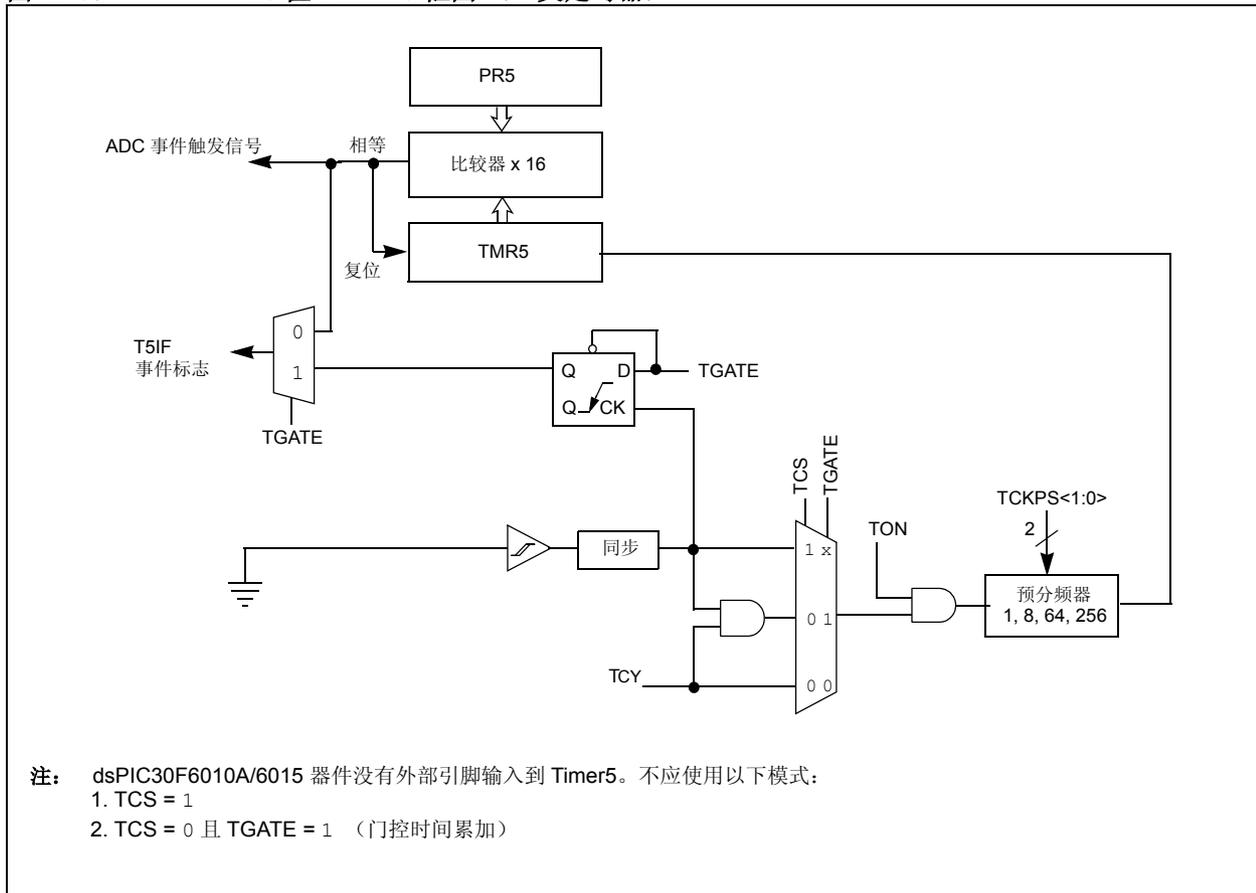


表 11-1: TIMER4/5 寄存器映射

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 (仅在 32 位定时器工作模式下使用)																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	T45	—	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

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

dsPIC30F6010A/6015

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，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本章对输入捕捉模块和相关的工作模式进行了介绍。此模块提供的工作特性可用于需要进行频率（周期）和脉冲测量的应用场合。图 12-1 给出了输入捕捉模块的框图。输入捕捉功能可用于以下模式：

- 频率 / 周期 / 脉冲测量
- 其他外部中断源

输入捕捉模块的主要工作特性是：

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

通过设定 ICxCON 寄存器（其中，x = 1, 2, ..., N）中相应的位，确定上述操作模式。dsPIC30F6010A 和 dsPIC30F6015 器件有 8 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：

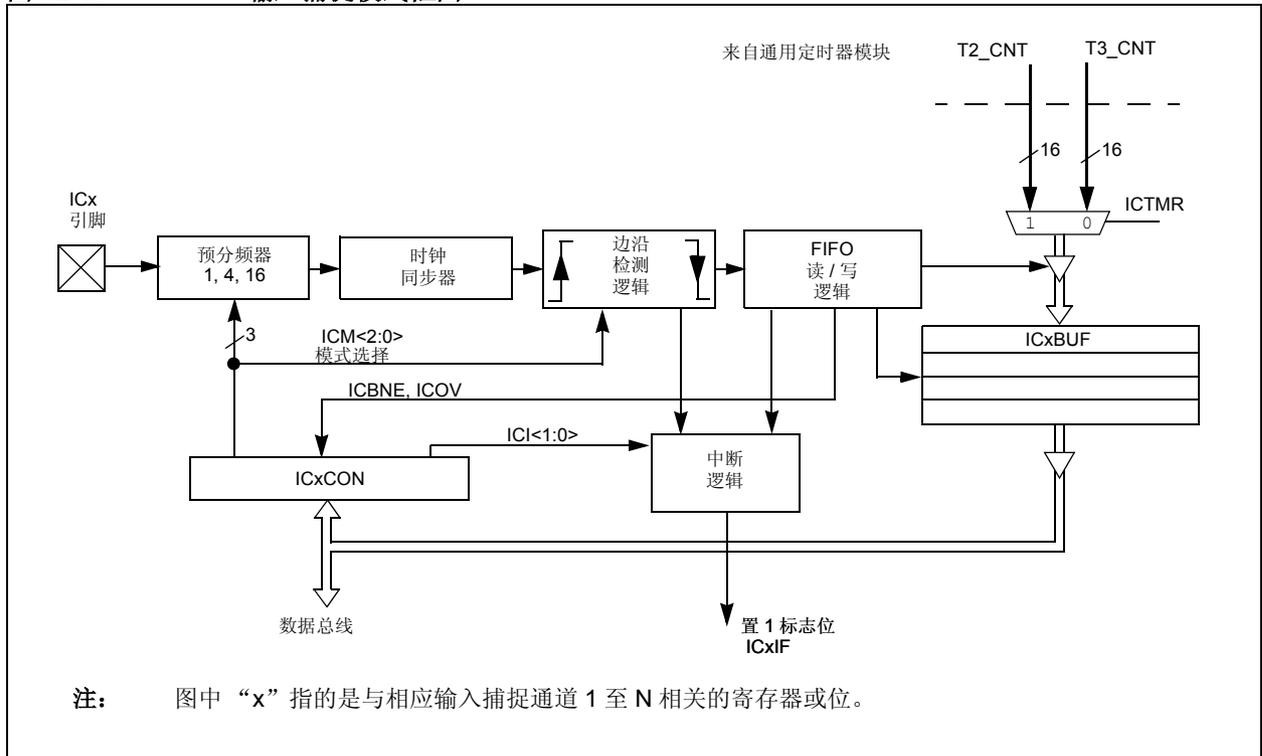
- 在每个下降沿进行捕捉
- 在每个上升沿进行捕捉
- 在每第 4 个上升沿进行捕捉
- 在每第 16 个上升沿进行捕捉
- 在每个上升沿和下降沿进行捕捉

通过设置适当的位 ICM<2:0> (ICxCON<2:0>)，选择上述简单输入捕捉模式。

12.1.1 捕捉预分频器

有四种输入捕捉预分频器设置，它们由位 ICM<2:0> (ICxCON<2:0>) 指定。每当捕捉通道关闭时，预分频器计数器将清零。另外，任何复位都将清零预分频器计数器。

图 12-1: 输入捕捉模式框图



12.1.2 捕捉缓冲器操作

每个捕捉通道都有一个相关的 FIFO 缓冲器，缓冲器宽 16 位，深度为 4 级。有两个状态标志，提供 FIFO 缓冲器的状态：

- ICBNE——输入捕捉缓冲器非空
- ICOV——输入捕捉溢出

ICBFNE 将在第一个输入捕捉事件发生时被置 1，并将保持置 1 状态，直到所有的捕捉事件都已从 FIFO 中读出。每次从 FIFO 中读出一个字，余下的字将在缓冲器中前进一个位置。

当 FIFO 已满（FIFO 中有 4 个捕捉事件）时，如果在读 FIFO 之前发生第五个捕捉事件，则将产生溢出，且 ICOV 位将被设置成逻辑 1。第五个捕捉事件将丢失，不会存入 FIFO。而且在从缓冲器读出全部四个事件之前不会再捕捉任何事件。

如果 FIFO 已空、且还没有收到新的捕捉事件，这时读 FIFO 将产生不确定的结果。

12.1.3 TIMER2 和 TIMER3 选择模式

每个捕捉通道均可选择两个定时器（Timer2 和 Timer3）之一作为时基。

通过特殊功能寄存器位 ICTMR（ICxCON<7>），来进行定时器资源的选择。对于输入捕捉模块，Timer3 是缺省的可用定时器资源。

12.1.4 霍尔传感器模式

当输入捕捉模块设置为在每个边沿（上升沿和下降沿，ICM<2:0> = 001）进行捕捉时，输入捕捉逻辑将执行下列操作：

- 输入捕捉中断标志在每个边沿（上升沿和下降沿）置 1。
- 捕捉时产生中断模式设置位 ICI<1:0> 被忽略，因为每次捕捉都产生中断。
- 在此模式下，不产生捕捉溢出。

12.2 休眠和空闲模式下的输入捕捉工作

当器件在 CPU 空闲或休眠模式下时，输入捕捉事件将使器件唤醒。如果允许中断，还将产生中断。

如果 ICM<2:0> = 111 且中断允许位置 1，那么当捕捉事件发生时，输入捕捉模块将使 CPU 从休眠或空闲模式中唤醒，这与使能的定时器无关。如果处理中断的条件已经得到满足的话，上述的唤醒还将产生中断。唤醒特性可以用来增加额外的外部引脚中断。

12.2.1 CPU 休眠模式下的输入捕捉

CPU 在休眠模式下时其输入捕捉模块能够工作，但功能有所限制。在 CPU 休眠模式下，ICI<1:0> 位不适用，输入捕捉模块只能用作外部中断源。

当器件处于休眠模式下时，为了使用输入捕捉模块，必须把捕捉模块设置成仅在上升沿产生中断（ICM<2:0> = 111）。在此模式下，4:1 或 16:1 的预分频比设定不适用。

12.2.2 CPU 空闲模式下的输入捕捉

CPU 在空闲模式时其输入捕捉模块能够工作，但功能有所限制。在 CPU 空闲模式下，通过 ICI<1:0> 位选择的的中断模式是适用的，由控制位 ICM<2:0> 定义的 4:1 和 16:1 捕捉预分频比设定同样也适用。这个模式要求使能所选择的定时器。此外，ICSIDL 位必须置为逻辑 0。

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

12.3 输入捕捉中断

输入捕捉通道具有根据选定的捕捉事件数来产生中断的能力。通过控制位 ICI<1:0>（ICxCON<6:5>）来设定选定的捕捉事件数。

每个通道都有一个中断标志位（ICxIF）。各捕捉通道中断标志位于相应的 IFSx 状态寄存器中。

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

表 12-1: 输入捕捉寄存器映射

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	输入捕捉 1 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	输入捕捉 2 寄存器																uuuu uuuu uuuu uuuu
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC3BUF	0148	输入捕捉 3 寄存器																uuuu uuuu uuuu uuuu
IC3CON	014A	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC4BUF	014C	输入捕捉 4 寄存器																uuuu uuuu uuuu uuuu
IC4CON	014E	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC5BUF	0150	输入捕捉 5 寄存器																uuuu uuuu uuuu uuuu
IC5CON	0152	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC6BUF	0154	输入捕捉 6 寄存器																uuuu uuuu uuuu uuuu
IC6CON	0156	—	—	ICSIDL	—	—	—	—	—	ICTMR	ICI<1:0>	ICOV	ICBNE	ICM<2:0>				0000 0000 0000 0000
IC7BUF	0158	输入捕捉 7 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	输入捕捉 8 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 = 未初始化位

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

dsPIC30F6010A/6015

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，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

本节对输出比较模块和相关工作模式进行了介绍。此模块提供的特性适用于需要以下工作模式的应用：

- 产生可变宽度输出脉冲
- 功率因数校正

图 13-1 给出了输出比较模块的框图。

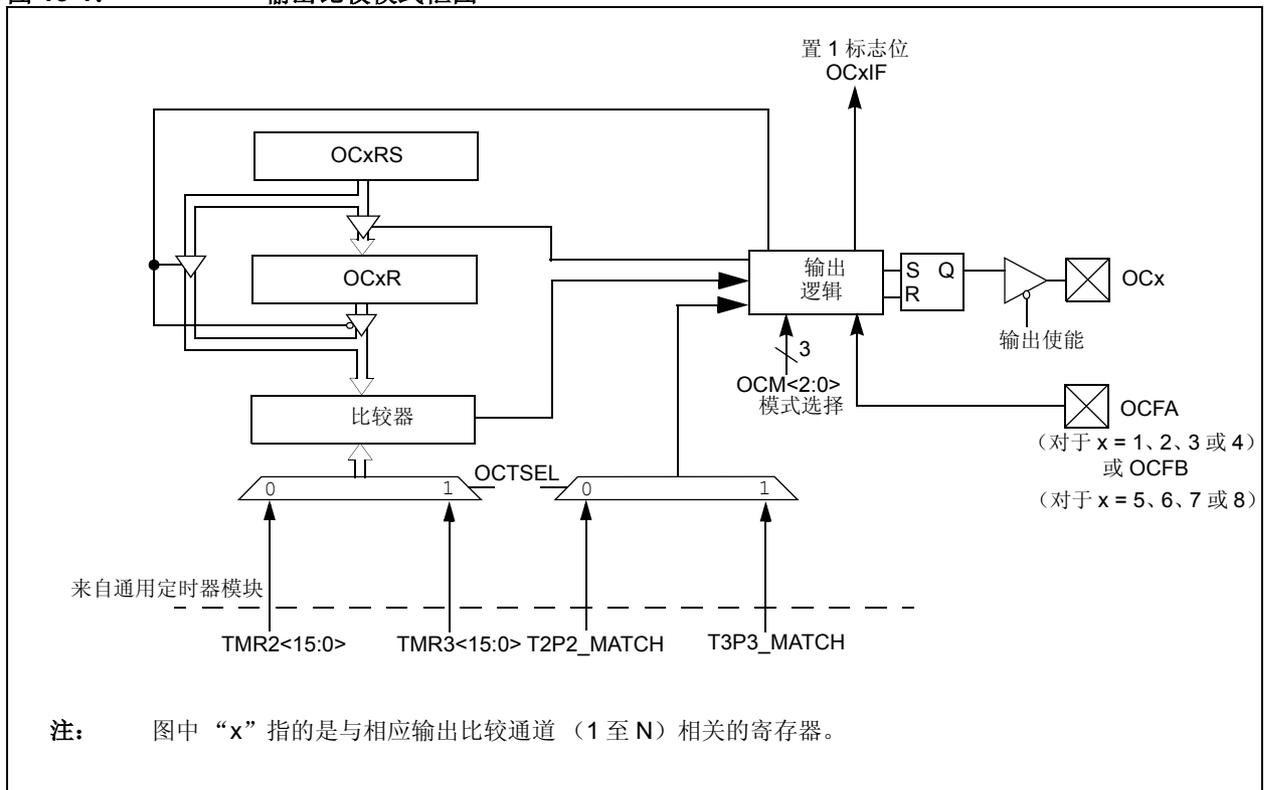
输出比较模块的主要工作特性包括：

- Timer2 和 Timer3 选择模式
- 简单输出比较匹配模式
- 双输出比较匹配模式
- 简单 PWM 模式
- 休眠和空闲模式下的输出比较
- 发生输出比较 /PWM 事件时将产生中断

通过设定 16 位 OCxCON SFR（其中 $x = 1, 2, 3 \dots N$ ）中相应的位可确定这些操作模式。dsPIC30F6010A 和 dsPIC30F6015 器件有 8 个比较通道。

图 13-1 中所示的 OCxRS 和 OCxR 表示双比较寄存器。在双比较模式下，OCxR 寄存器用于第一次比较而 OCxRS 用于第二次比较。

图 13-1: 输出比较模式框图



13.1 Timer2 和 Timer3 选择模式

每个输出比较通道均可选择两个 16 位定时器（Timer2 或 Timer3）中的一个。

通过 OCTSEL 位（OCxCON<3>）来控制定时器的选择。Timer2 是输出比较模块默认的定时器资源。

13.2 简单输出比较匹配模式

当控制位 OCM<2:0>（OCxCON<2:0>）= 001、010 或 011 时，所选输出比较通道被配置为下列三种简单输出比较匹配模式之一：

- 比较强制 I/O 引脚为低电平
- 比较强制 I/O 引脚为高电平
- 比较使 I/O 引脚状态翻转

上述模式中使用了 OCxR 寄存器。OCxR 寄存器将装入一个值，然后与所选的递增定时器的计数值作比较。进行比较时，将发生上述比较匹配模式之一。如果在取出 OCxR 中的值之前，计数器复位为零，则 OCx 引脚的状态保持不变。

13.3 双输出比较匹配模式

当控制位 OCM<2:0>（OCxCON<2:0>）= 100 或 101 时，所选输出比较通道被配置为下列两种双输出比较匹配模式之一：

- 单输出脉冲模式
- 连续输出脉冲模式

13.3.1 单脉冲模式

用户要将模块配置为生成单输出脉冲，需要执行下列步骤（假设定时器已关闭）：

- 确定指令周期时间 Tcy。
- 根据 Tcy 计算所需的脉冲宽度值。
- 计算从定时器起始值 0x0000 到脉冲开始的时间。
- 把脉冲宽度开始和停止时间分别写入 OCxR 和 OCxRS 比较寄存器（x 表示通道 1、2,...,N）。
- 设置定时器周期寄存器的值，使之等于或大于 OCxRS 比较寄存器中的值。
- 设置 OCM<2:0> = 100。
- 使能定时器，TON（TxCON<15>）= 1。

要启动另一个单脉冲，进行另一次写操作设置 OCM<2:0> = 100 即可。

13.3.2 连续脉冲模式

用户要将模块配置为生成连续输出脉冲，则需要执行下列步骤：

- 确定指令周期时间 Tcy。
- 根据 Tcy 计算所需的脉冲宽度值。
- 计算从定时器起始值 0x0000 到脉冲开始的时间。
- 把脉冲宽度开始和停止时间分别写入 OCxR 和 OCxRS 比较寄存器（x 表示通道 1、2,...,N）。
- 设置定时器周期寄存器的值，使之等于或大于 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，所选输出比较通道配置为 PWM 工作模式。当配置为 PWM 工作模式时，OCxR 是主锁存器（只读），OCxRS 是辅助锁存器。这能使 PWM 电平跳变时不会产生毛刺。

为了把输出比较模块配置为 PWM 工作模式，用户必须执行下列步骤：

1. 写适当的周期寄存器，设置 PWM 周期。
2. 写 OCxRS 寄存器，设置 PWM 占空比。
3. 把输出比较模块配置为 PWM 工作模式。
4. 设置 TMRx 预分频值，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，所选输出比较通道还是配置为 PWM 工作模式，但具有输入故障保护的附加功能。在此模式下，如果在 OCFA/B 引脚上检测到逻辑 0，则对应的 PWM 输出引脚将被置于高阻抗输入状态。OCFLT 位（OCxCON<4>）表明是否产生故障条件。此状态将保持到下列事件都已发生为止：

- 外部故障条件已经消除。
- 通过写适当的控制位，重新使能了 PWM 模式。

13.4.2 PWM 周期

通过写入 PRx 寄存器可以指定 PWM 周期。可使用公式 13-1 计算 PWM 周期。

公式 13-1: PWM 周期

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

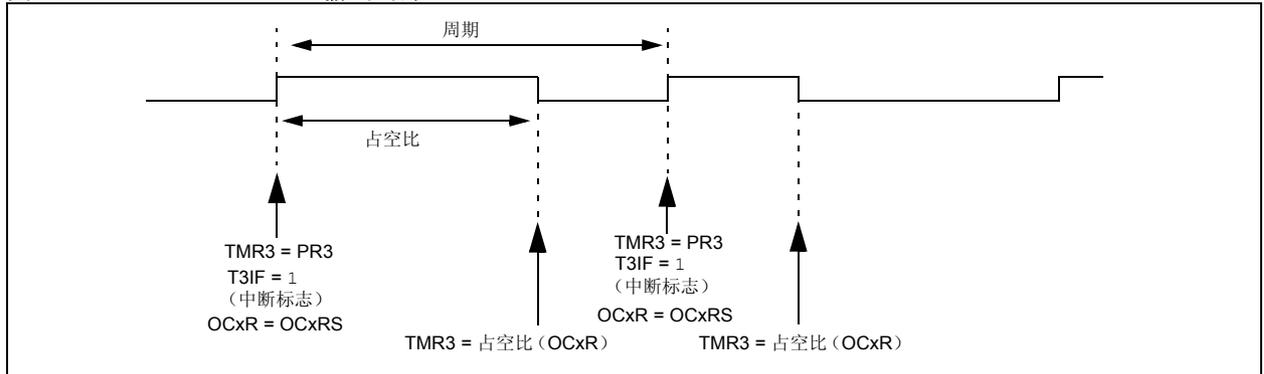
PWM 频率定义为 $1/[\text{PWM 周期}]$ 。

当所选 TMRx 等于其对应的周期寄存器 PRx 时，在下一个递增周期将发生如下四个事件：

- TMRx 清零。
- OCx 引脚置 1。
 - 例外 1: 如果 PWM 占空比为 0x0000，则 OCx 引脚将保持为低电平。
 - 例外 2: 如果占空比大于 PRx，则引脚将保持为高电平。
- 把 PWM 占空比从 OCxRS 锁存到 OCxR 中。
- 将相应的定时器中断标志置 1。

请参见图 13-2 获取关键的 PWM 周期比较。为了清晰起见，图中引用的是 Timer3。

图 13-2: PWM 输出时序



13.5 CPU 休眠模式下的输出比较操作

当 CPU 进入休眠模式时，所有的内部时钟都将停止。从而，在 CPU 进入休眠状态时，输出比较通道将把引脚驱动为 CPU 进入休眠模式之前的有效状态。

例如，当 CPU 进入休眠状态时，如果引脚为高电平，则引脚将继续保持为高电平。类似地，当 CPU 进入休眠状态时，如果引脚为低电平，它将继续保持为低电平。在这两种情况下，当器件唤醒时，输出比较模块将恢复工作。

13.6 CPU 空闲模式下的输出比较操作

当 CPU 进入空闲模式时，输出比较模块仍然能够全功能工作。

如果 OCSIDL 位 (OCxCON<13>) 为逻辑 0，所选时基 (Timer2 或 Timer3) 使能且所选定时器的 TSIDL 位为逻辑 0，输出比较通道将在 CPU 空闲模式下工作。

13.7 输出比较中断

输出比较通道具有在比较匹配时产生中断的能力，而与选择的匹配模式无关。

对于除 PWM 之外的所有模式，当比较事件发生时，相应的中断标志 (OCxIF) 置 1，如果允许中断的话还将产生中断。OCxIF 位位于相应的 IFS 状态寄存器中，且必须用软件清零。通过相应的比较中断允许位 (OCxIE)，来允许中断；OCxIE 位位于相应的 IEC 控制寄存器中。

对于 PWM 模式，当事件发生时，对应的定时器中断标志位 (T2IF 或 T3IF) 置 1，如果允许中断的话还将产生中断。IF 位在 IFS0 状态寄存器中，且必须用软件清零。通过位于 IEC0 控制寄存器中的定时器中断允许位 (T2IE 或 T3IE)，以允许相应的中断。在 PWM 工作模式下，输出比较中断标志始终不会置 1。

表 13-1: 输出比较寄存器映射

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	
OC5RS	0198	输出比较 5 辅助寄存器																0000 0000 0000 0000
OC5R	019A	输出比较 5 主寄存器																0000 0000 0000 0000
OC5CON	019C	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	
OC6RS	019E	输出比较 6 辅助寄存器																0000 0000 0000 0000
OC6R	01A0	输出比较 6 主寄存器																0000 0000 0000 0000
OC6CON	01A2	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	
OC7RS	01A4	输出比较 7 辅助寄存器																0000 0000 0000 0000
OC7R	01A6	输出比较 7 主寄存器																0000 0000 0000 0000
OC7CON	01A8	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	
OC8RS	01AA	输出比较 8 辅助寄存器																0000 0000 0000 0000
OC8R	01AC	输出比较 8 主寄存器																0000 0000 0000 0000
OC8CON	01AE	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000 0000 0000 0000	

图注: u = 未初始化位

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

14.0 正交编码器接口 (QEI) 模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

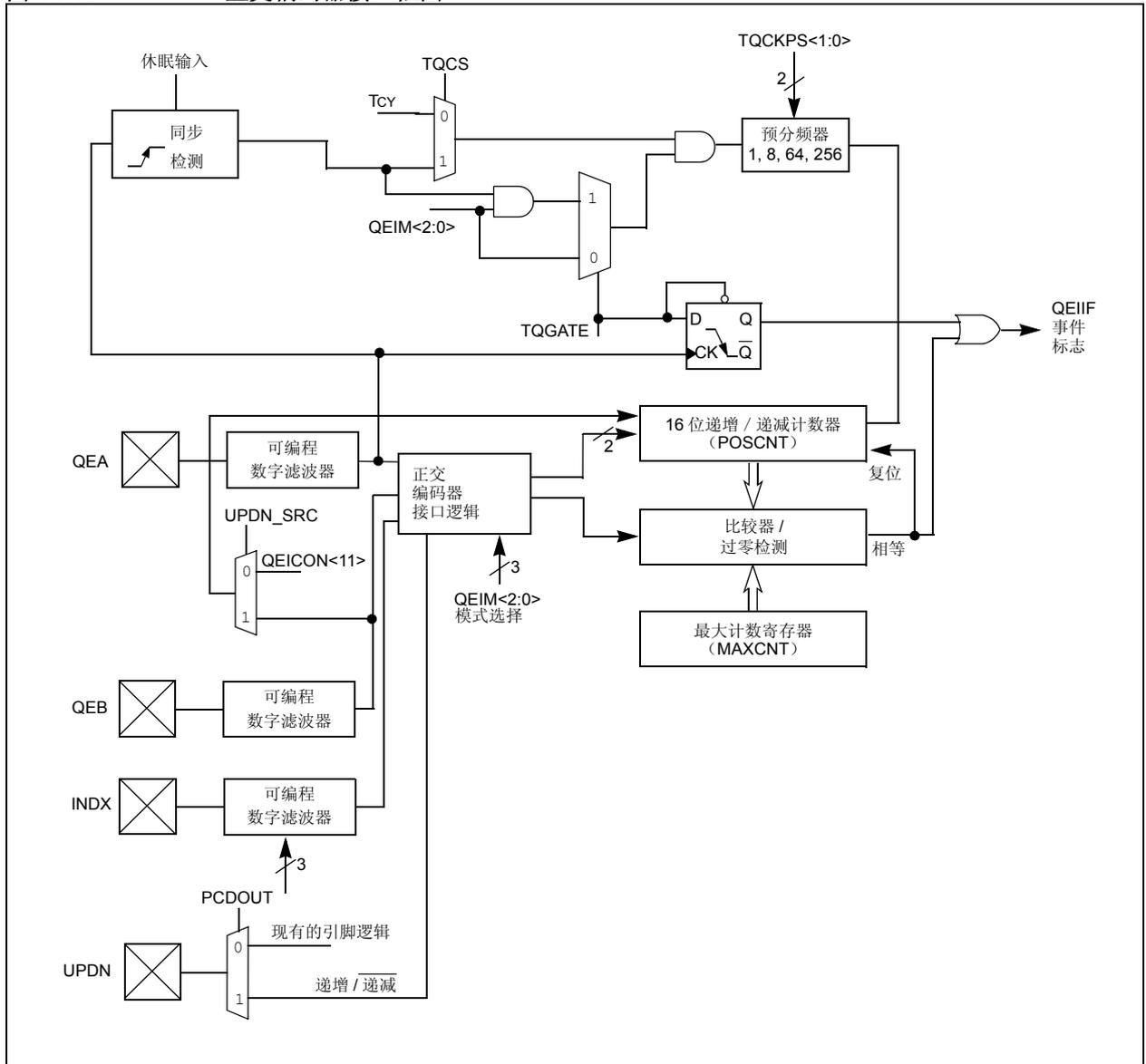
本章介绍了正交编码器接口 (Quadrature Encoder Interface, QEI) 模块和相关的工作模式。QEI 模块提供了与增量式编码器的接口以获得机械位置数据。

QEI 的工作特性包括：

- 三个输入通道，分别为两相信号和索引脉冲输入
- 16 位递增 / 递减位置计数器
- 计数方向状态
- 位置测量 (x2 和 x4) 模式
- 输入端上的可编程数字噪声滤波器
- 备用的 16 位定时器 / 计数器模式
- 正交编码器接口中断

通过设置相应位 $QEIM<2:0>$ ($QEICON<10:8>$) 可确定这些工作模式。图 14-1 给出了正交编码器接口的框图。

图 14-1: 正交编码器接口框图



14.1 正交编码器接口逻辑

典型的增量式 (incremental) 编码器 (也称为光电式编码器) 具有三个输出: A 相、B 相和索引脉冲。在 ACIM 和 SR 电机的位置和速度控制中, 这三个信号非常有用, 且通常是必须的。

A 相 (QEA) 和 B 相 (QEB) 通道具有特定的相位关系。如果 A 相超前 B 相, 那么认为电机正向旋转。如果 A 相滞后于 B 相, 那么认为电机反向旋转。

第三个通道称为索引脉冲, 每转一圈产生一个脉冲, 作为基准用来确定绝对位置。索引脉冲与 A 相和 B 相一致, 皆为低电平。

14.2 16 位递增 / 递减位置计数器模式

16 位递增 / 递减计数器在每一个计数脉冲递增或递减计数, 该脉冲信号由 A 相和 B 相输入信号的关系确定。计数器作为一个积分器, 其计数值与位置成正比。计数方向由 UPDN 信号确定, 该信号由正交编码器接口逻辑产生。

14.2.1 位置计数器错误校验

CNTERR 位 (QEICON<15>) 提供了 QEI 中的位置计数错误校验功能。只有当位置计数器配置为通过索引脉冲复位模式 (QEIM<2:0> = 110 或 100) 时, 才能使用错误校验功能。在这些模式下, POSCNT 寄存器中的内容将与值 (0xFFFF 或 MAXCNT + 1, 取决于转向) 进行比较。如果检测到这些值, CNTERR 位将置 1, 产生错误条件并产生 QEI 计数错误中断。通过置 1 CEID 位 (DFLTCON<8>) 可禁止 QEI 计数错误中断。在检测到错误之后, 位置计数器仍将继续对编码器脉冲边沿进行计数。POSCNT 寄存器继续递增 / 递减计数, 直至出现自然计满溢出 / 下溢。发生自然计满溢出 / 下溢事件时不产生中断。CNTERR 位是一个可读 / 写位, 由用户采用软件方式对其进行复位。

14.2.2 位置计数器复位

位置计数器复位使能位 POSRES (QEI<2>) 用于确定当检测到索引脉冲时是否对位置计数器进行复位。只有当 QEIM<2:0> = 100 或 110 时, 此位才可用。

如果 POSRES 位设置为 1, 则位置计数器在检测到索引脉冲时将被复位。如果 POSRES 位设置为 0, 则位置计数器在检测到索引脉冲时将不会复位。位置计数器将继续进行递增或递减计数, 且在发生计满溢出或下溢时进行复位。

当检测到索引脉冲时, 仍将产生中断, 但在检测到位置计数器发生计满溢出 / 下溢时不产生中断。

14.2.3 计数方向状态

如前所述, QEI 逻辑将根据 A 相和 B 相之间的关系产生 UPDN 信号。除输出引脚外, 还将此内部 UPDN 信号的状态传送到 SFR 的 UPDN 位 (QEICON<11>), 该位是只读位。要将该信号的状态施加到 I/O 引脚上, SFR 的 PCDOUT 位 (QEICON<6>) 必须为 1。

14.3 位置测量模式

QEI 支持两种测量模式, 分别称为 x2 和 x4 模式。通过位于 SFR QEICON<10:8> 中的模式选择位 QEIM<2:0> 可选择模式。

当控制位 QEIM<2:0> = 100 或 101 时, 将选择 x2 测量模式, 此时 QEI 逻辑将只通过 A 相输入信号来确定位置计数器递增速率。A 相信号的每个上升沿和下降沿都将导致位置计数器递增或递减。和在 x4 模式中一样, B 相信号仍然用于计数器方向的确定。

在 x2 测量模式中, 存在导致位置计数器复位的两种不同原因:

1. 检测到索引脉冲 QEIM<2:0> = 100, 导致位置计数器复位。
2. 计数值与 MAXCNT, QEIM<2:0> = 101 匹配时, 导致位置计数器复位。

当控制位 QEIM<2:0> = 110 或 111 时, 将选择 x4 测量模式, 此时 QEI 逻辑将通过 A 相和 B 相输入信号的每个边沿来确定位置计数器递增速率。A 相和 B 相输入信号的每个边沿都将导致位置计数器递增或递减。

在 x4 测量模式中, 存在导致位置计数器复位的两种不同原因:

1. QEIM<2:0> = 110 时, 检测到索引脉冲导致位置计数器复位。
2. QEIM<2:0> = 111 时, 计数值与 MAXCNT 匹配导致位置计数器复位。

x4 测量模式提供了更精细的分辨率数据 (更多次位置计数) 用以确定电机位置。

14.4 可编程数字噪声滤波器

数字噪声滤波器模块负责抑制输入捕捉或正交信号中的噪声。施密特触发器输入和三时钟周期延迟滤波器共同用来抑制低电平噪声和短时的大噪声尖峰，这些噪声干扰经常出现在易受噪声干扰的应用中，例如电机系统。

该滤波器可以确保在三个连续的滤波器周期内都获得同一个稳定值之后，才允许经过滤波的输出信号发生变化。

对于 QEA、QEB 和 INDX 引脚，可通过 QECK<2:0> (DFLTCON<6:4>) 位对数字滤波器的时钟分频频率进行编程设定，该频率源自基本指令周期 Tcy。

要使用 QEA、QEB 和 INDX 通道的滤波器输出功能，QEOUT 位必须为 1。在 POR 和 BOR 时，所有通道的滤波器网络将被禁止。

14.5 备用 16 位定时器 / 计数器

当 QEI 模块未被配置为 QEI 模式 QEIM<2:0> = 001 时，此模块可被配置为简单的 16 位定时器 / 计数器。通过 QEICON SFR 寄存器可实现辅助定时器的设置和控制。此定时器功能与 Timer1 相同。QEA 引脚用作定时器时钟输入。

当配置为定时器时，POSCNT 寄存器作为定时器计数寄存器，而 MAXCNT 寄存器用作周期寄存器。当定时器 / 周期寄存器匹配时，QEI 中断标志将被置 1。

此定时器和通用定时器之间唯一的差异是增加了外部递增 / 递减输入选择功能。当 UPDN 引脚为高电平时，定时器将进行递增计数。当 UPDN 引脚为低电平时，定时器将进行递减计数。

注： 工作模式的改变（即，从 QEI 改变到定时器模式或反之）将不会对定时器 / 位置计数寄存器内容造成影响。

UPDN 控制 / 状态位 (QEICON<11>) 可用来选择定时器寄存器的计数方向状态。当 UPDN = 1 时，定时器将进行递增计数。当 UPDN = 0 时，定时器将进行递减计数。

此外，控制位 UDSRC (QEICON<0>) 用来确定定时器计数方向状态是取决于写入 UPDN 控制 / 状态位 (QEICON<11>) 的逻辑状态还是 QE 引脚的状态。当 UDSRC = 1 时，定时器计数方向由 QEB 引脚控制。同样，当 UDSRC = 0 时，定时器计数方向由 UPDN 引脚控制。

注： 此定时器不支持外部异步计数器工作模式。如果使用外部时钟源，时钟将自动与内部指令周期同步。

14.6 CPU 休眠模式下的 QEI 模块操作

14.6.1 CPU 休眠模式下的 QEI 操作

在 CPU 休眠模式下 QEI 模块将停止工作。

14.6.2 CPU 休眠模式下的定时器操作

在 CPU 休眠模式期间，由于禁止了内部时钟，定时器将不工作。

14.7 CPU 空闲模式下的 QEI 模块操作

由于 QEI 模块可作为正交编码器接口，或 16 位定时器，以下部分介绍了这两种模式下的 QEI 模块操作。

14.7.1 CPU 空闲模式下的 QEI 操作

当 CPU 处于空闲模式时，如果 QEISIDL 位 (QEICON<13>) = 0，则 QEI 模块将工作。在 POR 和 BOR 时，该位默认为逻辑 0。在 CPU 空闲模式下若要使 QEI 模块停止工作，应将 QEISIDL 位设定为 1。

14.7.2 CPU 空闲模式下的定时器操作

当 CPU 处于空闲模式且 QEI 模块配置为 16 位定时器模式时，如果 QEISIDL 位 (QEICON<13>) = 0，则 16 位定时器将工作。在 POR 和 BOR 时，该位默认为逻辑 0。在 CPU 空闲模式下若要使定时器模块停止工作，应将 QEISIDL 位设定为 1。

如果 QEISIDL 位被清零，定时器将正常工作，就象 CPU 没有进入空闲模式一样。

14.8 正交编码器接口中断

正交编码器接口能够在发生以下事件时产生中断：

- 当 16 位递增 / 递减位置计数器发生计满溢出 / 下溢时产生中断
- 检测到合格的索引脉冲时，或 CNTERR 位置 1 时
- 定时器周期匹配事件（溢出 / 下溢）
- 门控累加事件

在发生以上任一事件时，QEI 中断标志位 QEIIF 将被置 1。QEIIF 位必须由软件清零。QEIIF 位于 IFS2 状态寄存器中。

可通过相应的中断允许位 QEIIE 允许中断。QEIIE 位于 IEC2 控制寄存器中。

表 14-1: QEI 寄存器映射

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
QEICON	0122	CNTERR	—	QEISIDL	INDX	UPDN	QEIM<2:0>		SWPAB	PCDOUT	TQGATE	TQCKPS<1:0>		POSRES	TQCS	UDSRC	0000 0000 0000 0000	
DFLTCON	0124	—	—	—	—	—	IMV<1:0>		CEID	QEOUT	QECK<2:0>		—	—	—	—	0000 0000 0000 0000	
POSCNT	0126	位置计数器 <15:0>															0000 0000 0000 0000	
MAXCNT	0128	最大计数值 <15:0>															1111 1111 1111 1111	

图注: u = 未初始化的位

注: 有关寄存器位字段的描述, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F6010A/6015

注:

15.0 电机控制 PWM 模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

此模块简化了产生多种同步脉宽调制输出 (Pulse Width Modulated, PWM) 的任务。特别是它还能支持以下电源和电机控制应用：

- 三相交流感应电机
- 开关磁阻 (Switched Reluctance, SR) 电机
- 直流无刷 (Brushless DC, BLDC) 电机
- 不间断电源 (Uninterruptible Power Supply, UPS)

PWM 模块具有以下特性：

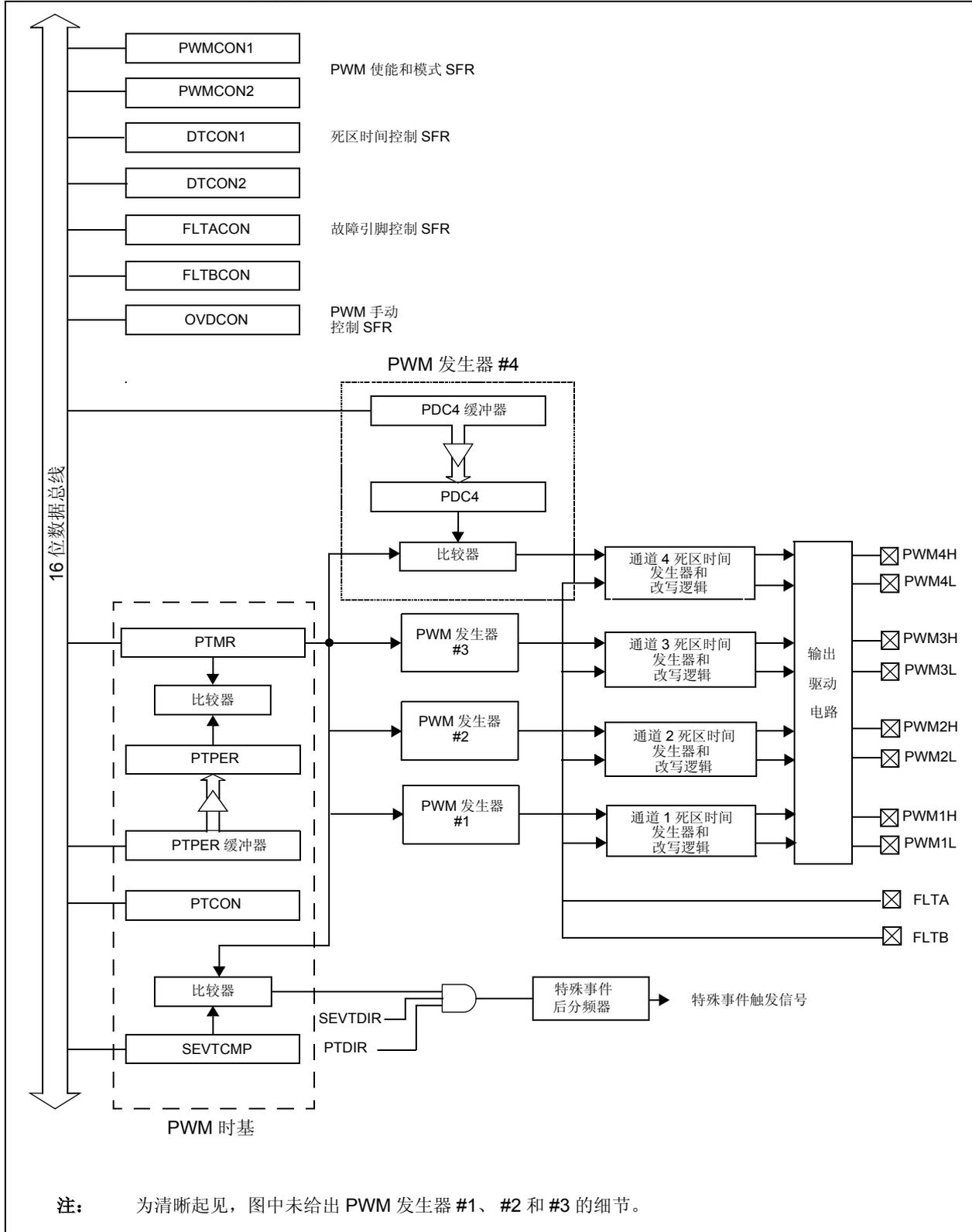
- 8 个具备 4 占空比发生器的 PWM I/O 引脚
- 最高 16 位分辨率
- 实时更改 PWM 频率
- 边沿对齐和中心对齐输出模式
- 单脉冲产生模式
- 对中心对齐模式下不对称更新的中断支持
- 用于电子换相电机 (Electrically Commutative Motor, ECM) 控制的输出改写控制
- 用于调度其他外设事件的“特殊事件”比较器
- 可选择将任何一个 PWM 输出引脚驱动为确定状态的故障引脚
- 占空比更新可配置为立即更新或与 PWM 时基同步

此模块包含 4 个占空比发生器，分别编号为 1 至 4。此模块具有 8 个 PWM 输出引脚，分别编号为 PWM1H/PWM1L 至 PWM4H/PWM4L。8 个 I/O 引脚可组合为 4 个高 / 低端引脚对，分别以下标 H 或 L 表示。对于互补的负载，低端 PWM 引脚的状态总是与高端 I/O 引脚的状态相反。

PWM 模块可以实现多种工作模式，有利于实现特定的电源控制应用。

dsPIC30F6010A/6015

图 15-1: PWM 模块框图



15.1 PWM 时基

PWM 时基由带预分频器和后分频器的 15 位定时器提供。通过 PTMR SFR 可对该时基进行访问。PTMR<15> 为只读状态位，而 PTDIR 用于表明 PWM 时基的当前计数方向。如果 PTDIR 被清零，表明 PTMR 正进行递增计数。如果 PTDIR 置 1，表明 PTMR 正进行递减计数。可通过 PTCON SFR 对 PWM 时基进行配置。可通过置 1 / 清零 PTCON SFR 中的 PTEN 位来使能 / 禁止时基。当用软件清零 PTEN 位时，PTMR 将不会被清零。

PTPER SFR 设定 PTMR 的计数周期。用户必须写入 15 位值至 PTPER<14:0>。当 PTMR<14:0> 中的值与 PTPER<14:0> 中的值相匹配时，时基将会被复位为 0 或在下一个时钟周期反转计数方向，这取决于时基的工作模式。

注： 如果周期寄存器设定为 0x0000，定时器将停止计数，且即使特殊事件值也是 0x0000，也不会产生中断和特殊事件触发信号。该模块将不会更新周期寄存器，即使它已是 0x0000；因此，用户必须禁止该模块以更新周期寄存器。

PWM 时基可配置为 4 种不同的工作模式：

- 自由运行模式
- 单事件模式
- 连续递增 / 递减计数模式
- 带双更新中断的连续递增 / 递减计数模式

通过 PTCON SFR 中的 PTMOD<1:0> 位可选择这 4 种模式。递增 / 递减计数模式支持产生中心对齐 PWM。单事件模式使得 PWM 模块可支持某些电子换相电机 (ECM) 的脉冲控制。

PWM 时基产生的中断信号取决于模式选择位 (PTMOD<1:0>) 和 PTCON SFR 中的后分频比控制位 (PTOPS<3:0>)。

15.1.1 自由运行模式

在自由运行模式下，PWM 时基进行递增计数直至与时基周期寄存器 (Time Base Period register, PTPER) 中的设定值相匹配。PTMR 寄存器将在下一输入时钟边沿复位且只要 PTEN 位保持置 1 时基将继续进行递增计数。

当 PWM 时基处于自由运行模式时 (PTMOD<1:0> = 00)，每当计数值与 PTPER 寄存器匹配时，将产生一个中断事件且 PTMR 寄存器复位到零。在此模式下，可使用后分频比选择位来降低中断事件的频率。

15.1.2 单事件模式

在单事件计数模式下，当 PTEN 位置 1 时 PWM 时基将开始递增计数。当 PTMR 寄存器中的值与 PTPER 寄存器匹配时，PTMR 寄存器将在下一个输入时钟边沿被复位，且 PTEN 位将被硬件清零以中止时基的操作。

当 PWM 时基处于单事件模式 (PTMOD<1:0> = 01) 时，在 PTMR 寄存器中的值与 PTPER 寄存器匹配时将产生一个中断事件，PTMR 寄存器会在下一个输入时钟边沿复位为零且 PTEN 位将被清零。在该模式下，后分频比选择位将不起作用。

15.1.3 连续递增 / 递减计数模式

在连续递增 / 递减计数模式下，PWM 时基将进行递增计数直至与 PTPER 寄存器值相匹配。定时器将在下一个输入时钟边沿开始递减计数。PTCON SFR 中的 PTDIR 位为只读位，用于表明计数方向。当定时器进行递减计数时，PTDIR 位将被置 1。

在递增 / 递减计数模式 (PTMOD<1:0> = 10) 下，每当 PTMR 寄存器值为零时都将产生中断事件且 PWM 时基将开始递增计数。在此模式下可使用后分频比选择位来降低中断事件的频率。

15.1.4 双更新模式

在双更新模式（PTMOD<1:0> = 11）下，每当 PTMR 寄存器等于零或发生周期匹配时都将产生中断事件。在该模式下，后分频比选择位将不起作用。

双更新模式向用户提供了另外两种功能。首先，控制环带宽可增加一倍，因为在每一个周期内 PWM 占空比可被更新两次。其次，可产生不对称的中心对齐 PWM 波形，可在某些电机控制应用中用于减少输出波形畸变。

注： 将周期寄存器编程设定为 0x0001 可产生一个连续的中断脉冲，因此必须加以避免。

15.1.5 PWM 时基预分频器

PTMR 的输入时钟（Fosc/4）可通过 PTCN SFR 中的 PTCKPS<1:0> 控制位选择 1:1、1:4、1:16 或 1:64 预分频比。当发生以下事件时，预分频器的计数器将被清零：

- 写入 PTMR 寄存器
- 写入 PTCN 寄存器
- 任何器件复位

当写入 PTCN 时，PTMR 寄存器将不会被清零。

15.1.6 PWM 时基后分频器

PTMR 的匹配输出可选择通过 4 位后分频器进行后分频（可实现 1:1 至 1:16 分频）。

当出现下列事件之一时，后分频器计数器将被清零：

- 写入 PTMR 寄存器
- 写入 PTCN 寄存器
- 任何器件复位

当写入 PTCN 时，PTMR 寄存器将不会被清零。

15.2 PWM 周期

PTPER 是一个 15 位、双缓冲寄存器，用来设定 PWM 时基的计数周期。在下列情况下，PTPER 缓冲器的内容将被装入 PTPER 寄存器：

- 自由运行和单事件模式：与 PTPER 寄存器匹配后当 PTMR 寄存器复位为零时。
- 递增 / 递减计数模式：PTMR 寄存器为零时。

当 PWM 时基被禁止时（PTEN = 0），PTPER 缓冲器中的值将被自动装入 PTPER 寄存器。

可使用公式 15-1 确定 PWM 周期。

公式 15-1: PWM 周期

$$T_{PWM} = \frac{T_{CY} \cdot (PTPER + 1)}{(PTMR \text{ 预分频比})}$$

如果 PWM 时基配置为递增 / 递减计数模式之一，则 PWM 周期可由公式 15-2 确定。

公式 15-2: PWM 周期（递增 / 递减计数模式）

$$T_{PWM} = \frac{T_{CY} \cdot 2 \cdot (PTPER + 0.75)}{(PTMR \text{ 预分频比})}$$

对于给定器件振荡器和 PWM 频率，最大分辨率（以位为单位）可使用公式 15-3 进行确定

公式 15-3: PWM 分辨率

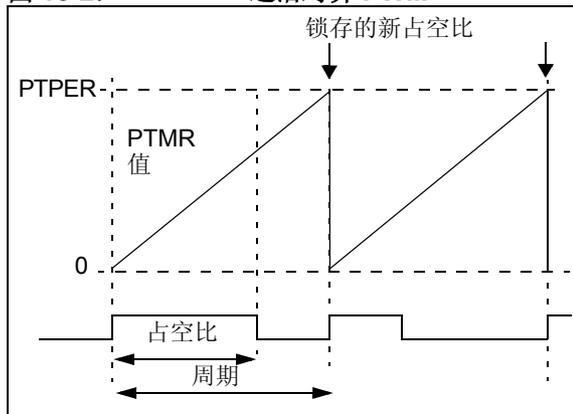
$$\text{分辨率} = \frac{\log(2 \cdot T_{PWM}/T_{CY})}{\log(2)}$$

15.3 边沿对齐 PWM

当 PWM 时基处于自由运行模式或单事件模式时，模块将产生边沿对齐 PWM 信号。对于边沿对齐 PWM 输出，输出信号的周期由 PTPER 中的值指定，而占空比由相应的占空比寄存器指定（见图 15-2）。PWM 输出将在周期开始时（PTMR = 0）驱动为有效状态，而当占空比寄存器中的值与 PTMR 匹配时被驱动为无效状态。

如果特定占空比寄存器的值为零，则对应 PWM 引脚上的输出会在整个 PWM 周期中处于无效状态。此外，如果占空比寄存器中的值大于 PTPER 寄存器中的值，则 PWM 引脚的输出将会在整个 PWM 周期中保持有效状态。

图 15-2: 边沿对齐 PWM



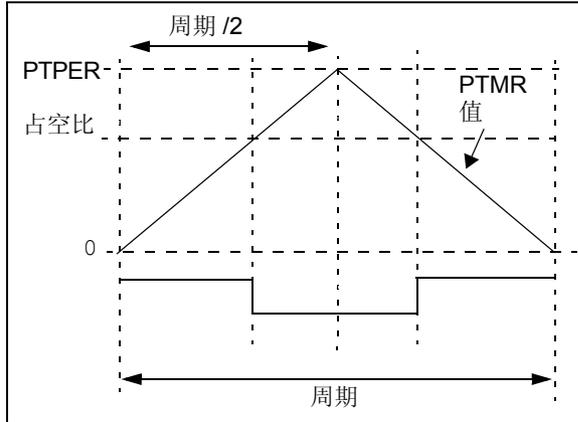
15.4 中心对齐 PWM

当 PWM 时基配置为递增/递减计数模式时（见图 15-3），模块将产生中心对齐的 PWM 信号。

当占空比寄存器值与 PTMR 的值相匹配且 PWM 时基进行递减计数（PTDIR = 1）时，PWM 比较输出将被驱动为有效状态。当 PWM 时基进行递增计数（PTDIR = 0）且 PTMR 寄存器中的值与占空比值匹配时，PWM 比较输出将被驱动为无效状态。

如果特定占空比寄存器中的值为零，则整个 PWM 周期内对应 PWM 引脚上的输出将为无效状态。此外，如果占空比寄存器中的值等于 PTPER 寄存器中的值，则整个 PWM 周期内 PWM 引脚上的输出将为有效状态。

图 15-3: 中心对齐 PWM



15.5 PWM 占空比较单元

有 4 个 16 位特殊功能寄存器（PDC1、PDC2、PDC3 和 PDC4）用来指定 PWM 模块的占空比值。

每一个占空比寄存器中的值负责确定 PWM 输出处于有效状态的时间。占空比寄存器为 16 位宽。占空比寄存器的 LSB 用来确定开始时是否出现 PWM 边沿。因此，PWM 的分辨率实际上提高了一倍。

15.5.1 占空比寄存器缓冲器

这 4 个 PWM 占空比寄存器采用双缓冲方式，因此可对 PWM 输出进行无故障更新。对于每一个占空比，都有两个占空比寄存器与之对应，一个可供用户访问，另一个用来保存当前 PWM 周期使用的实际比较值。

对于边沿对齐 PWM 输出，当计数值与 PTPER 寄存器值匹配且 PTMR 寄存器被复位时，将更新占空比值。当 PWM 时基被禁止（PTEN = 0）且 PWMCON2 寄存器中的 UDIS 位被清零时，占空比缓冲器的内容将被自动装入占空比寄存器。

如果 PWM 时基处于递增/递减计数模式，则当 PTMR 寄存器值为零且 PWM 时基开始递增计数时将采用新的占空比值。当 PWM 时基被禁止（PTEN = 0）时，占空比缓冲器中的值将被自动装入占空比寄存器。

如果 PWM 时基处于双更新功能的递增/递减计数模式，当 PTMR 寄存器值为零且 PTMR 的值与 PTPER 寄存器的值匹配时，将采用新的占空比值。当 PWM 时基被禁止（PTEN = 0），占空比缓冲器中的值将被自动装入占空比寄存器。

15.5.2 占空比立即更新

当立即更新使能位置 1（IUE = 1）时，对占空比寄存器的任何写入操作都将立即更新新的占空比值。这种功能为用户提供了一种选择，使其可以立即更新有效的 PWM 占空比寄存器，而不必等到当前时基周期结束。在闭环伺服应用中，如果使能了立即更新（IUE = 1），则系统稳定性将由于系统观察到发出系统校正命令之间的时间缩短而提高。

如果新占空比写入时 PWM 输出有效，且新占空比小于当前时基的值，则 PWM 脉宽将变窄。如果新占空比写入时 PWM 输出有效，且新占空比大于当前时基的值，则 PWM 脉宽将变宽。

如果新占空比写入时 PWM 输出无效，且新占空比大于当前时基的值，则 PWM 输出将立即有效且对新写入的占空比值保持有效。

dsPIC30F6010A/6015

15.6 互补 PWM 操作

在互补工作模式下，每对 PWM 输出都是互补的 PWM 信号。对于一段两个引脚的输出均为无效的短暂时间，可选择在器件开关过程中插入一个死区时间（见第 15.7 节“死区时间发生器”）。

在互补模式下，占空比较单元将按如下方式分配给 PWM 输出：

- PDC1 寄存器控制 PWM1H/PWM1L 输出
- PDC2 寄存器控制 PWM2H/PWM2L 输出
- PDC3 寄存器控制 PWM3H/PWM3L 输出
- PDC4 寄存器控制 PWM4H/PWM4L 输出

将 PWMCON1 SFR 中相应的 PMODx 位清零可使 PWM I/O 引脚对工作在互补模式下。在器件复位时，PWM I/O 引脚将被设定为互补模式。

15.7 死区时间发生器

当任一 PWM I/O 引脚对工作在互补输出模式时，就可使用死区时间发生功能。PWM 输出采用推挽驱动电路。由于功率输出器件不可能瞬时完成切换，所以必须在关闭互补中的一个 PWM 输出和开启另一个晶体管之间间隔一定的时间。

PWM 模块允许两种可编程的死区时间。这两种死区时间可用于以下两种方法从而增加用户灵活性：

- 对 PWM 输出信号进行优化以使互补晶体管对中的高边和低边晶体管的关断时间不同。在互补对中的低边晶体管的关断事件和高边晶体管的导通事件之间插入第一个死区时间。在高边晶体管的关断事件和低边晶体管的导通事件之间插入第二个死区时间。
- 两个死区时间可分配给两对不同的 PWM I/O 引脚。此工作模式可以使 PWM 模块对每一对 PWM I/O 引脚驱动不同的晶体管 / 负载。

15.7.1 死区时间发生器

PWM 模块的每一对互补输出都有一个 6 位的递减计数器，用于插入死区时间。如图 15-4 所示，每个死区时间单元都有与占空比较输出相连的上升沿和下降沿检测器。

15.7.2 死区时间分配

DTCON2 SFR 包含控制位，可以将两个死区时间分配给互补输出中的任何一个。表 15-1 总结了每个死区时间选择控制位的功能。

表 15-1: 死区时间选择位

位	功能
DTS1A	选择在 PWM1L/PWM1H 有效边沿处插入死区时间
DTS1I	选择在 PWM1L/PWM1H 无效边沿处插入死区时间
DTS2A	选择在 PWM2L/PWM2H 有效边沿处插入死区时间
DTS2I	选择在 PWM2L/PWM2H 无效边沿处插入死区时间
DTS3A	选择在 PWM3L/PWM3H 有效边沿处插入死区时间
DTS3I	选择在 PWM3L/PWM3H 无效边沿处插入死区时间
DTS4A	选择在 PWM4L/PWM4H 有效边沿处插入死区时间
DTS4I	选择在 PWM4L/PWM4H 无效边沿处插入死区时间

15.7.3 死区时间范围

通过指定输入时钟预分频比和一个 6 位无符号值可设定死区时间单元提供的死区时间值。每个单元提供的死区时间值可单独设定。

死区时间单元提供了 4 种输入时钟预分频比选项，用户可根据器件的工作频率选择适当的死区时间范围。可为两个死区时间分别选择时钟预分频比选项。使用 DTCON1 SFR 中的 DTAPS<1:0> 和 DTBPS<1:0> 控制位可选择死区时间时钟预分频比。可使用 4 种时钟预分频比选择之一（Tcy、2 Tcy、4 Tcy 或 8 Tcy）作为各死区时间值。

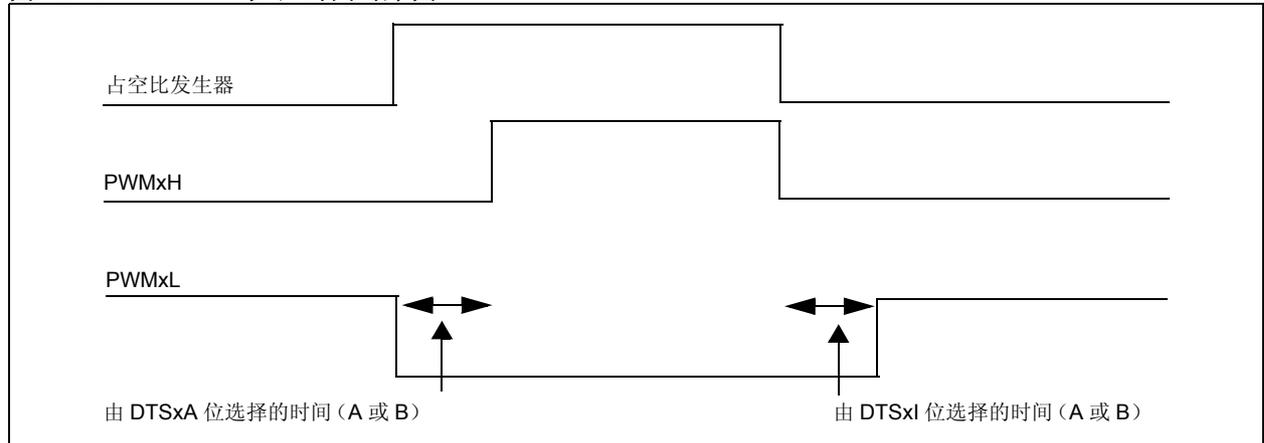
在选择预分频比之后，通过装入两个 6 位无符号值至 DTCON1 SFR 可对死区时间进行调整。

发生以下事件时，死区时间单元预分频器将被清零：

- 由于占空比较边沿事件导致装入递减定时器。
- 写入 DTCON1 或 DTCON2 寄存器。
- 任何器件复位。

注： 当 PWM 模块处于工作状态时（PTEN = 1），用户不应 DTCON1 或 DTCON2 值进行修改，否则可能导致意外结果的发生。

图 15-4: 死区时间时序图



15.8 独立的 PWM 输出

在驱动某些类型的负载时，需要采用独立的 PWM 输出。当 PWMCON1 寄存器中相应的 PMOD 位置 1 时，则相应的一对 PWM 输出将工作在独立输出模式。当模块工作在独立模式时，相邻 PWM I/O 引脚之间将不会实现死区时间控制，而允许一对 I/O 引脚同时处于有效状态。

在独立模式下，每一个占空比发生器将被同时连接到一对 PWM 输出中的两个 I/O 引脚。通过相关的占空比寄存器和 OVDCON 寄存器中的相应位，用户可为工作在独立模式下的每一个 PWM I/O 引脚选择以下的信号输出选项：

- I/O 引脚输出 PWM 信号
- I/O 引脚处于无效状态
- I/O 引脚处于有效状态

15.9 单脉冲 PWM 操作

当 PTCN 控制位 PTMOD<1:0> = 10 时，PWM 模块将产生单脉冲输出。在单脉冲模式下仅可产生边沿对齐输出。在单脉冲模式下，当 PTEN 位置 1 时，PWM I/O 引脚将被驱动为有效状态。当计数值与占空比寄存器设定值匹配时，PWM I/O 引脚将被驱动为无效状态。当计数值与 PTPER 寄存器匹配时，PTMR 寄存器将被清零，所有处于有效状态的 PWM I/O 引脚将被驱动为无效状态，PTEN 位将被清零且将产生中断。

15.10 PWM 输出改写

通过 PWM 输出改写控制位，用户可手动驱动 PWM I/O 引脚至指定逻辑状态，而不受占空比比较单元的影响。

OVDCON 寄存器中包含了所有与 PWM 输出改写功能相关的控制位。OVDCON 寄存器的高字节包含 8 个数据位，即 POVDxH<4:1> 和 POVDxL<4:1>，用于确定哪些 PWM I/O 引脚输出将被改写。OVDCON 寄存器的低字节包含 8 个数据位，即 POUTxH<4:1> 和 POUTxL<4:1>，用于确定当通过 POVD 位对特定输出进行改写时的 PWM I/O 引脚的状态。

15.10.1 互补输出模式

当通过 OVDCON 寄存器驱动 PWMxL 引脚为有效状态时，输出信号将被强制为与输出对应对应 PWMxH 引脚输出状态相反的状态。当通过手动方式对 PWM 通道进行改写时，死区时间插入仍然有效。

15.10.2 改写同步

如果 PWMCON2 寄存器中的 OSYNC 位置 1，所有通过 OVDCON 寄存器执行的输出改写操作都将与 PWM 时基同步。同步输出改写将发生在以下时刻：

- 若是边沿对齐模式，则当 PTMR 为零时。
- 若是中心对齐模式，则当 PTMR 为零且当 PTMR 值与 PTPER 匹配时。

15.11 PWM 输出和极性控制

有 3 个与 PWM 模块有关的器件配置位用来提供 PWM 输出引脚控制：

- HPOL 配置位
- LPOL 配置位
- PWMPIN 配置位

FPORBOR 配置寄存器中的这 3 个位（见第 21.0 节“系统集成”）将与位于 PWMCON1 SFR 中的 4 个 PWM 使能位（PENxH 和 PENxL）配合工作。这些配置位和 PWM 使能位确保在器件发生复位后 PWM 引脚能够处于正确的状态。配置位 PWMPIN 使得 PWM 模块输出可选择在器件复位时被使能。如果 PWMPIN = 0，在复位时 PWM 输出将被驱动为无效状态。若 PWMPIN = 1（默认），PWM 模块将为三态。HPOL 位用于指定 PWMxH 输出的极性，而 LPOL 位指定 PWMxL 输出的极性。

15.11.1 输出引脚控制

PWMCON1 SFR 中的 PEN<4:1>H 和 PEN<4:1>L 控制位将分别用于使能每一个高端 PWM 输出引脚和每一个低端 PWM 输出引脚。如果某个 PWM 输出引脚未被使能，则该引脚将被视作一般 I/O 引脚。

15.12 PWM 故障引脚

有两个与 PWM 模块相关的故障引脚（FLTA 和 FLTB）。当使能这两个引脚时，可选择将每个 PWM I/O 引脚驱动为定义的状态。

15.12.1 故障引脚使能位

FLTACON 和 FLTBCON 特殊功能寄存器各自都包含 4 个控制位，用来确定特定 PWM I/O 引脚对是否由故障输入引脚控制。欲使能某一 PWM I/O 引脚对的故障改写功能，必须将 FLTACON 或 FLTBCON 寄存器中的相应位置 1。

如果 FLTACON 或 FLTBCON 寄存器中所有的使能位都被清零，则相应的故障输入引脚将对 PWM 模块没有影响，且该引脚可用作一般的中断引脚或 I/O 引脚。

注： 故障引脚逻辑可独立于 PWM 逻辑。如果 FLTACON/FLTBCON 寄存器中所有的使能位都被清零，则故障引脚将用作通用中断引脚。每个故障引脚都有与之相关的中断向量、中断标志位和中断优先级位。

15.12.2 故障状态

FLTACON 和 FLTBCON 特殊功能寄存器各自都包含 8 个位，用来确定当通过故障输入改写 PWM I/O 引脚时该引脚的状态。当这些位被清零，PWM I/O 引脚被驱动为无效状态。如果这些位置 1，PWM I/O 引脚将被驱动为有效状态。有效和无效状态与为每个 PWM I/O 引脚所定义的极性有关（HPOL 和 LPOL 极性控制位）。

当 PWM 模块 I/O 对处于互补模式且这两个引脚被设定为在故障条件下为有效状态时，就存在特殊情况。由于 PWMxH 引脚在互补模式下通常都具有优先级，所以这两个引脚不能同时被驱动为有效状态。

15.12.3 故障引脚优先级

如果两个故障输入引脚都用于控制某个 PWM I/O 引脚，则为故障 A 输入引脚设定的故障状态将优先于故障 B 输入引脚的故障状态。

15.12.4 故障输入模式

每个故障输入引脚都有两种工作模式：

- **锁存模式：**当故障引脚驱动为低电平时，PWM 输出将进入 FLTACON/FLTBCON 寄存器所定义的状态。PWM 输出将保持在此状态，直到故障引脚被驱动为高电平并且相应的中断标志由软件清零。当这两种情况都发生时，PWM 输出将在下一个 PWM 周期开始时或在半周期边界返回到正常工作状态。如果中断标志在故障状态结束前清零，PWM 模块将等到故障引脚不再有效时才恢复输出。
- **逐周期模式：**当故障输入引脚被驱动为低电平时，只要故障引脚保持为低电平，PWM 输出将会一直保持在定义的故障状态。在故障引脚被驱动为高电平后，PWM 输出将在下一个 PWM 周期开始时或半周期边界返回正常状态。

故障输入引脚的工作模式可通过 FLTACON 和 FLTBCON 特殊功能寄存器中的 FLTAM 和 FLTBM 控制位进行选择。

故障引脚可通过软件方式进行手动控制。

15.13 PWM 更新锁定

对于复杂的 PWM 应用，用户可能需要在某一时刻对多达 4 个占空比寄存器以及时基周期寄存器 PTPER 进行写入操作。在某些应用中，在装入新的占空比和周期设定值供模块使用之前写入所有的缓冲寄存器是很重要的。

通过将 PWMCON2 SFR 中的 UDIS 控制位置 1 可使能 PWM 更新锁定功能。UDIS 位会影响所有占空比缓冲寄存器和 PWM 时基周期缓冲器 PTPER。当 UDIS = 1 时，占空比或周期值将不会发生变化。

如果置 1 IUE 位，则任何对占空比寄存器的更改都将导致对占空比寄存器的立即更新，与 UDIS 位的状态无关。PWM 周期寄存器 (PTPER) 的更新将不受 IUE 控制位的影响。

15.14 PWM 特殊事件触发器

PWM 模块具有一个特殊事件触发器，可使 A/D 转换与 PWM 时基同步。通过对 A/D 采样和转换的时间进行编程设定，可使其发生在 PWM 周期中的任何时刻。特殊事件触发器使用户将采集 A/D 转换结果的时间和更新占空比值的时间之间的延迟减到最小。

PWM 特殊事件触发器使用一个称为 SEVTCMP 的 SFR 以及 5 个控制位对其操作进行控制。用于产生特殊事件触发信号的 PTMR 值将被装入 SEVTCMP 寄存器。当 PWM 时基处于递增 / 递减计数模式时，还需要一个控制位指定特殊事件触发信号的计数方向。此计数方向是通过使用 SEVTCMP SFR 中的 SEVTDIR 控制位选择的。如果 SEVTDIR 位清零，特殊事件触发信号将在 PWM 时基的递增计数周期内产生。如果 SEVTDIR 位置 1，特殊事件触发信号将在 PWM 时基的递减计数周期内产生。除非 PWM 时基配置为递增 / 递减计数模式，否则 SEVTDIR 控制位将不起作用。

15.14.1 特殊事件触发器后分频器

PWM 特殊事件触发器有一个允许后分频比为 1:1 到 1:16 的后分频器。通过写入 PWMCON2 SFR 中的 SEVOPS<3:0> 控制位可配置后分频器。

特殊事件输出后分频器在发生下列事件时清零：

- 任何写入 SEVTCMP 寄存器的操作
- 任何器件复位

15.15 CPU 休眠模式下的 PWM 操作

故障 A 和故障 B 输入引脚能将 CPU 从休眠模式唤醒。在休眠模式下，如果任何一个故障引脚被驱动为低电平，PWM 模块将产生中断。

15.16 CPU 空闲模式下的 PWM 操作

PTCON SFR 包含 PTSIDL 控制位。该位用于确定当器件进入空闲模式时 PWM 模块将继续工作还是停止工作。如果 PTSIDL = 0，则模块将继续工作。如果 PTSIDL = 1，则只要 CPU 仍然处于空闲模式，模块将停止工作。

表 15-2: 8 输出 PWM 寄存器映射

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
PTCON	01C0	PTEN	—	PTSIDL	—	—	—	—	—	PTOPS<3:0>			PTCKPS<1:0>		PTMOD<1:0>		0000 0000 0000 0000	
PTMR	01C2	PTDIR	PWM 定时器计数值															0000 0000 0000 0000
PTPER	01C4	—	PWM 时基周期寄存器															0111 1111 1111 1111
SEVTCMP	01C6	SEVTDIR	PWM 特殊事件比较寄存器															0000 0000 0000 0000
PWMCON1	01C8	—	—	—	—	PTMOD4	PTMOD3	PTMOD2	PTMOD1	PEN4H	PEN3H	PEN2H	PEN1H	PEN4L	PEN3L	PEN2L	PEN1L	0000 0000 1111 1111
PWMCON2	01CA	—	—	—	—	SEVOPS<3:0>			—	—	—	—	—	IUE	OSYNC	UDIS	0000 0000 0000 0000	
DTCON1	01CC	DTBPS<1:0>		死区时间 B 值					DTAPS<1:0>		死区时间 A 值							0000 0000 0000 0000
DTCON2	01CE	—	—	—	—	—	—	—	—	DTS4A	DTS4I	DTS3A	DTS3I	DTS2A	DTS2I	DTS1A	DTS1I	0000 0000 0000 0000
FLTACON	01D0	FAOV4H	FAOV4L	FAOV3H	FAOV3L	FAOV2H	FAOV2L	FAOV1H	FAOV1L	FLTAM	—	—	—	FAEN4	FAEN3	FAEN2	FAEN1	0000 0000 0000 0000
FLTBCON	01D2	FBOV4H	FBOV4L	FBOV3H	FBOV3L	FBOV2H	FBOV2L	FBOV1H	FBOV1L	FLTBM	—	—	—	FBEN4	FBEN3	FBEN2	FBEN1	0000 0000 0000 0000
OVDCON	01D4	POVD4H	POVD4L	POVD3H	POVD3L	POVD2H	POVD2L	POVD1H	POVD1L	POUT4H	POUT4L	POUT3H	POUT3L	POUT2H	POUT2L	POUT1H	POUT1L	1111 1111 0000 0000
PDC1	01D6	PWM 占空比 #1 寄存器															0000 0000 0000 0000	
PDC2	01D8	PWM 占空比 #2 寄存器															0000 0000 0000 0000	
PDC3	01DA	PWM 占空比 #3 寄存器															0000 0000 0000 0000	
PDC4	01DC	PWM 占空比 #4 寄存器															0000 0000 0000 0000	

图注: u = 未初始化的位

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

16.0 SPI 模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册 I》(DS70046E_CN)。

串行外设接口 (Serial Peripheral Interface, SPI) 模块是同步串行接口。可用于与其他外设 (如 EEPROM、移位寄存器、显示驱动器和 A/D 转换器等) 或单片机进行通信。SPI 模块与 Motorola 的 SPI 和 SIOP 接口兼容。

16.1 操作功能说明

每个 SPI 模块由一个 16 位移位寄存器 SPIxSR (其中 $x = 1$ 或 2) 和一个缓冲器寄存器 SPIxBUF 组成；移位寄存器用来移入和移出数据。控制寄存器 SPIxCON 用来配置模块。此外，状态寄存器 SPIxSTAT 用来表明各种状态条件。

串行接口有 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) 以及 SSx (低电平有效从选择)。

在主模式下工作时，SCK 是时钟输出，但在从模式下为时钟输入。

一组 8 个或 16 个时钟脉冲将数据位从 SPIxSR 移出到 SDOx 引脚，同时将 SDIx 引脚的数据位移入 SPIxSR。当传输完成后将产生中断，相应的中断标志位 (SPI1IF 或 SPI2IF) 置 1。通过中断允许位 (SPI1IE 或 SPI2IE) 可以禁止该中断。

接收操作是双缓冲的。当接收完字节时，将字节从 SPIxSR 送到 SPIxBUF。

当从 SPIxSR 传输新数据到 SPIxBUF 时，如果接收缓冲器满，模块会将 SPIROV 位置 1，表明产生溢出条件。数据从 SPIxSR 向 SPIxBUF 的传送不会完成，新数据将丢失。当 SPIROV 为 1 时，模块将不会对 SCK 电平跳变进行响应；实际上会禁止模块，直到用户软件读 SPIxBUF 为止。

注：若模块配置为仅发送模式，用户必须对 SPIxBUF 进行读操作，以避免产生接收溢出条件 (SPIROV = 1)。

发送写操作同样也是双缓冲的。用户写入 SPIxBUF。当主器件或从器件传输完成后，移位寄存器 (SPIxSR) 的内容将送至接收缓冲器。如果已经向缓冲寄存器写了任何发送数据，发送缓冲器的内容将送至 SPIxSR。于是，接收到的数据存放在 SPIxBUF 中，而 SPIxSR 中的发送数据已经准备就绪，可进行下一次传输。

注：发送缓冲器 (SPIxTXB) 和接收缓冲器 (SPIxRXB) 都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来产生时钟。只要数据写入 SPIxBUF 就会被发送。在最后一个位传输的过程中产生中断。

在从模式下，SCK 上出现外部时钟脉冲时，进行数据的发送和接收。同样，当最后一个位被锁存时产生中断。如果使能了 SSx 控制，那么仅当 SSx 为低电平时使能发送和接收。在 SSx 模式下，当 SSx 为高电平时 SDOx 输出将被禁止。

提供给模块的时钟是 $F_{osc}/4$ 。而后，使用主预分频因子 (PPRE<1:0>) 和辅助预分频因子 (SPRE<2:0>) 对这个时钟进行分频。CKE 位决定是在时钟从有效状态跳变到空闲状态进行发送，还是反之。CKP 位选择时钟的空闲状态 (高电平或低电平)。

16.1.1 字和字节通信

控制位 MODE16 (SPIxCON<10>) 允许模块在 8 位或 16 位模式下通信。16 位通信，除了发送的位数是 16 而不是 8 以外，与 8 位通信相同。

在更改 MODE16 位之前，用户软件必须禁止模块。用户更改 MODE16 位时，SPI 模块将复位。

8 位和 16 位工作之间的基本区别是：8 位工作模式下数据是从 SPIxSR 的 bit 7 开始发送的；而在 16 位工作模式下，则是从 SPIxSR 的 bit 15 开始发送的。在两种模式下，数据都会移入 SPIxSR 的 bit 0。

16.1.2 SDOx 禁止

SPIxCON 寄存器中的控制位 DISSDO，用来禁止 SDOx 输出。这样可以使 SPI 模块配置为仅输入的工作模式。SDO 也可用作通用 I/O 口。

dsPIC30F6010A/6015

图 16-1: SPI 框图

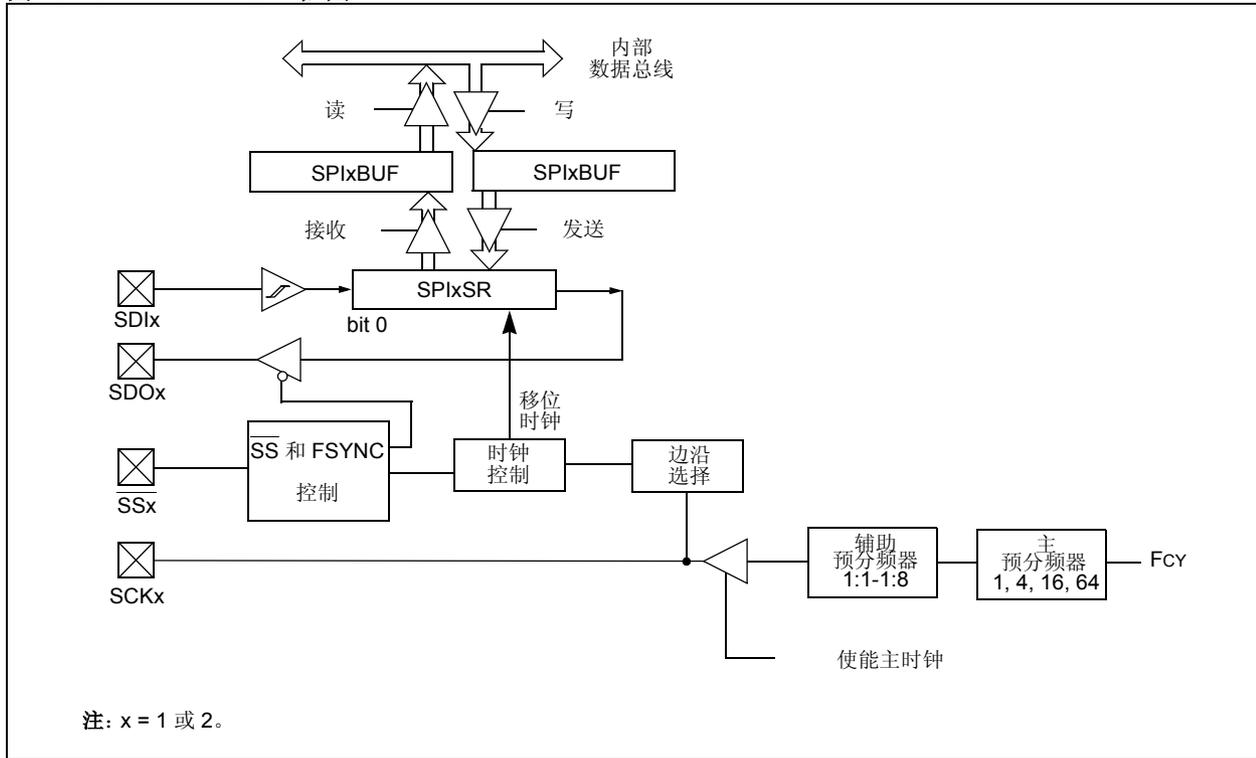
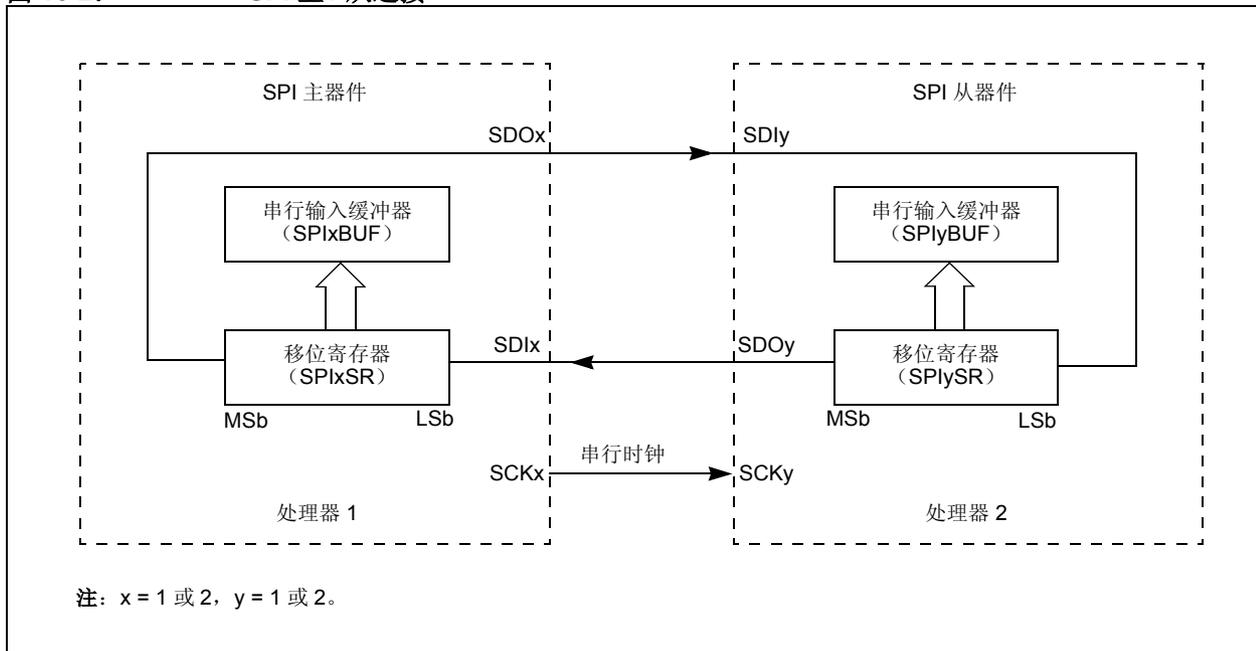


图 16-2: SPI 主 / 从连接



16.2 帧 SPI 支持

当工作在主模式或从模式时，模块支持基本的帧 SPI 协议。控制位 **FRMEN** 使能帧 SPI 支持，并使 **SSx** 引脚用作帧同步脉冲（**FSYNC**）功能。控制位 **SPIFSD** 决定 **SSx** 引脚用作输入还是输出（即，模块是接收还是产生帧同步脉冲）。帧脉冲在一个 SPI 时钟周期内是高电平有效脉冲。当帧同步使能时，仅在 SPI 时钟的下一个发送边沿开始发送数据。

16.3 从选择同步

SSx 引脚允许同步从模式。必须使用 **SSx** 引脚控制使能位（**SSEN** = 1）把 SPI 配置为 SPI 从模式。当 **SSx** 引脚为低电平时，使能发送和接收，并驱动 **SDOx** 引脚。当 **SSx** 引脚变为高电平时，将不再驱动 **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。于是，当 **SSx** 引脚再次为低电平时，发送 / 接收又将从 **MSb** 开始，即使 **SSx** 在发送 / 接收过程中变为高电平也是如此。

16.4 CPU 休眠模式下的 SPI 操作

在休眠期间，SPI 模块关闭。SPI 模块正在进行处理时，若 CPU 进入休眠模式，则发送 / 接收操作将中止。

在休眠模式下，发送器和接收器将停止工作。但进入或退出休眠模式时并不影响寄存器的内容。

16.5 CPU 空闲模式下的 SPI 操作

器件进入空闲模式时，所有时钟源还将继续工作。**SPISIDL** 位（**SPIxSTAT<13>**）选择 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 16-1: SPI1 寄存器映射

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
SPI1STAT	0220	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SPI1CON	0222	—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SPI1BUF	0224	发送和接收缓冲器																0000 0000 0000 0000

图注: u = 未初始化的位

注: 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: SPI2 寄存器映射

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
SPI2STAT	0226	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPIROV	—	—	—	—	SPITBF	SPIRBF	0000 0000 0000 0000
SPI2CON	0228	—	FRMEN	SPIFSD	—	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE2	SPRE1	SPRE0	PPRE1	PPRE0	0000 0000 0000 0000
SPI2BUF	022A	发送和接收缓冲器																0000 0000 0000 0000

图注: u = 未初始化的位

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

17.0 I²C™ 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

I²C™ 模块（带有 16 位接口）为 I²C 串行通信标准的从模式和多主器件模式提供完全的硬件支持。

模块具有以下主要特性：

- I²C 接口支持主模式和从模式工作。
- I²C 从模式支持 7 位和 10 位地址。
- I²C 主模式支持 7 位和 10 位地址。
- I²C 端口允许主器件和从器件之间的双向传输。
- I²C 端口的串行时钟同步可以用作握手机制来暂停和继续串行传输（SCLREL 控制）。
- I²C 支持多主器件工作；检测总线冲突并相应地进行仲裁。

17.1 操作功能说明

硬件完全实现了 I²C 标准和快速模式规范的全部主从功能，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 I²C 总线上的从器件工作。

17.1.1 各种 I²C 模式

支持 I²C 模块工作在以下模式：

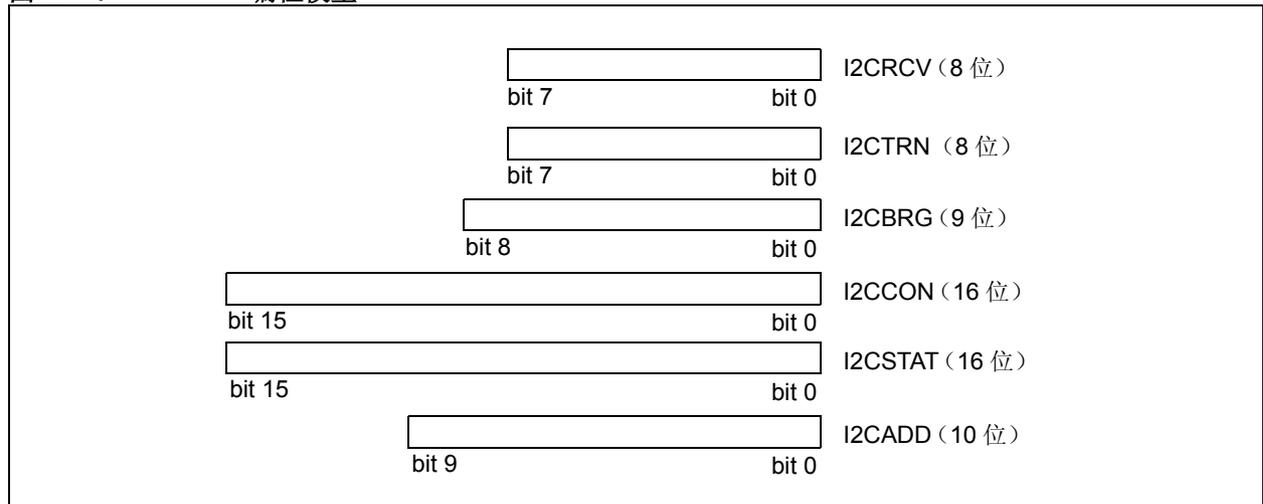
- 7 位地址的 I²C 从模式
- 10 位地址的 I²C 从模式
- 7 位或 10 位地址的 I²C 主模式

请参见图 17-1 中的 I²C 编程模型。

17.1.2 I²C 模式下的引脚配置

I²C 有一个双引脚接口；SCL 引脚为时钟线，SDA 引脚为数据线。

图 17-1: 编程模型



17.1.3 I²C 寄存器

I2CCON 和 I2CSTAT 分别是控制寄存器和状态寄存器。I2CCON 寄存器是可读且可写的。I2CSTAT 的低 6 位是只读的，其余位则是可读且可写的。

I2CRSR 是移位寄存器，用来对数据进行移位，而 I2CRCV 是接收缓冲寄存器，可写入和读出数据字节，如图 16-1 所示。I2CTRN 是发送寄存器，发送操作期间，字节将会写入 I2CTRN，如图 16-2 所示。

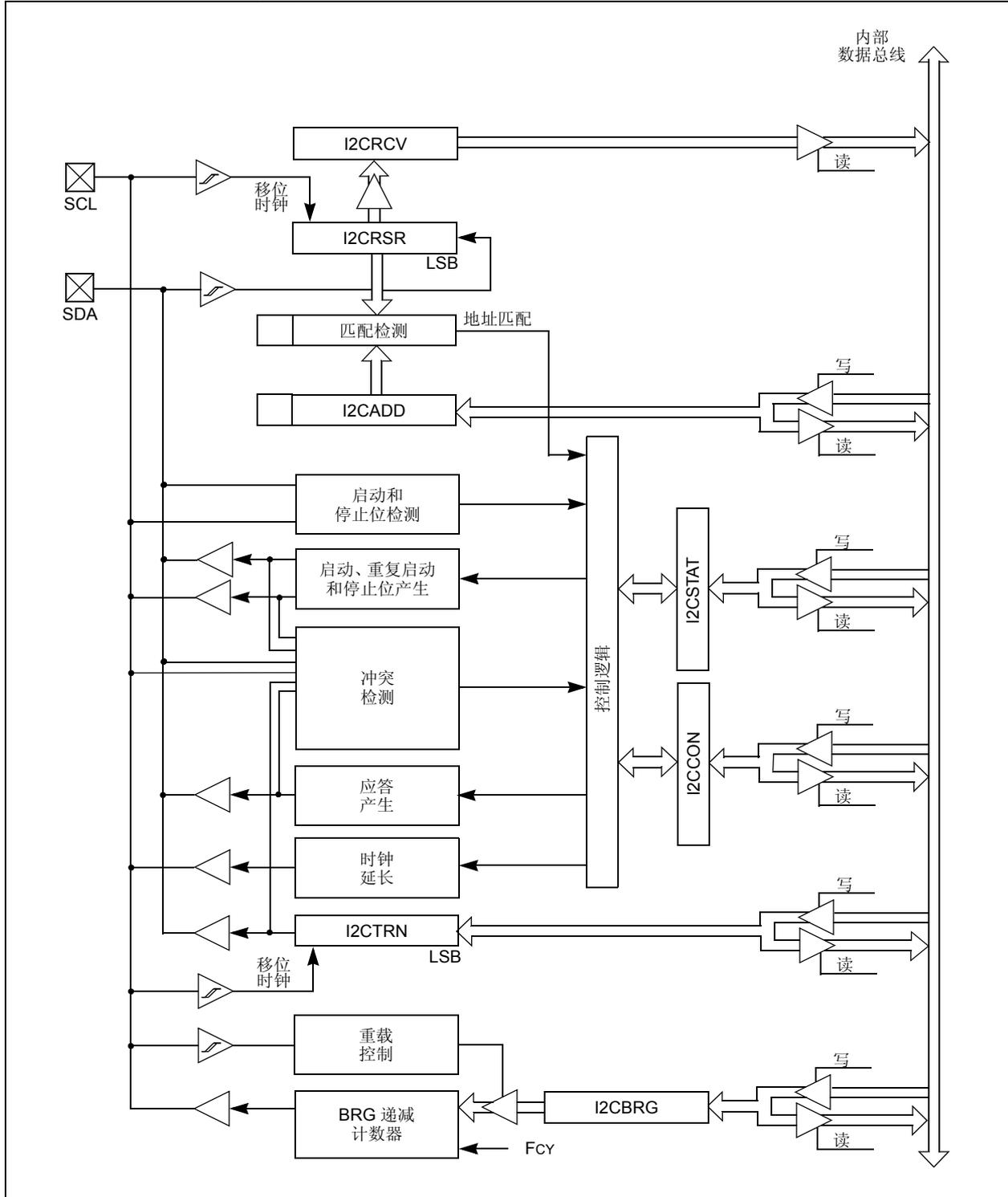
I2CADD 寄存器存放从地址。状态位 ADD10 表明是否为 10 位地址模式。I2CBRG 用来保存波特率发生器的重载值。

在进行接收时，I2CRSR 和 I2CRCV 一起形成双缓冲接收器。在 I2CRSR 接收到一个完整的字节后，字节被送至 I2CRCV 并产生中断脉冲。在发送期间，I2CTRN 不是双缓冲的。

注： 在 10 位模式下，重复启动条件发生后，用户只需匹配前 7 位地址。

dsPIC30F6010A/6015

图 17-2: I²C™ 框图



17.2 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。

如果 A10M 位 (I2CCON<10>) 为 0，模块把地址解释为 7 位地址。在接收到地址时，将收到的地址与 I2CADD 寄存器中的低 7 位进行比较。

如果 A10M 位为 1，将认为地址是 10 位地址。在接收地址时，将收到的地址与二进制数 “1 1 1 1 0 A9 A8” (其中 A9 和 A8 是 I2CADD 的高二位) 进行比较。如果匹配，则下一个地址将与 I2CADD 的低 8 位进行比较，这是 10 位寻址协议规定的。

表 17-1: dsPIC30F 支持的 7 位 I²C™ 从地址

0x00	广播呼叫地址或启动字节
0x01-0x03	保留
0x04-0x07	Hs 模式主机码
0x04-0x77	7 位有效地址
0x78-0x7b	10 位有效地址 (低 7 位)
0x7c-0x7f	保留

17.3 I²C 7 位从模式的工作

一旦使能 (I2CEN = 1)，从模块将等待一个启动位出现 (即，I²C 模块 “空闲”)。检测到启动位之后，会将 8 个位移入 I2CRSR，并将地址与 I2CADD 进行比较。在 7 位寻址方式下 (A10M = 0)，I2CADD<6:0> 位与 I2CRSR<7:1> 位进行比较，而 I2CRSR<0> 是 R_W 位。在 SCL 上升沿采样所有输入位。

如果地址匹配的话，将发送一个应答，在第九个位 (ACK) 的下降沿，从事件中断标志位 (SI2CIF) 置 1。地址匹配不影响 I2CRCV 缓冲器或 RBF 位的内容。

17.3.1 从模式发送

如果收到的 R_W 位为 1，串行端口将进入发送模式。模块将在第九位发送 ACK 信号，并使 SCL 保持为 0，直到 CPU 写 I2CTRN 进行响应为止。通过把 SCLREL 置 1 释放 SCL，而后数据的 8 个位被移出。数据位在 SCL 下降沿移出，因此 SCL 为高 (见时序图) 时 SDA 有效。将在第九个时钟脉冲的下降沿发出中断脉冲，而与从主器件接收到的 ACK 状态无关。

17.3.2 从模式接收

如果地址匹配时，接收到的 R_W 位是 0，就将启动接收模式。在 SCL 上升沿采样输入位。接收到 8 个位之后，如果 I2CRCV 未满或者 I2COV 没有置 1，I2CRSR 中的数据将被送至 I2CRCV。在第九个时钟发送 ACK。

如果 RBF 标志置 1，表明 I2CRCV 仍然存有前一次收到的数据 (RBF = 1)，则不发送 ACK；然而，仍将产生中断脉冲。如果出现溢出，I2CRSR 的内容不会被装载到 I2CRCV 中。

注： 如果 I2COV 位为 1 且 RBF 标志为 0，将装载 I2CRCV。在这种情况下，要对 I2CRCV 进行读操作，但在下一次接收操作发生之前，用户不会清除 I2COV 位的状态。不发送应答信号 (ACK = 1)，但 I2CRCV 位会被更新。

17.4 I²C 10 位从模式的工作

在 10 位模式下，基本的接收和发送操作与 7 位模式下的相同。不过，地址匹配的判据更加复杂。

I²C 规范要求写操作必须用启动位后的两个地址字节寻址从器件。

A10M 控制位置 1 表明 I2CADD 中的地址是 10 位地址，而不是 7 位地址。报文地址首字节的地址检测协议，对于 7 位和 10 位报文地址而言，是完全相同的，但首字节各位进行比较的机制不同。

I2CADD 存放完整的 10 位地址。在接收到启动位后的地址时，I2CRSR <7:3> 与立即数 11110 (缺省 10 位地址) 进行比较，I2CRSR <2:1> 与 I2CADD <9:8> 进行比较。如果出现匹配且 R_W = 0，则发出中断脉冲。ADD10 位清零，表明地址部分匹配。如果地址不匹配，或者 R_W = 1，则 ADD10 位清零，模块返回到空闲状态。

然后，接收地址的低字节与 I2CADD <7:0> 进行比较。如果出现地址匹配，则产生中断脉冲且置 1 ADD10 位，表明 10 位地址完全匹配。如果地址不匹配的话，ADD10 位清零，模块返回到空闲状态。

17.4.1 10 位地址从模式发送

一旦以上述方式使用完整的10位地址寻址到从器件（我们把这个状态叫做“PRIOR_ADDR_MATCH”），主器件就可以开始发送数据字节，供从器件接收。

17.4.2 10 位地址从模式接收

一旦寻址到从器件，主器件就可以产生重复的启动位，复位地址的高字节，并置1 R_W 位，但不产生停止位，于是开始进行从器件发送操作。

17.5 自动时钟延长

当模块工作在从模式下时，通过时钟延长，模块能够同步读缓冲器与写主器件。

17.5.1 发送时钟延长

在10位和7位地址发送模式下，如果TBF位清零（表明缓冲器为空），则在第九个时钟的下降沿后清零SCLREL位，就实现了时钟延长。

在从发送模式下，始终进行时钟延长，而与STREN位的状态无关。

在发送序列的第九个时钟之后，发生时钟同步。如果器件在第九个时钟的下降沿采样到ACK信号，且TBF位仍然清零的话，则自动清零SCLREL位，从而使SCL线变为低电平。在允许继续发送之前，用户中断服务程序必须置1SCLREL位。通过使SCL线保持为低电平，使用户在主器件启动下一个发送序列之前有时间执行ISR，并装载I2CTRN的内容。

注 1: 如果用户装入I2CTRN的内容，在第九个时钟下降沿之前置1TBF位，则SCLREL位将不会清零，也不会发生时钟延长。

2: SCLREL位可由软件置1，而无需考虑TBF位的状态。

17.5.2 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。当STREN位置1时，SCL引脚将在每个数据接收序列结束后保持低电平。

17.5.3 7 位寻址时的时钟延长（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。将SCL线保持为低电平，能使用户在主器件启动下一个接收序列之前，有时间执行ISR并读取I2CRCV的内容。这将防止发生缓冲器溢出。

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RBF位清零，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RBF位的状态。在下一个接收序列开始之前，用户应该在ISR中清零RBF位，以避免溢出。

17.5.4 10 位寻址时的时钟延长（STREN = 1）

在寻址过程中自动发生时钟延长。由于模块有寄存器存放完整的地址，协议不必等待地址更新。

寻址过程完成后，每个数据接收或发送序列都要进行时钟延长，如前所述。

17.6 软件控制的时钟延长（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。逻辑电路将写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输出的下降沿且采样到SCL为低。在SCL线采样为低电平时，如果用户清零SCLREL位，则SCL输出将保持为低，直到SCLREL位置1，且I²C总线上所有其他器件都已经释放了SCL。这确保了对SCLREL位的写操作不会违背SCL的最小高电平时间要求。

如果STREN位为0，软件对SCLREL位的写操作将被忽略，不会对SCLREL位产生影响。

17.7 中断

I²C 模块产生两个中断标志：MI2CIF（I²C 主中断标志）和 SI2CIF（I²C 从中断标志）。MI2CIF 中断标志，会在主器件报文事件完成时置 1（从而激活中断）。SI2CIF 中断标志，在检测到发往从器件的报文时置 1（从而激活中断）。

17.8 斜率控制

对于快速模式（400 kHz），I²C 标准要求对 SDA 和 SCL 信号进行斜率控制。如果需要的话，用户可通过控制位 DISSLW 来禁止斜率控制。对于 1 MHz 模式，必须禁止斜率控制。

17.9 IPMI 支持

控制位 IPMIEN 允许模块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。

17.10 广播呼叫地址支持

广播呼叫地址能寻址所有器件。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八个地址之一。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，识别广播呼叫地址。检测到启动位后，将 8 位数据移入 I2CRSR 寄存器，并将地址与 I2CADD 进行比较，同时与固化在硬件中的广播呼叫地址进行比较。

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地址还是广播呼叫地址。

17.11 I²C 主模式支持

用作主器件时，支持下列 6 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 / 地址的发送。
- 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。

17.12 I²C 主模式下的工作

主器件产生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次串行传输的开始，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，会接收到一个 $\overline{\text{ACK}}$ 位。输出启动和停止条件来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，后跟一个表明接收的位 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。在每个字节接收完之后，要发送一个 $\overline{\text{ACK}}$ 位。启动条件和停止条件表明发送的开始和结束。

17.12.1 I²C 主发送

数据字节、7 位地址或 10 地址后半部分的发送，通过简单地写入 I2CTRN 寄存器即可完成。当模块处于 WAIT 状态时，用户只能写 I2CTRN 寄存器。此操作会将缓冲器满标志（TBF）置 1，允许波特率发生器开始计数，并开始下一次发送。SCL 出现下降沿后，地址 / 数据的每一个位将被移出至 SDA 引脚。发送状态标志位 TRSTAT（I2CSTAT<14>）显示主器件是否正在发送。

17.12.2 I²C 主接收

通过设置接收使能位 RCEN（I2CCON<3>）可以启用主模式接收。在 RCEN 置 1 之前，I²C 模块必须是空闲的，否则 RCEN 位被忽略。波特率发生器开始计数，每次计满返回到零时，在每个时钟的上升沿，SCL 引脚状态发生翻转，数据移入至 I2CRSR 中。

17.12.3 波特率发生器

在 I²C 主模式下，波特率发生器的重载值位于 I2CBRG 寄存器中。波特率发生器装入该值后，发生器递减计数到 0，然后停止，直到发生再次装入。如果发生时钟仲裁，例如 SCL 引脚采样为高时，将重载波特率发生器。

dsPIC30F6010A/6015

根据 I²C 标准，F_{SCL} 可以是 100 kHz 或 400 kHz。然而，用户可以指定任何波特率，最高到 1 MHz。I²C_{BRG} 的值不能是 0 或 1。

公式 17-1: 串行时钟速率

$$I2C_{BRG} = \left(\frac{F_{CY}}{F_{SCL}} - \frac{F_{CY}}{1,111,111} \right) - 1$$

17.12.4 时钟仲裁

在任何接收、发送或重复启动 / 停止条件期间，当主器件释放了 SCL 引脚（允许 SCL 悬空为高电平），就会出现时钟仲裁。当允许 SCL 引脚悬空为高电平时，波特率发生器（BRG）将暂停计数，直到 SCL 引脚被实际采样到高电平为止。当 SCL 引脚被采样到高电平时，波特率发生器重新装载 I²C_{BRG} 的内容并开始计数。在外部器件使时钟保持为低时，这可以始终保证 SCL 高电平时间至少为一个 BRG 计满返回计数周期。

17.12.5 多主器件通信、总线冲突与总线仲裁

多主器件工作的支持，是通过总线仲裁来实现的。当主器件在 SDA 引脚上输出地址 / 数据位时，第一个主器件使 SDA 悬空为高电平从而输出一个 1，而另一个主器件要输出 0，就会发生仲裁。当 SCL 引脚悬空为高时，数据应该是稳定的。如果 SDA 上预期数据是 1，但从 SDA 引脚采样到的数据是 0，那么就发生了总线冲突。主器件将发出 MI²CIF 脉冲，并把 I²C 端口的主器件部分复位到空闲状态。

如果在进行发送时产生总线冲突，将停止发送，清除 TBF 标志，释放 SDA 和 SCL 线，而后可向 I²C_{TRN} 写入数据。当用户执行 I²C 主事件中中断服务程序时，如果 I²C 总线空闲（即 P 位置 1），用户可以通过产生一个启动条件继续通信。

当总线冲突发生在启动、重复启动、停止或应答条件期间，将中止这些条件，释放 SDA 和 SCL 线，清零 I²C_{CON} 寄存器中对应的控制位。当用户执行总线冲突中断服务程序时，如果 I²C 总线空闲，用户可以通过产生一个启动条件继续通信。

主器件将继续监视 SDA 和 SCL 引脚，当产生停止条件时，将置 1 MI²CIF 位。

写 I²C_{TRN} 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。

在多主器件环境中，在检测到启动和停止条件时产生中断，能够确定总线何时空闲。当 I²C_{STAT} 寄存器中的 P 位置 1 时，器件可以控制 I²C 总线；否则，总线空闲，且 S 和 P 位清零。

17.13 CPU 休眠和空闲模式期间 I²C 模块的工作

17.13.1 CPU 休眠模式期间 I²C 的工作

当器件进入休眠模式时，模块的全部时钟源都将关闭并保持在逻辑 0。如果休眠发生在发送过程中，且时钟停止时状态机部分进入发送，发送将中止。类似地，如果休眠出现在接收过程中，接收也将中止。

17.13.2 CPU 空闲模式期间 I²C 的工作

对于 I²C，I²C_{SIDL} 位选择 CPU 空闲时模块是停止还是继续工作。如果 I²C_{SIDL} = 0，在空闲模式时，模块将继续工作；如果 I²C_{SIDL} = 1，在空闲模式时，模块将停止工作。

表 17-2: I²C™ 寄存器映射

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: u = 未初始化的位

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

dsPIC30F6010A/6015

注:

18.0 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，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

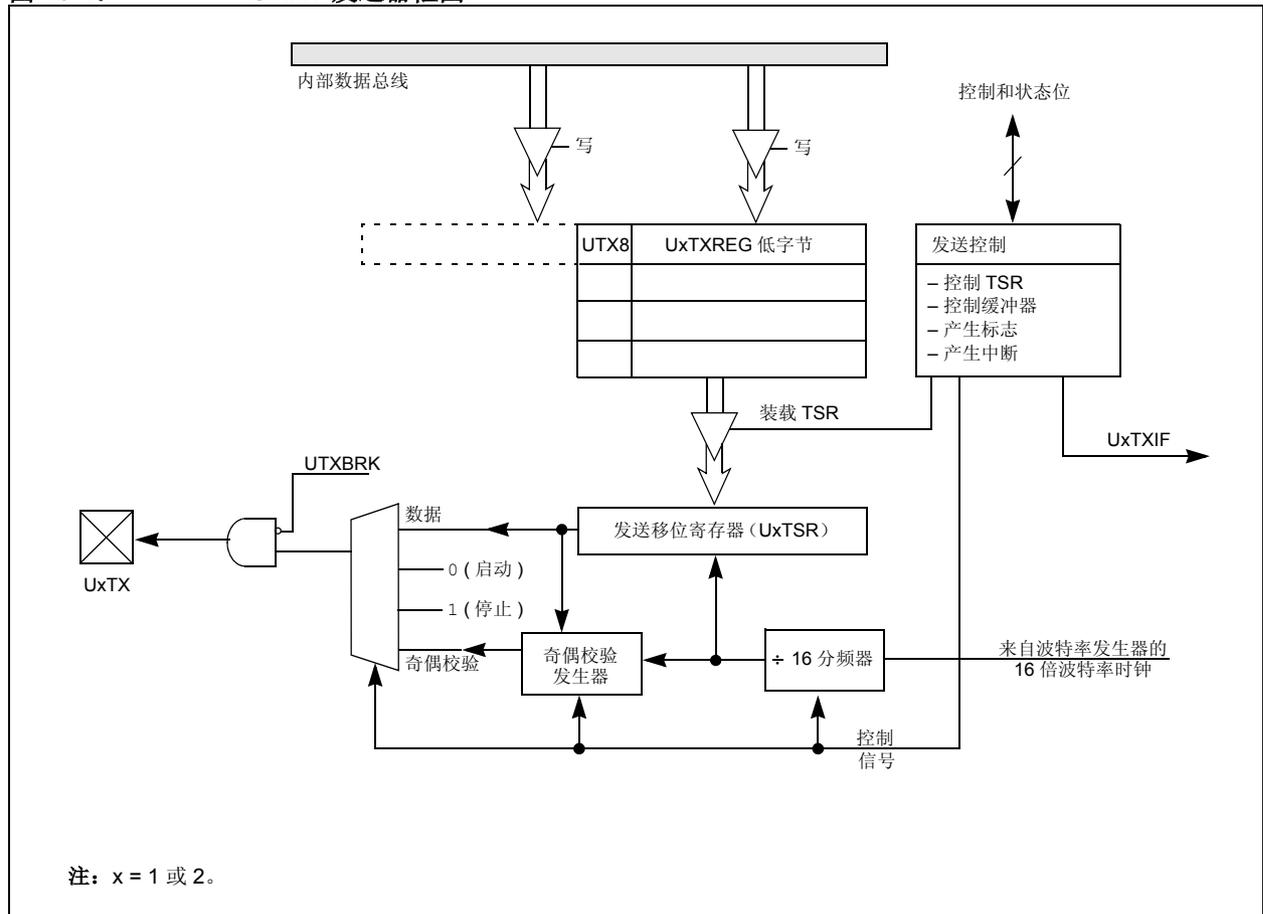
本节介绍了通用异步接收器 / 发送器通信模块。

18.1 UART 模块概述

UART 模块的主要特性是：

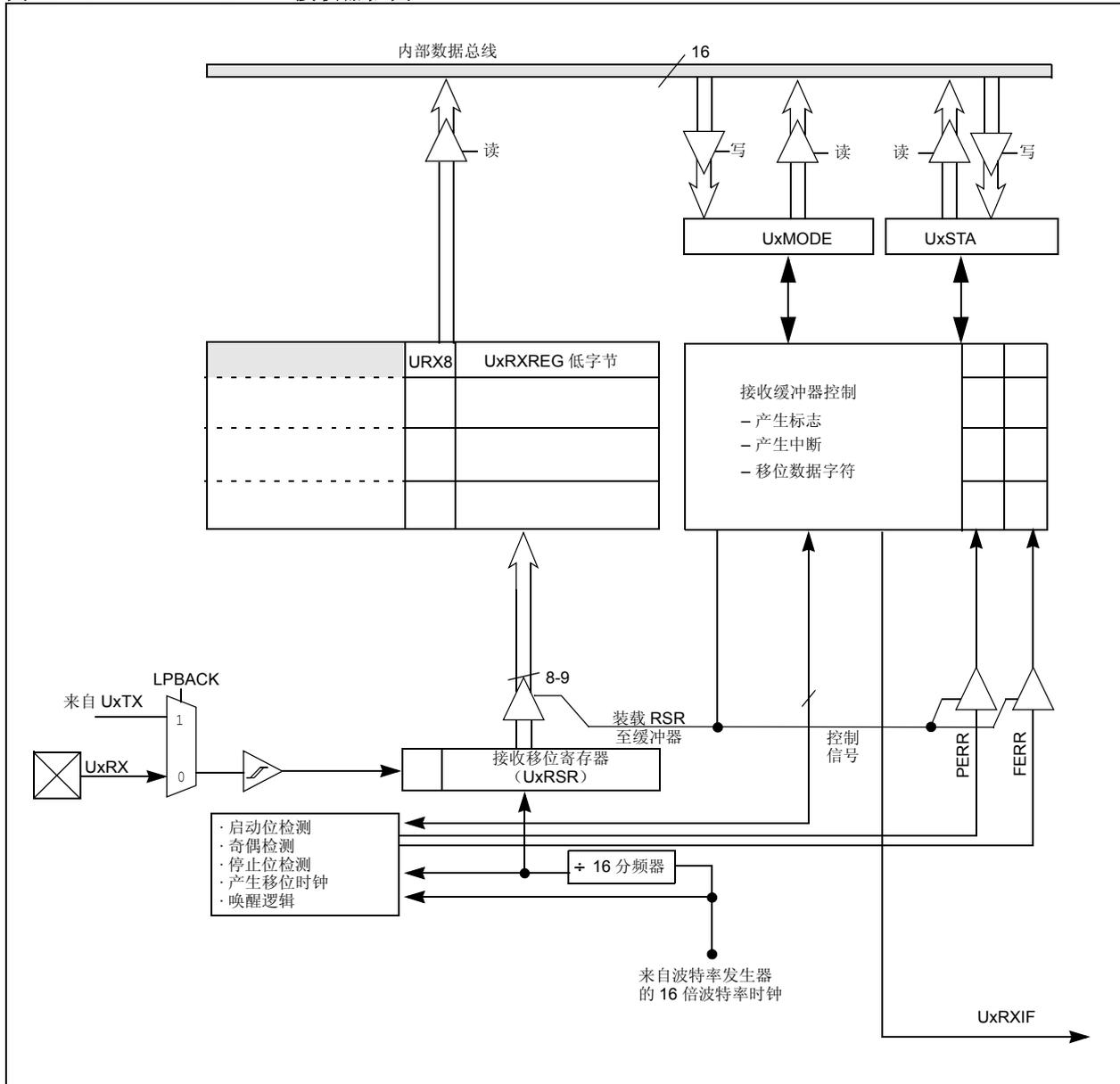
- 全双工 8 位或 9 位数据通信
- 偶校验、奇校验或无奇偶校验选项（对于 8 位数据）
- 一或两个停止位
- 完全集成的波特率发生器，具有 16 位预分频器
- 在 30 MHz 指令速度时，波特率范围为 38 bps 到 1.875 Mbps
- 4 字深度发送数据缓冲器
- 4 字深度接收数据缓冲器
- 奇偶、帧和缓冲溢出错误检测
- 支持仅在地址检测（第 9 位 = 1）时中断
- 独立的发送和接收中断
- 用于诊断支持的环回模式

图 18-1: UART 发送器框图



dsPIC30F6010A/6015

图 18-2: UART 接收器框图



18.2 使能和设置 UART

18.2.1 使能 UART

UART 模块的使能，是通过置 1 UxMODE 寄存器（其中，x = 1 或 2）中的 UARTEN 位来进行的。一旦使能，UxTX 和 UxRX 引脚被分别配置为输出和输入，这改写了相应 I/O 端口引脚的 TRIS 和 LAT 寄存器位的设置。在没有传输发生时，UxTX 引脚状态为逻辑 1。

18.2.2 禁止 UART

通过清零 UxMODE 寄存器中的 UARTEN 位，可禁止 UART 模块。这是任何复位后的默认状态。如果 UART 被禁止，那么所有的 I/O 引脚均用作端口引脚，受相应端口引脚的 LAT 和 TRIS 位的控制。

禁止 UART 模块将使缓冲器复位为空状态。缓冲器中的所有字符都将丢失，同时波特率计数器也将复位。

当 UART 模块禁止时，所有与之相关的错误和状态标志都将复位。URXDA、OERR、FERR、PERR、UTXEN、UTXBRK 和 UTXBF 位被清零，而 RIDLE 和 TRMT 被置 1。其他的控制位，包括 ADDEN、URXISEL<1:0> 和 UTXISEL，以及 UxMODE 和 UxBRG 寄存器都不受影响。

当 UART 处于工作状态时，对 UARTEN 位清零将中止所有等待的发送和接收，同时还将像上面定义的那样将模块复位。再次使能 UART 将使用同样的配置来重新启动 UART。

18.2.3 设定数据、奇偶校验以及停止位选择

UxMODE 寄存器中的控制位 PDSEL<1:0> 用来选择发送时所使用的数据长度和奇偶校验。数据可以是具有偶校验位、奇校验位或无奇偶校验位的 8 位数据，也可以是无奇偶校验位的 9 位数据。

STSEL 位决定在数据发送期间，是使用一个还是两个停止位。

UART 的默认（上电）设置是 8 位、无奇偶校验位以及 1 个停止位（通常表示为 8, N, 1）。

18.3 发送数据

18.3.1 8 位数据模式下的发送

为了发送 8 位数据，必须执行下列步骤：

1. 设置 UART：
首先，必须选择数据长度、奇偶校验和停止位的个数。然后，在 UxMODE 和 UxSTA 寄存器中设置发送和接收中断允许位以及优先级位。另外，必须把适当的波特率值写入 UxBRG 寄存器。
 2. 将 UARTEN 位（UxMODE<15>）置 1，使能 UART。
 3. 将 UTXEN 位（UxSTA<10>）置 1，使能发送。
- 注：** 在 UARTEN 位置 1 后必须将 UTXEN 位置 1，以使能 UART 发送。
4. 向 UxTXREG 的低字节写入要发送的字节。它将被立即送往发送移位寄存器（UxTSR），在波特率时钟的下一个上升沿期间，串行位流将开始移出移位寄存器。另一种方法是，在 UTXEN = 0 时可写入数据字节，而后，用户可以把 UTXEN 置 1。这将使串行位流立即开始，因为波特率时钟将从清零状态启动。
 5. 将产生发送中断，这取决于中断控制位 UTXISEL（UxSTA<15>）的值。

18.3.2 9 位数据模式下的发送

9 位数据发送步骤类似于 8 位数据发送，不同之处在于必须将一个 16 位数据字（其中高 7 位始终为 0）写入 UxTXREG 寄存器。

18.3.3 发送缓冲器（UxTXB）

发送缓冲器为 9 位宽且 4 个字符深。加上发送移位寄存器（UxTSR），用户实际上有一个 5 级深度的 FIFO（先进先出）缓冲器。UTXBF 状态位（UxSTA<9>）表明发送缓冲器是否已满。

如果用户试图写已满的缓冲器，新数据将不会被 FIFO 接收，缓冲器内也不会有数据移位，这确保了能够从缓冲器溢出条件中恢复。

FIFO 在任何器件复位时复位，但当器件进入节能模式或从节能模式唤醒时，FIFO 不受影响。

18.3.4 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置 1 UxTXIF 位。产生中断的条件取决于 UTXISEL 控制位:

- 如果 UTXISEL = 0, 当从发送缓冲器传输一个字到发送移位寄存器 (UxTSR) 时, 将产生中断。这意味着, 发送缓冲器中至少有一个空字。
- 如果 UTXISEL = 1, 当从发送缓冲器传输一个字到发送移位寄存器 (UxTSR) 且缓冲器为空时, 将产生中断。

工作期间可以在这两种中断方式之间切换, 有时这提供了更大的灵活性。

18.3.5 发送间隔字符

把 UTXBRK 位 (UxSTA<11>) 置 1, 将驱动 UxTX 线为逻辑 0。UTXBRK 位可改写所有发送器活动。因此, 通常用户应该在将 UTXBRK 置 1 前等待发送器空闲。

为了发送间隔字符, 必须由软件将 UTXBRK 位置 1, 而且保持该位置 1 至少 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。在再次加载 UTXBUF 或开始其他发送活动前, 用户必须等待至少一或两个波特率时钟周期, 以便保证产生有效的停止位。中止符的发送不会产生发送中断。

18.4 接收数据

18.4.1 在 8 位或 9 位数据模式下接收

在接收 8 位或 9 位数据时, 必须执行下列步骤:

- 设置和使能 UART (见第 18.3 节 "发送数据")。
- 当接收到一个或多个数据字时, 将产生接收中断, 这取决于由 URXISEL 位 (UxSTA<7:6>) 指定的接收中断设置。
- 读 OERR 位, 判断是否产生溢出错误。OERR 位必须通过软件来复位。
- 从 UxRXREG 中读取接收数据。对 UxRXREG 的读操作, 将使下一个字移动到接收 FIFO 的栈顶, 且将更新 PERR 和 FERR 值。

18.4.2 接收缓冲器 (UxRXB)

接收缓冲器是 4 字深缓冲器。加上接收移位寄存器 (UxRSR), 用户实际上有一个 5 字深的 FIFO 缓冲器。

URXDA (UxSTA<0>) = 1 表明接收缓冲器中还有数据。URXDA = 0 意味着缓冲器为空。如果用户试图读空缓冲器, 读取的是缓冲器中的原有值, 在 FIFO 中不会产生数据移位。

任何器件复位时, FIFO 都将复位。当器件进入节能模式或从节能模式中唤醒时, 它不受影响。

18.4.3 接收中断

可以从相应的中断标志寄存器中读取接收中断标志 (U1RXIF 或 U2RXIF)。中断标志由接收器产生的边沿来进行置 1。置 1 接收中断标志的条件, 取决于 URXISEL<1:0> (UxSTA<7:6>) 控制位的设定。

- 如果 URXISEL<1:0> = 00 或 01, 每当一个数据字从接收移位寄存器 (UxRSR) 送往接收缓冲器后就会产生中断。接收缓冲器中可以有一个或多个字符。
- 如果 URXISEL<1:0> = 10, 当一个字从接收移位寄存器 (UxRSR) 送往接收缓冲器, 使得缓冲器中有 3 个字符时, 就会产生中断。
- 如果 URXISEL<1:0> = 11, 当一个字从接收移位寄存器 (UxRSR) 送往接收缓冲器, 使得缓冲器中有 4 个字符 (即缓冲器满) 时, 就会产生中断。

工作期间可以在这两种中断方式之间切换, 尽管正常工作期间这样做并不可取。

18.5 接收错误处理

18.5.1 接收缓冲器溢出错误 (OERR 位)

如果满足下列全部条件, OERR 位 (UxSTA<1>) 将置 1:

- 接收缓冲器满。
- 接收移位寄存器满, 但不能把字符送往接收缓冲器。
- 检测到 UxRSR 中字符的停止位, 这表明 UxRSR 需要把字符送往缓冲器。

一旦 OERR 置 1, 就不会有数据移入 UxRSR (除非使用软件将 OERR 位清零, 或者发生了复位)。存放在 UxRSR 和 UxRXREG 中的数据继续有效。

18.5.2 帧错误 (FERR)

如果检测到 0 而不是停止位, 则 FERR 位 (UxSTA<2>) 置 1。如果选用两个停止位, 则两个停止位都必须是 1, 否则将置 1 FERR。FERR 位是只读的, 它将和收到的数据一起被送入缓冲器。任何复位时都将清零 FERR 位。

18.5.3 奇偶校验错误 (PERR)

如果接收到的字奇偶校验错误的话, PERR 位 (UxSTA<3>) 将置 1。此错误位仅在选取了奇偶校验模式 (奇或偶) 时适用。PERR 位是只读的, 它将和收到的数据一起被送入缓冲器。任何复位时都将清零 PERR 位。

18.5.4 空闲状态

当接收器处于活动状态 (即, 在初次检测到启动位与停止位完成之间时), RIDLE 位 (UxSTA<4>) 为 0。在停止位完成与检测到下一个启动位之间时, RIDLE 位为 1, 表明 UART 空闲。

18.5.5 接收间隔字符

接收器会根据在 PDSEL (UxMODE<2:1>) 和 STSEL (UxMODE<0>) 中设置的值, 计数并等待一定的位时间数。

如果间隔超过 13 个位时间, 则在 PDSEL 和 STSEL 指定的位时间数之后, 认为接收完成。URXDA 位置 1, FERR 置 1, 接收 FIFO 中装载零, 同时产生中断 (若允许中断) 且 RIDLE 位置 1。

如果模块收到长间隔信号, 同时接收器检测到了启动位、数据位和无效的停止位 (这将把 FERR 置 1), 那么接收器在找寻下一个启动位之前, 必须等待有效停止位出现。接收器不能假定线上的间隔条件就是下一个启动位。

间隔符被视为一个全 0 的字符, 且 FERR 位置 1。间隔符将被装载到缓冲器中。除非接收到停止位, 否则不会再进行接收。注意, 当停止位还没有接收到时, RIDLE 将置 1。

18.6 地址检测模式

置 1 ADDEN 位 (UxSTA<5>) 来使能这个特殊的模式, 在此模式下, 如果第 9 位 (URX8) 值为 1, 则表明接收到的字是一个地址, 而不是数据。此模式仅适用于 9 位数据通信。在此模式下, URXISEL 控制位对中断产生没有任何影响, 因为中断 (如果允许的话) 仅在接收到的字的第 9 位为 1 时才产生。

18.7 环回模式

置 1 LPBACK 位来使能这个特殊的模式, 在该模式下 UxTX 引脚从内部连接到 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。但是, UxTX 引脚仍然正常工作。

要选择该模式:

- 将 UART 配置为所需的工作模式。
- 设置 LPBACK = 1, 使能环回模式。
- 根据第 18.3 节 “发送数据” 中所述使能发送。

18.8 波特率发生器 (BRG)

UART 有一个 16 位波特率发生器, 以便在波特率发生时获得最大的灵活性。波特率发生器寄存器 (UxBRG) 是可读写的。波特率的计算如下:

BRG = UxBRG 寄存器中的 16 位值
(0 到 65535)

F_{CY} = 指令时钟速率 (1/T_{CY})

波特率由公式 18-1 给出。

公式 18-1: 波特率

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

因此, 可能的最大波特率是

F_{CY}/16 (如果 BRG = 0),

可能的最小波特率是

F_{CY}/(16 * 65536)。

对于以 30 MIPS 工作的 16 位波特率发生器, 能够达到的最小波特率是 28.5 bps。

18.9 自动波特率支持

为了让系统确定接收字符的波特率，可选择将输入连接到所选择的捕捉输入通道。为使能该模式，用户必须对输入捕捉模块进行设置，以检测启动位的上升沿和下降沿。

18.10 UART 在 CPU 休眠和空闲模式下的工作

18.10.1 UART 在 CPU 休眠模式下的工作

当器件进入休眠模式时，模块的所有时钟源都将关闭，并保持在逻辑 0 状态。如果器件在发送进行期间进入休眠模式，发送将中止，并且 UxTX 引脚被驱动为逻辑 1。类似地，如果接收进行期间进入休眠模式，接收将中止。休眠模式不影响 UxSTA、UxMODE、发送和接收的寄存器和缓冲器以及 UxBRG 寄存器。

器件进入休眠模式前，如果 WAKE 位 (UxMODE<7>) 置 1，那么 UxRX 引脚上的下降沿将产生接收中断。接收中断选择模式位 (URXISEL) 对该功能没有影响。如果接收中断允许，那么这将把器件从休眠中唤醒。为了产生唤醒中断，UARTEN 位必须置 1。

18.10.2 UART 在 CPU 空闲模式下的工作

对于 UART，USIDL 位选择当器件进入空闲模式时，模块是停止还是继续工作。如果 USIDL = 0，模块将在空闲模式期间继续工作。如果 USIDL = 1，模块将在空闲模式下停止工作。

表 18-1: UART1 寄存器映射

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	—	—	—	—	—	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 = 未初始化位

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

表 18-2: UART2 寄存器映射

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 = 未初始化位

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

dsPIC30F6010A/6015

注:

19.0 CAN 模块

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

19.1 概述

控制器局域网 (Controller Area Network, CAN) 模块是一个串行接口，用于同其他 CAN 模块或单片机器件进行通信。此接口协议是针对允许在噪声环境下通信而设计的。dsPIC30F6010A 具有两个 CAN 模块。dsPIC30F6015 仅具有一个 CAN 模块。

CAN 模块是一个通信控制器，实现了 BOSCH 规范中定义的 CAN 2.0A 或 CAN 2.0B 协议。该模块将支持 CAN 1.2、CAN 2.0A、CAN 2.0B Passive 和 CAN 2.0B Active 版本的协议。该模块实现了一种完整的 CAN 系统。但本数据手册不讨论 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
- 支持标准和扩展数据帧
- 0-8 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 2 个区分优先级的接收报文存储缓冲器 (每个缓冲器最多可容纳 8 字节的数据)
- 6 个完全 (标准 / 扩展标识符) 验收过滤器，其中 2 个与高优先级的接收缓冲器相关联，4 个与低优先级的接收缓冲器相关联
- 2 个完全验收屏蔽寄存器，其中一个与高优先级的接收缓冲器相关联，另一个与低优先级的接收缓冲器相关联。
- 3 个发送缓冲器，可由应用指定优先级和中止功能 (每个缓冲器最多可容纳 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
- 通过中断功能在出现所有的 CAN 接收器和发送器错误条件时发出信号
- 可编程时钟源
- 可编程连接到定时器模块，用于时间标记和网络同步
- 低功耗休眠和空闲模式

CAN 总线模块由协议引擎以及报文缓冲 / 控制模块组成。CAN 协议引擎处理在 CAN 总线上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能开始发送报文。通过读取相应的寄存器可以检测状态和错误。将对 CAN 总线上接收到的任何报文进行错误检测，然后与过滤器进行匹配以决定是否接收该报文并将其存储在接收寄存器中。

19.2 帧类型

CAN 模块可发送多种类型的帧，其中包括数据报文、用户发出的远程发送请求以及自动生成的用于控制的其他帧。支持下列帧类型：

- 标准数据帧

当节点要发送数据时会生成一个标准数据帧。它包含一个 11 位的标准标识符 (SID)，而不是 18 位的扩展标识符 (EID)。

- 扩展数据帧

扩展数据帧与标准数据帧类似，但包含的是扩展标识符。

- 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，目标节点必须发送一个包含与所需数据帧的标识符相匹配的标识符的远程帧。随后相应的数据源节点会发送一个数据帧作为对该远程请求的响应。

- 错误帧

错误帧是由检测到总线错误的任一节点生成的。错误帧包含 2 个字段：错误标志字段和错误定界符字段。

- 过载帧

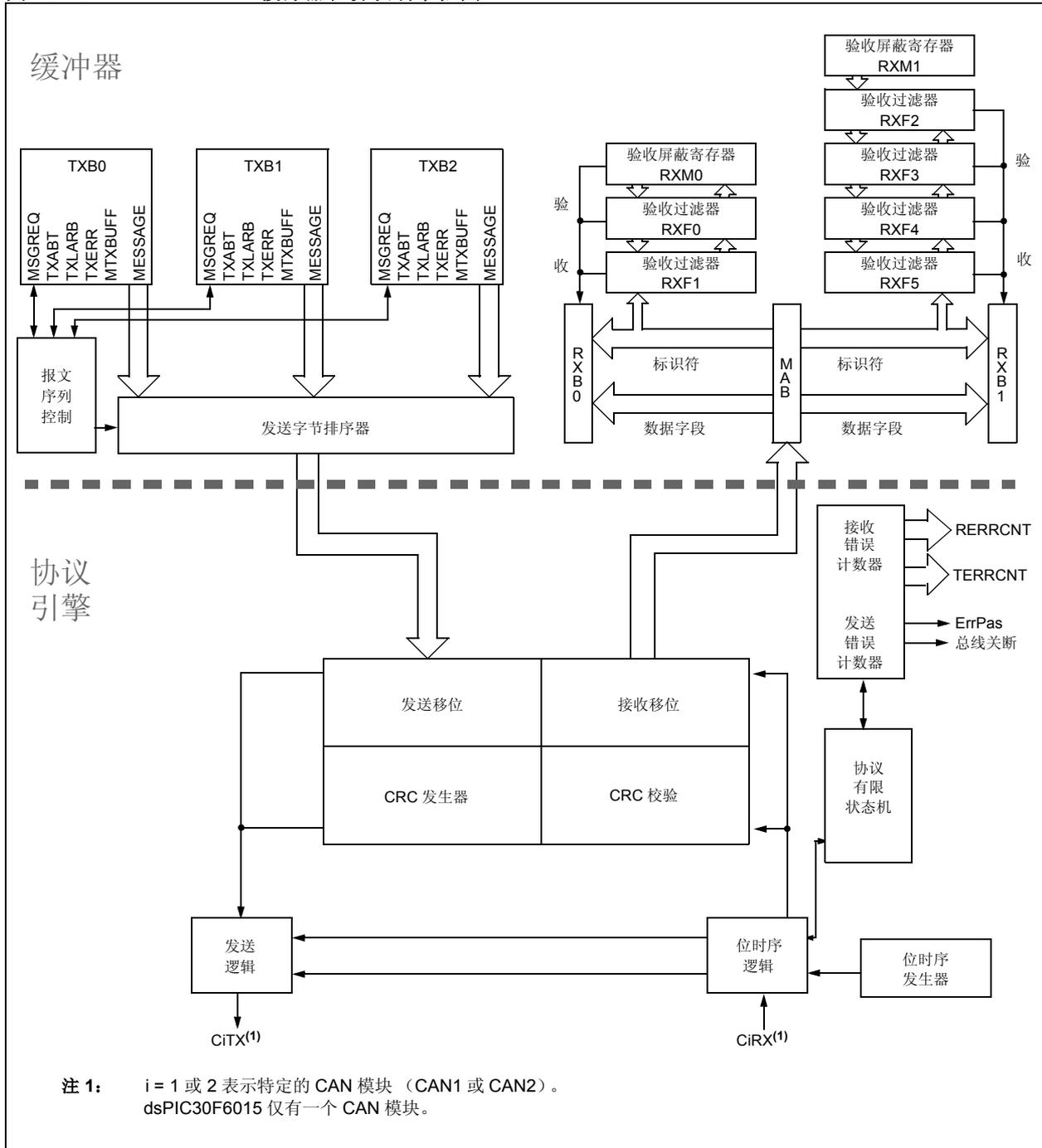
节点在两种条件下会生成过载帧。一种情况是节点在帧间间隔期间检测到一个显性位，这是一种非法的情况。另一种情况是由于内部原因，节点尚无法开始接收下一条报文。节点最多可生成两个连续过载帧来延迟下一条报文的接收。

- 帧间间隔

帧间间隔将当前帧 (无论何种类型) 与随后的数据或远程帧分隔开。

dsPIC30F6010A/6015

图 19-1: CAN 缓冲器和协议引擎框图



19.3 工作模式

用户可以选择使 CAN 模块工作在下列几种模式之一。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>) 可选择所需模式。通过监视 OPMODE<2:0> 位 (CiCTRL<7:5>) 确定所进入的模式。在允许模式改变之前，通常在由至少 11 位连续的隐性位定义的总线空闲时间内，模块不会改变工作模式和 OPMODE 位。

19.3.1 初始化模式

在初始化模式下，模块不会进行发送或接收。错误计数器将被清零且中断标志位保持不变。编程人员可以访问在其他模式下不可访问的配置寄存器。模块会防止用户因为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。当进行发送的时候，不允许 CAN 模块进入配置模式。配置模式会作为锁来保护以下寄存器：

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

19.3.2 禁止模式

在禁止模式下，模块不会进行发送或接收操作。由于总线活动，模块可将 WAKIF 位置 1，但待处理的中断将继续等待，且错误计数器的值也将保持不变。

如果 REQOP<2:0> 位 (CiCTRL<10:8>) = 001，模块将进入禁止模式。如果模块处于工作状态，它将等待 CAN 总线上出现 11 个隐性位以表明总线空闲后执行模块禁止命令。当 OPMODE<2:0> 位 (CiCTRL<7:5>) = 001 时，表明模块成功进入了禁止模式。模块处于禁止模式时，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，可对模块进行编程以通过 CiRX 输入线提供低通滤波器功能。WAKFIL 位 (CiCFG2<14>) 用于使能或禁止该滤波器。

注：通常，如果 CAN 模块在某种工作模式下允许发送，并且在 CAN 模块进入该工作模式后立即请求发送，则模块发送之前，将等待 11 个连续的隐性位。如果用户在此 11 个隐性位期间切换到禁止模式，则中止发送。且将相应的 TXABT 位置 1，TXREQ 位清零。

19.3.3 正常工作模式

当 REQOP<2:0> = 000 时选择正常工作模式。在此模式下激活模块，I/O 引脚将承担 CAN 总线功能。模块将通过 CiTX 和 CiRX 引脚发送和接收 CAN 总线报文。

19.3.4 监听模式

如果激活仅监听模式，CAN 总线上的模块处于被动状态。发送缓冲器恢复为端口 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，错误计数器不再工作。仅监听模式可用于检测 CAN 总线上的波特率。要使用该模式，必须保证至少有两个可以相互通信的节点。

19.3.5 错误识别模式

该模块可被设置为忽略所有错误并接受任何报文。通过将 RXM<1:0> 位 (CiRXnCON<6:5>) 设置为 11 激活错误识别模式，在此模式下，发生错误之前一直位于报文组装缓冲器中的数据被复制到接收缓冲器并可通过 CPU 接口读数据。

19.3.6 环回模式

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

19.4 报文接收

19.4.1 接收缓冲器

CAN 总线模块具有 3 个接收缓冲器。但其中一个接收缓冲器总是用于监视总线上是否有进入的报文。此缓冲器被称为报文组装缓冲器 (MAB)。因此, 只有 2 个接收缓冲器 RXB0 和 RXB1 可见, 它们本质上可以瞬间接收来自协议引擎的完整报文。

MAB 将组装所有的报文, 并将满足接收过滤条件的报文传输到 RXBn 缓冲器中。当接收到报文时, 将 RXnIF 标志 (CiINTF<0> 或 CiINTF<1>) 位置 1。此模块仅在接收到报文时才将该位置 1。用户处理完缓冲器中的报文后由 CPU 将该位清零。如果 RXnIE 位 (CiINTE<0> 或 CiINTE<1>) 被置 1, 则接收报文时产生中断。

RXF0、RFX1 过滤器以及屏蔽寄存器 RXM0 与 RXB0 相关联。过滤器 RFX2、RFX3、RFX4 和 RFX5 以及屏蔽寄存器 RXM1 与 RXB1 相关联。

19.4.2 报文验收过滤器

报文验收过滤器和屏蔽寄存器用于决定报文组合缓冲器中的报文是否应该被装入某个接收缓冲器中。报文组装过滤器接收一条有效报文后, 就会将报文的标识字段与过滤器的值进行匹配。如果匹配, 就将报文装入相应的接收缓冲器。

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, 报文是标准帧, 则只与 EXIDE 位 (CiRXFnSID<0>) 清零的过滤器比较。如果 EXIDE 位置 1, 报文是扩展帧, 则只与 EXIDE 位置 1 的过滤器比较。将 RXM<1:0> 位配置为 01 或 10 可改写 EXIDE 位。

19.4.3 报文验收过滤器屏蔽寄存器

屏蔽位主要决定将对哪些位应用过滤器。如果任何屏蔽位被设置为零, 无论过滤位为何值该位都会被自动接收。2 个可编程接收过滤器屏蔽寄存器分别与 2 个接收缓冲器相关联。

19.4.4 接收溢出

当报文组装缓冲器组装了一个有效的接收报文后, 由接收过滤器接收该报文, 此时如果与该过滤器相关的接收缓冲器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 (CiINTF<5>) 将被置 1, 并丢弃 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 RXOVR 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 接收的为有效报文且 RFXFUL = 1, 则表明 RXB0 为满, 如果 RFXFUL = 0, 则表明 RXB1 为空, 将 RXB0 的报文装入 RXB1 中。RXB0 不会产生溢出错误。如果 RXB0 接收为有效报文且 RFXFUL = 1, 则表明 RXB0 和 RXB1 都为满, 报文丢失且 RXB1 发生溢出。

19.4.5 接收错误

CAN 模块将会检测到以下接收错误:

- 循环冗余校验 (CRC) 错误
- 位填充错误
- 无效报文接收错误

这些接收错误不会产生中断。但发生上述错误之一时, 接收错误计数器会递增 1。RXWAR 位 (CiINTF<9>) 表明接收错误计数器已经达到 CPU 警告的上限值 96, 并由此产生中断。

19.4.6 接收中断

接收中断可被分成 3 个主要组, 每组包含了产生中断的不同条件:

- 接收中断

报文已被成功接收并被装入其中一个接收缓冲器。接收到帧结束 (End-of-Frame, EOF) 字段后立即激活中断。读 RXnIF 标志位将指示引起中断的接收缓冲器。

- 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

接收错误中断由 ERRIF 位指示。该位显示有错误条件产生。通过校验 CAN 中断状态寄存器中的 CiINTF 位可以确定错误源。

- 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 位都将指示有错误发生。

- 接收器溢出

RXnOVR 位指示有溢出条件产生。

- 接收器警告

RXWAR 位表明接收错误计数器 (RERRCNT<7:0>) 已经达到 CPU 警告的上限值 96。

- 接收器错误被动

RXEP 位表明接收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。

19.5 报文发送

19.5.1 发送缓冲器

CAN 模块具有 3 个发送缓冲器。每个缓冲器可容纳 14 字节的数据。其中的 8 个字节用于存储发送报文（最大 8 个字节）。另外 5 个字节用来存放标准或扩展报文标识符和其他报文仲裁信息。

19.5.2 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。如果某个报文缓冲器的 TXPRI<1:0> (CiTXnCON<1:0>, 其中 n = 0、1 或 2, 表示特定发送缓冲器) 被设置为 11, 则该缓冲器具有最高优先级。如果指定报文缓冲器的 TXnPRI<1:0> 位被设置为 10 或 01, 则该缓冲器具有中等优先级。如果指定报文缓冲器的 TXnPRI<1:0> 位被设置为 00, 则该缓冲器具有最低优先级。

19.5.3 发送过程

要启动报文发送, 必须将 TXREQ 位 (CiTXnCON<3>) 置 1。CAN 总线模块解决了将 TXREQ 位置 1 与起始帧 (Start-of-Frame, SOF) 之间的时序冲突, 确保当优先级改变时, 能在 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 即可将某个报文缓冲器标志为正在排队等待发送。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 TXIE 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, 则将 TXLARB 位置 1, 此时不会产生任何告知仲裁失败的中断。

19.5.4 中止报文发送

系统可通过清零与每个报文缓冲器相关的 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仲裁失败或错误而被中断, 则系统将对中止进行处理。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。

19.5.5 发送错误

CAN 模块将会检测到以下发送错误:

- 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。

19.5.6 发送中断

发送中断可被分成 2 个主要组，每组包括了产生中断的不同情况：

- 发送中断

三个发送缓冲器中至少有一个为空（非预定）并且可以装入预定发送的报文。读 TXnIF 标志位可以确定可用的发送缓冲器和引起中断的缓冲器。

- 发送错误中断

发送错误中断由 ERRIF 标志位表示。该位显示有错误条件产生。通过检测 CAN 中断状态寄存器中的 CiINTF 错误标志位，就可以确定错误源。该寄存器中的标志位与接收和发送错误有关。

- 发送器警告中断

TXWAR 位表明发送错误计数器已经达到 CPU 警告的上限值 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>）表示发送错误计数器的值已经超过了 255，且该模块已经进入总线关断状态。

19.6 波特率设置

任意指定 CAN 总线上的所有节点都必须具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

19.6.1 位时序

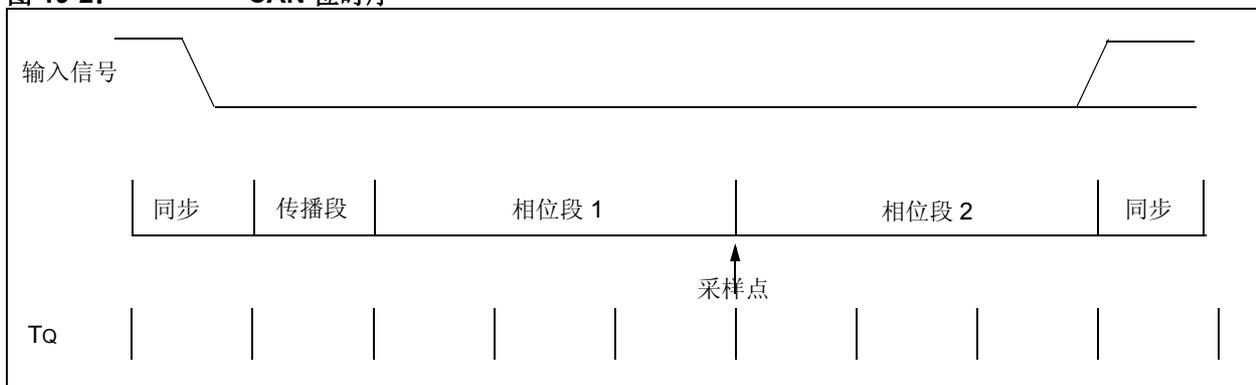
CAN 总线上的所有控制器必须使用相同的波特率和位长度。但并不要求所有的控制器使用相同的主振荡器时钟。由于各个控制器的时钟频率不同，必须通过调节各段的时间份额数来调节波特率。

标称位时间可被看成是按图 19-2 所分成的几个互不重叠的时间段。

- 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 T_Q。根据定义，标称位时间最少由 8 个 T_Q 组成，最多由 25 个 T_Q 组成。同理根据定义，最小标称位时间为 1 μs，对应最大比特率为 1 MHz。

图 19-2: CAN 位时序



19.6.2 预分频比设置

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位，如公式 19-1 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。如果 CANCKS = 0，那么 FCY 不能超过 7.5 MHz。

公式 19-1: 时钟发生器的时间份额

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

19.6.3 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 到 8 TQ。

19.6.4 相位段

相位段用于将接收到位的采样点放置在发送位时间内的最佳位置。采样点位于相位段 1 和相位段 2 之间。可通过重新同步延长或缩短这两个段。相位段 1 的末尾决定一个位周期内的采样点。可编程将相位段 1 的持续时间设置为 1 TQ 到 8 TQ。相位段 2 提供在发送下一数据前的延时，也可通过编程将其持续时间设置为 1 TQ 到 8 TQ，或定义为相位段 1 与信息处理时间（2 TQ）两者中的较大者。将 SEG1PH<2:0> 位（CiCFG2<5:3>）置 1 以初始化相位段 1，将 SEG2PH<2:0>（CiCFG2<10:8>）置 1 以初始化相位段 2。

设置相位段的长度时必须满足以下要求：

- 传播时间 + 相位段 1 ≥ 相位段 2

19.6.5 采样点

采样点是读总线电平并将其解析为相应的位值的时间点。采样点位于相位段 1 的末尾。若位时序较慢而且包含很多 TQ，可以在采样点对总线线路进行多次采样。由 CAN 总线确定接收位的值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，具体位置取决于系统参数。

19.6.6 同步

为了补偿总线上各节点间振荡频率的相移，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，逻辑电路会将该沿的位置与预期时间（同步段）比较。电路将随后调整相位段 1 和相位段 2 的值。有两种同步机制。

19.6.6.1 硬同步

总线空闲期间只要出现从隐性到显性的边沿，才执行硬同步，表明报文传输的开始。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。

19.6.6.2 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。同步跳转宽度值将添加到相位段 1 或从相位段 2 中减去。重新同步跳转宽度可以编程为 1 TQ 到 4 TQ。

在设置 SJW<1:0> 位时，必须满足下列要求：

- 相位段 2 > 同步跳转宽度

表 19-1: dsPIC30F6010A 和 6015 器件的 CAN1 寄存器映射

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	—	—	—	接收接收过滤器 0 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C1RXF0EIDH	0302	—	—	—	接收接收过滤器 0 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXF0EIDL	0304	接收接收过滤器 0 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF1SID	0308	—	—	—	接收接收过滤器 1 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C1RXF1EIDH	030A	—	—	—	接收接收过滤器 1 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXF1EIDL	030C	接收接收过滤器 1 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF2SID	0310	—	—	—	接收接收过滤器 2 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C1RXF2EIDH	0312	—	—	—	接收接收过滤器 2 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXF2EIDL	0314	接收接收过滤器 2 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF3SID	0318	—	—	—	接收接收过滤器 3 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C1RXF3EIDH	031A	—	—	—	接收接收过滤器 3 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXF3EIDL	031C	接收接收过滤器 3 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF4SID	0320	—	—	—	接收接收过滤器 4 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C1RXF4EIDH	0322	—	—	—	接收接收过滤器 4 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXF4EIDL	0324	接收接收过滤器 4 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXF5SID	0328	—	—	—	接收接收过滤器 5 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C1RXF5EIDH	032A	—	—	—	接收接收过滤器 5 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXF5EIDL	032C	接收接收过滤器 5 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM0SID	0330	—	—	—	接收接收屏蔽寄存器 0 标准标识符 <10:0>											—	MIDE	000u uuuu uuuu uu0u	
C1RXM0EIDH	0332	—	—	—	接收接收屏蔽寄存器 0 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXM0EIDL	0334	接收接收屏蔽寄存器 0 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1RXM1SID	0338	—	—	—	接收接收屏蔽寄存器 1 标准标识符 <10:0>											—	MIDE	000u uuuu uuuu uu0u	
C1RXM1EIDH	033A	—	—	—	接收接收屏蔽寄存器 1 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C1RXM1EIDL	033C	接收接收屏蔽寄存器 1 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C1TX2SID	0340	发送缓冲器 2 标准标识符 <10:6>						—	—	—	发送缓冲器 2 标准标识符 <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX2EID	0342	发送缓冲器 2 扩展标识符 <17:14>						—	—	—	发送缓冲器 2 扩展标识符 <13:6>								uuuu 0000 uuuu uuuu
C1TX2DLC	0344	发送缓冲器 2 扩展标识符 <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>				—	—	—	uuuu uuuu uuuu u000	
C1TX2B1	0346	发送缓冲器 2 字节 1						发送缓冲器 2 字节 0										uuuu uuuu uuuu uuuu	
C1TX2B2	0348	发送缓冲器 2 字节 3						发送缓冲器 2 字节 2										uuuu uuuu uuuu uuuu	
C1TX2B3	034A	发送缓冲器 2 字节 5						发送缓冲器 2 字节 4										uuuu uuuu uuuu uuuu	
C1TX2B4	034C	发送缓冲器 2 字节 7						发送缓冲器 2 字节 6										uuuu uuuu uuuu uuuu	
C1TX2CON	034E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>		0000 0000 0000 0000	
C1TX1SID	0350	发送缓冲器 1 标准标识符 <10:6>						—	—	—	发送缓冲器 1 标准标识符 <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C1TX1EID	0352	发送缓冲器 1 扩展标识符 <17:14>						—	—	—	发送缓冲器 1 扩展标识符 <13:6>								uuuu 0000 uuuu uuuu
C1TX1DLC	0354	发送缓冲器 1 扩展标识符 <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>				—	—	—	uuuu uuuu uuuu u000	
C1TX1B1	0356	发送缓冲器 1 字节 1						发送缓冲器 1 字节 0										uuuu uuuu uuuu uuuu	

图注: u = 未初始化的位

注: 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 19-1: dsPIC30F6010A 和 6015 器件的 CAN1 寄存器映射 (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
C1TX1B2	0358	发送缓冲器 1 字节 3							发送缓冲器 1 字节 2							uuuu uuuu uuuu uuuu		
C1TX1B3	035A	发送缓冲器 1 字节 5							发送缓冲器 1 字节 4							uuuu uuuu uuuu uuuu		
C1TX1B4	035C	发送缓冲器 1 字节 7							发送缓冲器 1 字节 6							uuuu uuuu uuuu uuuu		
C1TX1CON	035E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>	0000 0000 0000 0000	
C1TX0SID	0360	发送缓冲器 0 标准标识符 <10:6>							发送缓冲器 0 标准标识符 <5:0>							SRR TXIDE	uuuu u000 uuuu uuuu	
C1TX0EID	0362	发送缓冲器 0 扩展标识符 <17:14>							发送缓冲器 0 扩展标识符 <13:6>							uuuu 0000 uuuu uuuu		
C1TX0DLC	0364	发送缓冲器 0 扩展标识符 <5:0>							TXRTR	TXRB1	TXRB0	DLC<3:0>			—	—	—	uuuu uuuu uuuu u000
C1TX0B1	0366	发送缓冲器 0 字节 1							发送缓冲器 0 字节 0							uuuu uuuu uuuu uuuu		
C1TX0B2	0368	发送缓冲器 0 字节 3							发送缓冲器 0 字节 2							uuuu uuuu uuuu uuuu		
C1TX0B3	036A	发送缓冲器 0 字节 5							发送缓冲器 0 字节 4							uuuu uuuu uuuu uuuu		
C1TX0B4	036C	发送缓冲器 0 字节 7							发送缓冲器 0 字节 6							uuuu uuuu uuuu uuuu		
C1TX0CON	036E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>	0000 0000 0000 0000	
C1RX1SID	0370	接收缓冲器 1 标准标识符 <10:0>							接收缓冲器 1 扩展标识符 <17:6>							SRR RXIDE	000u uuuu uuuu uuuu	
C1RX1EID	0372	接收缓冲器 1 扩展标识符 <5:0>							RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>			uuuu uuuu 000u uuuu
C1RX1B1	0376	接收缓冲器 1 字节 1							接收缓冲器 1 字节 0							uuuu uuuu uuuu uuuu		
C1RX1B2	0378	接收缓冲器 1 字节 3							接收缓冲器 1 字节 2							uuuu uuuu uuuu uuuu		
C1RX1B3	037A	接收缓冲器 1 字节 5							接收缓冲器 1 字节 4							uuuu uuuu uuuu uuuu		
C1RX1B4	037C	接收缓冲器 1 字节 7							接收缓冲器 1 字节 6							uuuu uuuu uuuu uuuu		
C1RX1CON	037E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	FILHIT<2:0>		0000 0000 0000 0000	
C1RX0SID	0380	接收缓冲器 0 标准标识符 <10:0>							接收缓冲器 0 扩展标识符 <17:6>							SRR RXIDE	000u uuuu uuuu uuuu	
C1RX0EID	0382	接收缓冲器 0 扩展标识符 <5:0>							RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>			uuuu uuuu 000u uuuu
C1RX0B1	0386	接收缓冲器 0 字节 1							接收缓冲器 0 字节 0							uuuu uuuu uuuu uuuu		
C1RX0B2	0388	接收缓冲器 0 字节 3							接收缓冲器 0 字节 2							uuuu uuuu uuuu uuuu		
C1RX0B3	038A	接收缓冲器 0 字节 5							接收缓冲器 0 字节 4							uuuu uuuu uuuu uuuu		
C1RX0B4	038C	接收缓冲器 0 字节 7							接收缓冲器 0 字节 6							uuuu uuuu uuuu uuuu		
C1RX0CON	038E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	DBEN	JTOFF	FILHIT0	0000 0000 0000 0000
C1CTRL	0390	CANCAP	—	CSIDLE	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	TXOIF	RX1IF	RX0IF	0000 0000 0000 0000
C1INTE	0398	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	TX2IE	TX1IE	TXOIE	RX1IE	RX0IE	0000 0000 0000 0000
C1EC	039A	发送错误计数寄存器							接收错误计数寄存器							0000 0000 0000 0000		

图注: u = 未初始化的位

注: 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 19-2: dsPIC30F6010A 的 CAN2 寄存器映射

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	
C2RXF0SID	03C0	—	—	—	接收验收过滤器 0 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C2RXF0EIDH	03C2	—	—	—	接收验收过滤器 0 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXF0EIDL	03C4	接收验收过滤器 0 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2RXF1SID	03C8	—	—	—	接收验收过滤器 1 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C2RXF1EIDH	03CA	—	—	—	接收验收过滤器 1 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXF1EIDL	03CC	接收验收过滤器 1 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2RXF2SID	03D0	—	—	—	接收验收过滤器 2 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C2RXF2EIDH	03D2	—	—	—	接收验收过滤器 2 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXF2EIDL	03D4	接收验收过滤器 2 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2RXF3SID	03D8	—	—	—	接收验收过滤器 3 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C2RXF3EIDH	03DA	—	—	—	接收验收过滤器 3 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXF3EIDL	03DC	接收验收过滤器 3 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2RXF4SID	03E0	—	—	—	接收验收过滤器 4 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C2RXF4EIDH	03E2	—	—	—	接收验收过滤器 4 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXF4EIDL	03E4	接收验收过滤器 4 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2RXF5SID	03E8	—	—	—	接收验收过滤器 5 标准标识符 <10:0>											—	EXIDE	000u uuuu uuuu uu0u	
C2RXF5EIDH	03EA	—	—	—	接收验收过滤器 5 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXF5EIDL	03EC	接收验收过滤器 5 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2RXM0SID	03F0	—	—	—	接收验收屏蔽寄存器 0 标准标识符 <10:0>											—	MIDE	000u uuuu uuuu uu0u	
C2RXM0EIDH	03F2	—	—	—	接收验收屏蔽寄存器 0 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXM0EIDL	03F4	接收验收屏蔽寄存器 0 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2RXM1SID	03F8	—	—	—	接收验收屏蔽寄存器 1 标准标识符 <10:0>											—	MIDE	000u uuuu uuuu uu0u	
C2RXM1EIDH	03FA	—	—	—	接收验收屏蔽寄存器 1 扩展标识符 <17:6>													0000 uuuu uuuu uuuu	
C2RXM1EIDL	03FC	接收验收屏蔽寄存器 1 扩展标识符 <5:0>							—	—	—	—	—	—	—	—	—	—	uuuu uu00 0000 0000
C2TX2SID	0400	发送缓冲器 2 标准标识符 <10:6>						—	—	—	发送缓冲器 2 标准标识符 <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C2TX2EID	0402	发送缓冲器 2 扩展标识符 <17:14>						—	—	—	发送缓冲器 2 扩展标识符 <13:6>								uuuu 0000 uuuu uuuu
C2TX2DLC	0404	发送缓冲器 2 扩展标识符 <5:0>						TXRTR	TXRB1	TXRB0	DLC<3:0>			—	—	—			uuuu uuuu uuuu u000
C2TX2B1	0406	发送缓冲器 2 字节 1						发送缓冲器 2 字节 0										uuuu uuuu uuuu uuuu	
C2TX2B2	0408	发送缓冲器 2 字节 3						发送缓冲器 2 字节 2										uuuu uuuu uuuu uuuu	
C2TX2B3	040A	发送缓冲器 2 字节 5						发送缓冲器 2 字节 4										uuuu uuuu uuuu uuuu	
C2TX2B4	040C	发送缓冲器 2 字节 7						发送缓冲器 2 字节 6										uuuu uuuu uuuu uuuu	
C2TX2CON	040E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>		0000 0000 0000 0000	
C2TX1SID	0410	发送缓冲器 1 标准标识符 <10:6>						—	—	—	发送缓冲器 1 标准标识符 <5:0>						SRR	TXIDE	uuuu u000 uuuu uuuu
C2TX1EID	0412	发送缓冲器 1 扩展标识符 <17:14>						—	—	—	发送缓冲器 1 扩展标识符 <13:6>								uuuu 0000 uuuu uuuu
C2TX1DLC	0414	发送缓冲器 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

注: 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 19-2: dsPIC30F6010A 的 CAN2 寄存器映射 (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
C2TX1B1	0416	发送缓冲器 1 字节 1							发送缓冲器 1 字节 0							uuuu uuuu uuuu uuuu		
C2TX1B2	0418	发送缓冲器 1 字节 3							发送缓冲器 1 字节 2							uuuu uuuu uuuu uuuu		
C2TX1B3	041A	发送缓冲器 1 字节 5							发送缓冲器 1 字节 4							uuuu uuuu uuuu uuuu		
C2TX1B4	041C	发送缓冲器 1 字节 7							发送缓冲器 1 字节 6							uuuu uuuu uuuu uuuu		
C2TX1CON	041E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>	0000 0000 0000 0000	
C2TX0SID	0420	发送缓冲器 0 标准标识符 <10:6>							发送缓冲器 0 标准标识符 <5:0>							SRR	TXIDE	uuuu u000 uuuu uuuu
C2TX0EID	0422	发送缓冲器 0 扩展标识符 <17:14>							发送缓冲器 0 扩展标识符 <13:6>									uuuu 0000 uuuu uuuu
C2TX0DLC	0424	发送缓冲器 0 扩展标识符 <5:0>							TXRTR	TXRB1	TXRB0	DLC<3:0>			—	—	—	uuuu uuuu uuuu u000
C2TX0B1	0426	发送缓冲器 0 字节 1							发送缓冲器 0 字节 0							uuuu uuuu uuuu uuuu		
C2TX0B2	0428	发送缓冲器 0 字节 3							发送缓冲器 0 字节 2							uuuu uuuu uuuu uuuu		
C2TX0B3	042A	发送缓冲器 0 字节 5							发送缓冲器 0 字节 4							uuuu uuuu uuuu uuuu		
C2TX0B4	042C	发送缓冲器 0 字节 7							发送缓冲器 0 字节 6							uuuu uuuu uuuu uuuu		
C2TX0CON	042E	—	—	—	—	—	—	—	—	—	TXABT	TXLARB	TXERR	TXREQ	—	TXPRI<1:0>	0000 0000 0000 0000	
C2RX1SID	0430	—	—	—	接收缓冲器 1 标准标识符 <10:0>							SRR	RXIDE	000u uuuu uuuu uuuu				
C2RX1EID	0432	—	—	—	接收缓冲器 1 扩展标识符 <17:6>									0000 uuuu uuuu uuuu				
C2RX1DLC	0434	接收缓冲器 1 扩展标识符 <5:0>							RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>			uuuu uuuu 000u uuuu
C2RX1B1	0436	发送缓冲器 1 字节 1							发送缓冲器 1 字节 0							uuuu uuuu uuuu uuuu		
C2RX1B2	0438	发送缓冲器 1 字节 3							发送缓冲器 1 字节 2							uuuu uuuu uuuu uuuu		
C2RX1B3	043A	发送缓冲器 1 字节 5							发送缓冲器 1 字节 4							uuuu uuuu uuuu uuuu		
C2RX1B4	043C	发送缓冲器 1 字节 7							发送缓冲器 1 字节 6							uuuu uuuu uuuu uuuu		
C2RX1CON	043E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	FILHIT<2:0>		0000 0000 0000 0000	
C2RX0SID	0440	—	—	—	接收缓冲器 0 标准标识符 <10:0>							SRR	RXIDE	000u uuuu uuuu uuuu				
C2RX0EID	0442	—	—	—	接收缓冲器 0 扩展标识符 <17:6>									0000 uuuu uuuu uuuu				
C2RX0DLC	0444	接收缓冲器 0 扩展标识符 <5:0>							RXRTR	RXRB1	—	—	—	RXRB0	DLC<3:0>			uuuu uuuu 000u uuuu
C2RX0B1	0446	发送缓冲器 0 字节 1							发送缓冲器 0 字节 0							uuuu uuuu uuuu uuuu		
C2RX0B2	0448	发送缓冲器 0 字节 3							发送缓冲器 0 字节 2							uuuu uuuu uuuu uuuu		
C2RX0B3	044A	发送缓冲器 0 字节 5							发送缓冲器 0 字节 4							uuuu uuuu uuuu uuuu		
C2RX0B4	044C	发送缓冲器 0 字节 7							发送缓冲器 0 字节 6							uuuu uuuu uuuu uuuu		
C2RX0CON	044E	—	—	—	—	—	—	—	—	RXFUL	—	—	—	RXRTRRO	DBEN	JTOFF	FILHITO	0000 0000 0000 0000
C2CTRL	0450	CANCAP	—	CSIDLE	ABAT	CANCKS	REQOP<2:0>			OPMODE<2:0>			—	ICODE<2:0>			—	0000 0100 1000 0000
C2CFG1	0452	—	—	—	—	—	—	—	—	SJW<1:0>			BRP<5:0>					0000 0000 0000 0000
C2CFG2	0454	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>			0u00 0uuu uuuu uuuu
C2INTF	0456	RX0OVR	RX1OVR	TXBO	TXEP	RXEP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	TX2IF	TX1IF	TX0IF	RX1IF	RX0IF	0000 0000 0000 0000
C2INTE	0458	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	TX2IE	TX1IE	TX0IE	RX1IE	RX0IE	0000 0000 0000 0000
C2EC	045A	发送错误计数寄存器							接收错误计数寄存器							0000 0000 0000 0000		

图注: u = 未初始化的位

注: 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)。

dsPIC30F6010A/6015

注:

20.0 10 位高速模数转换器 (ADC) 模块

注 本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

10 位高速模数转换器 (ADC) 可将模拟输入信号转换为 10 位数字值。此模块基于逐次逼近型寄存器 (Successive Approximation Register, SAR) 架构，可实现最大 1 Msps 的采样速率。ADC 模块具有 16 个模拟输入通道，它们通过多路开关连接到 4 个采样 / 保持放大器。采样 / 保持电路的输出端用作产生转换结果的转换器的输入端。可通过软件将模拟参考电压选择为器件供电电压 (AVDD/AVSS) 或 (VREF+/VREF-) 引脚上的电压。当器件处在休眠模式时，A/D 转换器仍可工作，这是它独有的特性。

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)

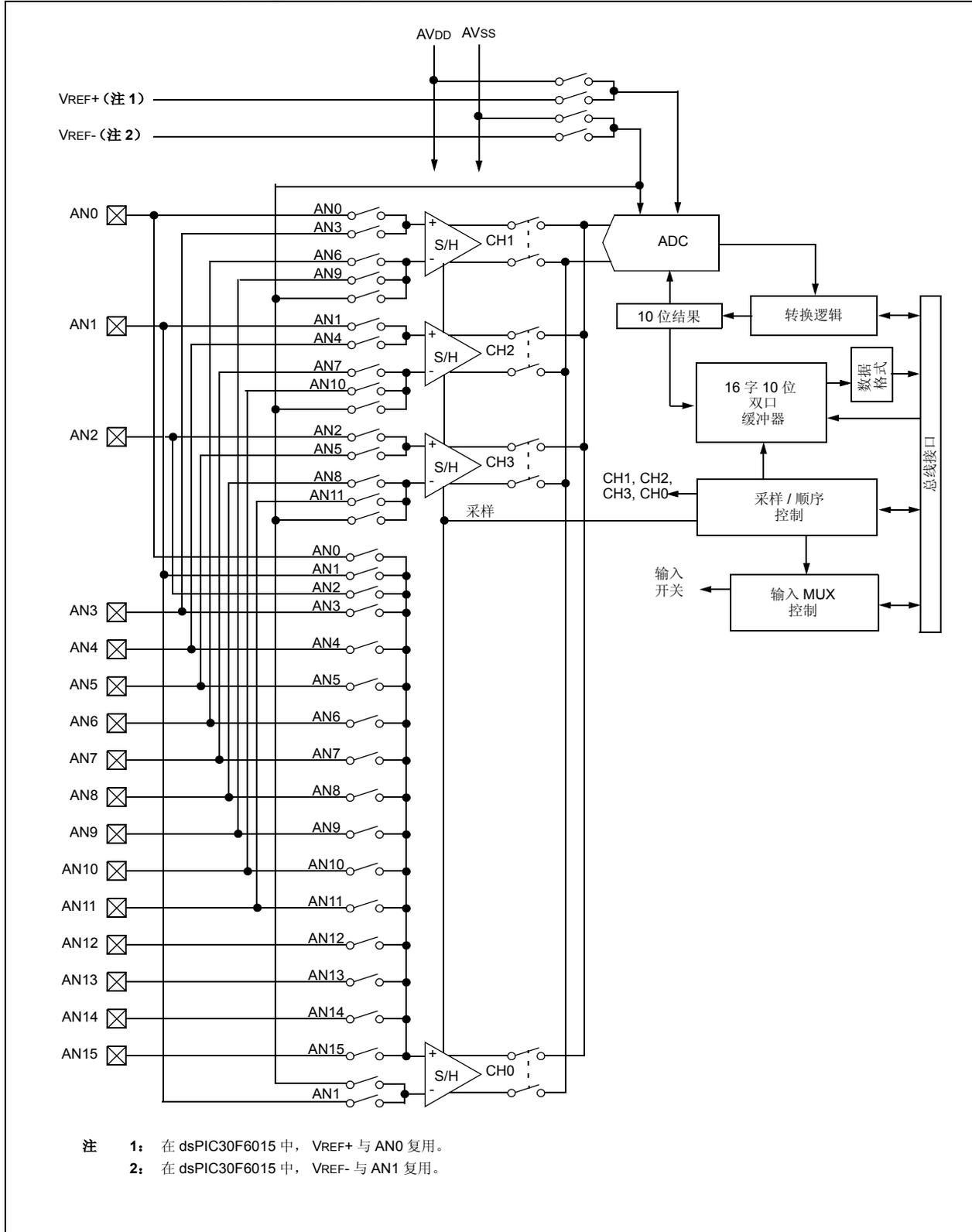
ADCON1、ADCON2 和 ADCON3 寄存器控制 A/D 模块的工作。ADCHS 寄存器用于选择要进行转换的输入通道。ADPCFG 寄存器用于将端口引脚配置为模拟输入或数字 I/O。ADCSSL 寄存器用于选择要扫描的输入通道。

注： 当 ADON = 1 时，不应在 SSRC<2:0>、ASAM、SIMSAM、SMPI<3:0>、BUFM 和 ALTS 位以及 ADCON3 和 ADCSSL 寄存器进行写入操作，否则将导致不确定的结果。

图 20-1 给出了 A/D 模块的框图。

dsPIC30F6010A/6015

图 20-1: 10 位高速 A/D 功能框图



20.1 A/D 结果缓冲器

模块包含一块称为 ADCBUF0 至 ADCBUFF 的 16 字双口只读缓冲器，对 A/D 转换结果进行缓冲。RAM 为 10 位宽，但其存储的数据可为不同格式的 16 位数据。用户软件不能对 16 个 A/D 转换结果缓冲寄存器 ADCBUF0 至 ADCBUFF 的内容进行写入操作。

20.2 转换操作

配置 A/D 模块后，将 SAMP 位置 1 以开始采样。编程位、定时器超时和外部事件等触发源将终止采集并启动转换。A/D 转换结束后，将转换结果装载到 ADCBUF0 至 ADCBUFF 中，且经过由 SMPI 位指定的采样数后将 A/D 中断标志位 ADIF 和 DONE 位置 1。

A/D 转换应遵循以下步骤：

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 转换触发源
 - 开启 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
3. 启动采样。
4. 等待所需的采集时间。
5. 由触发中止采集，启动转换
6. 可通过下列方式等待 A/D 转换结束：
 - 等待 A/D 中断
7. 读 A/D 结果缓冲器，清零 ADIF（如果需要）。

20.3 选择转换顺序

几组控制位可用来对 A/D 将输入连接至采样/保持通道、转换通道、写入缓冲器存储器和产生中断的过程进行选择控制。该过程将由采样时钟来控制。

SIMSAM 位用于控制多个通道的采集 / 转换过程。当 SIMSAM 位为 0 时，在 2 个或 4 个采样时钟内对 2 个或 4 个选中的通道进行采集，并按顺序进行转换。当 SIMSAM 位为 1 时，在一个采样时钟内对 2 个或 4 个选中的通道进行采集。然后按顺序转换通道。显然在只选中一个通道的情况下，SIMSAM 位不可用。

CHPS 位用于选择采样的通道数。通道数目可为 1、2 或 4 个通道不等。如果 CHPS 选择 1 个通道，则会在采样时钟控制下对 CH0 通道采样并进行转换。结果将存放在缓冲器中。如果 CHPS 选择 2 个通道，则对 CH0 和 CH1 通道进行采样并进行转换。如果 CHPS 选择 4 个通道，则对 CH0、CH1、CH2 和 CH3 通道进行采样并进行转换。

SMPI 位用于选择在中断发生前进行采集 / 转换的次数。中断产生前的采样 / 转换次数可为 1 至 16 次。

用户不能通过编程将 CHPS 和 SMPI 位的组合设置为每次中断进行超过 16 次或 8 次的转换，具体是 16 次还是 8 次取决于 BUFM 位的状态。BUFM 置 1 时，将把 16 字结果缓冲器分成 2 个 8 字缓冲器。每个中断事件将交替写入这两个 8 字缓冲器。BUFM 位的使用取决于中断发生后从缓冲器移出数据的可用时间，该时间是由应用程序决定的。

如果处理器可在对一个通道进行采样和转换的时间内快速卸空一个满的缓冲器，则 BUFM 位可为 0 且每次中断可进行最多 16 次转换。处理器在一次采样和转换的时间内移出 16 个转换结果。

如果在采样和转换时间内处理器不能完全卸空一个满缓冲器，BUFM 位应为 1。例如，如果 $SMPI<3:0>$ ($ADCON2<5:2>$) = 0111，则先将 8 个转换结果装入其中一个 8 字缓冲器，此后将产生中断。再将另外的 8 个转换结果装入另一个 8 字缓冲器。处理器将利用相邻中断之间的全部时间来移出 8 个转换结果。

ALTS 位用于在采样过程中切换选中的输入通道。输入多路开关具有 2 组采样输入：MUX A 和 MUX B。如果 ALTS 位为 0，仅选择 MUX A 输入用于采样。如果 ALTS 位为 1 且 $SMPI<3:0>$ = 0000，在第一个采样 / 转换过程中选择 MUX A 输入通道，而在接下来的采样 / 转换过程中选择 MUX B 输入通道。

CSCNA 位 ($ADCON2<10>$) 可使 CH0 通道的各输入被交替扫描，这些输入为选定数量的 MUX A 类模拟输入。这些输入由 ADCSSL 寄存器选择。如果 ADCSSL 寄存器中的某个位为 1，则将选择相应的输入。将在每次中断之后，始终按照输入编号从低至高的顺序对输入进行扫描。如果选择的输入数目大于每次中断进行的采样数，则编号高于采样数的输入将不会被使用。

20.4 通过编程触发转换

转换触发将终止采集并启动所需的转换。

SSRC<2:0> 位用于选择转换触发源。

SSRC 位可提供最多 5 个备用的转换触发源。

当 SSRC<2:0> = 000 时，转换触发是由软件控制的。

清零 SAMP 位将引发转换触发。

当 SSRC<2:0> = 111（自启动模式）时，转换触发是由 A/D 时钟控制的。SAMC 位用于选择从采样开始至转换开始之间的 A/D 时钟数。从而提供了最快的多通道转换速率。SAMC 必须至少等于 1 个时钟周期。

其他触发源可能来自定时器模块、电机控制 PWM 模块或外部中断。

注： 要使 A/D 工作在最大指定转换速率，必须选中自动转换触发选项（SSRC = 111）且必须将自动采样时间设置为 1 个 TAD（SAMC = 00001）。此配置下的全部转换周期（采样时间 + 转换时间）为 13 个 TAD。

使用其他类型的转换触发信号将导致增加额外的 TAD 周期，增加的时间用于同步外部事件和 A/D。

20.5 中止转换

转换过程中清零 ADON 位将中止当前的转换并停止采样。不会用部分完成的 A/D 采样转换来更新 ADCBUF。即，ADCBUF 仍然保持上一次转换完成后的值（即上一次写入该缓冲器的值）。

如果清零 ADON 位与自启动同时发生，清零 ADON 位具有更高的优先级。

A/D 转换中止后，通过将 SAMP 位置 1 启动下一次采样之前必须等待 2 个 TAD。

在顺序采样的情况下，A/D 将在与下一个待转换的通道相对应的下一个采样脉冲处继续工作。在同步采样的情况下，A/D 将继续处理下一个多通道组转换过程。

20.6 选择 A/D 转换时钟

A/D 转换需要 12 个 TAD。通过使用 6 位计数器，由软件选择 A/D 转换时钟，共有 64 种可能的选择。

公式 20-1: A/D 转换时钟

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

$$ADCS<5:0> = 2 \frac{TAD}{T_{CY}} - 1$$

通过将 ADRC 位置 1 可选择内部 RC 振荡器作为转换时钟。

为了 A/D 转换的正确性，必须选择合适的 A/D 转换时钟（TAD）以确保 83.33 ns 的最小 TAD 时间（VDD = 5V 时）。其他工作条件下的最小 TAD，请参见第 24.0 节“电气特性”。

例 20-1 中给出了计算 ADCS<5:0> 位的示例，假设器件的工作速度为 30 MIPS。

表 20-1: A/D 转换时钟计算

$TAD = 84 \text{ ns}$ $T_{CY} = 33 \text{ ns} \text{ (30 MIPS)}$
$ADCS<5:0> = 2 \frac{TAD}{T_{CY}} - 1$ $= 2 \cdot \frac{84 \text{ ns}}{33 \text{ ns}} - 1$ $= 4.09$
<p>因此， 设置 ADCS<5:0> = 5</p>
$\text{实际 TAD} = \frac{T_{CY}}{2} (ADCS<5:0> + 1)$ $= \frac{33 \text{ ns}}{2} (5 + 1)$ $= 99 \text{ ns}$

20.7 A/D 转换速度

dsPIC30F 的 10 位 A/D 转换器规范所允许的最大采样速率为 1 Msps。表 20-1 总结了 dsPIC30F 10 位 A/D 转换器的工作速度以及所需的工作条件。

表 20-1: 10 位 A/D 转换速率参数

dsPIC30F 10 位 A/D 转换器转换速率						
A/D 速度	TAD 最小值	采样时间最小值	Rs 最大值	VDD	温度	A/D 通道配置
最大 1 Msps ⁽¹⁾	83.33 ns	12 TAD	500Ω	4.5V 至 5.5V	-40°C 至 +85°C	
最大 750 ksps ⁽¹⁾	95.24 ns	2 TAD	500Ω	4.5V 至 5.5V	-40°C 至 +85°C	
最大 600 ksps ⁽¹⁾	138.89 ns	12 TAD	500Ω	3.0V 至 5.5V	-40°C 至 +125°C	
最大 500 ksps	153.85 ns	1 TAD	5.0 kΩ	4.5V 至 5.5V	-40°C 至 +125°C	
最大 300 ksps	256.41 ns	1 TAD	5.0 kΩ	3.0V 至 5.5V	-40°C 至 +125°C	

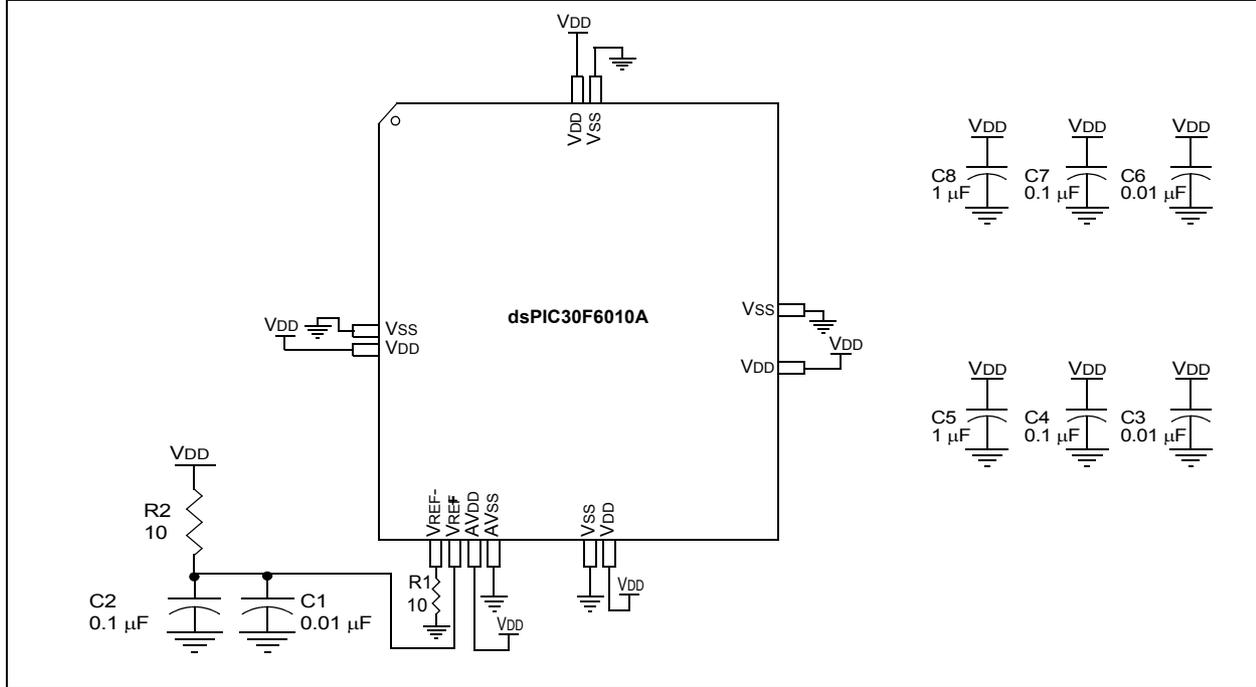
注 1: 为确保正常工作, 必须使用外部 VREF- 和 VREF+ 引脚。建议电路请参见图 20-2。

dsPIC30F6010A/6015

以下配置指导原则中给出了转换速度高于500 ksps时的所需设置值。因为它们需要使用外部 VREF 引脚，且它们在配置过程中有些不同。此处忽略了与转换速度关系不大的配置细节。

下图描述了转换速率高于 500 ksps 时的建议电路。

图 20-2: A/D 转换器参考电压原理图



20.7.1 1 Msps 时的配置指导原则

工作在 1 Msps 下的配置取决于对单输入引脚还是多引脚进行采样。

20.7.1.1 单路模拟输入

若要以 1 Msps 的速度对单个模拟输入进行转换，至少要使能两个采样 / 保持通道。必须配置模拟输入多路开关，以将同一个输入引脚既连接到采样通道，又连接到保持通道。A/D 在对一个 S/H 通道保持的值进行转换的同时，另外一个 S/H 通道采集新的输入。

20.7.1.2 多路模拟输入

A/D 转换器还可以使用多个采样 / 保持通道对多路模拟输入进行采样。在这种情况下，由不同的输入信号共享总的 1 Msps 的采样速率。例如，对 4 路输入进行采样，每个信号的采样速率为 250 ksps；或对 2 路输入进行采样，每个信号的采样速率为 500 ksps。此配置必须使用顺序采样以使每路输入有足够的采样时间。

20.7.1.3 1 Msps 配置项

为了获得 1 Msps 的转换速率，需要如下配置项。

- 符合表 20-2 中的工作条件
- 按图 20-2 中的建议电路连接外部 VREF+ 和 VREF- 引脚。
- 将 ADCON1 寄存器中的 SSRC<2:0> 设置为 111 以使能自动转换选项。
- 将 ADCON1 寄存器的 ASAM 控制位置 1 使能自动采样
- 将 ADCON1 寄存器的 SIMSAM 位清零使能顺序采样
- 通过写 ADCON2 寄存器的 CHPS<1:0> 控制位使能至少两个采样 / 保持通道
- 写 ADCON2 寄存器的 SMPI<3:0> 控制位设置两次中断之间的目标转换次数。至少要将 SMPI<3:0> 设置为 0001，因为至少要使能 2 个采样和保持通道。
- 通过写 ADCON3 寄存器的 ADCS<5:0> 控制位将 A/D 时钟周期配置为：

$$\frac{1}{12 \times 1,000,000} = 83.33 \text{ ns}$$

- 通过写 SAMC<4:0> = 00010，将采样时间配置为 2 TAD
- 通过写 ADCHS 寄存器为每个模拟输入引脚选择至少两个通道

20.7.2 750 ksps 时的配置指导原则

为获得 750 ksps 的转换速率，需要如下配置项。此配置假设对单路模拟输入进行采样。

- 符合表 20-2 中提供的条件
- 按图 20-2 中的建议电路连接外部 VREF+ 和 VREF- 引脚
- 将 ADCON1 寄存器中的 SSRC<2:0> 设置为 111 以使能自动转换选项。
- 将 ADCON1 寄存器的 ASAM 控制位置 1 使能自动采样
- 通过设置 ADCON2 寄存器中的 CHPS<1:0> = 00 使能采样和保持通道
- 写 ADCON2 寄存器的 SMPI<3:0> 控制位以获得两次中断之间的目标转换次数。
- 通过写 ADCON3 寄存器的 ADCS<5:0> 控制位将 A/D 时钟周期配置为：

$$\frac{1}{(12 + 2) \times 750,000} = 95.24 \text{ ns}$$

- 通过写 SAMC<4:0> = 00010，将采样时间配置为 2 TAD

20.7.3 600 ksps 时的配置指导原则

600 ksps 时的配置取决于要采样单个输入引脚还是多个引脚。

20.7.3.1 单路模拟输入

若要以 600 ksps 的速度对单个模拟输入进行转换，至少要使能两个采样 / 保持通道。必须配置模拟输入多路开关，以将同一个输入引脚既连接到采样通道，又连接到保持通道。A/D 在对一个 S/H 通道保持的值进行转换的同时，另外一个 S/H 通道采集新的输入。

20.7.3.2 多路模拟输入

A/D 转换器还可以使用多个采样 / 保持通道对多路模拟输入进行采样。在这种情况下，不同的输入信号共享总的 600 ksps 的采样速率。例如，对 4 路输入进行采样，每个信号的采样速率为 150 ksps；对 2 路输入进行采样，每个信号的采样速率为 300 ksps。此配置必须使用顺序采样以为每路输入提供足够的采样时间。

20.7.3.3 600 ksps 时的配置项

为获得 600 ksps 的转换速率，需要以下配置项。

- 符合表 20-2 中的工作条件
- 按图 20-2 中的建议电路连接外部 VREF+ 和 VREF- 引脚。
- 将 ADCON1 寄存器中的 SSRC<2:0> 设置为 111 以使能自动转换选项。
- 将 ADCON1 寄存器的 ASAM 控制位置 1 使能自动采样
- 将 ADCON1 寄存器的 SIMSAM 位清零使能顺序采样
- 通过写 ADCON2 寄存器的 CHPS<1:0> 控制位使能至少两个采样 / 保持通道
- 写 ADCON2 寄存器的 SMPI<3:0> 控制位设置两次中断之间的目标转换次数。至少将 SMPI<3:0> 设置为 0001，因为至少要使能 2 个采样和保持通道。
- 通过写 ADCON3 寄存器的 ADCS<5:0> 控制位将 A/D 时钟周期配置为：

$$\frac{1}{12 \times 600,000} = 138.89 \text{ ns}$$

- 通过写 SAMC<4:0> = 00010，将采样时间配置为 2 个 TAD
- 通过写 ADCHS 寄存器为每个模拟输入引脚选择最少两个通道

dsPIC30F6010A/6015

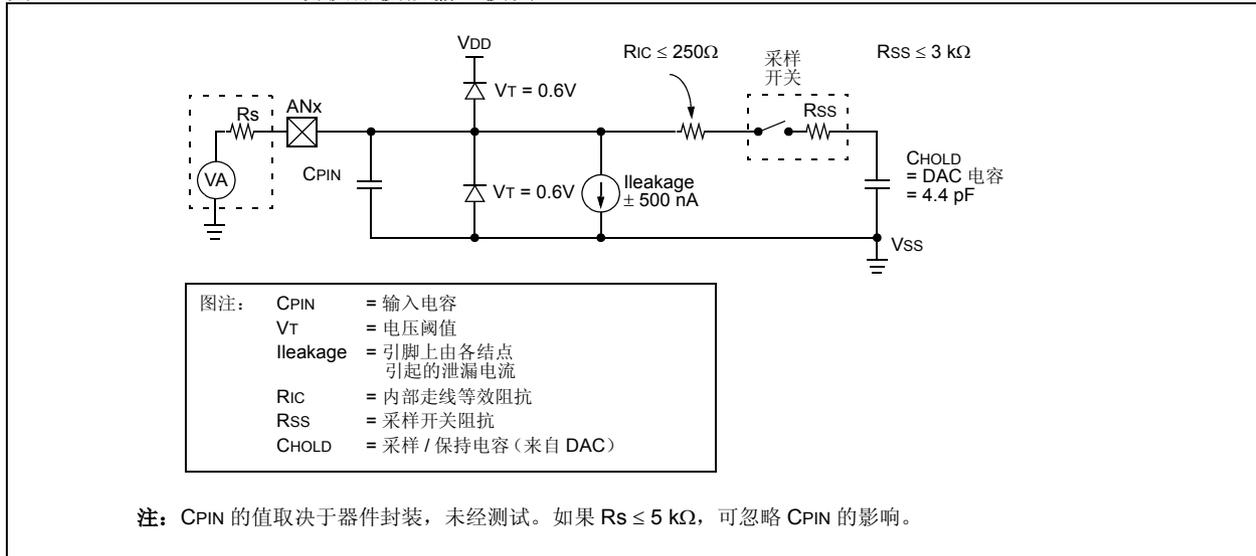
20.8 A/D 采集要求

图 20-3 中给出了 10 位 A/D 转换器的模拟输入模型。A/D 采样的总时间是内部放大器稳定时间、器件 VDD 和保持电容充电时间的函数。

为了使 A/D 转换器达到规定的精度，必须让充电保持电容 (CHOLD) 充分充电至模拟输入引脚上的电平。模拟输出源阻抗 (R_s)、内部走线等效阻抗 (R_{ic}) 和内部采样开关阻抗 (R_{ss}) 组合在一起直接影响给电容 CHOLD 充电所需的时间。该组合阻抗必须足够小以便在选择的采样时间内对保持电容充分充电。为了最大程度地减小引脚泄漏电路对 A/D 转换器精度的影响，在最大转换速率为 500 ksp/s 的情况下，建议信号源阻抗 R_s 的最大值为 5 k Ω ，在最大转换速率为 1 Msps 的情况下，建议信号源阻抗 R_s 的最大值为 500 Ω 。选中 (改变) 了模拟输入通道后，采样必须在转换开始前完成。在每次采样操作前，内部保持电容将处于放电状态。

两次转换之间应该至少留出 1 个 TAD 周期作为采样时间 TSAMP，采样时间的控制可通过置 1/ 清零 SAMP 位由软件手动完成或由 A/D 转换器自动完成。在自动配置中，用户必须在触发两次转换之间保留足够时间以满足最小采样时间。有关 TAD 和采样时间要求请参见第 24.0 节“电气特性”。

图 20-3: A/D 转换器模拟输入模块



20.9 模块掉电模式

模块具有 3 种内部功耗模式。当 ADON 位为 1 时，模块处于工作模式，并处于完全供电和工作状态。当 ADON 位为 0 时，模块处于关断模式。此时将禁止电路的数字和模拟部分以最大程度地节省电流。要想从关断模式回到工作模式，用户必须等待 ADC 电路进入稳定状态。

20.10 CPU 休眠和空闲模式下的 A/D 操作

20.10.1 CPU 休眠模式下的 A/D 操作

当器件进入休眠模式时，模块的所有时钟源将被关闭且保持为逻辑 0 状态。

如果在转换过程中进入休眠模式，转换将中止。当器件退出休眠模式时，转换器将不会继续进行部分完成的转换。

器件进入或退出休眠模式将不会对寄存器内容造成影响。

如果将 RC 用作 A/D 的时钟源 (ADRC = 1)，在休眠模式下，A/D 模块仍可继续工作。选中 RC 作为时钟源时，A/D 模块在开始转换之前将等待一个指令周期。以允许执行 SLEEP 指令，这将消除转换过程中所有的数字开关噪声。转换结束后将 DONE 位置 1，并将结果装入 ADCBUF 寄存器。

如果允许 A/D 中断，则器件将从休眠状态唤醒。如果 A/D 中断被禁止，ADC 模块将被关闭，尽管 ADON 位仍然保持置 1 状态。

20.10.2 CPU 空闲模式下的 A/D 操作

ADSIDL 位选择模块在空闲模式下停止工作还是继续工作。如果 ADSIDL 为零 (= 0)，器件进入空闲模式时，模块继续工作。如果 ADSIDL = 1，模块在空闲模式下将停止工作。

20.11 复位的影响

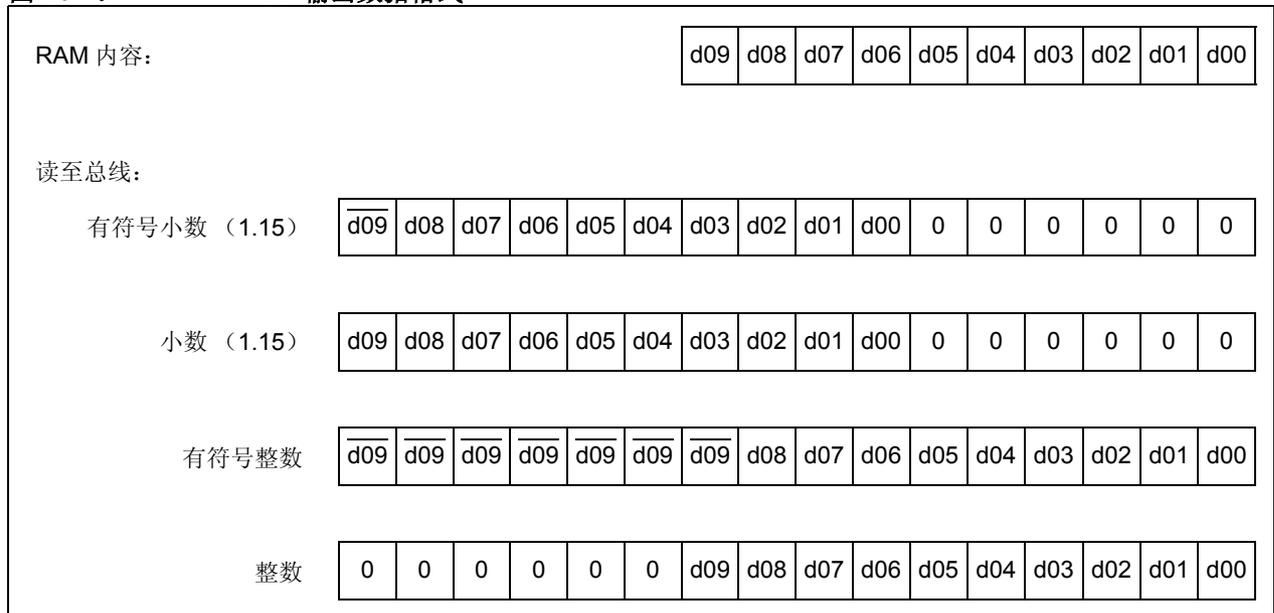
器件复位强制所有寄存器进入复位状态。这将强制关闭 A/D 模块，并中止任何正在进行的转换和采集。ADCBUF 寄存器中的值不会改变。上电复位后，A/D 结果寄存器将包含未知的数据。

20.12 输出格式

A/D 结果寄存器是 10 位宽的。数据缓冲器 RAM 也是 10 位宽的。当对该缓冲器执行读操作时，读出数据将以四种不同格式之一进行表示。FORM<1:0> 位用于选择格式。每一种输出格式将读出数据转换为数据总线上的 16 位结果。

写数据总是采用右对齐（整数）的格式。

图 20-4: A/D 输出数据格式



20.13 配置模拟端口引脚

ADPCFG 和 TRIS 寄存器用于控制 A/D 端口引脚的操作。将端口引脚用作模拟输入时，必须将其对应的 TRIS 位置 1（输入）。如果 TRIS 位清零（输出），将对数字输出电平（V_{OH} 或 V_{OL}）进行转换。

A/D 操作不受 CH0SA<3:0>/CH0SB<3:0> 位以及 TRIS 位状态的影响。

当对 PORT 寄存器进行读操作时，所有配置为模拟输入通道的引脚将被读为零。

配置为数字输入的引脚将不会对模拟输入进行转换。任何定义为数字输入的引脚（包括 AN_x 引脚）上的模拟电平可能导致输入缓冲器消耗的电流超过器件规范限值。

20.14 连接注意事项

模拟输入与 V_{DD} 和 V_{SS} 之间连接有二极管用于 ESD 保护。这就要求模拟输入电压必须介于 V_{DD} 和 V_{SS} 之间。如果输入电压超出此范围 0.3V 以上（任一方向上），就会有一个二极管正偏，而且如果超过输入电流规范值可能会损坏器件。

有时可外接 RC 滤波器来对输入信号进行抗混叠滤波。R 元件的选择要确保满足采样时间要求。应确保通过高阻抗连接到模拟输入引脚上的任何外部元件（如电容和齐纳二极管等）在引脚上只有极小的泄漏电流。

表 20-2: ADC 寄存器映射

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 数据缓冲器 0										0000 uuuu uuuu uuuu
ADCBUF1	0282	—	—	—	—	—	—	ADC 数据缓冲器 1										0000 uuuu uuuu uuuu
ADCBUF2	0284	—	—	—	—	—	—	ADC 数据缓冲器 2										0000 uuuu uuuu uuuu
ADCBUF3	0286	—	—	—	—	—	—	ADC 数据缓冲器 3										0000 uuuu uuuu uuuu
ADCBUF4	0288	—	—	—	—	—	—	ADC 数据缓冲器 4										0000 uuuu uuuu uuuu
ADCBUF5	028A	—	—	—	—	—	—	ADC 数据缓冲器 5										0000 uuuu uuuu uuuu
ADCBUF6	028C	—	—	—	—	—	—	ADC 数据缓冲器 6										0000 uuuu uuuu uuuu
ADCBUF7	028E	—	—	—	—	—	—	ADC 数据缓冲器 7										0000 uuuu uuuu uuuu
ADCBUF8	0290	—	—	—	—	—	—	ADC 数据缓冲器 8										0000 uuuu uuuu uuuu
ADCBUF9	0292	—	—	—	—	—	—	ADC 数据缓冲器 9										0000 uuuu uuuu uuuu
ADCBUFA	0294	—	—	—	—	—	—	ADC 数据缓冲器 10										0000 uuuu uuuu uuuu
ADCBUFB	0296	—	—	—	—	—	—	ADC 数据缓冲器 11										0000 uuuu uuuu uuuu
ADCBUFC	0298	—	—	—	—	—	—	ADC 数据缓冲器 12										0000 uuuu uuuu uuuu
ADCBUFD	029A	—	—	—	—	—	—	ADC 数据缓冲器 13										0000 uuuu uuuu uuuu
ADCBUFE	029C	—	—	—	—	—	—	ADC 数据缓冲器 14										0000 uuuu uuuu uuuu
ADCBUFF	029E	—	—	—	—	—	—	ADC 数据缓冲器 15										0000 uuuu uuuu uuuu
ADCON1	02A0	ADON	—	ADSIDL	—	—	—	FORM<1:0>	SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE	0000 0000 0000 0000	
ADCON2	02A2	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>	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	CH123NB<1:0>		CH123SB	CH0NB	CHOSB<3:0>			CH123NA<1:0>	CH123SA	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

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

dsPIC30F6010A/6015

注:

21.0 系统集成

注：本数据手册总结了 dsPIC30 系列器件的功能，但是并不作为参考大全使用。有关 CPU、外设、寄存器说明和通用器件功能的更多信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。欲知有关指令集和编程的更多详细信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B_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，通过减少外部元件的使用来降低成本，提供节能操作模式以及代码保护功能：

- 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)
- 省电模式 (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，它可以通过设置配置位永久使能，或用软件进行控制。它依靠自身的 RC 振荡器运行，以便提升可靠性。两个定时器提供必要的上电延时：一个是振荡器起振定时器 (OST)，确保在晶体振荡器达到稳定前，器件始终处于复位状态；另一个是上电延时定时器 (PWRT)，仅提供一个上电延时，用来在电源稳定过程中使器件保持在复位状态。有了这两个片内定时器，大部分的应用不再需要外部复位电路。

休眠模式用来提供一个低电流的掉电模式。用户可通过外部复位、看门狗定时器唤醒或中断将器件从休眠状态唤醒。某些振荡器也可用于这种模式，使此模式适合多种应用要求。在空闲模式下，时钟源依然有效，但 CPU 处于关闭状态。选择 RC 振荡器模式可节省系统的成本，而 LP 晶体模式可降低系统功耗。

21.1 振荡器系统概述

dsPIC30F 振荡器系统包含以下模块和功能：

- 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
- 在不同时钟源之间进行切换的时钟切换机制
- 可节省系统功耗的可编程时钟后分频器
- 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

配置位决定上电复位 (POR) 和欠压复位 (BOR) 时的时钟源。因此，可在允许的时钟源间切换时钟源。OSCCON 寄存器控制时钟切换并包含与系统时钟相关的状态位。

表 21-1 总结了 dsPIC30F 的振荡器工作模式。图 21-1 给出了振荡器系统的简化框图。

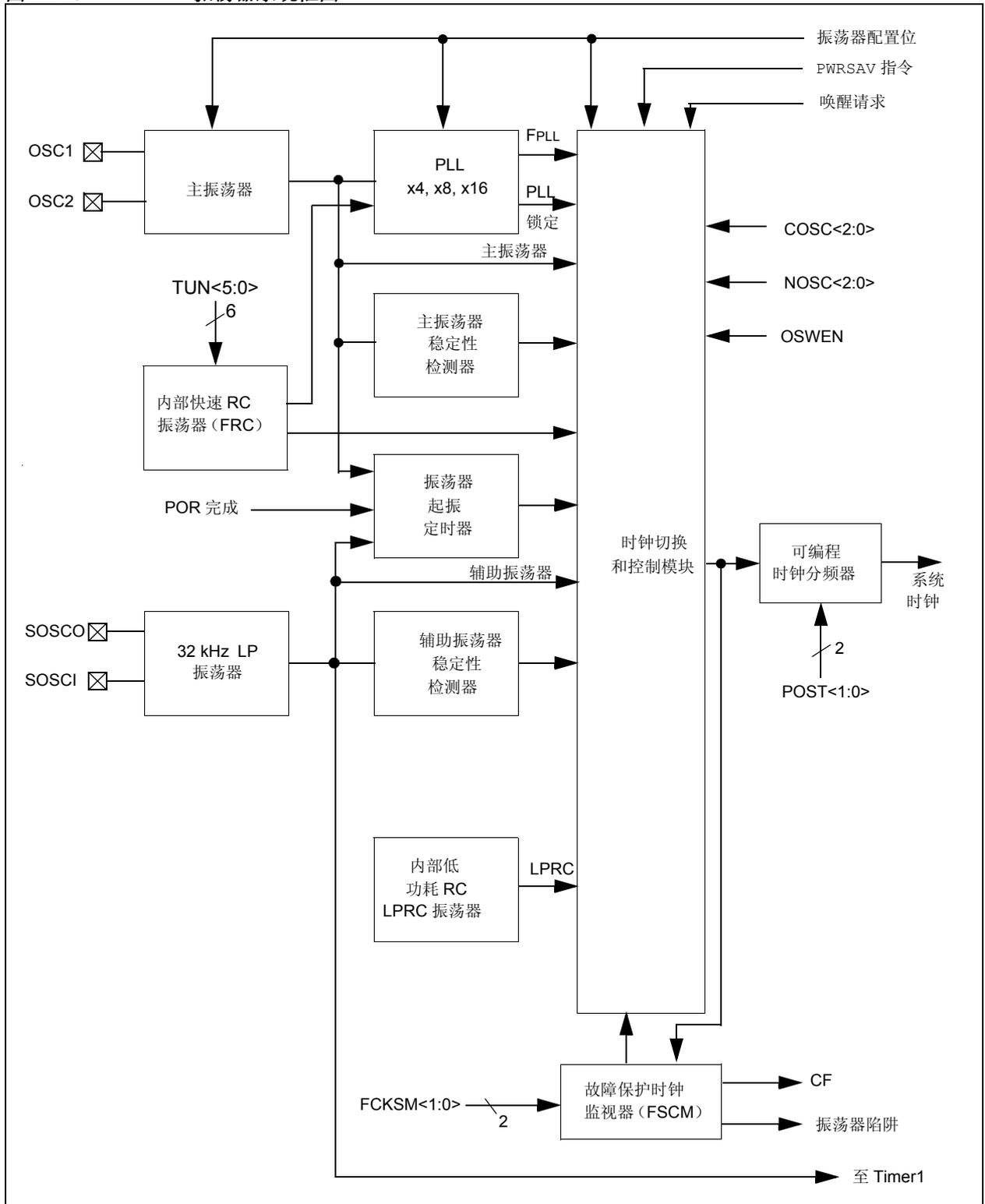
dsPIC30F6010A/6015

表 21-1: 振荡器工作模式

振荡器模式	说明
XTL	OSC1:OSC2 上的晶振频率为 200 kHz 至 4 MHz
XT	OSC1:OSC2 上的晶振频率为 4 MHz 至 10 MHz
带 4 倍频 PLL 的 XT	OSC1:OSC2 上的晶振频率为 4 MHz 至 10 MHz, 使能 4x PLL
带 8 倍频 PLL 的 XT	OSC1:OSC2 上的晶振频率为 4 MHz 至 10 MHz, 使能 8x PLL。
带 16 倍频 PLL 的 XT	OSC1:OSC2 上的晶振频率为 4 MHz 至 7.5 MHz, 使能 16x PLL ⁽¹⁾
LP	SOSCO:SOSCI ⁽²⁾ 上的晶振频率为 32 kHz
HS	晶振频率为 10 MHz 至 25 MHz。
带 4 倍频 PLL 的 HS/2	晶振频率为 10 MHz 至 20 MHz, 2 分频, 使能 4x PLL ⁽³⁾
带 8 倍频 PLL 的 HS/2	晶振频率为 10 MHz 至 20 MHz, 2 分频, 使能 8x PLL ⁽³⁾
带 16 倍频 PLL 的 HS/2	晶振频率为 10 MHz 至 15 MHz, 2 分频, 使能 16x PLL ⁽¹⁾ 。
带 4 倍频 PLL 的 HS/3	晶振频率为 12 MHz 至 25 MHz, 3 分频, 使能 4x PLL ⁽⁴⁾
带 8 倍频 PLL 的 HS/3	晶振频率为 12 MHz 至 25 MHz, 3 分频, 使能 8x PLL ⁽⁴⁾
带 16 倍频 PLL 的 HS/3	晶振频率为 12 MHz 至 22.5 MHz, 3 分频, 使能 16x PLL ⁽¹⁾
EC	外部时钟输入 (0 至 40 MHz)
ECIO	外部时钟输入 (0 至 40 MHz), OSC2 引脚为 I/O
带 4 倍频 PLL 的 EC	外部时钟输入 (4 至 10 MHz), OSC2 引脚为 I/O, 使能 4x PLL
带 8 倍频 PLL 的 EC	外部时钟输入 (4 至 10 MHz), OSC2 引脚为 I/O, 使能 8x PLL
带 16 倍频 PLL 的 EC	外部时钟输入 (4 至 7.5 MHz), OSC2 引脚为 I/O, 使能 16x PLL ⁽¹⁾
ERC	外部 RC 振荡器, OSC2 引脚为 Fosc/4 输出 ⁽⁵⁾
ERCIO	外部 RC 振荡器, OSC2 引脚为 I/O ⁽⁵⁾
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, 使能 4x PLL
带 8 倍频 PLL 的 FRC	7.37 MHz 内部 RC 振荡器, 使能 8x PLL
带 16 倍频 PLL 的 FRC	7.37 MHz 内部 RC 振荡器, 使能 16x PLL
LPRC	512 kHz 内部 RC 振荡器

- 注 1: 更高频率将超出工作频率范围。
 2: LP 振荡器可方便地为系统时钟或 Timer1 实时时钟所共用。
 3: 更高频率将超出 PLL 输入范围。
 4: 更低频率将超出 PLL 输入范围。
 5: 需要外部 R 和 C。最大工作频率为 4 MHz。

图 21-1: 振荡器系统框图



dsPIC30F6010A/6015

21.2 振荡器配置

21.2.1 初始时钟源选择

当退出上电复位或欠压复位状态后，器件将根据以下原则选择时钟源：

- FOS<2:0> 配置位选择四种振荡器组合中的一种，
- FPR<4:0> 配置位选择主振荡器组中的 16 个振荡器之一。

具体选择如表 21-2 所示。

表 21-2: 用于时钟选择的配置位值

振荡器模式	振荡器源	FOS<2:0>			FPR<4:0>					OSC2 功能
		1	1	1	0	1	1	0	1	
带 4 倍频 PLL 的 ECIO	PLL	1	1	1	0	1	1	0	1	I/O
带 8 倍频 PLL 的 ECIO	PLL	1	1	1	0	1	1	1	0	I/O
带 16 倍频 PLL 的 ECIO	PLL	1	1	1	0	1	1	1	1	I/O
带 4 倍频 PLL 的 FRC	PLL	1	1	1	0	0	0	0	1	I/O
带 8 倍频 PLL 的 FRC	PLL	1	1	1	0	1	0	1	0	I/O
带 16 倍频 PLL 的 FRC	PLL	1	1	1	0	0	0	1	1	I/O
带 4 倍频 PLL 的 XT	PLL	1	1	1	0	0	1	0	1	OSC2
带 8 倍频 PLL 的 XT	PLL	1	1	1	0	0	1	1	0	OSC2
带 16 倍频 PLL 的 XT	PLL	1	1	1	0	0	1	1	1	OSC2
带 4 倍频 PLL 的 HS/2	PLL	1	1	1	1	0	0	0	1	OSC2
带 8 倍频 PLL 的 HS/2	PLL	1	1	1	1	0	0	1	0	OSC2
带 16 倍频 PLL 的 HS/2	PLL	1	1	1	1	0	0	1	1	OSC2
带 4 倍频 PLL 的 HS/3	PLL	1	1	1	1	0	1	0	1	OSC2
带 8 倍频 PLL 的 HS/3	PLL	1	1	1	1	0	1	1	0	OSC2
带 16 倍频 PLL 的 HS/3	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
EC	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, 2)
FRC	内部 FRC	0	0	1	x	x	x	x	x	(注 1, 2)
LPRC	内部 LPRC	0	1	0	x	x	x	x	x	(注 1, 2)

注 1: 由主振荡器模式选择位 (FPR<4:0>) 决定 OSC2 引脚的功能。

2: 即使始终选择辅助振荡器或内部时钟源，OSC1 引脚仍不能用作 I/O 引脚。

21.2.2 振荡器起振定时器 (OST)

为了确保晶体振荡器（或陶瓷谐振器）已起振并达到稳定状态，振荡器中包含了一个振荡器起振定时器。该振荡器仅是一个 10 位计数器，在允许振荡器时钟应用于系统其他部分之前，将计数 1024 个 T_{OST} 周期。 T_{OST} 表示超时周期。每次振荡器重启（即，POR、BOR 和从休眠中唤醒）时都要涉及 T_{OST} 时间。振荡器起振定时器可应用于主振荡器的 LP、XT、XTL 和 HS 模式（从休眠中唤醒、POR 和 BOR 时）。

21.2.3 LP 振荡器控制

由以下两个元素控制 LP 振荡器的使能：

1. 当前振荡器组控制位 $COSC<2:0>$
2. $LPOSCEN$ 位 ($OSCCON$ 寄存器)

如果 $LPOSCEN = 1$ ，则 LP 振荡器开启（即使在休眠模式期间）。以下条件下，LP 振荡器为器件时钟：

- $COSC<2:0> = 000$ （选择 LP 作为主振荡器）
且
- $LPOSCEN = 1$

保持 LP 振荡器始终开启可快速切换至 32 KHz 系统时钟以进入低功耗工作状态。返回至更快速的主振荡器仍然需要一个起振延时。

21.2.4 锁相环 (PLL)

PLL 可倍频放大由主振荡器产生的时钟信号。PLL 的可选增益有 $\times 4$ 、 $\times 8$ 和 $\times 16$ 。表 21-3 总结了输入和输出频率范围。

表 21-3: PLL 频率范围

F_{in}	PLL 乘法器	F_{out}
4 MHz 至 10 MHz	$\times 4$	16 MHz 至 40 MHz
4 MHz 至 10 MHz	$\times 8$	32 MHz 至 80 MHz
4 MHz 至 7.5 MHz	$\times 16$	64 MHz 至 120 MHz

PLL 具有一个锁定输出，当 PLL 进入相位锁定状态时，锁定输出有效。如果锁相环锁定失败（例如由于噪声原因），则锁定信号将被视作无效。此信号的状态由 $OSCCON$ 寄存器中的只读位 $LOCK$ 反映。

21.2.5 快速 RC 振荡器 (FRC)

FRC 振荡器是一种快速（标称值为 7.37 MHz）的内部 RC 振荡器。该振荡器用于在不使用外部振荡器、陶瓷谐振器或 RC 网络的条件下，提供合理的器件工作频率。FRC 振荡器可与 PLL 一同使用以获得更高的时钟频率。

一旦 $OSCCON$ 寄存器的当前振荡器选择位 ($OSCCON<14:12>$) 设置为 001，dsPIC30F 就工作在 FRC 振荡器下。

由 $TUN<5:0>$ ($OSCTUN<5:0>$) 指定的 6 个位域允许用户调节内部快速 RC 振荡器（标称值为 7.37 MHz）。用户可以 0.4% 的步长在出厂校准值的 +12.6% (930 kHz) 至 -13% (960 kHz) 范围内对 FRC 振荡器进行调节，请参见表 20-4。

如果设置 $OSCCON<14:12>$ 为 111，且将 $FPR<4:0>$ 设置为 00101、00110 或 00111，则 PLL 倍频数将分别为 4、8 和 16。

注： 使用 16x PLL 时，不能将 FRC 的频率调节为大于 7.5 MHz。

表 21-4: FRC 调节

$TUN<5:0>$ 位	FRC 频率
01 1111	+12.6%
01 1110	+12.2%
01 1101	+11.8%
...	...
00 0100	+1.6%
00 0011	+1.2%
00 0010	+0.8%
00 0001	+0.4%
00 0000	中心频率（振荡器运行在经过校准的频率下）
11 1111	-0.4%
11 1110	-0.8%
11 1101	-1.2%
11 1100	-1.6%
...	...
10 0011	-11.8%
10 0010	-12.2%
10 0001	-12.6%
10 0000	-13.0%

21.2.6 低功耗 RC 振荡器 (LPRC)

LPRC 振荡器是看门狗定时器 (WDT) 的一个组件, 其标称振荡频率 512 kHz。LPRC 振荡器是上电延时定时器 (PWRT) 电路、WDT 和时钟监视器电路的时钟源。它也可用作功耗要求严格但时序精度要求不高的应用场合的低频时钟源。

因为 LPRC 振荡器是 PWRT 的时钟源, 因此上电复位时总会使能 LPRC 振荡器。PWRT 超时后, 如果以下任何一个条件为真, 则 LPRC 振荡器将保持为开启状态:

- 已使能故障保护时钟监视器
- 已使能 WDT
- 通过设置 OSCCON 寄存器中的 COSC<2:0> 控制位选择 LPRC 振荡器作为系统时钟

如果不满足以上条件中的任何一条, 则 LPRC 将在 PWRT 到期后关闭。

注 1: 由主振荡器模式选择位 (FPR<4:0>) 决定 OSC2 引脚的功能。

2: 注意, 即使始终选择辅助振荡器或内部时钟源, OSC1 引脚仍不能用作 I/O 引脚。

21.2.7 故障保护时钟监视器

故障保护时钟监视器 (FSCM) 允许器件在发生振荡器故障时继续运行。可通过正确编程 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) 使能 FSCM 功能。如果使能了 FSCM 功能, LPRC 内部振荡器将始终保持运行 (休眠模式除外), 且不受 SWDTEN 的控制。

发生振荡器故障时, FSCM 产生时钟故障陷阱事件, 并将系统时钟切换到 FRC 振荡器。然后用户可尝试重启振荡器或执行用户控制下的关闭。用户可将陷阱作为热复位来处理, 这只要把复位地址装入振荡器故障陷阱向量即可。这种情况下, 每次识别到时钟故障时, CF (时钟故障) 状态位 (OSCCON<3>) 也会置 1。

发生时钟故障时, WDT 不受影响, 且继续运行在 LPRC 时钟下。

从 POR、BOR 或休眠状态退出后, 如果振荡器的起振时间非常长的话, 则有可能出现这样的情况, 在振荡器启动前 PWRT 定时器就已超时。这种情况下, FSCM 将被激活且 FSCM 将启动一个时钟故障陷阱,

COSC<2:0> 位将载入 FRC 振荡器选项。这实际上是关闭了先前正在试图启动的振荡器。

在时钟故障陷阱 ISR 中, 用户可检测此情况并重启振荡器。

检测到时钟故障时, FSCM 模块将按如下步骤将时钟切换至 FRC 振荡器:

1. 把 FRC 振荡器选择位的值装入 COSC 位 (OSCCON<14:12>)。
2. 将 CF 位 (OSCCON<3>) 置 1。
3. 清零 OSWEN 控制位 (OSCCON<0>)。

出于时钟切换的目的, 可将时钟源分为以下四类:

1. 主时钟源
2. 辅助时钟源
3. 内部 FRC 时钟源
4. 内部 LPRC 时钟源

用户可在这些功能分类的时钟源间进行切换, 但不能在同一类中切换。如果选择了主钟源, 则在该类时钟源中的选择始终由 FPR<4:0> 配置位决定。

OSCCON 寄存器中包含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一类振荡器。
- NOSC<2:0>: 控制位, 向其写入值可表示新选择的一类振荡器。
 - POR 和 BOR 后, COSC<2:0> 和 NOSC<2:0> 均载入配置位的值 FOS<2:0>。
- LOCK: LOCK 状态位, 表示 PLL 是否锁定。
- CF: 只读状态位, 表示是否检测到了时钟故障。
- OSWEN: 控制位, 当启动一个时钟转换序列时, 由 0 变为 1。清零 OSWEN 控制位将中止正在进行的时钟转换 (用于挂起情形)。

如果配置位 FCKSM<1:0> = 1x, 则禁止时钟切换功能和故障保护时钟监视器功能。这是默认配置位设置。

如果禁止时钟切换, 则 FOS<2:0> 和 FPR<4:0> 位直接控制振荡器选择, 且 COSC<2:0> 位不控制时钟选择。但是, 这些位将反映时钟源选择。

注: 使能故障保护时钟监视器时, 应用不应试图将时钟频率切换至低于 100 kHz。如果执行了这样的时钟切换, 则器件可能产生一个振荡器故障陷阱, 并切换至快速 RC 振荡器。

21.2.8 防止意外写入 OSCCON

因为 OSCCON 寄存器控制时钟切换和时钟分频，所以有意将该寄存器的写操作设计得较为困难。

要写 OSCCON 低字节，必须执行以下代码序列，且中间不能插入其他指令：

向 OSCCON 的低字节写入字节 “0x46”
向 OSCCON 的低字节写入字节 “0x57”

规定字节写操作占用一个指令周期。写入期望值或使用位操作类指令。

要写 OSCCON 高字节，必须执行以下代码序列，且中间不能插入其他指令：

向 OSCCON 的高字节写入字节 “0x78”
向 OSCCON 的高字节写入字节 “0x9A”

规定字节写操作占用一个指令周期。写入期望值或使用位操作类指令。

21.3 复位

dsPIC30F 有以下几种复位方式：

- a) 上电复位 (POR)
- b) 正常工作时的 MCLR 复位
- c) 休眠时的 MCLR 复位
- d) 看门狗定时器 (WDT) 复位 (正常工作时)
- e) 可编程欠压复位 (BOR)
- f) RESET 指令
- g) 由陷阱锁定引起的复位 (TRAPR)
- h) 由非法操作码或由于将未初始化的 W 寄存器用作地址指针 (IOPUWR) 而引起的复位

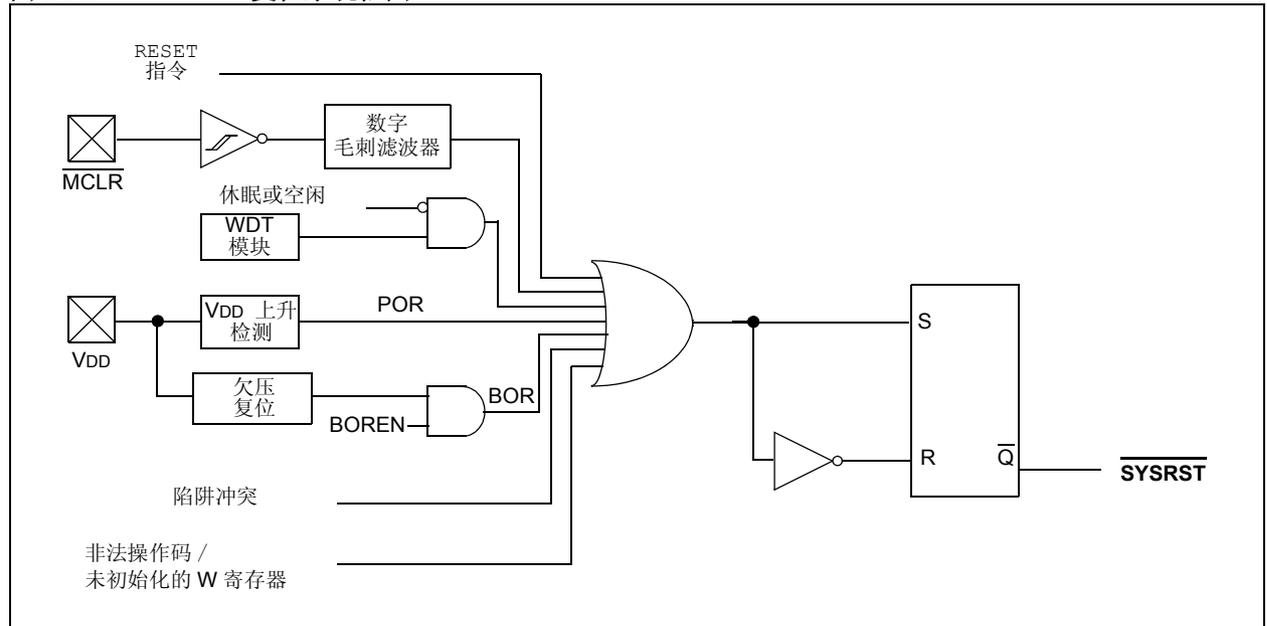
不同复位条件对不同寄存器的影响方式也不同。大多数寄存器不受 WDT 唤醒的影响，因为它被视作恢复正常工作。在不同复位状态下，RCON 寄存器的状态位的置位或清零情况也不相同，如表 21-5 所示。在软件中使用这些位可确定复位类型。

片上复位电路的框图如图 21-2 所示。

在 MCLR 复位路径上提供了一个 MCLR 噪声滤波器。该滤波器可检测并滤出小脉冲。

内部产生的复位不会将 MCLR 引脚驱动为低电平。

图 21-2: 复位系统框图



dsPIC30F6010A/6015

21.3.1 POR: 上电复位

当检测到 VDD 电平上升时，上电事件会产生一个内部 POR 脉冲信号。当电压达到 POR 电路门限电压 (VPOR, 标称值为 1.85V) 时，产生复位脉冲信号。器件供电电压特性必须满足规定的起始电压和上升速率要求。POR 脉冲信号将复位 POR 定时器并使器件进入复位状态。POR 也可通过振荡器配置位选择相应的器件时钟源。

POR 电路会插入标称值为 10 μ s 的一小段延时 TPOR，并确保器件偏置电路稳定。另外，还施加了用户选择的上电延时 (TPWRT)。TPWRT 参数由器件配置位设定，可以为 0 ms (无延时)、4 ms、16 ms 或 64 ms。器件上电延时的总延时为 TPOR + TPWRT。当这些延时超时后，SYSRST 将在 Q1 时钟周期的下一上升沿反相，PC 将跳转至复位向量。

$\overline{\text{SYSRST}}$ 信号的时序如图 21-3 至图 21-5 所示。

图 21-3: 上电延序列 ($\overline{\text{MCLR}}$ 连接至 VDD)

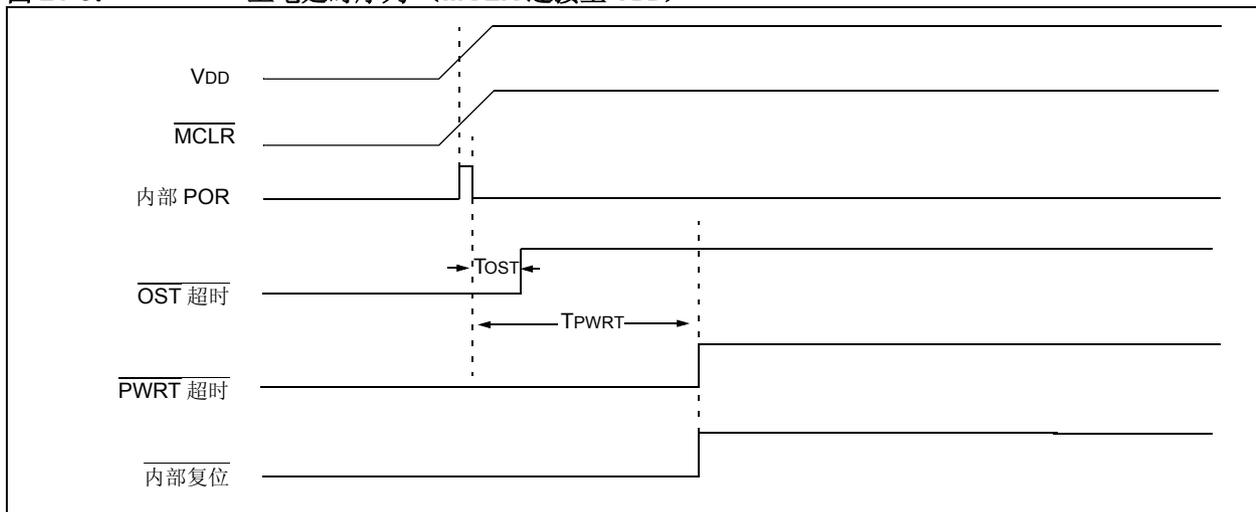


图 21-4: 上电延序列 ($\overline{\text{MCLR}}$ 未连接至 VDD): 情形 1

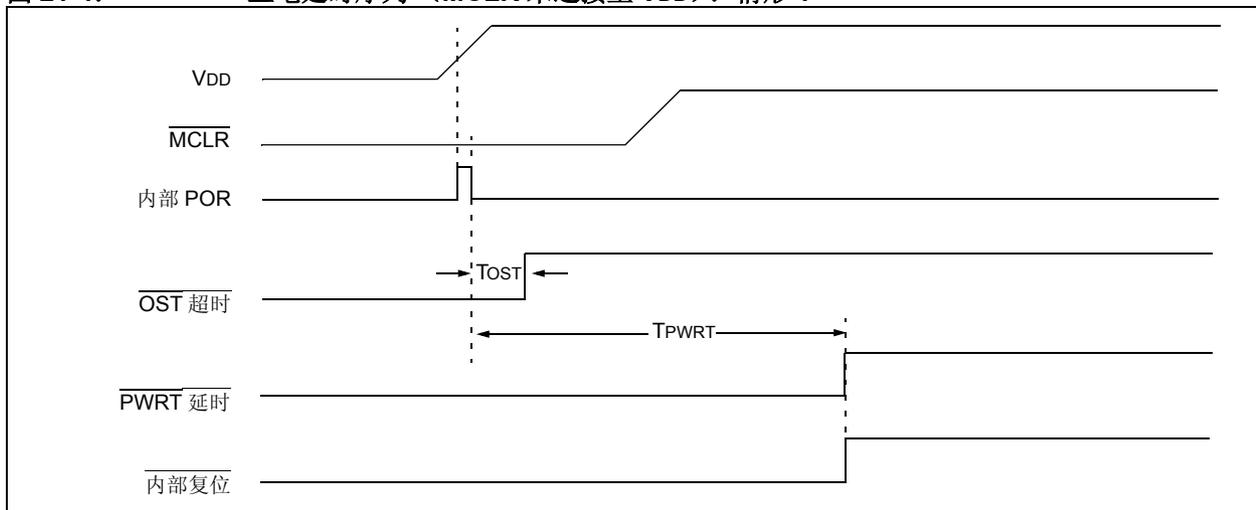
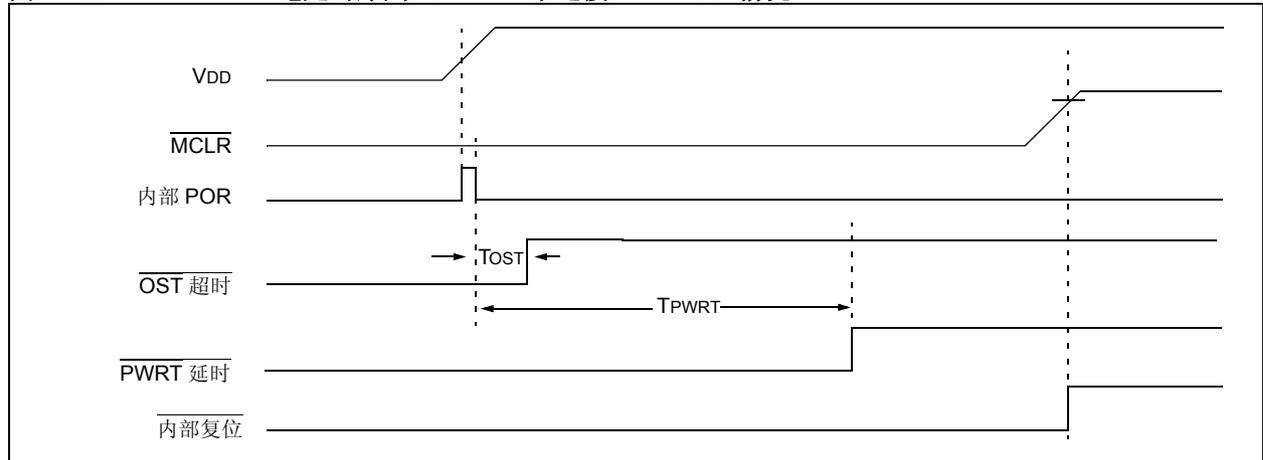


图 21-5: 上电延序列 (MCLR 未连接至 VDD): 情况 2



21.3.1.1 具有长晶振起振时间的 POR (使能 FSCM)

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) 具有相对较长的起振时间。因此, 当 POR 定时器和 PWRT 超时后, 有可能出现下列情况中的一种或多种:

- 振荡电路还未开始振荡。
- 振荡器起振定时器尚未超时 (如果使用了晶振)。
- PLL 还未锁定 (如果使用了 PLL)。

如果使能了 FSCM 且满足以上条件之一, 则将发生时钟故障陷阱。器件将自动切换至 FRC 振荡器, 且用户可在陷阱 ISR 中切换至所需的晶振。

21.3.1.2 FSCM 和 PWRT 禁止时的工作

如果禁止了 FSCM 和上电延时定时器 (PWRT), 则上电时器件将快速地退出复位。如果时钟源为 FRC、LPRC、EXTRC 或 EC, 则将立即激活。

如果禁止了 FSCM 且系统时钟尚未启动, 则在系统时钟启动前器件将在复位向量处处于冻结状态。从用户的角度来讲, 在系统时钟可用前, 器件似乎处于复位状态。

21.3.2 BOR: 可编程欠压复位

BOR (欠压复位) 模块基于一个内部电压参考电路。BOR 模块的主要作用是当发生欠压条件时产生器件复位。欠压条件通常由 AC 电源上的干扰信号 (即由于不良的电源传输线路导致的 AC 周期波形丢失, 或介入大感性负载时过电流导致的电压下降) 产生的。

BOR 模块允许选择以下电压跳变点其中之一:

- 2.6V-2.71V
- 4.1V-4.4V
- 4.58V-4.73V

注: 以上 BOR 电压跳变点为标称值, 仅供设计参考之用。

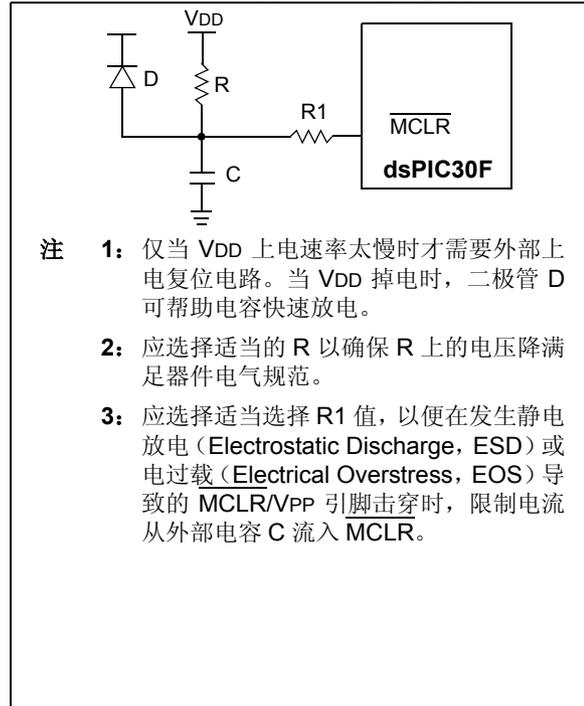
BOR 将产生复位器件的复位脉冲。BOR 将根据器件配置位值 (FOS<2:0> 和 FPR<4:0>) 选择时钟源。另外, 如果选择了振荡器模式, 则 BOR 将激活振荡器起振定时器 (OST)。OST 超时前一直使用系统时钟。如果使用了 PLL, 则在 LOCK 位 (OSCCON<5>) 变为 1 前一直使用该时钟。

dsPIC30F6010A/6015

同时，在释放内部复位前，将施加 POR 延时（TPOR）和 PWRT 延时（TPWRT）。如果 TPWRT = 0，且正在使用晶振，则使用标称值为 100 μs 的 TFSCM 延时。这种情况下的总延时为（TPOR + TFSCM）。

BOR 状态位（RCON<1>）将置 1 以表明已发生了 BOR。如果 BOR 电路已使能，则它将在休眠或空闲模式下继续工作，当 VDD 降至低于 BOR 门限电压时，它将复位器件。

图 21-6: 外部上电复位电路（用于缓慢 VDD 上电的情况）



注: 也可使用专用监控器件，如 MCP1XX 和 MCP8XX，作为外部上电复位电路。

表 21-5 显示了 RCON 寄存器的复位条件。由于 RCON 寄存器中的控制位为可读写的，所以此表中的信息表示在“条件”列中指定的操作之前，已对所有位进行了求反操作。

表 21-5: RCON 寄存器的初始化条件: 情形 1

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	空闲	休眠	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
正常工作时的 $\overline{\text{MCLR}}$ 操作	0x000000	0	0	1	0	0	0	0	0	0
正常工作时的软件复位	0x000000	0	0	0	1	0	0	0	0	0
休眠时的 $\overline{\text{MCLR}}$ 复位	0x000000	0	0	1	0	0	0	1	0	0
空闲时的 $\overline{\text{MCLR}}$ 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
从休眠中通过中断唤醒	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 = 未改变, x = 未知, - = 未实现位, 读为 0

注 1: 当唤醒由允许的中断引起时, PC 将装载相应的中断向量。

表 21-6 显示了 RCON 寄存器复位条件的第二个示例。在此示例中, 假定用户在“复位”列中所指定的操作前未置 1/ 清零特定位。

表 21-6: RCON 寄存器的初始化条件: 情形 2

条件	程序计数器	TRAPR	IOPUWR	EXTR	SWR	WDTO	空闲	休眠	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
正常工作时的 $\overline{\text{MCLR}}$ 操作	0x000000	u	u	1	0	0	0	0	u	u
正常工作时的软件复位	0x000000	u	u	0	1	0	0	0	u	u
休眠时的 $\overline{\text{MCLR}}$ 复位	0x000000	u	u	1	u	0	0	1	u	u
空闲时的 $\overline{\text{MCLR}}$ 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
从休眠中通过中断唤醒	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
非法操作复位	0x000000	u	1	u	u	u	u	u	u	u

图注: u = 未改变, x = 未知, - = 未用位, 读为 0

注 1: 当唤醒由已使能的中断引起时, PC 将装载相应的中断向量。

21.4 看门狗定时器 (WDT)

21.4.1 看门狗定时器操作

看门狗定时器 (WDT) 的主要功能是在发生软件故障时复位处理器。WDT 是独立运行的定时器，它使用片上 RC 振荡器，不需要外部组件。因此，即使主处理器时钟（如晶振）发生故障，WDT 定时器也将继续工作。

21.4.2 使能和禁止 WDT

只能通过配置寄存器 FWDEN 中的配置位 (FWDEN) “使能”或“禁止”看门狗定时器。

设置 FWDEN = 1 将使能看门狗定时器。在编程器件时完成使能。默认情况下，片擦除后，FWDEN 位 = 1。任何能够编程 dsPIC30F 器件的编程器均可编程此位及其他配置位。

如何使能了 WDT，它将递增，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）。要防止 WDT 超时，用户必须使用 CLRWDEN 指令清零看门狗定时器。

WDT 超时将强制器件复位（休眠期间除外）。RCON 寄存器中的 WDT0 位将清零以表明唤醒由 WDT 超时引起。

清零 FWDEN 可使用户软件通过 SWDEN (RCON<5>) 控制位使能/禁止看门狗定时器。

21.5 省电模式

可通过执行一条特殊的 PWRSAV 指令可进入两种省电模式：休眠和空闲。

PWRSAV 指令的格式如下：

PWRSAV <parameter>, 其中“parameter”定义空闲或休眠模式。

21.5.1 休眠模式

在休眠模式中，CPU 和外设的时钟被关闭。如果在使用片上振荡器，也将被关闭。

休眠期间，由于没有时钟要监视，所以故障保护时钟监视器不工作。但是，如果 WDT 在休眠中正常工作，则 LPRC 时钟仍然有效。

如果使能了欠压保护电路和低电压检测电路，则它们在休眠期间正常工作。

如果使能了看门狗定时器，则在 WDT 超时，处理器将从休眠模式唤醒。

发生以下任一事件时，处理器就会从休眠中唤醒：

- 任何中断，如果被单独允许且满足所需优先级
- 任何复位 (POR、BOR 和 MCLR)
- WDT 超时

从休眠模式唤醒时，处理器将使用在进入休眠模式之前处于工作状态的时钟源重新开始工作。使能时钟切换时，COSC<2:0> 位将确定唤醒时要使用的振荡器源。如果禁止了时钟切换，则仅有一个系统时钟。

注： 如果发生了 POR 或 BOR，则根据 FOS<2:0> 和 FPR<4:0> 配置位选择振荡器。

如果时钟源是一个振荡器，则 OST 超时（表明振荡器已稳定）前器件时钟保持关闭状态。如果使用了 PLL，则 LOCK = 1（表明 PLL 已稳定）前，系统时钟保持关闭状态。两种情况下，均使用了 TPOR、TLOCK 和 TPWRT 延时。

如果使用了 EC、FRC、LPRC 或 ERC 振荡器，则使用 TPOR (~ 10 μs) 延时。这是从休眠中唤醒时的最小可能延时。

另外，如果 LP 振荡器在休眠时有效，且唤醒时使用 LP 振荡器，则起振延时将等于 TPOR。未使用 PWRT 延时和 OST 定时器延时。为了获得从休眠中唤醒时的最小可能起振延时，应在进入休眠前选择这些快速唤醒选项之一。

任何单独允许（使用相应的 IE 位）并满足主要优先级要求的中断都唤醒处理器。处理器将处理中断并转移至 ISR。唤醒时，RCON 寄存器中的休眠状态位置 1。

注： 尽管使用了多种延时 (TPOR、TLOCK 和 TPWRT)，在延时结束时，晶体振荡器（和 PLL）可能仍未生效。如果已使能了 FSCM，器件将把这一情况检测为时钟故障，并将处理时钟故障陷阱，将使能 FRC 振荡器而用户需要重新使能晶体振荡器。如果未使能 FSCM，则器件将只是简单地暂停代码执行直到时钟稳定为止，且在振荡器时钟启动前保持休眠状态。

所有复位将使处理器从休眠模式中唤醒。除 POR 之外的任何复位，将置 1 休眠状态位。POR 时，休眠位被清零。

休眠和 WDT0 状态位均被置 1。

21.5.2 空闲模式

在空闲模式中，CPU 的时钟被关闭，而外设继续运行。与休眠模式不同，时钟源保持工作状态。

对于一些外设而言，每个模块都有一个控制位，允许外设在空闲模式中工作。

如果使能了时钟故障检测，则 LPRC 故障保护时钟保持工作状态。

发生以下任一事件时，处理器就会从空闲中唤醒：

- 任何中断，已单独允许（IE 位为 1）且满足所需优先级
- 任何复位（POR、BOR 和 MCLR）
- WDT 超时

从空闲模式唤醒时，时钟再次为 CPU 所有，且立即从 PWRSAV 指令之后的下一条指令开始执行指令。

任何单独允许（使用 IE 位）且优先级占先的中断都能唤醒处理器。处理器将处理中断并转移至 ISR。唤醒时 RCON 寄存器中的空闲状态位置 1。

除 POR 之外的任何复位，将置 1 空闲状态位。POR 时，空闲位被清零。

如果已使能了看门狗定时器，则处理器将在 WDT 超时后从空闲模式中唤醒。空闲和 WDTO 状态位均被置位。

与从休眠中唤醒不同，从空闲中唤醒时不存在延时。

21.6 器件配置寄存器

各器件配置寄存器中的配置位指定某些器件的模式，可由器件编程器或使用器件的在线串行编程（ICSP™）编程。各器件配置寄存器为 24 位寄存器，但各寄存器仅低 16 位可用于保存配置数据。有 6 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. FBS (0xF80006)：引导代码段配置寄存器
5. FBS (0xF80006)：安全代码段配置寄存器
6. FGS (0xF8000A)：通用代码段配置寄存器

在器件编程器中选择器件时，配置位会自动设置。可在源代码（取决于所使用的语言工具）中或通过编程界面指定所需的配置位状态。器件编程后，应用软件可使用表读指令读配置位值。更多信息，请参见《dsPIC30F 系列参考手册》(DS70157B_CN) 和《dsPIC30F 系列参考手册》(DS70046E_CN)。

- 注 1：** 如果已编程了代码保护配置位（FBS<BSS<2:0>）、FSS<SSS<2:0>）、FGS<GCP> 和 FGS<GWRP>），则只能在电压 $V_{DD} \geq 4.5V$ 下擦除整个代码保护器件。
- 2：** 此器件支持 CodeGuard™ 安全性的高级实现方式。有关如何在具体应用中使用 CodeGuard 安全性的信息，请参见《CodeGuard 安全性》(DS70180A_CN)。

21.7 外设模块禁止（PMD）寄存器

外设模块禁止（Peripheral Module Disable, PMD）寄存器通过停止所有提供给模块的时钟源来提供一种禁止外设模块的方法。当通过相应的 PMD 控制位禁止外设时，外设就进入了功耗最低的状态。与外设相关的控制寄存器和状态寄存器也被禁止，因此写入那些寄存器不会有影响，且读取值无效。

只有在 PMD 寄存器中的相应位被清零且特定的 dsPIC DSC 器件支持外设时，才会使能外设模块。如果外设包含在器件中，那么默认情况下，它在 PMD 寄存器中是使能的。

注： 如果 PMD 位置 1，那么相应的模块将在 1 个指令周期的延时后被禁止。类似地，如果 PMD 位清零，那么相应的模块将在 1 个指令周期的延时后被使能（假设已将模块控制寄存器配置为使能模块的工作）。

21.8 在线调试器

当选择 MPLAB® ICD 2 作为调试器时，使能在线调试功能。该功能允许与 MPLAB IDE 配合使用进行简单地调试。当使能该功能时，某些资源就不能用于一般用途了。这些资源包括数据 RAM 的前 80 个字节和两个 I/O 引脚。

用户可在 MPLAB IDE 中使用配置选项选择四对调试 I/O 引脚其中之一。这些引脚对的名称分别为 EMUD/EMUC、EMUD1/EMUC1、EMUD2/EMUC2 和 EMUD3/EMUC3。

各情况下，所选的 EMUD 引脚为仿真 / 调试数据线，而 EMUC 引脚为仿真 / 调试时钟线。这些引脚将与 Microchip 的 MPLAB ICD 2 模块连接。MPLAB ICD 2 使用所选的调试 I/O 引脚对发送命令、接收响应，以及发送和接收数据。要使用器件的在线调试器功能，就必须在设计中对 MCLR、V_{DD}、V_{SS}、PGC、PGD 和所选的 EMUD_x/EMUC_x 引脚对实现 ICSP 连接。

这样便产生了两种可能：

1. 如果选择 EMUD/EMUC 作为调试 I/O 引脚对，则仅需要一个 5 引脚的接口，因为在所有 dsPIC30F 器件中，EMUD 和 EMUC 引脚功能都与 PGD 和 PGC 引脚功能复用。
2. 如果选择 EMUD1/EMUC1、EMUD2/EMUC2 或 EMUD3/EMUC3 作为调试 I/O 引脚对，则需要一个 7 引脚的接口，因为 EMUD_x/EMUC_x（x=1、2或3）引脚功能未与 PGD 和 PGC 引脚功能复用。

表 21-7: 系统集成寄存器映射 (dsPIC30F6010A)

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	—	—	—	—	—	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	复位状态由复位类型决定。
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>			POST<1:0>		LOCK	—	CF	—	LPOSCEN	OSWEN	复位状态由配置位决定。
OSCTUN	0744	—	—	—	—	—	—	—	—	—	—	TUN<5:0>					0000 0000 0000 0000	
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QEIMD	PWMMD	—	I2CMD	U2MD	U1MD	SPI2MD	SPI1MD	C2MD	C1MD	ADCMD	0000 0000 0000 0000
PMD2	0772	IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000 0000 0000 0000

图注: u = 未初始化位

注: 有关寄存器位域的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 21-8: 系统集成寄存器映射 (dsPIC30F6015)

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	—	—	—	—	—	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	复位状态由复位类型决定。
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>			POST<1:0>		LOCK	—	CF	—	LPOSCEN	OSWEN	复位状态由配置位决定。
OSCTUN	0744	—	—	—	—	—	—	—	—	—	—	TUN<5:0>					0000 0000 0000 0000	
PMD1	0770	T5MD	T4MD	T3MD	T2MD	T1MD	QEIMD	PWMMD	—	I2CMD	U2MD	U1MD	SPI2MD	SPI1MD	—	C1MD	ADCMD	0000 0000 0000 0000
PMD2	0772	IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000 0000 0000 0000

图注: u = 未初始化位

注: 有关寄存器位域的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

表 21-9: 器件配置寄存器映射

寄存器名称	地址	Bit 23-16	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	HPOL	LPOL	BOREN	—	BORV<1:0>		—	—	FPWRT<1:0>		
FBS	F80006	—	—	—	RBS1	RBS0	—	—	—	EBS	—	—	—	—	BSS<2:0>		BWRP		
FSS	F80008	—	—	—	RSS1	RSS0	—	—	—	ESS1	ESS0	—	—	—	SSS<2:0>		SWRP		
FGS	F8000A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GSS<1:0>		GWRP	

图注: u = 未初始化位

注: 有关寄存器位域的说明, 请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

dsPIC30F6010A/6015

注:

22.0 指令集汇总

注：本数据手册总结了 dsPIC30F 系列器件的功能，但并不作为参考大全使用。有关 CPU、外设、寄存器说明和一般器件功能的更多信息，请参阅《dsPIC30F 系列参考手册》(DS70046E_CN)。有关器件指令集和编程的更多信息，请参阅《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

dsPIC30F 指令集的众多增强性能扩展了以往的 PIC[®] MCU 指令集，同时可以方便地从 PIC MCU 指令集移植到 dsPIC30F 指令集。

大部分指令的长度为一个程序存储字（24 位）。只有三条指令需要两个程序存储单元。

每一条单字指令长 24 位，分为一个指定指令类型的 8 位操作码和进一步指定指令操作的一个或多个操作数。

指令集是高度正交的，分为五个基本类别：

- 针对字或字节的操作
- 针对位的操作
- 立即数操作
- DSP 操作
- 控制操作

表 22-1 给出了在说明指令时使用的通用符号。

表 22-2 是 dsPIC30F 指令集汇总，还给出了每条指令影响的状态标志位。

大多数针对字或字节的 W 寄存器指令（包括桶形移位指令）有三个操作数：

- 第一个源操作数通常是寄存器 Wb，不带任何地址修改符
- 第二个源操作数通常是寄存器 Ws，带或不带地址修改符
- 运算结果的目的寄存器，通常是寄存器 Wd，带或不带地址修改符

不过，针对字或字节的文件寄存器指令只有两个操作数：

- 文件寄存器由“f”值指定
- 目的寄存器可以是文件寄存器“f”或 W0 寄存器（用“WREG”表示）

大多数位操作类指令（包括简单的循环/移位指令）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（由一个立即数指定，或者由“Wb”寄存器的内容间接指定）

涉及数据传送的立即数指令，可以使用下列操作数：

- 要被装入到 W 寄存器或文件寄存器中的立即数（由“k”值指定）
- 要装入立即数的 W 寄存器或文件寄存器（由“Wb”或“f”指定）

然而，涉及算术或逻辑操作的立即数指令，使用如下的操作数：

- 第一个源操作数是寄存器 Wb，不带任何地址修改符
- 第二个源操作数是立即数
- 操作结果的目的寄存器（仅在与第一个源操作数不同时）通常是寄存器 Wd（带或不带地址修改符）

MAC 类 DSP 指令可使用下列操作数：

- 要使用的累加器（A 或 B）（必须的操作数）
- 要用作第二个操作数的 W 寄存器
- X 和 Y 地址空间预取操作
- X 和 Y 地址空间预取目的寄存器
- 累加器回写目的寄存器

与乘法无关的其他 DSP 指令使用的操作数可能包括：

- 要使用的累加器（必须）
- 源操作数或目的操作数（分别由 Wso 或 Wdo 指定），带或不带地址修改符
- 移位位数，由 W 寄存器“Wn”或立即数指定

控制指令可以使用下列操作数：

- 程序存储地址
- 表读和表写指令的模式

除了某些双字指令外，所有指令都是单字指令；双字指令之所以是双字长的（48 位），是因为要用 48 位来提供所需信息。在第二个字中，8 个 MSb 全为 0。如果指令自身把第二个字当作一条指令来执行的话，它将作为一条 NOP 指令来执行。

dsPIC30F6010A/6015

大多数单字长指令都在一个指令周期内执行，除非条件测试为真、或者指令运行结果要改变程序计数器。在这样的情形下，执行就需要两个指令周期，多出的一个指令周期，就如同执行了一条 NOP 指令。值得注意的例外是 BRA（无条件 / 计算转移）、间接 CALL/GOTO、所有的表读和表写以及 RETURN/RETFIE 指令，它们是单字长指令，但执行需要两个或三个周期。某些与跳过后续指令有关的指令，如果要执行跳过的话，可能需要两个

或三个周期，这取决于被跳过的指令是单字还是双字指令。此外，双字传送需要两个周期。双字指令执行需要两个指令周期。

注： 关于指令集的更多细节，可参见《dsPIC30F/33F 程序员参考手册》(DS70157B_CN)。

表 22-1: 操作码描述中使用的符号

字段	说明
#text	表示由“text”定义的立即数
(text)	表示“text 的内容”
[text]	表示“text 寻址的地址单元”
{ }	可选的位域或操作
<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	累加器回写目的地址寄存器 $\in \{W13, [W13] + = 2\}$
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、溢出和全零位
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，对于字节模式， $\in \{0...255\}$ ；对于字模式， $\in \{0:1023\}$
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	字段无需内容，可为空
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 22-1: 操作码描述中使用的符号 (续)

字段	说明
Wb	基准 W 寄存器 $\in \{W0..W15\}$
Wd	目的 W 寄存器 $\in \{Wd, [Wd], [Wd++] , [Wd--], [++Wd], [--Wd]\}$
Wdo	目的 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 个目的工作寄存器之一 $\in \{W0..W15\}$
Wns	16 个源工作寄存器之一 $\in \{W0..W15\}$
WREG	W0 (文件寄存器指令中使用的工作寄存器)
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 $\in \{[W8]^+ = 6, [W8]^+ = 4, [W8]^+ = 2, [W8], [W8]^- = 6, [W8]^- = 4, [W8]^- = 2, [W9]^+ = 6, [W9]^+ = 4, [W9]^+ = 2, [W9], [W9]^- = 6, [W9]^- = 4, [W9]^- = 2, [W9+W12], \text{无}\}$
Wxd	用于 DSP 指令的 X 数据空间预取目的寄存器 $\in \{W4..W7\}$
Wy	用于 DSP 指令的 Y 数据空间预取地址寄存器 $\in \{[W10]^+ = 6, [W10]^+ = 4, [W10]^+ = 2, [W10], [W10]^- = 6, [W10]^- = 4, [W10]^- = 2, [W11]^+ = 6, [W11]^+ = 4, [W11]^+ = 2, [W11], [W11]^- = 6, [W11]^- = 4, [W11]^- = 2, [W11+W12], \text{无}\}$
Wyd	用于 DSP 指令的 Y 数据空间预取目的寄存器 $\in \{W4..W7\}$

dsPIC30F6010A/6015

表 22-2: 指令集汇总表

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
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 位有符号立即数加到累加器	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 中的指定位清零	1	1	无
		BCLR Ws, #bit4	将 Ws 中的指定位清零	1	1	无
6	BRA	BRA C, Expr	如果进位位为 1, 则转移	1	1 (2)	无
		BRA GE, Expr	如果有符号大于或等于, 则转移	1	1 (2)	无
		BRA GEU, Expr	如果无符号大于或等于, 则转移	1	1 (2)	无
		BRA GT, Expr	如果有符号大于, 则转移	1	1 (2)	无
		BRA GTU, Expr	如果无符号大于, 则转移	1	1 (2)	无
		BRA LE, Expr	如果有符号小于或等于, 则转移	1	1 (2)	无
		BRA LEU, Expr	如果无符号小于或等于, 则转移	1	1 (2)	无
		BRA LT, Expr	如果有符号小于, 则转移	1	1 (2)	无
		BRA LTU, Expr	如果无符号小于, 则转移	1	1 (2)	无
		BRA N, Expr	如果为负, 则转移	1	1 (2)	无
		BRA NC, Expr	如果进位位为 0, 则转移	1	1 (2)	无
		BRA NN, Expr	如果非负, 则转移	1	1 (2)	无
		BRA NOV, Expr	如果不溢出, 则转移	1	1 (2)	无
		BRA NZ, Expr	如果非零, 则转移	1	1 (2)	无
		BRA OA, Expr	如果累加器 A 溢出, 则转移	1	1 (2)	无
		BRA OB, Expr	如果累加器 B 溢出, 则转移	1	1 (2)	无
		BRA OV, Expr	如果溢出, 则转移	1	1 (2)	无
		BRA SA, Expr	如果累加器 A 饱和, 则转移	1	1 (2)	无
		BRA SB, Expr	如果累加器 B 饱和, 则转移	1	1 (2)	无
		BRA Expr	无条件转移	1	2	无
BRA Z, Expr	如果为零, 则转移	1	1 (2)	无		
BRA Wn	计算转移	1	2	无		
7	BSET	BSET f, #bit4	将 f 中的指定位置 1	1	1	无
		BSET Ws, #bit4	将 Ws 中的指定位置 1	1	1	无
8	BSW	BSW.C Ws, Wb	把 C 位内容写入 Ws<Wb>	1	1	无
		BSW.Z Ws, Wb	把 Z 位内容写入 <Wb>	1	1	无
9	BTG	BTG f, #bit4	将 f 中的指定位翻转	1	1	无
		BTG Ws, #bit4	将 Ws 中的指定位翻转	1	1	无
10	BTSC	BTSC f, #bit4	对 f 中的指定位进行测试, 如果清零则跳过	1	1 (2 或 3)	无
		BTSC Ws, #bit4	对 Ws 中的指定位进行测试, 如果清零则跳过	1	1 (2 或 3)	无

表 22-2: 指令集汇总表 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
11	BTSS	BTSS $f, \#bit4$	对 f 中的指定位进行测试, 如果为 1 则跳过	1	1 (2 或 3)	无
		BTSS $Ws, \#bit4$	对 Ws 中的指定位进行测试, 如果为 1 则跳过	1	1 (2 或 3)	无
12	BTST	BTST $f, \#bit4$	对 f 中的指定位进行测试	1	1	Z
		BTST.C $Ws, \#bit4$	对 Ws 中的指定位进行测试, 并将其值存储到 C	1	1	C
		BTST.Z $Ws, \#bit4$	对 Ws 中的指定位进行测试, 并将其反码存储到 Z	1	1	Z
		BTST.C Ws, Wb	对 $Ws < Wb >$ 位进行测试, 并将其值存储到 C	1	1	C
		BTST.Z Ws, Wb	对 $Ws < Wb >$ 进行测试, 并将其反码存储到 Z	1	1	Z
13	BTSTS	BTSTS $f, \#bit4$	对 f 中的指定位进行测试, 并将 f 中的该位置 1	1	1	Z
		BTSTS.C $Ws, \#bit4$	对 Ws 中的指定位进行测试, 并将其值存储到 C, 随后将 Ws 中的该位置 1	1	1	C
		BTSTS.Z $Ws, \#bit4$	对 Ws 中的指定位进行测试, 并将其反码存储到 Z, 随后将 Ws 中的该位置 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 $Acc, Wx, Wxd, Wy, Wyd, AWB$	将累加器清零	1	1	OA, OB, SA, SB
16	CLRWDT	CLRWDT	将看门狗定时器清零	1	1	WDTO, Sleep
17	COM	COM f	$f = f$	1	1	N, Z
		COM $f, WREG$	WREG = f	1	1	N, Z
		COM Ws, Wd	$Wd = Ws$	1	1	N, Z
18	CP	CP f	比较 f 与 WREG	1	1	C, DC, N, OV, Z
		CP $Wb, \#lit5$	比较 Wb 与 $lit5$	1	1	C, DC, N, OV, Z
		CP Wb, Ws	比较 Wb 与 Ws ($Wb - Ws$)	1	1	C, DC, N, OV, Z
19	CP0	CP0 f	比较 f 与 0x0000	1	1	C, DC, N, OV, Z
		CP0 Ws	比较 Ws 与 0x0000	1	1	C, DC, N, OV, Z
20	CPB	CPB f	带借位比较 f 与 WREG	1	1	C, DC, N, OV, Z
		CPB $Wb, \#lit5$	带借位比较 Wb 与 $lit5$	1	1	C, DC, N, OV, Z
		CPB Wb, Ws	带借位比较 Wb 与 Ws ($Wb - Ws - C$)	1	1	C, DC, N, OV, Z
21	CPSEQ	CPSEQ Wb, Wn	比较 Wb 与 Wn , 如果相等则跳过	1	1 (2 或 3)	无
22	CPSGT	CPSGT Wb, Wn	比较 Wb 与 Wn , 如果大于则跳过	1	1 (2 或 3)	无
23	CPSLT	CPSLT Wb, Wn	比较 Wb 与 Wn , 如果小于则跳过	1	1 (2 或 3)	无
24	CPSNE	CPSNE Wb, Wn	比较 Wb 与 Wn , 如果不相等则跳过	1	1 (2 或 3)	无
25	DAW	DAW Wn	$Wn =$ 十进制调整 Wn	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
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 $\#lit14, Expr$	执行 DO 循环代码到 $PC + Expr$, 执行次数为 $lit14 + 1$ 次	2	2	无
		DO $Wn, Expr$	执行 DO 循环代码到 $PC + Expr$, 执行次数为 $(Wn) + 1$ 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	交换 Wns 与 Wnd 的内容	1	1	无

dsPIC30F6010A/6015

表 22-2: 指令集汇总表 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
35	FBCL	FBCL Ws, Wnd	从左边 (MSb) 查找第一个位变化	1	1	C
36	FF1L	FF1L Ws, Wnd	从左边 (MSb) 查找第一个 1	1	1	C
37	FF1R	FF1R Ws, Wnd	从右边 (LSb) 查找第一个 1	1	1	C
38	GOTO	GOTO Expr	转移到地址	2	2	无
		GOTO Wn	间接转移到地址	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
42	LAC	LAC Ws, #lit5, Wd	$Wd = Wb .IOR.lit5$	1	1	N, Z
		LAC Wso, #Slit4, Acc	装载累加器	1	1	OA, OB, OAB, SA, SB, SAB
43	LNK	LNK #lit14	分配堆栈帧	1	1	无
44	LSR	LSR f	$f = f$ 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
45	MAC	MAC Wm*Wn, Acc, Wx, Wxd, Wy, Wyd, AWB	相乘并累加	1	1	OA, OB, OAB, SA, SB, SAB
		MAC Wm*Wm, Acc, Wx, Wxd, Wy, Wyd	平方并累加	1	1	OA, OB, OAB, SA, SB, SAB
46	MOV	MOV f, Wn	将 f 中的内容送入 Wn	1	1	无
		MOV f	将 f 中的内容送入目的寄存器	1	1	N, Z
		MOV f, WREG	将 f 中的内容送入 WREG	1	1	N, Z
		MOV #lit16, Wn	将 16 位立即数送入 Wn	1	1	无
		MOV.b #lit8, Wn	将 8 位立即数送入 Wn	1	1	无
		MOV Wn, f	将 Wn 中的内容送入 f	1	1	无
		MOV Wso, Wdo	将 Ws 中的内容送入 Wd	1	1	无
		MOV WREG, f	将 WREG 中的内容送入 f	1	1	N, Z
		MOV.D Wns, Wd	将 W(ns):W(ns+1) 中的双字内容送入 Wd	1	2	无
MOV.D Ws, Wnd	将 Ws 中的双字内容送入 W(nd+1):W(nd)	1	2	无		
MOV.D Ws, Wnd	将 Ws 中的双字内容送入 W(nd+1):W(nd)	1	2	无		
47	MOVSAC	MOVSAC Acc, Wx, Wxd, Wy, Wyd, AWB	预取操作并保存累加器	1	1	无
48	MPY	MPY Wm*Wn, Acc, Wx, Wxd, Wy, Wyd	Wm 与 Wn 相乘, 结果存入累加器	1	1	OA, OB, OAB, SA, SB, SAB
		MPY Wm*Wm, Acc, Wx, Wxd, Wy, Wyd	Wm 平方, 结果存入累加器	1	1	OA, OB, OAB, SA, SB, SAB
49	MPY.N	MPY.N Wm*Wn, Acc, Wx, Wxd, Wy, Wyd	Wm 与 Wn 相乘并取反, 结果存入累加器	1	1	无
50	MSC	MSC Wm*Wm, Acc, Wx, Wxd, Wy, Wyd, AWB	相乘并从累加器中减去	1	1	OA, OB, OAB, SA, SB, SAB
51	MUL	MUL.SS Wb, Ws, Wnd	$\{Wnd + 1, Wnd\} = signed(Wb) * signed(Ws)$	1	1	无
		MUL.SU Wb, Ws, Wnd	$\{Wnd + 1, Wnd\} = signed(Wb) * unsigned(Ws)$	1	1	无
		MUL.US Wb, Ws, Wnd	$\{Wnd + 1, Wnd\} = unsigned(Wb) * signed(Ws)$	1	1	无
		MUL.UU Wb, Ws, Wnd	$\{Wnd + 1, Wnd\} = unsigned(Wb) * unsigned(Ws)$	1	1	无
		MUL.SU Wb, #lit5, Wnd	$\{Wnd + 1, Wnd\} = signed(Wb) * unsigned(lit5)$	1	1	无
		MUL.UU Wb, #lit5, Wnd	$\{Wnd + 1, Wnd\} = unsigned(Wb) * unsigned(lit5)$	1	1	无
	MUL f		$W3:W2 = f * WREG$	1	1	无

表 22-2: 指令集汇总表 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
52	NEG	NEG <i>Acc</i>	将累加器内容求补	1	1	OA,OB,OAB,SA,SB,SAB
		NEG <i>f</i>	$f = f + 1$	1	1	C,DC,N,OV,Z
		NEG <i>f, WREG</i>	$WREG = f + 1$	1	1	C,DC,N,OV,Z
		NEG <i>Ws, Wd</i>	$Wd = Ws + 1$	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 <i>f</i>	从栈顶 (TOS) 的内容弹出至 <i>f</i>	1	1	无
		POP <i>Wdo</i>	从栈顶 (TOS) 的内容弹出至 <i>Wdo</i>	1	1	无
		POP.D <i>Wnd</i>	从栈顶 (TOS) 的内容弹出至 $W(nd):W(nd+1)$	1	2	无
		POP.S	把影子寄存器的内容弹出到主寄存器	1	1	全部
55	PUSH	PUSH <i>f</i>	把 <i>f</i> 的内容压入栈顶 (TOS)	1	1	无
		PUSH <i>Wso</i>	把 <i>Wso</i> 的内容压入栈顶 (TOS)	1	1	无
		PUSH.D <i>Wns</i>	把 $W(ns):W(ns+1)$ 的双字压入栈顶 (TOS)	1	2	无
		PUSH.S	把主寄存器中的双字内容压入影子寄存器	1	1	无
56	PWRSVAV	PWRSVAV #lit1	进入休眠或空闲模式	1	1	WDTO,Sleep
57	RCALL	RCALL <i>Expr</i>	相对调用	1	2	无
		RCALL <i>Wn</i>	计算调用	1	2	无
58	REPEAT	REPEAT #lit14	将下一条指令重复执行 lit14 + 1 次	1	1	无
		REPEAT <i>Wn</i>	将下一条指令重复执行 (Wn) + 1 次	1	1	无
59	RESET	RESET	软件器件复位	1	1	无
60	RETFIE	RETFIE	从中断返回	1	3 (2)	无
61	RETLW	RETLW #lit10, Wn	返回, 并将立即数存储入 <i>Wn</i>	1	3 (2)	无
62	RETURN	RETURN	从子程序返回	1	3 (2)	无
63	RLC	RLC <i>f</i>	$f =$ 对 <i>f</i> 执行带进位的循环左移	1	1	C,N,Z
		RLC <i>f, WREG</i>	$WREG =$ 对 <i>f</i> 执行带进位的循环左移	1	1	C,N,Z
		RLC <i>Ws, Wd</i>	$Wd =$ 对 <i>Ws</i> 执行带进位的循环左移	1	1	C,N,Z
64	RLNC	RLNC <i>f</i>	$f =$ 循环左移 <i>f</i> (不带进位)	1	1	N,Z
		RLNC <i>f, WREG</i>	$WREG =$ 循环左移 <i>f</i> (不带进位)	1	1	N,Z
		RLNC <i>Ws, Wd</i>	$Wd =$ 循环左移 <i>Ws</i> (不带进位)	1	1	N,Z
65	RRC	RRC <i>f</i>	$f =$ 对 <i>f</i> 执行带进位的循环右移	1	1	C,N,Z
		RRC <i>f, WREG</i>	$WREG =$ 对 <i>f</i> 执行带进位的循环右移	1	1	C,N,Z
		RRC <i>Ws, Wd</i>	$Wd =$ 对 <i>Ws</i> 执行带进位的循环右移	1	1	C,N,Z
66	RRNC	RRNC <i>f</i>	$f =$ 循环右移 <i>f</i> (不带进位)	1	1	N,Z
		RRNC <i>f, WREG</i>	$WREG =$ 循环右移 <i>f</i> (不带进位)	1	1	N,Z
		RRNC <i>Ws, Wd</i>	$Wd =$ 循环右移 <i>Ws</i> (不带进位)	1	1	N,Z
67	SAC	SAC <i>Acc, #Slit4, Wdo</i>	保存累加器内容	1	1	无
		SAC.R <i>Acc, #Slit4, Wdo</i>	保存舍入后的累加器内容	1	1	无
68	SE	SE <i>Ws, Wnd</i>	$Wnd =$ 对 <i>Ws</i> 进行符号扩展	1	1	C,N,Z
69	SETM	SETM <i>f</i>	$f = 0xFFFF$	1	1	无
		SETM <i>WREG</i>	$WREG = 0xFFFF$	1	1	无
		SETM <i>Ws</i>	$Ws = 0xFFFF$	1	1	无
70	SFTAC	SFTAC <i>Acc, Wn</i>	对累加器算术移位 (Wn) 次	1	1	OA,OB,OAB,SA,SB,SAB
		SFTAC <i>Acc, #Slit6</i>	对累加器算术移位 Slit6 次	1	1	OA,OB,OAB,SA,SB,SAB
71	SL	SL <i>f</i>	$f =$ 左移 <i>f</i>	1	1	C,N,OV,Z
		SL <i>f, WREG</i>	$WREG =$ 左移 <i>f</i>	1	1	C,N,OV,Z
		SL <i>Ws, Wd</i>	$Wd =$ 左移 <i>Ws</i>	1	1	C,N,OV,Z
		SL <i>Wb, Wns, Wnd</i>	$Wnd =$ 将 <i>Wb</i> 左移 <i>Wns</i> 位	1	1	N,Z
		SL <i>Wb, #lit5, Wnd</i>	$Wnd =$ 将 <i>Wb</i> 左移 lit5 位	1	1	N,Z

dsPIC30F6010A/6015

表 22-2: 指令集汇总表 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
72	SUB	SUB <i>Acc</i>	从累加器减去	1	1	OA,OB,OAB,SA,SB,SAB
		SUB <i>f</i>	$f = f - WREG$	1	1	C,DC,N,OV,Z
		SUB <i>f, WREG</i>	$WREG = f - WREG$	1	1	C,DC,N,OV,Z
		SUB <i>#lit10, Wn</i>	$Wn = Wn - lit10$	1	1	C,DC,N,OV,Z
		SUB <i>Wb, Ws, Wd</i>	$Wd = Wb - Ws$	1	1	C,DC,N,OV,Z
		SUB <i>Wb, #lit5, Wd</i>	$Wd = Wb - lit5$	1	1	C,DC,N,OV,Z
73	SUBB	SUBB <i>f</i>	$f = f - WREG - (C)$	1	1	C,DC,N,OV,Z
		SUBB <i>f, WREG</i>	$WREG = f - WREG - (C)$	1	1	C,DC,N,OV,Z
		SUBB <i>#lit10, Wn</i>	$Wn = Wn - lit10 - (C)$	1	1	C,DC,N,OV,Z
		SUBB <i>Wb, Ws, Wd</i>	$Wd = Wb - Ws - (C)$	1	1	C,DC,N,OV,Z
		SUBB <i>Wb, #lit5, Wd</i>	$Wd = Wb - lit5 - (C)$	1	1	C,DC,N,OV,Z
74	SUBR	SUBR <i>f</i>	$f = WREG - f$	1	1	C,DC,N,OV,Z
		SUBR <i>f, WREG</i>	$WREG = WREG - f$	1	1	C,DC,N,OV,Z
		SUBR <i>Wb, Ws, Wd</i>	$Wd = Ws - Wb$	1	1	C,DC,N,OV,Z
		SUBR <i>Wb, #lit5, Wd</i>	$Wd = lit5 - Wb$	1	1	C,DC,N,OV,Z
75	SUBBR	SUBBR <i>f</i>	$f = WREG - f - (C)$	1	1	C,DC,N,OV,Z
		SUBBR <i>f, WREG</i>	$WREG = WREG - f - (C)$	1	1	C,DC,N,OV,Z
		SUBBR <i>Wb, Ws, Wd</i>	$Wd = Ws - Wb - (C)$	1	1	C,DC,N,OV,Z
		SUBBR <i>Wb, #lit5, Wd</i>	$Wd = lit5 - Wb - (C)$	1	1	C,DC,N,OV,Z
76	SWAP	SWAP.b <i>Wn</i>	$Wn =$ 半字节交换 Wn 的内容	1	1	无
		SWAP <i>Wn</i>	$Wn =$ 将 Wn 的两个字节相交换	1	1	无
77	TBLRDH	TBLRDH <i>Ws, Wd</i>	将程序存储单元的 <23:16> 读入 $Wd<7:0>$	1	2	无
78	TBLRDL	TBLRDL <i>Ws, Wd</i>	将程序存储单元的 <15:0> 读入 Wd	1	2	无
79	TBLWTH	TBLWTH <i>Ws, Wd</i>	将 $Ws<7:0>$ 写入程序存储单元的 <23:16>	1	2	无
80	TBLWTL	TBLWTL <i>Ws, Wd</i>	将 Ws 写入程序存储单元的 <15:0>	1	2	无
81	ULNK	ULNK	释放堆栈帧	1	1	无
82	XOR	XOR <i>f</i>	$f = f .XOR.WREG$	1	1	N,Z
		XOR <i>f, WREG</i>	$WREG = f .XOR.WREG$	1	1	N,Z
		XOR <i>#lit10, Wn</i>	$Wd = lit10 .XOR.Wd$	1	1	N,Z
		XOR <i>Wb, Ws, Wd</i>	$Wd = Wb .XOR.Ws$	1	1	N,Z
		XOR <i>Wb, #lit5, Wd</i>	$Wd = Wb .XOR. lit5$	1	1	N,Z
83	ZE	ZE <i>Ws, Wnd</i>	$Wnd =$ 对 Ws 进行零扩展	1	1	C,Z,N

23.0 开发支持

一系列硬件及软件开发工具对 PIC® 单片机提供支持：

- 集成开发环境
 - MPLAB® IDE 软件
- 汇编器 / 编译器 / 链接器
 - MPASM™ 汇编器
 - MPLAB C18 和 MPLAB C30 C 编译器
 - MPLINK™ 目标链接器 / MPLIB™ 目标库管理器
 - MPLAB ASM30 汇编器 / 链接器 / 库
- 模拟器
 - MPLAB SIM 软件模拟器
- 仿真器
 - MPLAB ICE 2000 在线仿真器
 - MPLAB REAL ICE™ 在线仿真器
- 在线调试器
 - MPLAB ICD 2
- 器件编程器
 - PICSTART® Plus 开发编程器
 - MPLAB PM3 器件编程器
 - PICKit™ 2 开发编程器
- 低成本演示和开发板及评估工具包

23.1 MPLAB 集成开发环境软件

MPLAB IDE 软件为 8/16 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）
 - 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
- 可视化器件初始化程序，便于进行寄存器的初始化
- 鼠标停留在变量上进行查看的功能
- 通过拖放把变量从源代码窗口拉到观察窗口
- 丰富的在线帮助
- 集成了可选的第三方工具，如 HI-TECH 软件 C 编译器和 IAR C 编译器

MPLAB IDE 可以让您：

- 编辑源文件（汇编语言或 C 语言）
- 点击一次即可完成汇编（或编译）并将代码下载到 PIC MCU 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（汇编语言或 C 语言）
 - 混合汇编语言和 C 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更强大的工具时的学习时间。

23.2 MPASM 汇编器

MPASM 汇编器是全功能通用宏汇编器，适用于所有的 PIC 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 汇编器具有如下特征：

- 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

23.3 MPLAB C18 和 MPLAB C30 C 编译器

MPLAB C18 和 MPLAB C30 代码开发系统是完全的 ANSI C 编译器，分别适用于 Microchip 的 PIC18 和 PIC24 系列单片机及 dsPIC30F 和 dsPIC33 系列数字信号控制器。这些编译器可提供其他编译器并不具备的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，编译器提供了针对 MPLAB IDE 调试器的优化符号信息。

23.4 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中。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征：

- 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

23.5 MPLAB ASM30 汇编器、链接器和库管理器

MPLAB ASM30 汇编器为 dsPIC30F 器件提供转换自符号汇编语言的可重定位机器码。MPLAB C30 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，与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征：

- 支持整个 dsPIC30F 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

23.6 MPLAB SIM 软件模拟器

MPLAB SIM 软件模拟器在指令级对 PIC MCU 和 dsPIC® DSC 进行模拟，使得用户可以在 PC 主机的环境下进行代码开发。对于任何给定的指令，用户均可对数据区进行检查或修改，并通过各种触发机制来产生激励。可以将各寄存器的情况记录在文件中，以便进行进一步地运行时分析。跟踪缓冲器和逻辑分析器的显示使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的状况。

MPLAB SIM 软件模拟器完全支持使用 MPLAB C18 和 MPLAB C30 C 编译器以及 MPASM 和 MPLAB ASM30 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。

23.7 MPLAB ICE 2000 高性能在线仿真器

MPLAB ICE 2000 在线仿真器旨在为产品开发工程师提供一整套用于 PIC 单片机的设计工具。MPLAB ICE 2000 在线仿真器的软件控制由 MPLAB 集成开发环境平台提供，它允许在单一环境下进行编辑、编译、下载以及源代码调试。

MPLAB ICE 2000 是全功能仿真器系统，它具有增强的跟踪、触发和数据监控功能。处理器模块可插拔，使系统可轻松进行重新配置以适应各种不同处理器的仿真需要。MPLAB ICE 2000 在线仿真器的架构允许对其进行扩展以支持新的 PIC 单片机。

MPLAB ICE 2000 在线仿真器系统设计为一款实时仿真系统，该仿真系统具备通常只有昂贵的开发工具中才有的高级功能。选择 PC 平台和 Microsoft® Windows® 32 位操作系统可使这些功能在一个简单而统一的应用中得到很好的利用。

23.8 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® DSC 进行调试和编程。IDE 是随每个工具包一起提供的。

MPLAB REAL ICE 探针通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与常用 MPLAB ICD 2 系统兼容的连接器 (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 MPLAB REAL ICE 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，如软件断点和汇编代码跟踪等。在同类仿真器中，MPLAB REAL ICE 的优势十分明显：低成本、高速仿真、实时变量监视、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。

23.9 MPLAB ICD 2 在线调试器

Microchip 的在线调试器 MPLAB ICD 2 是一款功能强大而成本低廉的运行时开发工具，通过 RS-232 或高速 USB 接口与 PC 主机相连。该工具基于闪存 PIC MCU，可用于开发本系列及其他 PIC MCU 和 dsPIC DSC。MPLAB ICD 2 使用了闪存器件中内建的在线调试功能。该功能结合 Microchip 的在线串行编程 (In-Circuit Serial Programming™, ICSP™) 协议，可在 MPLAB 集成开发环境的图形用户界面上提供成本效益很高的在线闪存调试。这使设计人员可通过设置断点、单步运行以及对变量、CPU 状态以及外设寄存器进行监视的方法实现源代码的开发和调试。其全速运行特性可对硬件和应用进行实时测试。MPLAB ICD 2 还可用作某些 PIC 器件的开发编程器。

23.10 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款通用的、符合 CE 规范的器件编程器，其可编程电压设置在 VDDMIN 和 VDDMAX 之间时可靠性最高。它有一个用来显示菜单和错误信息的大 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、验证和编程。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，可对存储器很大的器件进行快速编程，它还采用 SD/MMC 卡用作文件存储及数据安全应用。

23.11 PICSTART Plus 开发编程器

PICSTART Plus 开发编程器是一款易于使用而成本低廉的原型编程器。它通过 COM (RS-232) 端口与 PC 相连。MPLAB 集成开发环境软件使得该编程器的使用简便、高效。PICSTART Plus 开发编程器支持采用 DIP 封装的大部分 PIC 器件，其引脚数最多可达 40 个。引脚数更多的器件，如 PIC16C92X 和 PIC17C76X，可通过连接一个转接插槽来获得支持。PICSTART Plus 开发编程器符合 CE 规范。

23.12 PICkit 2 开发编程器

PICkit™ 2 开发编程器是一个低成本编程器；对于某些选定闪存器件，它也是一个调试器，通过其易于使用的接口可对众多 Microchip 的低档、中档和 PIC18F 系列闪存单片机进行编程。PICkit 2 入门工具包中包含一个有实验布线区的开发板、十二堂系列课程、软件和 HI-TECH 的 PICC™ Lite C 编译器，有助于用户快速掌握 PIC® 单片机的使用。这一工具包为使用 Microchip 功能强大的中档闪存系列单片机进行编程、评估和应用开发，提供了所需的一切。

23.13 演示、开发和评估板

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，用于测试和修改。

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。

有关演示、开发和评估工具包的完整列表，请查阅 Microchip 公司网页 (www.microchip.com)。

24.0 电气特性

本节将对 dsPIC30F 系列器件的电气特性进行概括介绍。在本文档的后续版本中还将提供更多信息。

欲知有关 dsPIC30F 架构和内核的更多详细信息，请参见《dsPIC30F 系列参考手册》(DS70046E_CN)。

下面列出了 dsPIC30F 系列器件的绝对最大额定值。器件长时间工作在最大额定值条件下，其可靠性会受到影响。上述值仅为运行条件极大值，我们不建议器件在该规范规定的范围以外运行。

绝对最大额定值^(†)

环境温度.....	-40°C 至 +125°C
储存温度.....	-65°C 至 +150°C
除 VDD 和 MCLR 外的任一引脚相对于 Vss 的电压 (注 1).....	-0.3V 至 (VDD + 0.3V)
VDD 引脚相对于 Vss 的电压.....	-0.3V 至 +5.5V
MCLR 引脚相对于 Vss 的电压.....	0V 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
输出钳位电流, I _{OK} (V _O < 0 或 V _O > VDD).....	±20 mA
任一 I/O 引脚的最大输出灌电流.....	25mA
任一 I/O 引脚的最大输出拉电流.....	25 mA
所有端口的最大灌电流.....	200 mA
所有端口的最大拉电流 (注 2).....	200 mA

注 1: MCLR/VPP 引脚上尖峰电压低于 Vss，导致感应电流高于 80 mA 时，可能引起锁死。因此，当把一个“低”电平加到 MCLR/VPP 引脚上时，应串联一个 50-100Ω 的电阻，而不要将这个引脚直接接到 Vss。

2: 允许的最大电流由器件最大功耗决定，请参见表 24-6。

[†] 注意：如果器件工作条件超过上述“绝对最大额定值”，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，我们不建议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，其可靠性会受到影响。

dsPIC30F6010A/6015

24.1 直流特性

表 24-1: dsPIC30F6010A 的工作速度 (MIPS) 与电压

VDD 范围 (单位: V)	温度范围 (单位: °C)	最大 MIPS	
		dsPIC30F6010A-30I	dsPIC30F6010A-20E
4.5-5.5	-40 至 +85	30	—
4.5-5.5	-40 至 +125	—	20
3.0-3.6	-40 至 +85	20	—
3.0-3.6	-40 至 +125	—	15
2.5-3.0	-40 至 +85	10	—

表 24-2: dsPIC30F6015 的工作速度 (MIPS) 与电压

VDD 范围 (单位: V)	温度范围 (单位: °C)	最大 MIPS	
		dsPIC30F6015-30I	dsPIC30F6015-20E
4.5-5.5	-40 至 +85	30	—
4.5-5.5	-40 至 +125	—	20
3.0-3.6	-40 至 +85	20	—
3.0-3.6	-40 至 +125	—	15
2.5-3.0	-40 至 +85	10	—

表 24-3: 热工作条件

参数	符号	最小值	典型值	最大值	单位
dsPIC30F6010A-30I/dsPIC30F6015-30I					
工作结温范围	TJ	-40		+125	°C
工作环境温度范围	TA	-40		+85	°C
dsPIC30F6010A-20E/dsPIC30F6015-20E					
工作结温范围	TJ	-40		+150	°C
工作环境温度范围	TA	-40		+125	°C
功耗: 芯片的内部功耗: $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$ I/O 引脚的功耗: $P_{I/O} = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	PD			P _{INT} + P _{I/O}	W
允许的最大功耗	PD _{MAX}			(T _J - T _A)/θ _{JA}	W

表 24-4: 热封装特性

特性	符号	典型值	最大值	单位	注
封装热阻, 80 引脚 TQFP (14x14x1mm)	θ _{JA}	36		°C/W	1
封装热阻, 80 引脚 TQFP (12x12x1 mm)	θ _{JA}	39		°C/W	1
封装热阻, 64 引脚 TQFP (10x10x1 mm)	θ _{JA}	39		°C/W	1

注 1: 通过封装模拟获得结点与环境的热阻值 θ_{JA}。

表 24-5: 直流温度和电压规范

直流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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	电源电压	2.5	—	5.5	V	工业级温度
DC11	VDD	电源电压	3.0	—	5.5	V	扩展级温度
DC12	VDR	RAM 数据保持电压⁽³⁾	1.5	—	—	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 上升率 确保能够产生 内部上电复位信号	0.05	—	—	V/ms	在 0.1s 内可从 0V 上升至 5V 在 60 ms 内可从 0V 上升至 3V

- 注 1:** 除非另外声明, 否则“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, 在生产时未经测试。
注 3: 该电压是在不丢失 RAM 数据前提下的最小 VDD。

dsPIC30F6010A/6015

表 24-6: 直流特性: 工作电流 (IDD)

直流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 (1)	最大值	单位	条件
工作电流 (IDD) (2)				
DC31a	9.5	15	mA	25°C
DC31b	9.5	15	mA	85°C
DC31c	9.4	15	mA	125°C
DC31e	18	27	mA	25°C
DC31f	17	27	mA	85°C
DC31g	17	27	mA	125°C
DC30a	15	23	mA	25°C
DC30b	15	23	mA	85°C
DC30c	14	23	mA	125°C
DC30e	30	45	mA	25°C
DC30f	29	45	mA	85°C
DC30g	27	45	mA	125°C
DC23a	40	50	mA	25°C
DC23b	40	50	mA	85°C
DC23c	36	50	mA	125°C
DC23e	44	64	mA	25°C
DC23f	43	64	mA	85°C
DC23g	43	64	mA	125°C
DC24a	50	75	mA	25°C
DC24b	51	75	mA	85°C
DC24c	51	75	mA	125°C
DC24e	85	125	mA	25°C
DC24f	84	125	mA	85°C
DC24g	84	125	mA	125°C
DC27a	89	115	mA	25°C
DC27b	89	115	mA	85°C
DC27d	147	185	mA	25°C
DC27e	146	185	mA	85°C
DC27f	145	185	mA	125°C
DC29a	206	255	mA	25°C
DC29b	205	255	mA	85°C

- 注 1: 除非另外声明, 否则“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: 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 引脚均被配置为输入引脚, 并拉至 VDD。MCLR = VDD, WDT、FSCM、LVD 和 BOR 被禁止。CPU、SRAM、程序存储器 and 数据存储器均处于工作状态。外设模块均不工作。

表 24-7: 直流特性: 空闲电流 (IDLE)

直流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 ^(1,2)	最大值	单位	条件	
工作电流 (IDD)⁽³⁾					
DC51a	9.0	14	mA	25°C	3.3V 0.128 MIPS LPRC (512 kHz)
DC51b	9.0	14	mA	85°C	
DC51c	9.0	14	mA	125°C	
DC51e	17	26	mA	25°C	
DC51f	16	26	mA	85°C	
DC51g	16	26	mA	125°C	
DC50a	11	18	mA	25°C	3.3V (1.8 MIPS) FRC (7.37 MHz)
DC50b	12	18	mA	85°C	
DC50c	11	18	mA	125°C	
DC50e	25	38	mA	25°C	
DC50f	24	38	mA	85°C	
DC50g	23	38	mA	125°C	
DC43a	19	30	mA	25°C	3.3V 4 MIPS
DC43b	20	30	mA	85°C	
DC43c	20	30	mA	125°C	
DC43e	34	51	mA	25°C	
DC43f	33	51	mA	85°C	
DC43g	33	51	mA	125°C	
DC44a	34	53	mA	25°C	3.3V 10 MIPS
DC44b	35	53	mA	85°C	
DC44c	35	53	mA	125°C	
DC44e	59	89	mA	25°C	
DC44f	59	89	mA	85°C	
DC44g	59	89	mA	125°C	
DC47a	59	70	mA	25°C	3.3V 20 MIPS
DC47b	60	70	mA	85°C	
DC47d	99	115	mA	25°C	
DC47e	99	115	mA	85°C	
DC47f	100	115	mA	125°C	
DC49a	138	155	mA	25°C	
DC49b	139	155	mA	85°C	5V 30 MIPS

- 注 1:** 除非另外声明, 否则“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:** 基本 IDLE 电流是在内核停止工作但时钟有效并且所有模块均关闭时测得的。
- 注 3:** 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 引脚均被配置为输入引脚, 并拉至 Vss。MCLR = VDD, WDT、FSCM、LVD 和 BOR 被禁止。CPU、SRAM、程序存储器和数据存储器均处于工作状态。外设模块均不工作。

dsPIC30F6010A/6015

表 24-8: 直流特性: 掉电电流 (IPD)

直流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 (1)	最大值	单位	条件		
掉电电流 (IPD) (2)						
DC60a	0.2	—	μA	25°C	3.3V 基本掉电电流 (3)	
DC60b	1.2	40	μA	85°C		
DC60c	12	65	μA	125°C		
DC60e	0.4	—	μA	25°C		5V
DC60f	1.7	55	μA	85°C		
DC60g	15	90	μA	125°C		
DC61a	9	15	μA	25°C		3.3V 看门狗定时器电流: ΔI _{WDT} (3)
DC61b	9	15	μA	85°C		
DC61c	9	15	μA	125°C		
DC61e	18	30	μA	25°C	5V	
DC61f	17	30	μA	85°C		
DC61g	16	30	μA	125°C		
DC62a	4	10	μA	25°C		
DC62b	5	10	μA	85°C		
DC62c	4	10	μA	125°C		
DC62e	4	15	μA	25°C	5V	
DC62f	6	15	μA	85°C		
DC62g	5	15	μA	125°C		
DC63a	29	52	μA	25°C		3.3V BOR 工作: ΔI _{BOR} (3)
DC63b	32	52	μA	85°C		
DC63c	33	52	μA	125°C		
DC63e	34	60	μA	25°C	5V	
DC63f	39	60	μA	85°C		
DC63g	38	60	μA	125°C		

- 注 1: 除非另外声明, 否则“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: 基本 IPD 电流是在关闭所有外设和时钟的条件下测得的。所有 I/O 引脚都配置为输入且被拉至高电平。BOR、WDT 等外设也都被关闭。
- 注 3: Δ 电流是使能模块时额外消耗的电流。掉电时外设模块的电流消耗是这一电流与基本 IPD 电流之和。

表 24-9: 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 (除非另外声明) 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	
D110	V _{IL}	输入低电压⁽²⁾ I/O 引脚: 带有施密特触发器缓冲器	V _{SS}	—	0.2 V _{DD}	V		
D115		MCLR	V _{SS}	—	0.2 V _{DD}	V		
D116		OSC1 (在 XT、HS 和 LP 模式下)	V _{SS}	—	0.2 V _{DD}	V		
D117		OSC1 (在 RC 模式下) ⁽³⁾	V _{SS}	—	0.3 V _{DD}	V		
D118		SDA 和 SCL	V _{SS}	—	0.3 V _{DD}	V		禁止 SMBus
D119		SDA 和 SCL	V _{SS}	—	0.2 V _{DD}	V		使能 SMBus
D120	V _{IH}	输入高电压⁽²⁾ I/O 引脚: 带有施密特触发器缓冲器	0.8 V _{DD}	—	V _{DD}	V		
D125		MCLR	0.8 V _{DD}	—	V _{DD}	V		
D126		OSC1 (在 XT、HS 和 LP 模式下)	0.7 V _{DD}	—	V _{DD}	V		
D127		OSC1 (在 RC 模式下) ⁽³⁾	0.9 V _{DD}	—	V _{DD}	V		
D128		SDA 和 SCL	0.7 V _{DD}	—	V _{DD}	V		禁止 SMBus
D129		SDA 和 SCL	0.8 V _{DD}	—	V _{DD}	V		使能 SMBus
D130	I _{CNPU}	CNxx 上拉电流⁽²⁾	50	250	400	μA	V _{DD} = 5V, V _{PIN} = V _{SS}	
D150	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	
D151		模拟输入引脚	—	0.50	—	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态	
D155		MCLR	—	0.05	±5	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD}	
D156		OSC1	—	0.05	±5	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , XT、HS 和 LP 振荡 模式	

- 注 1:** 除非另外声明, 否则“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, 在生产时未经测试。
- 注 3:** 在 RC 振荡器配置中, OSC1/CLKI 引脚为一个施密特触发器输入。当处于 RC 模式时, 建议不要使用外部时钟驱动 dsPIC30F 器件。
- 注 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测得更高的泄漏电流。
- 注 5:** 负电流定义为引脚的拉电流。

dsPIC30F6010A/6015

表 24-10: 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 (除非另外声明) 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
DO16		OSC2/CLKO (RC 或 EC 振荡模式)	—	—	TBD	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
DO26		OSC2/CLKO (RC 或 EC 振荡器模式)	TBD	—	—	V	IoH = -2.0 mA, VDD = 3V
DO50	Cosc2	输出引脚上的容性负载规范 ⁽²⁾ OSC2/SOSC2 引脚	—	—	15	pF	当采用外部时钟驱动 OSC1 时, 处于 XTL、XT、HS 和 LP 模式。
DO56	Cio	所有 I/O 引脚和 OSC2	—	—	50	pF	(RC 或 EC 振荡模式)
DO58	Cb	SCL 和 SDA	—	—	400	pF	在 I ² C™ 模式下

图注: TBD = 待定

注 1: 除非另外声明, 否则“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, 在生产时未经测试。

图 24-1: 欠压复位特性

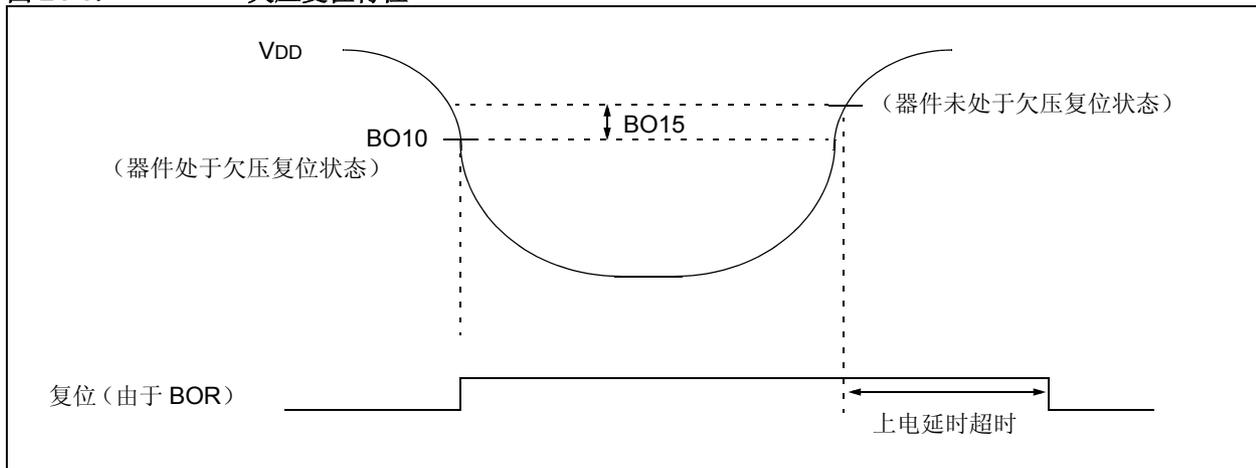


表 24-11: 电气特性 BOR

直流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 由高水平翻转为低电平时的 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: 除非另外声明, 否则“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: 参数仅为特征值, 在生产时未经测试。
 3: 11 值不在可用的工作范围内。

表 24-12: DC 特性: 程序和 EEPROM

直流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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
D121	VDRW	用于读 / 写的 VDD	V _{MIN}	—	5.5	V	
D122	TDEW	擦写周期时间	—	2	—	ms	假设没有违反其他规范 行擦除
D123	TRETD	特性保持时间	40	100	—	年	
D124	IDEW	编程时的 IDD	—	10	30	mA	
		编程闪存程序存储器⁽²⁾					
D130	EP	单元耐擦写能力	10K	100K	—	E/W	-40°C ≤ TA ≤ +85°C V _{MIN} = 最小工作电压
D131	VPR	用于读的 VDD	V _{MIN}	—	5.5	V	
D132	VEB	用于块擦除的 VDD	4.5	—	5.5	V	假设没有违反其他规范 行擦除 块擦除
D133	VPEW	用于擦 / 写的 VDD	3.0	—	5.5	V	
D134	TPEW	擦 / 写周期时间	—	2	—	ms	
D135	TRETD	特性保持时间	40	100	—	年	
D136	TEB	ICSP™ 块擦除时间	—	4	—	ms	
D137	IPEW	编程时的 IDD	—	10	30	mA	
D138	IEB	编程时的 IDD	—	10	30	mA	

- 注 1: 除非另外声明, 否则“典型值”栏中的数据均为 5V、25°C 下的值。
 2: 参数仅为特征值, 在生产时未经测试。

dsPIC30F6010A/6015

24.2 交流特性和时序参数

本节所包含的信息定义了 dsPIC30F 的交流特性和时序参数。

表 24-13: 温度和电压规范——交流

交流特性	标准工作条件: 2.5V 至 5.5V (除非另外声明)
	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} 范围如第 24.0 节所示。

图 24-2: 器件时序规范的负载条件

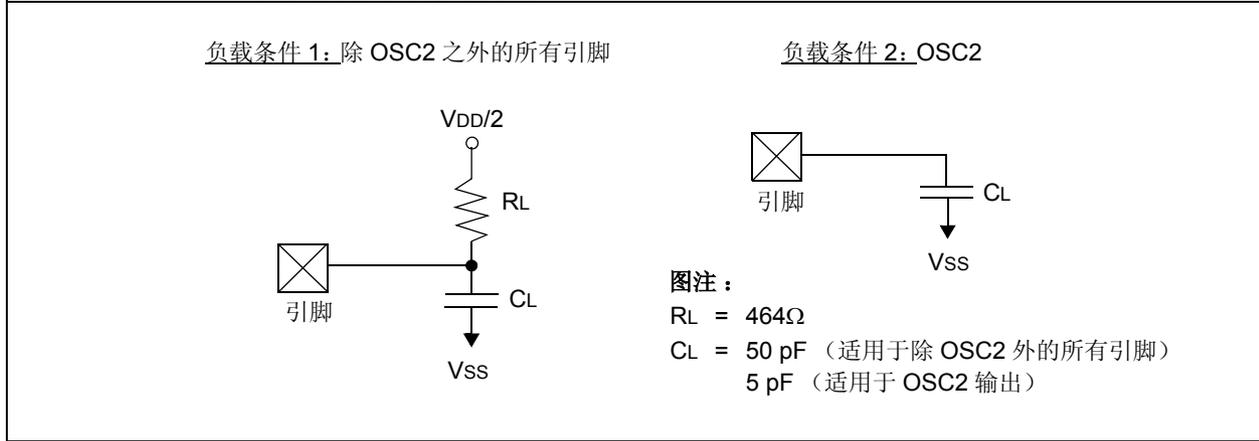


图 24-3: 外部时钟时序

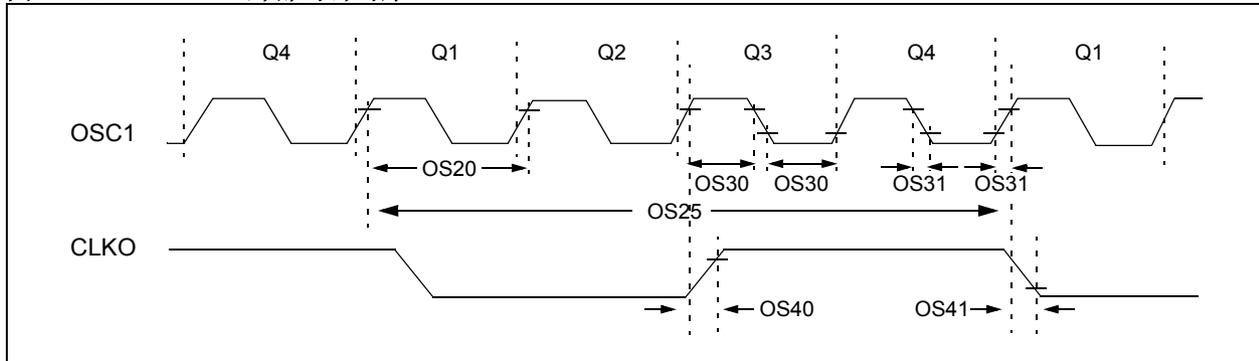


表 24-14: 外部时钟时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 CLKN 频率 ⁽²⁾ (仅在 EC 模式下允许使用外部时钟)	DC	—	40	MHz	EC
			4	—	10	MHz	带 4x PLL 的 EC
			4	—	10	MHz	带 8x PLL 的 EC
			4	—	7.5 ⁽³⁾	MHz	带 16x PLL 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 4x PLL 的 XT	
		4	—	10	MHz	带 8x PLL 的 XT	
		4	—	7.5 ⁽³⁾	MHz	带 16x PLL 的 XT	
		10	—	25	MHz	HS	
		10	—	20 ⁽⁴⁾	MHz	带 4x PLL 的 HS/2	
		10	—	20 ⁽⁴⁾	MHz	带 8x PLL 的 HS/2	
		10	—	15 ⁽³⁾	MHz	带 16x PLL 的 HS/2	
12 ⁽⁴⁾	—	25	MHz	带 4x PLL 的 HS/3			
12 ⁽⁴⁾	—	25	MHz	带 8x PLL 的 HS/3			
12 ⁽⁴⁾	—	22.5 ⁽³⁾	MHz	带 16x PLL 的 HS/3			
—	32.768	—	kHz	LP			
OS20	Tosc	Tosc = 1/Fosc	—	—	—	—	关于 Fosc 的值, 见参数 OS10
OS25	Tcy	指令周期时间 ⁽²⁾⁽⁵⁾	33	—	DC	ns	见表 24-16
OS30	TosL,	外部时钟输入 (OSC1) 高电平或低电平时间 ⁽²⁾	.45 x Tosc	—	—	ns	EC
	TosH						
OS31	TosR,	外部时钟输入 (OSC1) 上升或下降时间 ⁽²⁾	—	—	20	ns	EC
	TosF						
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: 除非另外声明, 否则“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: 参数仅为特征值, 在生产时未经测试。
- 3: 受 PLL 输出频率范围的限制。
- 4: 受 PLL 输入频率范围的限制。
- 5: 指令周期时间 (Tcy) 等于输入振荡器时钟周期的 4 倍。所有规定值均为基于标准工作条件下器件执行代码时对应特定振荡器类型的特征数据。超过规定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 引脚接入了外部时钟。当使用外部时钟输入时, 所有器件的“最大”周期时间极限为“DC”(没有时钟)。
- 6: 测量在 EC 或 ERC 模式下进行。在 OSC2 引脚上测量 CLKO 信号。CLKO 在 Q1-Q2 周期 (1/2 Tcy) 中为低电平, 在 Q3-Q4 周期 (1/2 Tcy) 中为高电平。

dsPIC30F6010A/6015

表 24-15: PLL 时钟时序规范 (VDD = 2.5V 至 5.5V)

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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
OS50	FPLLI	PLL 输入频率范围 (2)	4	—	10	MHz	带 4x PLL 的 EC
			4	—	10	MHz	带 8x PLL 的 EC
			4	—	7.5 ⁽⁴⁾	MHz	带 16x PLL 的 EC
			4	—	10	MHz	带 4x PLL 的 XT
			4	—	10	MHz	带 8x PLL 的 XT
			4	—	7.5 ⁽⁴⁾	MHz	带 16x PLL 的 XT
			5 ⁽³⁾	—	10	MHz	带 4x PLL 的 HS/2
			5 ⁽³⁾	—	10	MHz	带 8x PLL 的 HS/2
			5 ⁽³⁾	—	7.5 ⁽⁴⁾	MHz	带 16x PLL 的 HS/2
			4	—	8.33 ⁽³⁾	MHz	带 4x PLL 的 HS/3
			4	—	8.33 ⁽³⁾	MHz	带 8x PLL 的 HS/3
			4	—	7.5 ⁽⁴⁾	MHz	带 16x PLL 的 HS/3
OS51	Fsys	片上 PLL 输出 (2)	16	—	120	MHz	带有 PLL 的 EC、XT、HS/2 和 HS/3 模式
OS52	TLOC	PLL 起振时间 (锁定时间)	—	20	50	μs	

- 注 1: 参数仅为特征值, 在生产时未经测试。
 2: 除非另外声明, 否则“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。
 3: 受振荡器频率范围的限制。
 4: 受器件工作频率范围的限制。

表 24-16: PLL 抖动

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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: 参数仅为特征值, 在生产时未经测试。

表 24-17: 内部时钟时序示例

时钟 振荡器 模式	Fosc (MHz) (1)	Tcy (μs) (2)	MIPS ⁽³⁾ 无 PLL	MIPS ⁽³⁾ PLL x4	MIPS ⁽³⁾ PLL x8	MIPS ⁽³⁾ 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: 振荡器后分频器分频比为 1。

2: 指令执行时间: $T_{CY} = 1/MIPS$ 。

3: 指令执行频率: $MIPS = (F_{osc} * PLLx)/4$ (因为每个指令周期有 4 个 Q 时钟)

dsPIC30F6010A/6015

表 24-18: 交流特性: 内部 FRC 抗抖动性、准确性和漂移 (2)

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 抗抖动性 @FRC 频率 = 7.37 MHz ⁽¹⁾							
OS62	FRC	—	±0.04	±0.16	%	-40°C ≤ TA ≤ +85°C	
		—	±0.07	±0.23	%	-40°C ≤ TA ≤ +125°C	
	带 4x PLL 的 FRC	—	±0.31	±0.62	%	-40°C ≤ TA ≤ +85°C	
		—	±0.34	±0.77	%	-40°C ≤ TA ≤ +125°C	
	带 8x PLL 的 FRC	—	±0.44	±0.87	%	-40°C ≤ TA ≤ +85°C	
		—	±0.48	±1.08	%	-40°C ≤ TA ≤ +125°C	
带 16x PLL 的 FRC	—	±0.71	±1.23	%	-40°C ≤ TA ≤ +125°C		
内部 FRC 精度 @FRC 频率 = 7.37 MHz ⁽¹⁾							
OS63	FRC	—	—	±0.4	%	-40°C ≤ TA ≤ +125°C	
内部 FRC 漂移 @FRC 频率 = 7.37 MHz ⁽¹⁾							
OS64		-0.7	—	0.5	%	-40°C ≤ TA ≤ +85°C	
		-0.7	—	0.7	%	-40°C ≤ TA ≤ +125°C	
		-0.7	—	0.5	%	-40°C ≤ TA ≤ +85°C	
		-0.7	—	0.7	%	-40°C ≤ TA ≤ +125°C	

- 注 1: 频率校准的条件为 25°C 和 5V。TUN 位可用于补偿温度漂移。
 注 2: FRC 总的偏差可通过将抖动、精度和漂移百分比的绝对值相加来计算。

表 24-19: 内部 LPRC 精度

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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⁾							
OS65		-35	—	+35	%	—	

- 注 1: LPRC 频率随 VDD 的改变而改变。

图 24-4: CLKOUT 和 I/O 时序特性

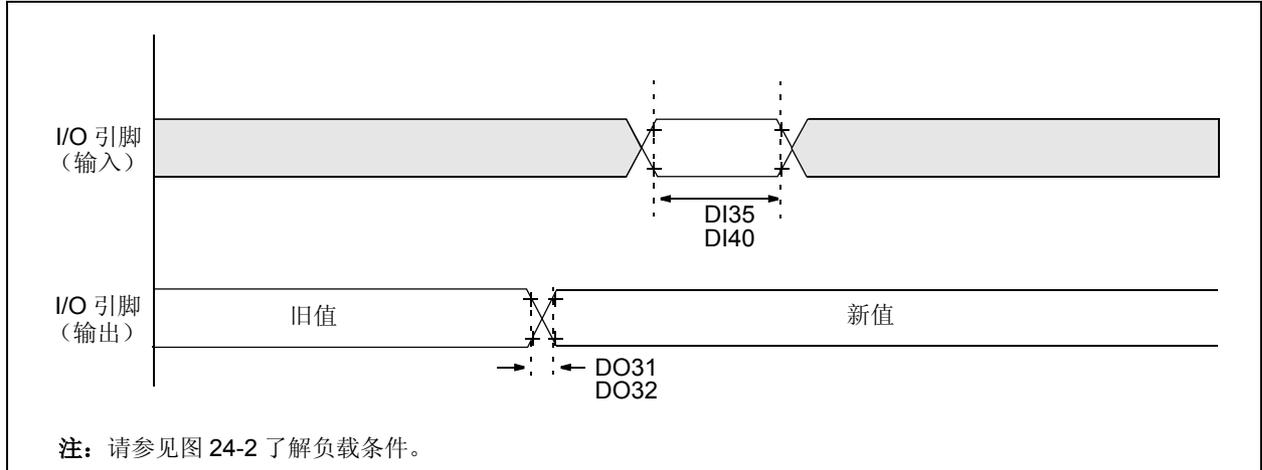


表 24-20: CLKOUT 和 I/O 时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)					
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 高电平或低电平时间 (输入)	2 Tcy	—	—	—	—

- 注
- 1: 这些参数是与任何内部时钟边沿无关的异步事件。
 - 2: 测量在 RC 模式和 EC 模式下进行, 其中 CLK0 输出为 $4 \times T_{osc}$ 。
 - 3: 参数仅为特征值, 在生产时未经测试。
 - 4: 除非另外声明, 否则“典型值”栏中的数据均为 5V、25°C 下的值。

dsPIC30F6010A/6015

图 24-5: 复位、看门狗定时器、振荡器起振定时器和上电延时定时器的时序特性

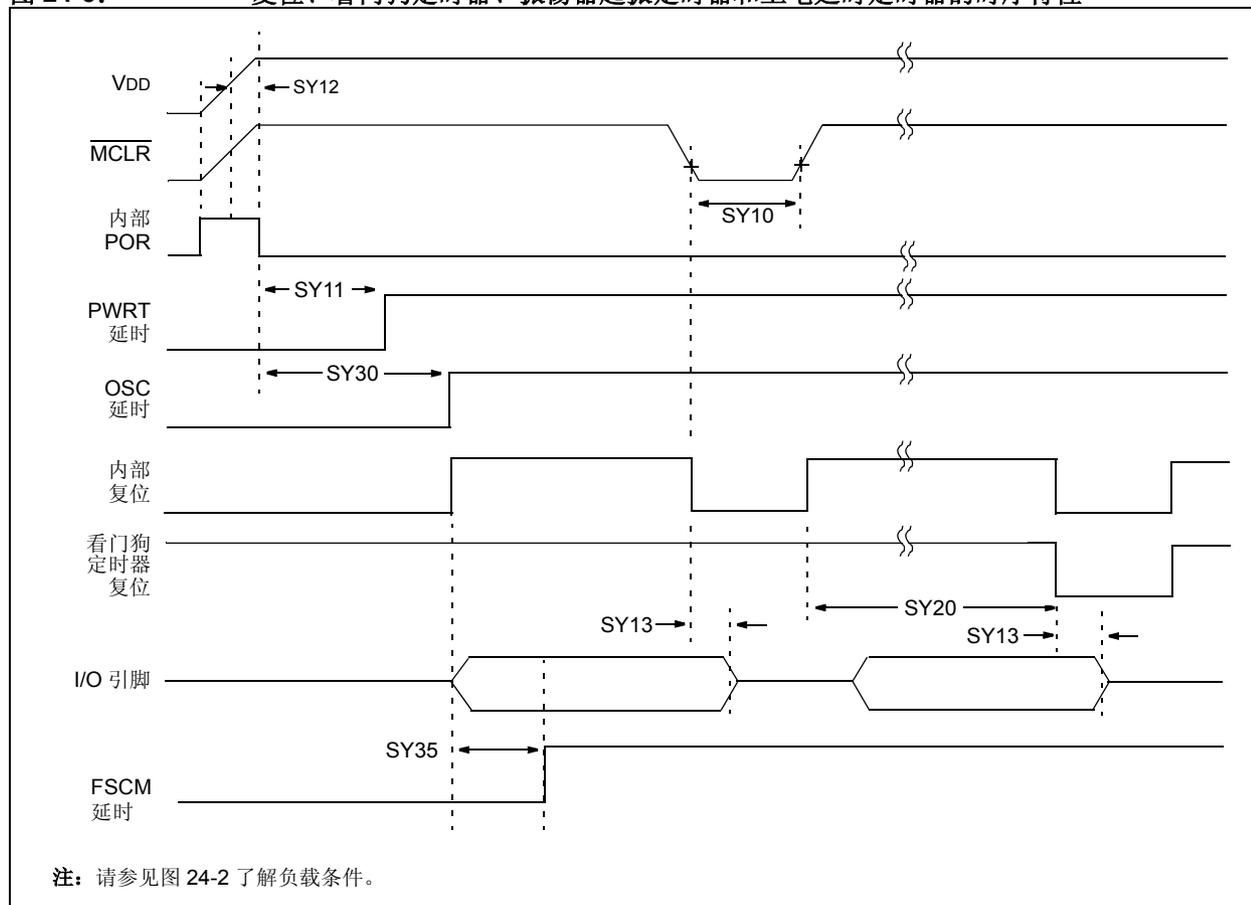


表 24-21: 复位、看门狗定时器、振荡器起振定时器、上电延时定时器和欠压复位时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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	3 12 50	4 16 64	6 22 90	ms	-40°C 至 +85°C 可由用户编程
SY12	TPOR	上电复位延时 (4)	3	10	30	μs	-40°C 至 +85°C
SY13	TIOZ	自 MCLR 低电平或看门狗定时器复位起 I/O 处于高阻态的时间	—	0.8	1.0	μs	
SY20	TWDT1	看门狗定时器超时周期 (无预分频器)	1.4	2.1	2.8	ms	VDD = 5V, -40°C 至 +85°C
	TWDT2		1.4	2.1	2.8	ms	VDD = 3V, -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: 参数仅为特征值, 在生产时未经测试。
 2: 除非另外声明, 否则“典型值”栏中的数据均为 5V、25°C 下的值。
 3: 关于 BOR, 请参见图 24-1 和表。
 4: 特征值, 仅供设计参考, 未经测试。

图 24-6: 带隙启动时间特性

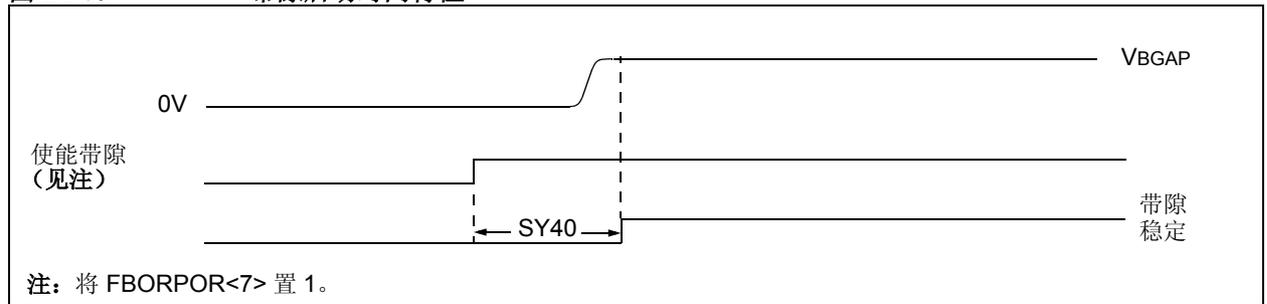


表 24-22: 带隙启动时间要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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
SY40	TBGAP	带隙启动时间	—	40	65	μs	定义为自带隙使能至带隙参考电压稳定 (由 RCON<13> 状态位指示) 的时间段。

- 注 1: 参数仅为特征值, 在生产时未经测试。

dsPIC30F6010A/6015

图 24-7: TIMER1、2、3、4 和 5 外部时钟时序特性

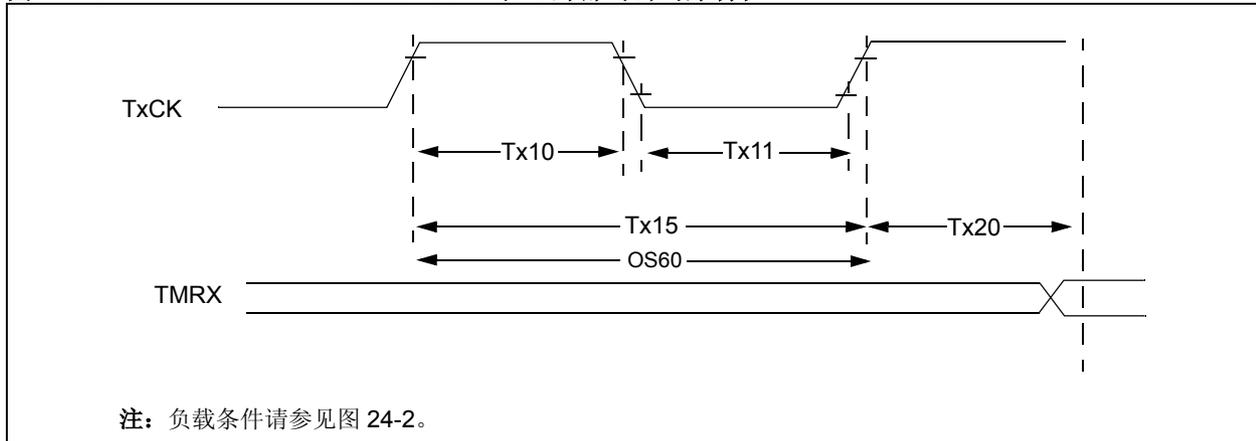


表 24-23: TIMER1 外部时钟时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)						
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TA10	TtxH	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, 有预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA11	TtxL	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, 有预分频器	10	—	—	ns	
			异步	10	—	—	ns	
TA15	TtxP	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, 有预分频器	取 20 ns 或 $(T_{CY} + 40)/N$ 中的较大值	—	—	—	
			异步	20	—	—	ns	
OS60	Ft1	SOSC1/T1CK 振荡器输入频率范围 (通过将 TCS 位 (T1CON 的 bit 1) 置 1 来使能振荡器)		DC	—	50	kHz	
TA20	TckEXTMRL	从出现外部 TxCK 时钟边沿到定时器递增的延时		$0.5 T_{CY}$	—	$1.5 T_{CY}$	—	

注: Timer1 属于 A 类定时器。

表 24-24: 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 (除非另外声明) 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, 有预分频器	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, 有预分频器	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, 有预分频器	取 20 ns 或 (Tcy + 40)/N 中的较大值	—	—		
TB20	TCKEXTMRL	从出现外部 TxCK 时钟边沿到定时器递增的延时		0.5 Tcy	—	1.5 Tcy	—	

表 24-25: 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 (除非另外声明) 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, 有预分频器	取 20 ns 或 (Tcy + 40)/N 中的较大值	—	—		
TC20	TCKEXTMRL	从出现外部 TxCK 时钟边沿到定时器递增的延时		0.5 Tcy	—	1.5 Tcy	—	

dsPIC30F6010A/6015

图 24-8: TIMERQ (QEI 模块) 外部时钟时序特性

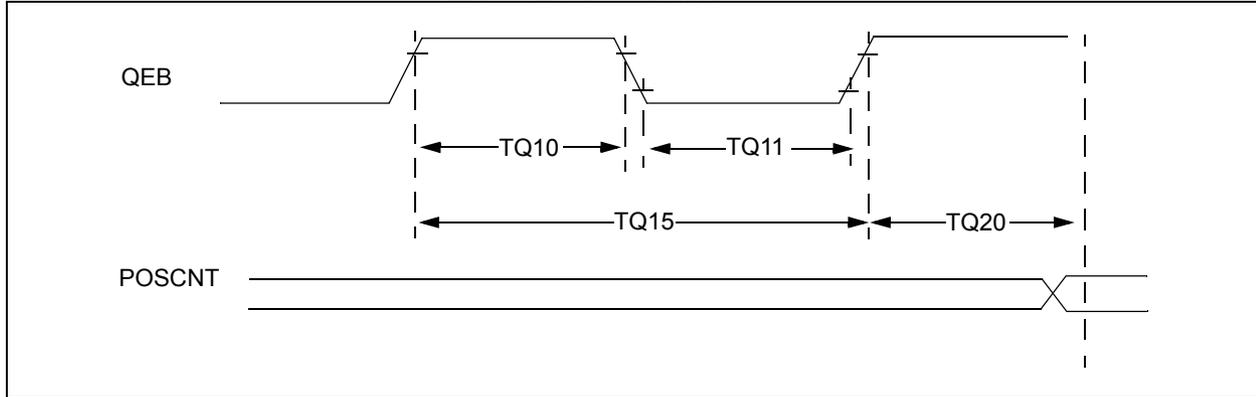


表 24-26: QEI 模块外部时钟时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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
TQ10	TtQH	TQCK 高电平时间	同步, 有预分频器	Tcy + 20		—	ns	还必须满足参数 TQ15
TQ11	TtQL	TQCK 低电平时间	同步, 有预分频器	Tcy + 20		—	ns	还必须满足参数 TQ15
TQ15	TtQP	TQCP 输入周期	同步, 有预分频器	2 * Tcy + 40		—	ns	—
TQ20	TCKEXTMRL	从出现外部 TxCK 时钟边沿到定时器递增的延时		0.5 Tcy		1.5 Tcy	—	—

注 1: 参数仅为特征值, 在生产时未经测试。

图 24-9: 输入捕捉 (CAPx) 时序特性

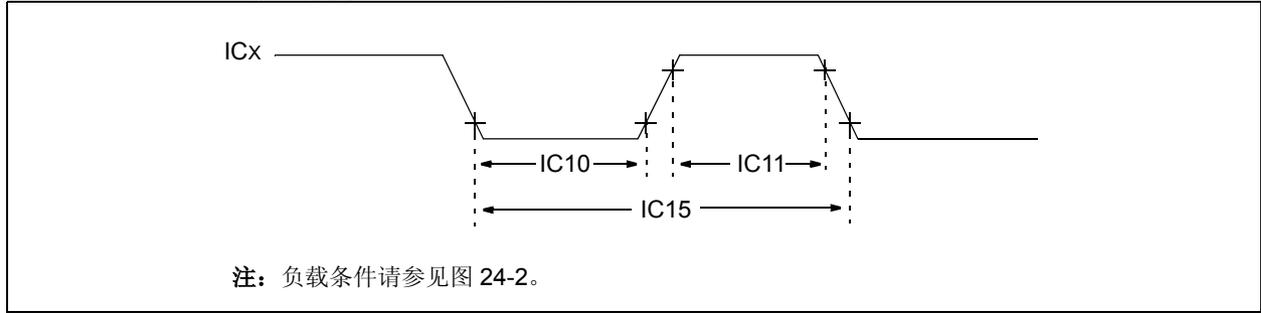


表 24-27: 输入捕捉时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)				
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	$0.5 T_{CY} + 20$	—	ns
			有预分频器	10	—	ns
IC11	TccH	ICx 输入高电平时间	无预分频器	$0.5 T_{CY} + 20$	—	ns
			有预分频器	10	—	ns
IC15	TccP	ICx 输入周期	$(2 T_{CY} + 40)/N$	—	ns	N = 预分频值 (1, 4, 16)

注 1: 参数仅为特征值, 在生产时未经测试。

图 24-10: 输出比较 (OCx) 时序特性

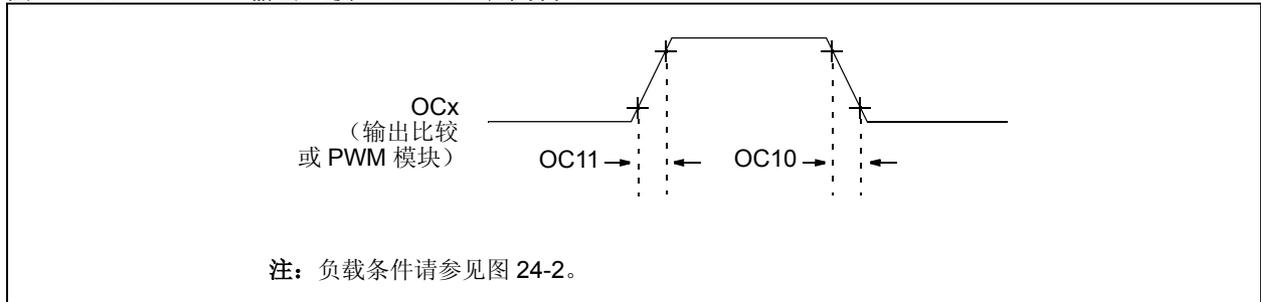


表 24-28: 输出比较模块时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)					
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	见参数 DO32
OC11	TccR	OCx 输出上升时间	—	—	—	ns	见参数 DO31

注 1: 参数仅为特征值, 在生产时未经测试。

注 2: 除非另外声明, 否则“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。

dsPIC30F6010A/6015

图 24-11: OC/PWM 模块时序特性

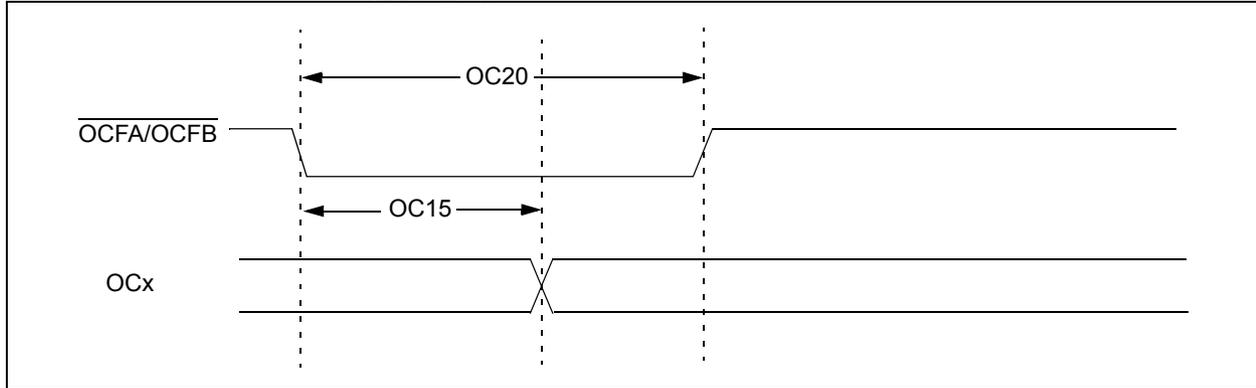


表 24-29: 简单 OC/PWM 模式时序要求

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 改变状态的时间	—	—	50	ns	—
OC20	TFLT	故障输入脉冲宽度	50	—	—	ns	—

注 1: 参数仅为特征值, 在生产时未经测试。

注 2: 除非另外声明, 否则“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。

图 24-12: 电机控制 PWM 模块故障时序特性

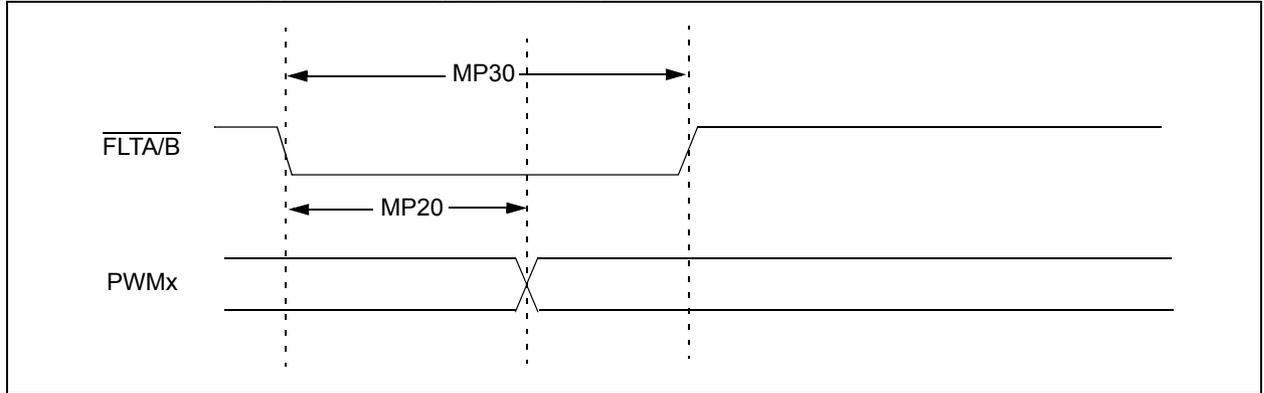


图 24-13: 电机控制 PWM 模块时序特性

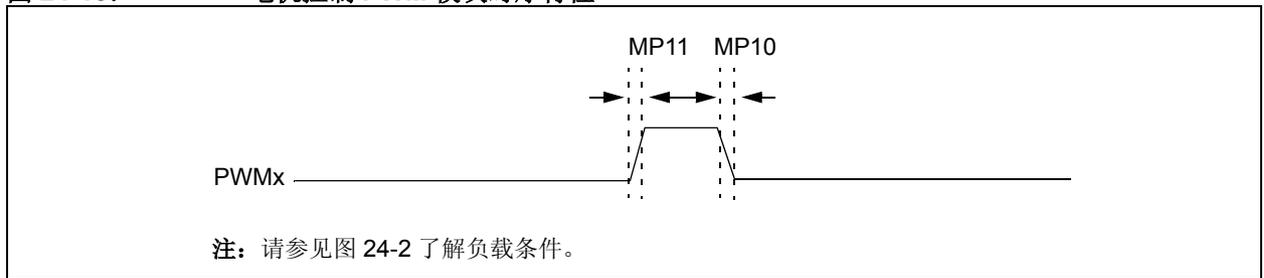


表 24-30: 电机控制 PWM 模块时序要求

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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
MP10	TFPWM	PWM 输出下降时间	—	—	—	ns	见参数 DO32
MP11	TRPWM	PWM 输出上升时间	—	—	—	ns	见参数 DO31
MP20	TFD	故障输入下降到 PWM I/O 变化的时间	—	—	50	ns	—
MP30	TFH	最小脉冲宽度	50	—	—	ns	—

注 1: 参数仅为特征值, 在生产时未经测试。

注 2: 除非另外声明, 否则“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。

dsPIC30F6010A/6015

图 24-14: QEA/QEB 输入特性

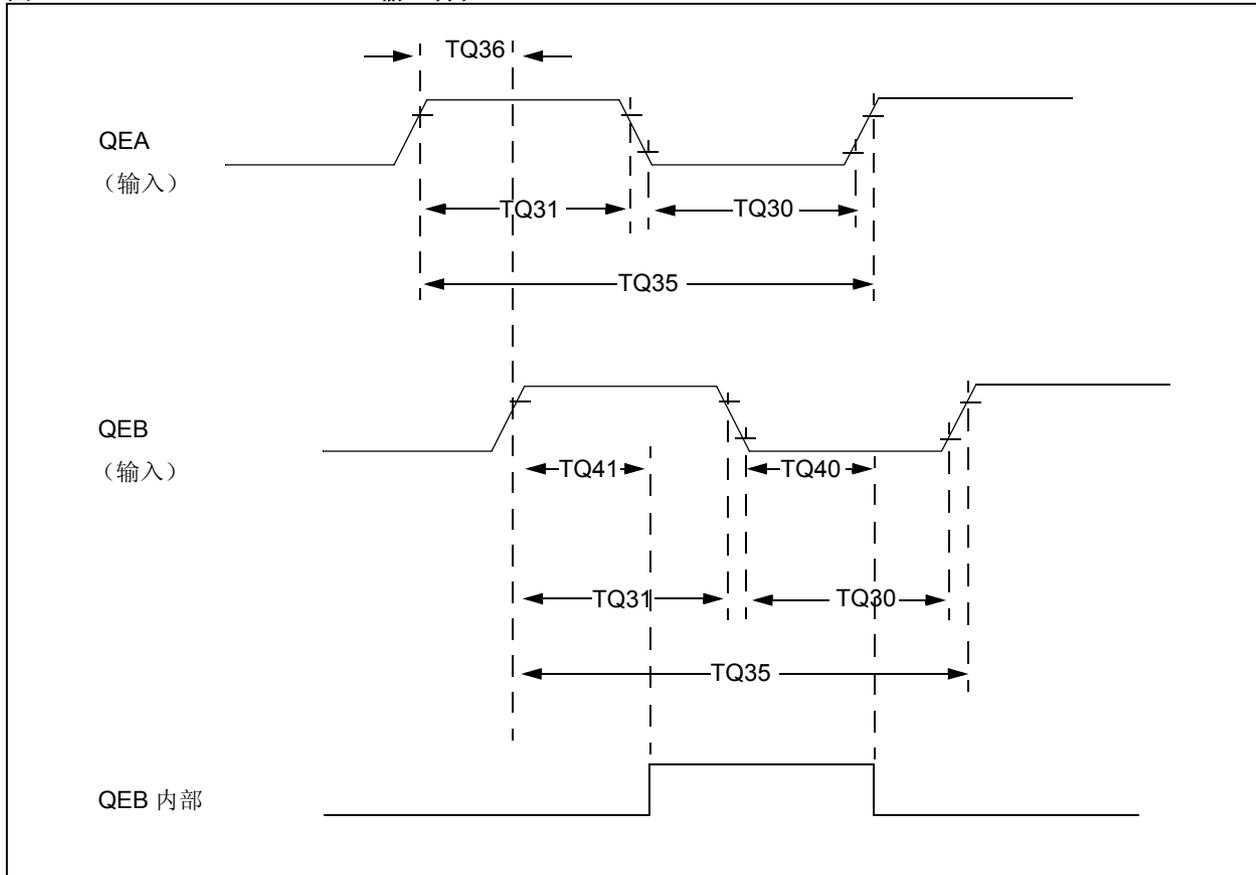


表 24-31: 正交编码器时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)				
参数编号	符号	特性 (1)	典型值 (2)	最大值	单位	条件
TQ30	TquL	正交输入低电平时间	6 Tcy	—	ns	—
TQ31	TquH	正交输入高电平时间	6 Tcy	—	ns	—
TQ35	TquIN	正交输入周期	12 Tcy	—	ns	—
TQ36	TquP	正交相周期	3 Tcy	—	ns	—
TQ40	TquFL	数字滤波器确认低电平的时间	$3 * N * Tcy$	—	ns	N = 1, 2, 4, 16, 32, 64, 128 和 256 (注 2)
TQ41	TquFH	数字滤波器确认高电平的时间	$3 * N * Tcy$	—	ns	N = 1, 2, 4, 16, 32, 64, 128 和 256 (注 2)

注 1: 参数仅为特征值, 在生产时未经测试。

注 2: N = 索引通道数字滤波器时钟分频选择位。请参见《dsPIC30F 系列参考手册》(DS70046E_CN) 中的“正交编码器接口 (QEI)”一章。

图 24-15: QEI 模块索引脉冲时序特性

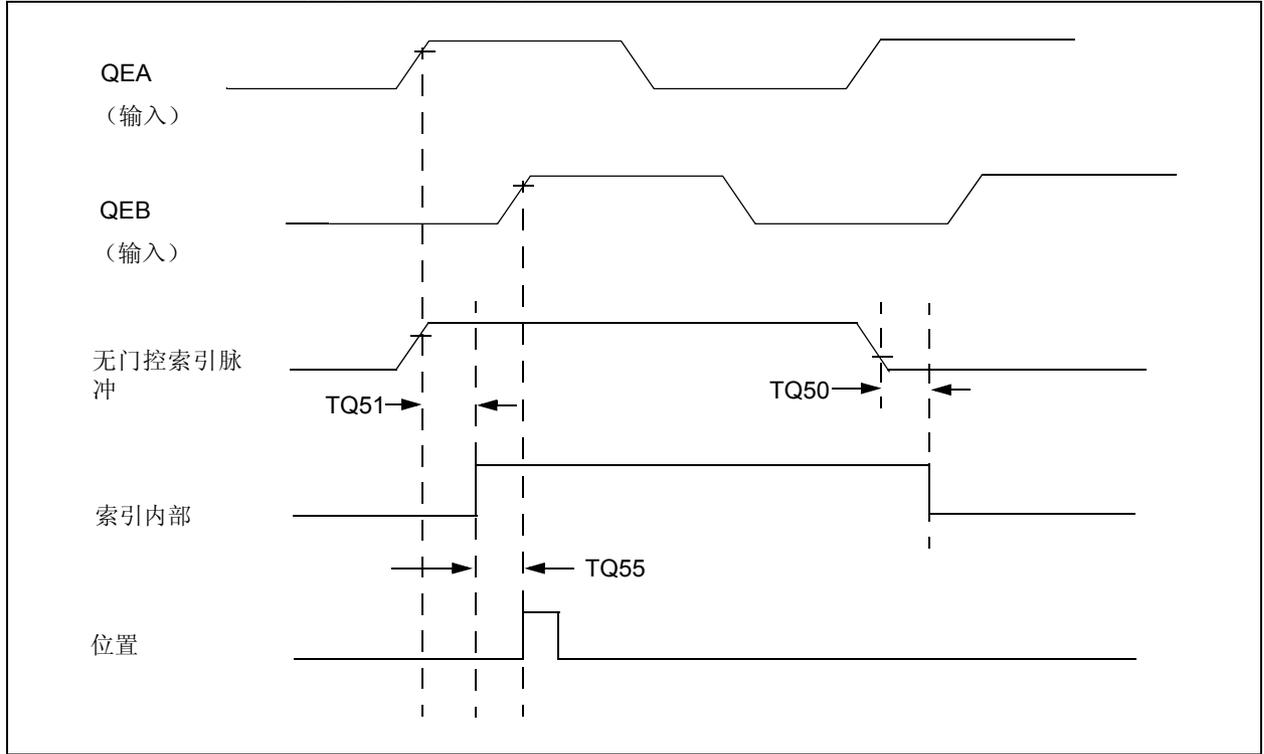


表 24-32: QEI 索引脉冲时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)				
参数编号	符号	特性 (1)	最小值	最大值	单位	条件
TQ50	TqiL	数字滤波器确认低电平的时间	$3 * N * T_{CY}$	—	ns	N = 1, 2, 4, 16, 32, 64, 128 和 256 (注 2)
TQ51	TqiH	数字滤波器确认高电平的时间	$3 * N * T_{CY}$	—	ns	N = 1, 2, 4, 16, 32, 64, 128 和 256 (注 2)
TQ55	Tqidxr	索引脉冲确认至位置计数器复位) (无门控索引)	$3 T_{CY}$	—	ns	—

注 1: 参数仅为特征值, 在生产时未经测试。

注 2: 索引脉冲与 QEA 和 QEB 对齐仅适用于位置计数器复位时序和正向旋转方式 (QEA 超前 QEB)。同样的时序适用于反向旋转的情形 (QEA 滞后于 QEB), 但索引脉冲确认出现在下降沿。

dsPIC30F6010A/6015

图 24-16: SPI 模块主模式 (CKE = 0) 时序特性

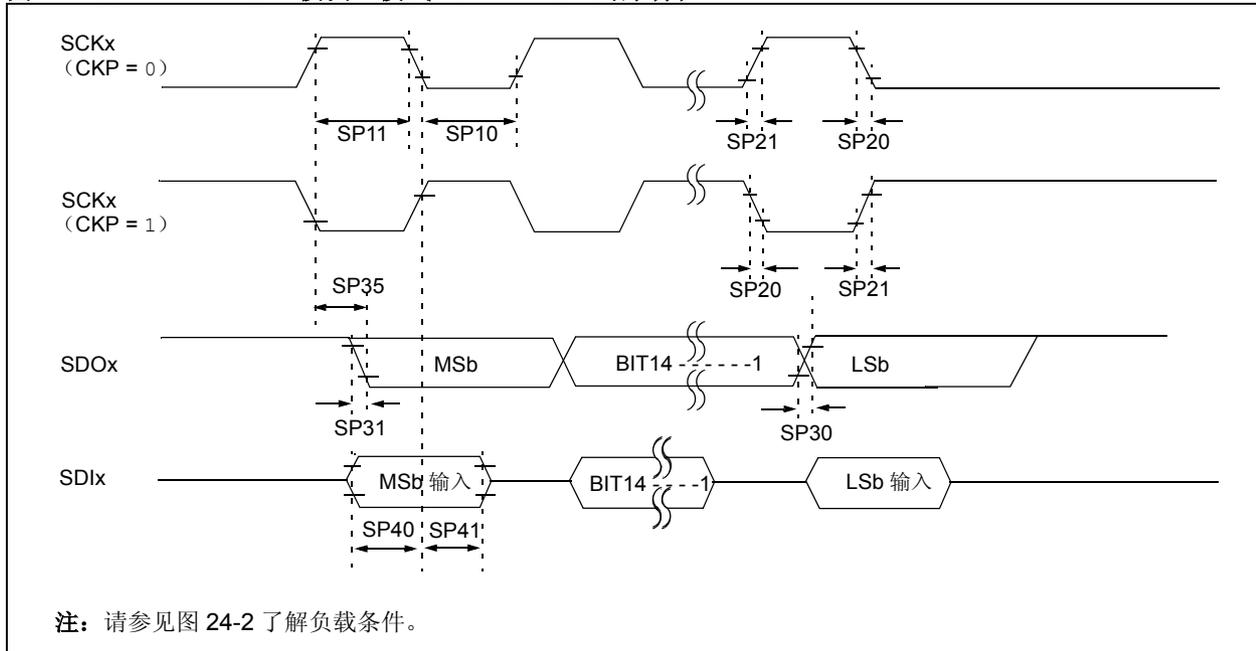


表 24-33: SPI 主模式 (CKE = 0) 时序特性

交流特性				标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 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	—

- 注
- 1: 参数仅为特征值, 在生产时未经测试。
 - 2: 除非另外声明, 否则“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。
 - 3: SCK 信号的最小时钟周期为 100 ns。因此, 在主模式下产生的时钟不能违反这一规范。
 - 4: 假设所有 SPI 引脚的负载均为 50 pF。

图 24-17: SPI 模块主模式 (CKE = 1) 时序特性

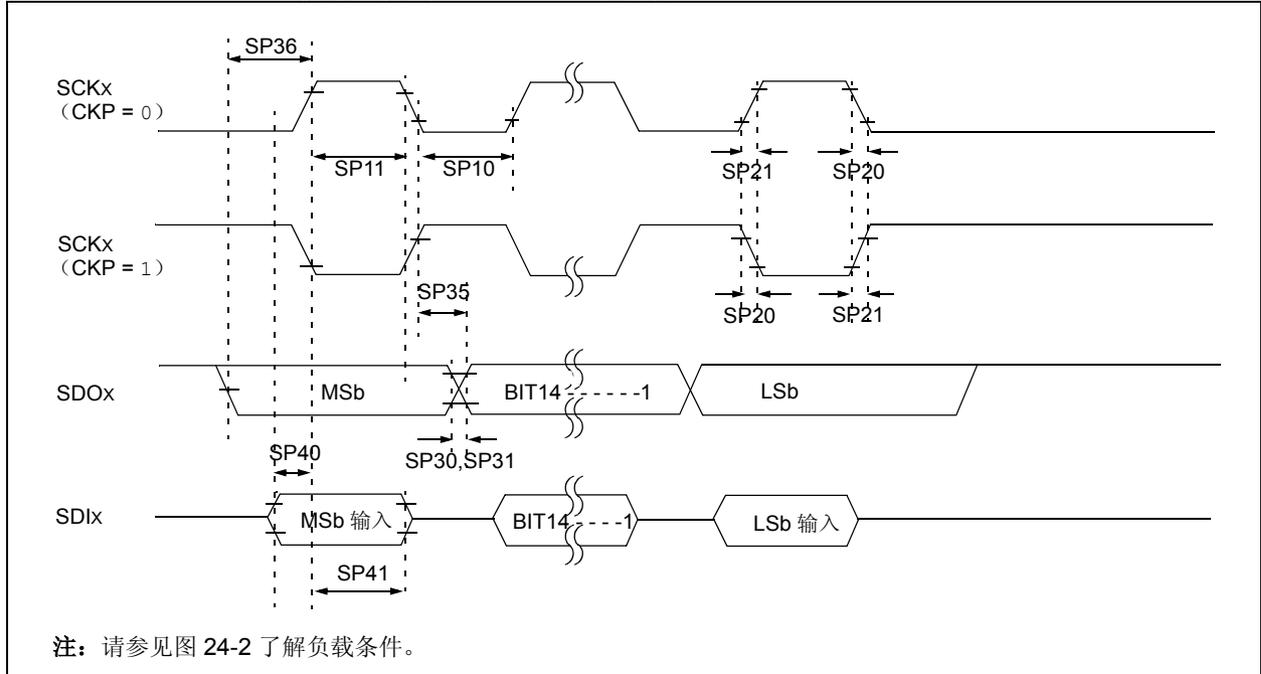


表 24-34: SPI 模块主模式 (CKE = 1) 时序特性

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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 边沿后 SDOx 数据输出有效时间	—	—	—	ns	—
SP36	TdoV2sc, TdoV2scL	SDOx 数据输出建立到第一个 SCKx 边沿的时间	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	—

注 1: 参数仅为特征值, 在生产时未经测试。

2: 除非另外声明, 否则“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。

3: SCK 的最小时钟周期为 100 ns。因此, 在主模式下产生的时钟不能违反这一规范。

4: 假设所有 SPI 引脚的负载均为 50 pF。

dsPIC30F6010A/6015

图 24-18: SPI 模块从模式 (CKE = 0) 时序特性

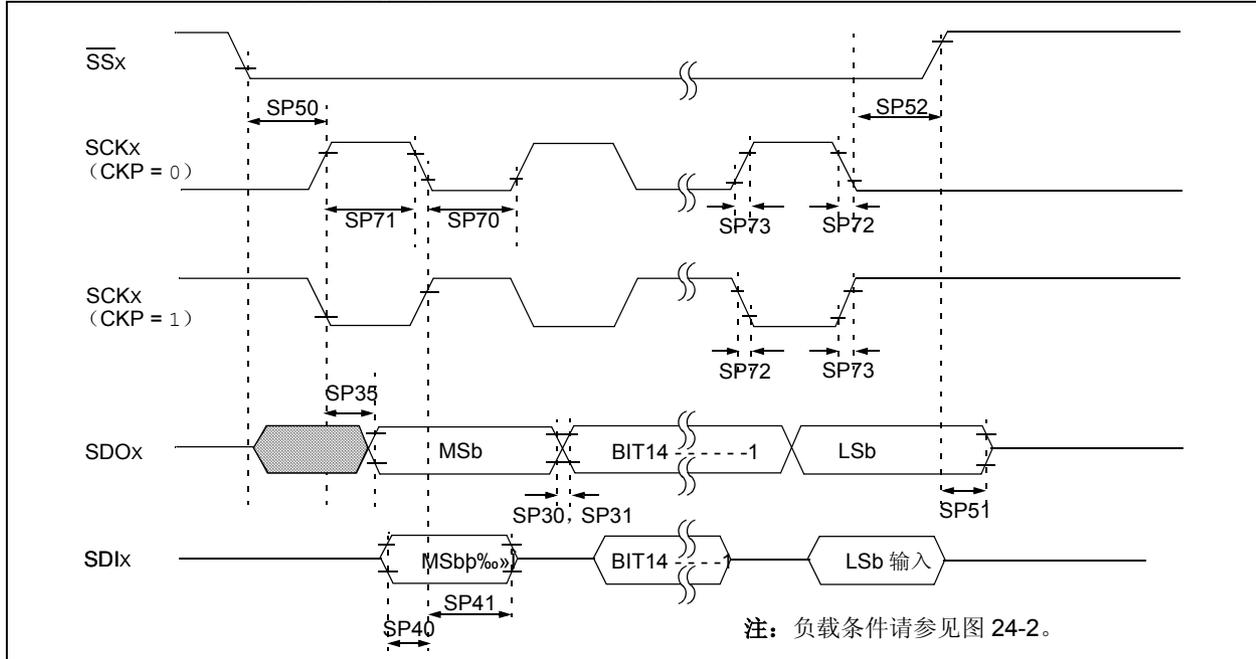


表 24-35: SPI 模块从模式 (CKE = 0) 时序要求

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	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	Tsch2doV, 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↓ 至 SCKx↑ 或 SCKx↓ 输入 的时间	120	—	—	ns	—
SP51	TssH2doZ	SSx↑ 至 SDOx 输出 高阻态的时间 (3)	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	SCK 边沿后的 SSx	1.5 Tcy +40	—	—	ns	—

- 注 1: 参数仅为特征值, 在生产时未经测试。
 2: 除非另外声明, 否则“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。
 3: 假设所有 SPI 引脚的负载均为 50 pF。

图 24-19: SPI 模块从模式 (CKE = 1) 时序特性

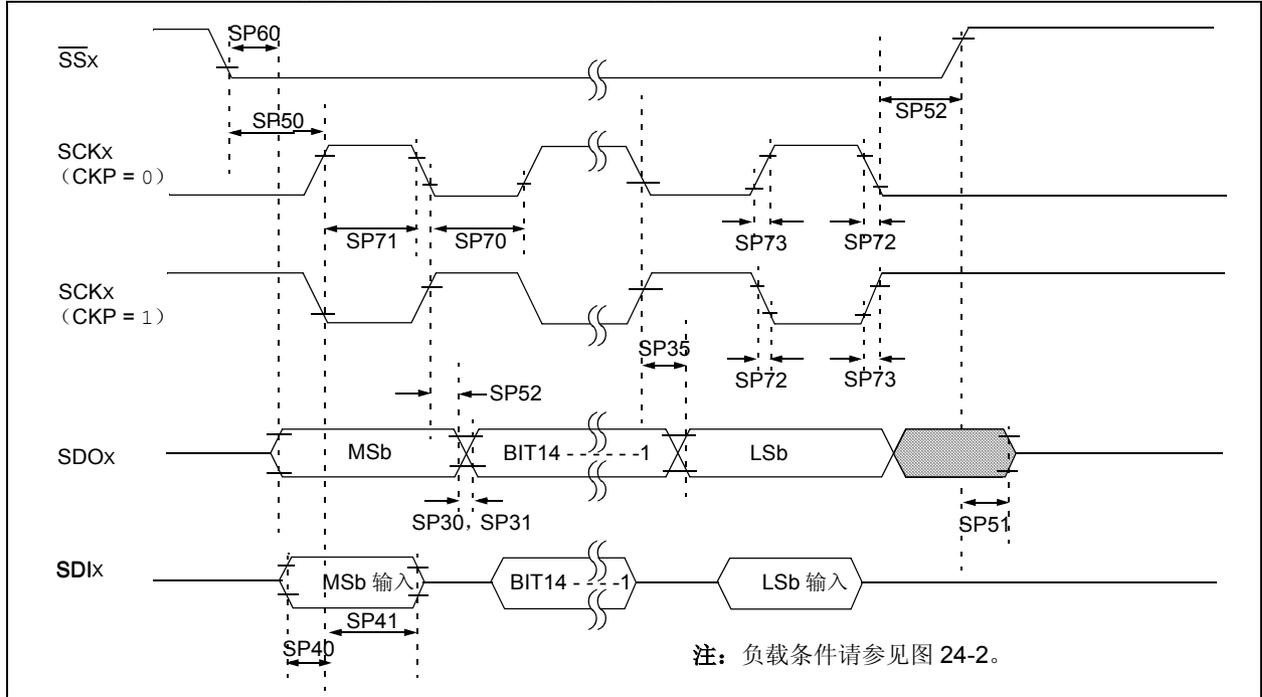


表 24-36: SPI 模块从模式 (CKE = 1) 时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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)					
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)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	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	Tsch2doV, TscL2doV	SCKx 边沿后 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	—
SP50	Tssl2scH, Tssl2scL	$\overline{\text{SS}}_x \downarrow$ 至 SCKx \downarrow 或 SCKx \uparrow 输入的时间	120	—	—	ns	—

- 注
- 1: 参数仅为特征值, 在生产时未经测试。
 - 2: 除非另外声明, 否则“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。
 - 3: SCK 信号的最小时钟周期为 100 ns。因此, 在主模式下产生的时钟不能违反这一规范。
 - 4: 假设所有 SPI 引脚的负载均为 50 pF。

dsPIC30F6010A/6015

表 24-36: SPI 模块从模式 (CKE = 1) 时序要求 (续)

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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
SP51	TssH2doZ	SS \uparrow 至 SDOx 输出高阻态的时间 (4)	10	—	50	ns	—
SP52	Tsch2ssH TscL2ssH	SCKx 边沿后 SSx \uparrow 的时间。	1.5 Tcy + 40	—	—	ns	—
SP60	Tssl2doV	SSx 边沿后 SDOx 数据输出有效的时间	—	—	50	ns	—

- 注
- 1: 参数仅为特征值, 在生产时未经测试。
 - 2: 除非另外声明, 否则“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。
 - 3: SCK 信号的最小时钟周期为 100 ns。因此, 在主模式下产生的时钟不能违反这一规范。
 - 4: 假设所有 SPI 引脚的负载均为 50 pF。

图 24-20: I²C™ 总线启动/停止位时序特性 (主模式)

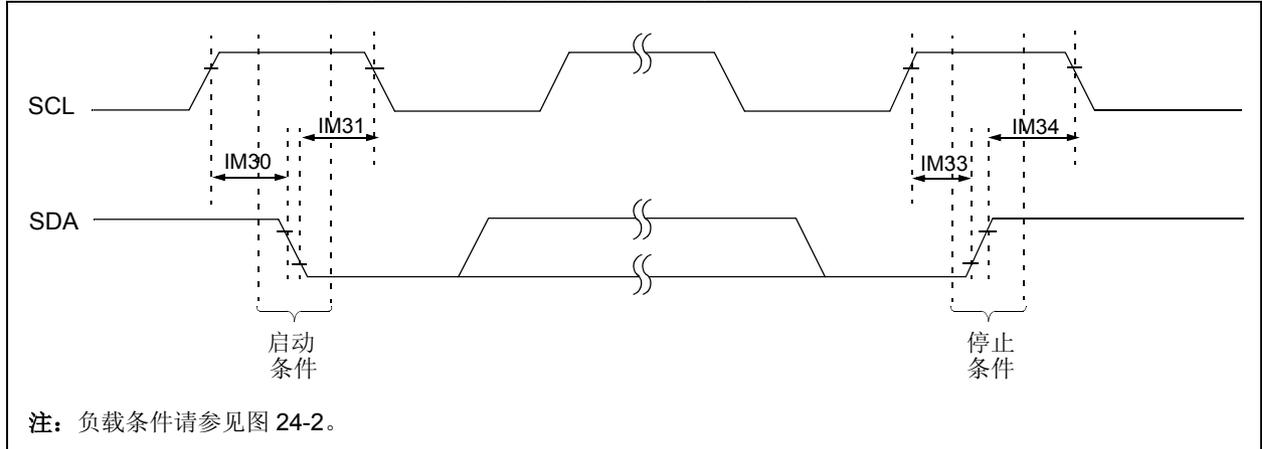
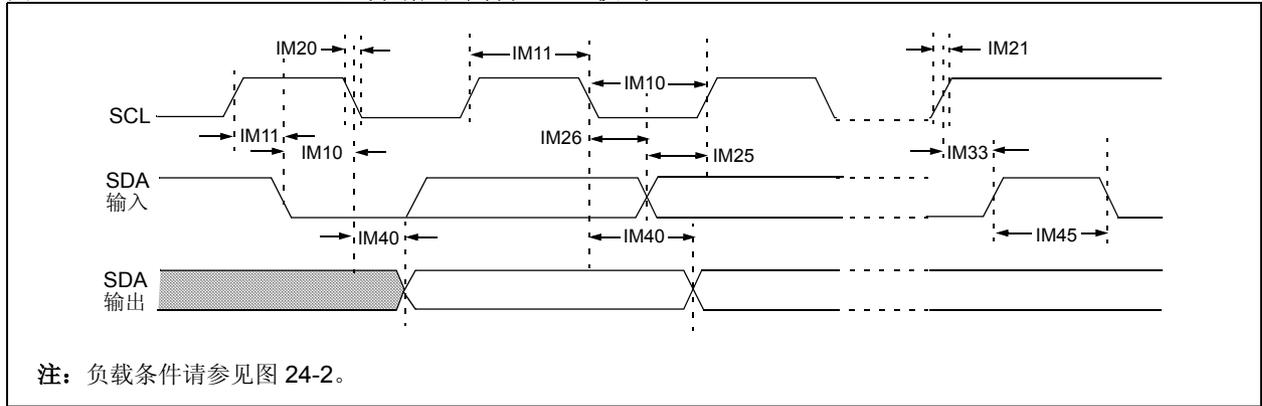


图 24-21: I²C™ 总线数据时序特性 (主模式)



dsPIC30F6010A/6015

表 24-37: I²C™ 总线数据时序要求 (主模式)

交流特性			标准工作条件: 2.5V 至 5.5V (除非另外声明) 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	
IM10	TLO:SCL	时钟低电平时间	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	—
IM11	THI:SCL	时钟高电平时间	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	—
IM20	TF:SCL	SDA 和 SCL 下降时间	100 kHz 模式	—	300	ns	规定 C _B 的值介于 10 至 400 pF
			400 kHz 模式	20 + 0.1 C _B	300	ns	
			1 MHz 模式 ⁽²⁾	—	100	ns	
IM21	TR:SCL	SDA 和 SCL 上升时间	100 kHz 模式	—	1000	ns	规定 C _B 的值介于 10 至 400 pF
			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⁾	TBD	—	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⁾	TBD	—	ns	
IM30	TSU:STA	启动条件建立时间	100 kHz 模式	T _{CY} /2 (BRG + 1)	—	μs	仅与重复启动条件相关
			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	此周期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	自时钟边沿到输出有效的时间	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	在开始新的发送之前总线必须处于空闲的时间
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 ⁽²⁾	TBD	—	μs	
IM50	C _B	总线的容性负载	—	400	pF	—	

图注: TBD = 待定

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 “I²C™” 一节。

2: 所有 I²C 引脚的最大电容值 = 10 pF (仅适用于频率为 1 MHz 的模式)。

图 24-22: I²C™ 总线启动 / 停止位时序特性 (从模式)

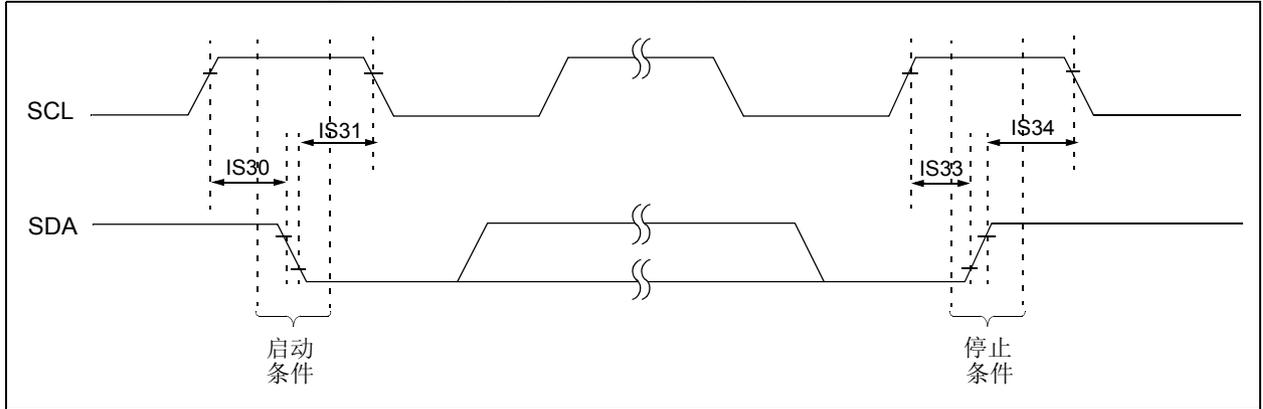
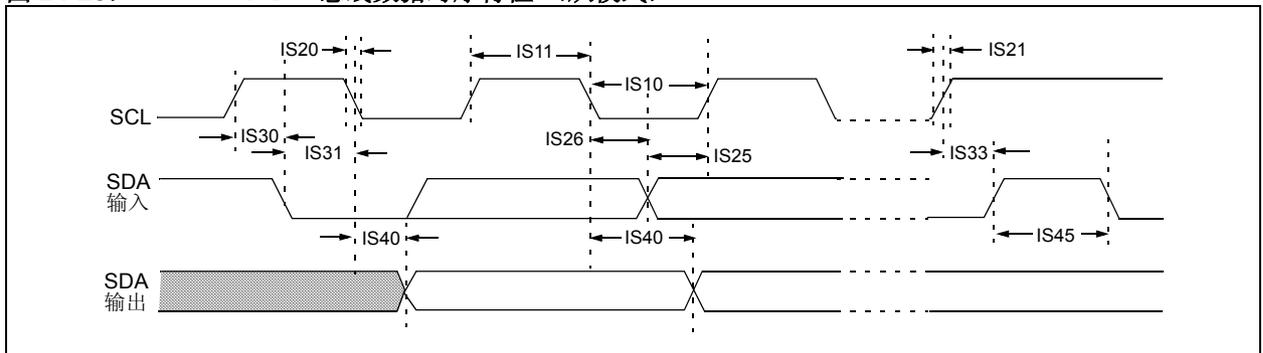


图 24-23: I²C™ 总线数据时序特性 (从模式)



dsPIC30F6010A/6015

表 24-38: I²C™ 总线数据时序要求 (从模式)

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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	器件的工作频率不得低于 1.5 MHz
			400 kHz 模式	1.3	—	μs	器件的工作频率不得低于 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	器件的工作频率不得低于 1.5 MHz
			400 kHz 模式	0.6	—	μs	器件的工作频率不得低于 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 的值介于 10 到 400 pF
			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 的值介于 10 到 400 pF
			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	仅与重复启动条件相关
			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	此周期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	自时钟边沿到输出有效的的时间	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	在开始新的发送之前总线必须处于空闲的时间
			400 kHz 模式	1.3	—	μs	
			1 MHz 模式 (1)	0.5	—	μs	
IS50	CB	总线的容性负载	—	400	pF	—	

注 1: 所有 I²C 引脚的最大引脚电容值 = 10 pF (仅适用于频率为 1 MHz 的模式)。

图 24-24: CAN 模块 I/O 时序特性

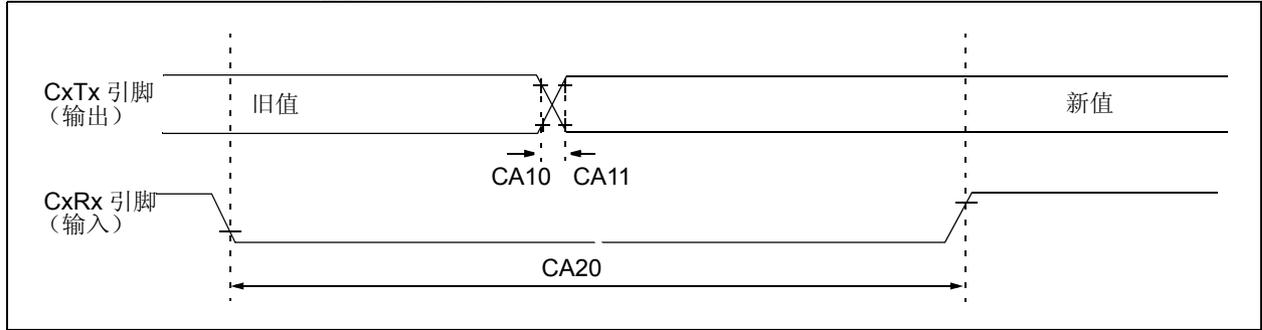


表 24-39: CAN 模块 I/O 时序要求

交流特性		标准工作条件: 2.5V 至 5.5V (除非另外声明) 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	—	—	—	ns	见参数 DO32
CA11	TioR	端口输出上升时间	—	—	—	ns	见参数 DO31
CA20	Tcwf	触发 CAN 唤醒过滤器的脉冲宽度	500			ns	—

注 1: 参数仅为特征值, 在生产时未经测试。

注 2: 除非另外声明, 否则“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。

dsPIC30F6010A/6015

表 24-40: 10 位 / 高速 A/D 模块规范 (1)

交流特性		标准工作条件: 2.7V 至 5.5V (除非另外声明) 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
器件供电电压							
AD01	AVDD	模块 VDD 电源	取 VDD - 0.3 和 2.7 中的 较大值		取 VDD + 0.3 和 5.5 中的 较小值	V	—
AD02	AVSS	模块 Vss 电源	Vss - 0.3		Vss + 0.3	V	—
参考输入							
AD05	VREFH	参考电压高电平	AVss + 2.7		AVDD	V	—
AD06	VREFL	参考电压低电平	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 3	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	—
AD12	—	泄漏电流	—	±0.001	±0.244	μA	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V 源阻抗 = 5 kΩ
AD13	—	泄漏电流	—	±0.001	±0.244	μA	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V 源阻抗 = 5 kΩ
AD17	RIN	模拟电压源的推荐阻抗	—	—	—	Ω	(见表 20-2)
直流精度							
AD20	Nr	分辨率	10 个数据位			位	—
AD21	INL	积分非线性误差 (2)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD21A	INL	积分非线性误差 (2)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD22	DNL	微分非线性误差 (2)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD22A	DNL	微分非线性误差 (2)	—	±1	±1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD23	GERR	增益误差 (2)	±1	±5	±6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 5V
AD23A	GERR	增益误差 (2)	±1	±5	±6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V

- 注 1: 参数仅为特征值, 在生产时未经测试。
 2: 测量是在使用外部 VREF+ 和 VREF- 作为 ADC 参考电压的情况下进行的。
 3: A/D 转换结果不会因输入电压的增加而减少, 并且不会丢失码。

表 24-40: 10 位 / 高速 A/D 模块规范⁽¹⁾ (续)

交流特性			标准工作条件: 2.7V 至 5.5V (除非另外声明) 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
AD24	E _{OFF}	失调误差 ⁽²⁾	±1	±2	±3	LSb	V _{INL} = AV _{SS} = V _{REFL} = 0V, AV _{DD} = V _{REFH} = 5V
AD24A	E _{OFF}	失调误差 ⁽²⁾	±1	±2	±3	LSb	V _{INL} = AV _{SS} = V _{REFL} = 0V, AV _{DD} = V _{REFH} = 3V
AD25	—	单调性 ⁽³⁾	—	—	—	—	保证
动态性能							
AD30	THD	总谐波失真	—	-64	-67	dB	—
AD31	SINAD	信噪比和失真	—	57	58	dB	—
AD32	SFDR	伪空闲动态范围	—	67	71	dB	—
AD33	F _{NYQ}	输入信号带宽	—	—	500	kHz	—
AD34	ENOB	有效位数	9.29	9.41	—	位	—

- 注 1: 参数仅为特征值, 在生产时未经测试。
 2: 测量是在使用外部 V_{REF+} 和 V_{REF-} 作为 ADC 参考电压的情况下进行的。
 3: A/D 转换结果不会因输入电压的增加而减少, 并且不会丢失码。

dsPIC30F6010A/6015

图 24-25: 10 位高速 A/D 转换时序特性
(CHPS = 01、SIMSAM = 0、ASAM = 0 且 SSRC = 000)

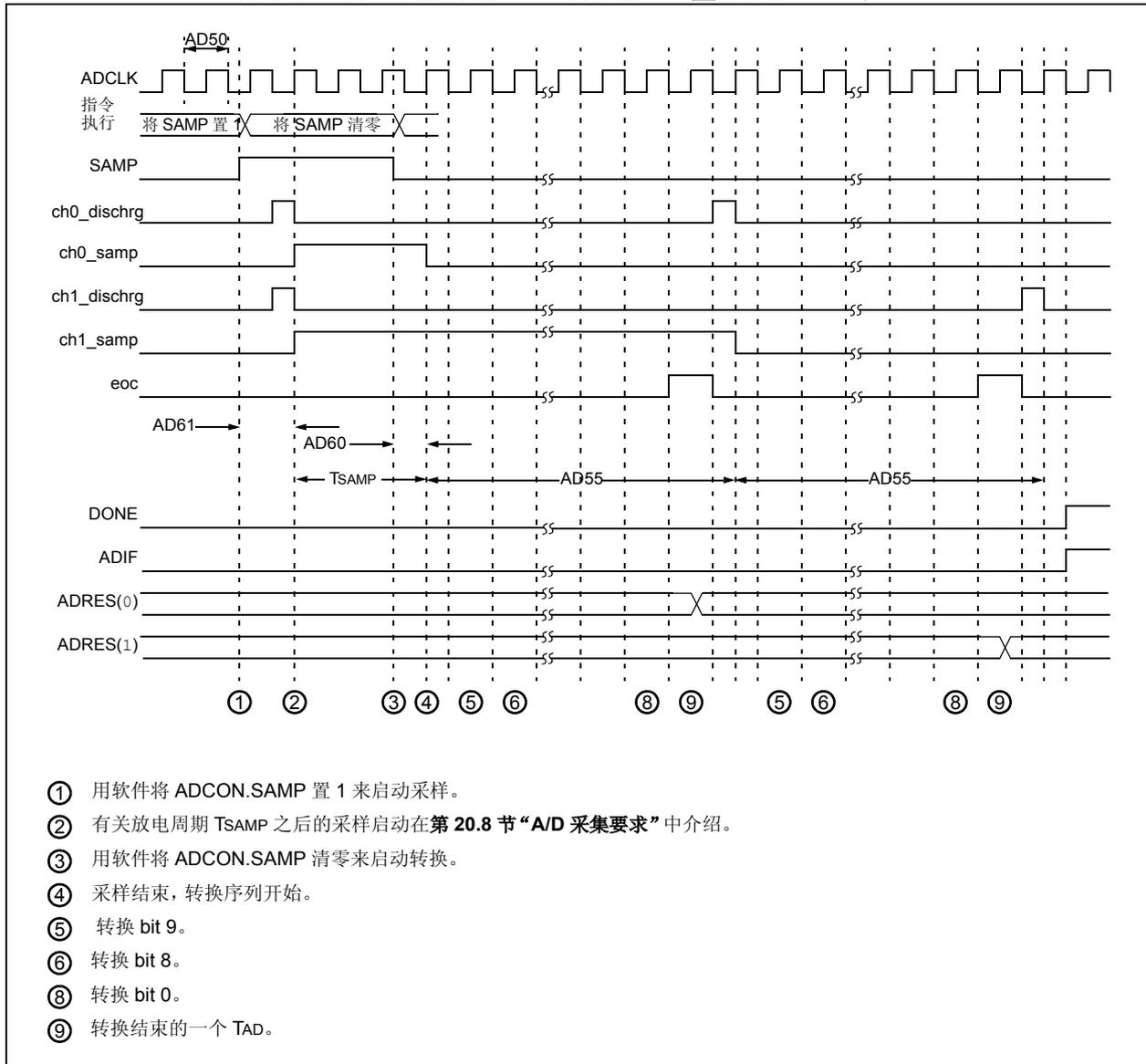
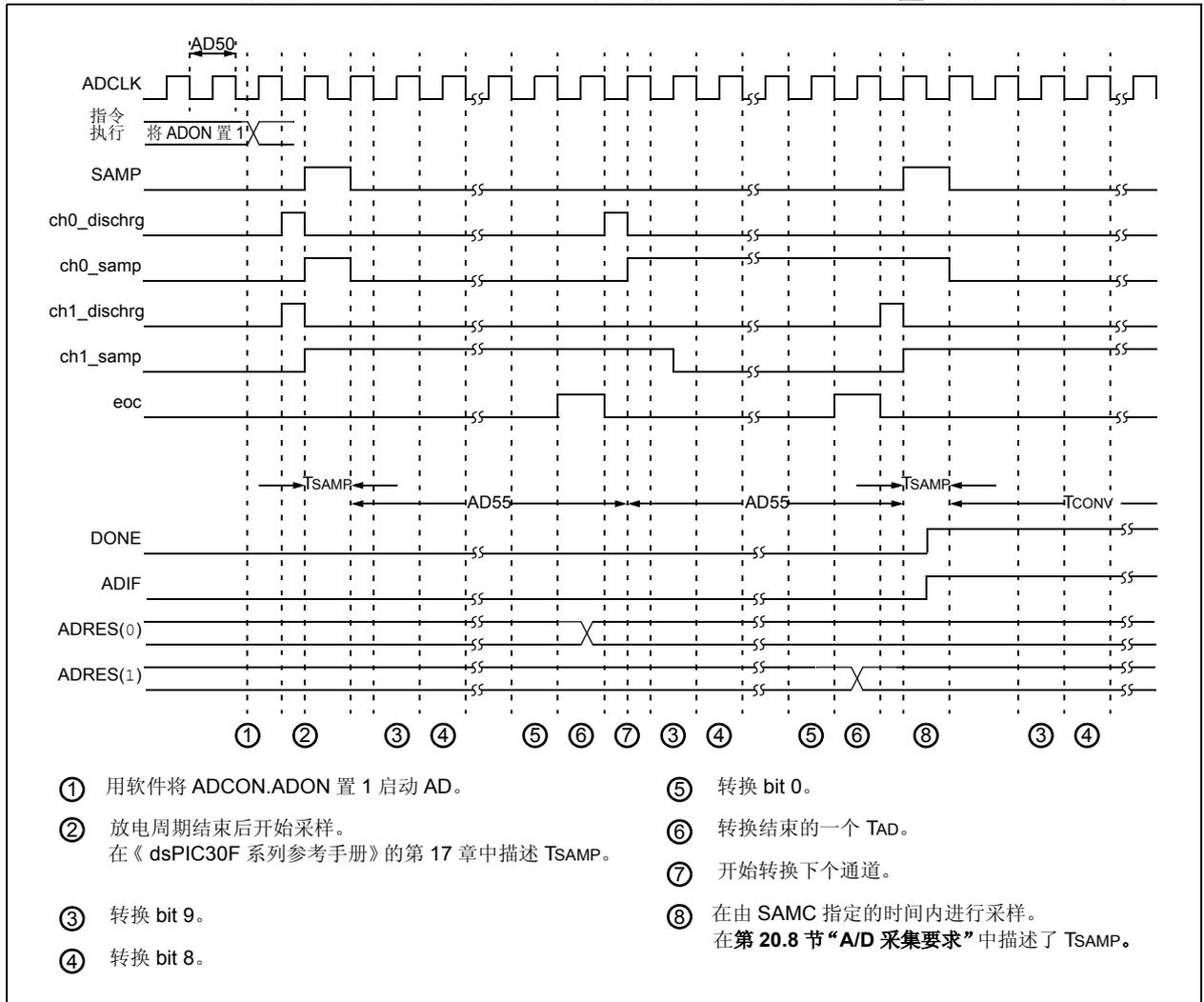


图 24-26: 10 位高速 A/D 转换时序特性
(CHPS = 01、SIMSAM = 0、ASAM = 1、SSRC = 111 且 SAMC = 00001)



dsPIC30F6010A/6015

表 24-41: 10 位高速 A/D 转换时序要求

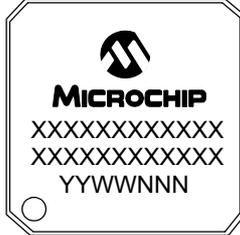
交流特性		标准工作条件: 2.7V 至 5.5V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
时钟参数							
AD50	TAD	A/D 时钟周期	—	84	—	ns	见表 20-2 ⁽²⁾ 。
AD51	tRC	A/D 内部 RC 振荡器周期	700	900	1100	ns	—
转换率							
AD55	tCONV	转换时间	—	12 TAD	—	—	—
AD56	FCNV	吞吐率	—	1.0	—	Msp/s	见表 20-2 ⁽²⁾ 。
AD57	TSAMP	采样时间	—	1 TAD	—	—	见表 20-2 ⁽²⁾ 。
时序参数							
AD60	tPCS	从触发采样到启动转换的时间 ⁽³⁾	—	1.0 TAD	—	—	没有选择自动转换触发 (SSRC = 111)
AD61	tPSS	从采样位 (SAMP) 置 1 到开始采样的时间	0.5 TAD	—	1.5 TAD	—	—
AD62	tCSS	转换结束至采样启动 (ASAM = 1) 的时间 ⁽³⁾	—	0.5 TAD	—	—	—
AD63	tDPU	从 A/D 关闭到 A/D 开始工作使模拟级稳定的时间 ⁽³⁾	—	20	—	μs	—

- 注 1: 参数仅为特征值, 在生产时未经测试。
 2: 因为采样电容最终将无法保持电荷, 因此低于 10 kHz 的时钟频率可能影响线性性能, 尤其是在温度较高时。
 3: 特征值, 仅供设计参考, 未经测试。

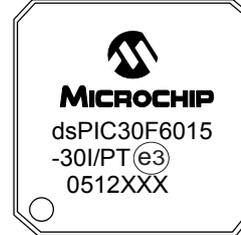
25.0 封装信息

25.1 封装标识信息

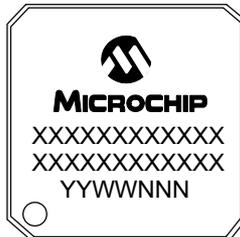
64 引脚 TQFP



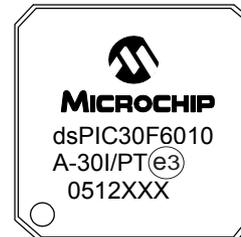
示例



80 引脚 TQFP



示例



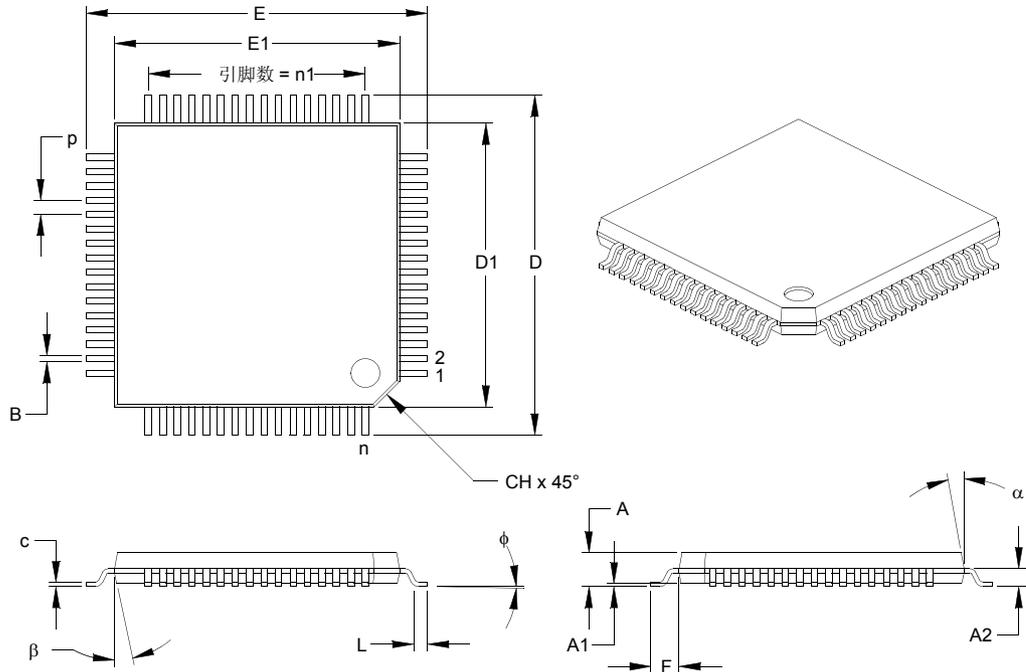
图注: XX...X 客户信息
Y 年份代码（日历年的最后一位数字）
YY 年份代码（日历年的最后两位数字）
WW 星期代码（一月一日的星期代码为“01”）
NNN 以字母数字排序的追踪代码
(e3) 雾锡（Matte Tin, Sn）的 JEDEC 无铅标志
* 此封装为无铅封装。JEDEC 无铅标志（(e3)）
标示于此种封装的外包装上。

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。

dsPIC30F6010A/6015

64 引脚塑封薄型正方扁平封装 主体 10x10x1 mm， 1.0/0.10 mm 引脚形式 (TQFP)

注： 最新的封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



尺寸范围	单位	英寸			毫米*		
		最小	正常	最大	最小	正常	最大
引脚数	n		64			64	
引脚间距	p		.020			0.50	
每侧引脚数	n1		16			16	
总高度	A	.039	.043	.047	1.00	1.10	1.20
塑模封装厚度	A2	.037	.039	.041	0.95	1.00	1.05
悬空间隙	A1	.002	.006	.010	0.05	0.15	0.25
底脚长度	L	.018	.024	.030	0.45	0.60	0.75
底脚占位	F	.039 REF.			1.00 REF.		
底脚倾角	ϕ	0°	3.5°	7°	0°	3.5°	7°
总宽度	E	.463	.472	.482	11.75	12.00	12.25
总长度	D	.463	.472	.482	11.75	12.00	12.25
塑模封装宽度	E1	.390	.394	.398	9.90	10.00	10.10
塑模封装长度	D1	.390	.394	.398	9.90	10.00	10.10
引脚厚度	c	.005	.007	.009	0.13	0.18	0.23
引脚宽度	B	.007	.009	.011	0.17	0.22	0.27
引脚 1 处角斜面	CH	.025	.035	.045	0.64	0.89	1.14
塑模顶部锥度	a	5°	10°	15°	5°	10°	15°
塑模底部锥度	b	5°	10°	15°	5°	10°	15°

* 控制参数

注：

尺寸 D 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸 (0.254 毫米)。

REF: 参考尺寸, 通常无公差, 仅供参考。

参见 ASME Y14.5M

等同于 JEDEC 号: MS-026

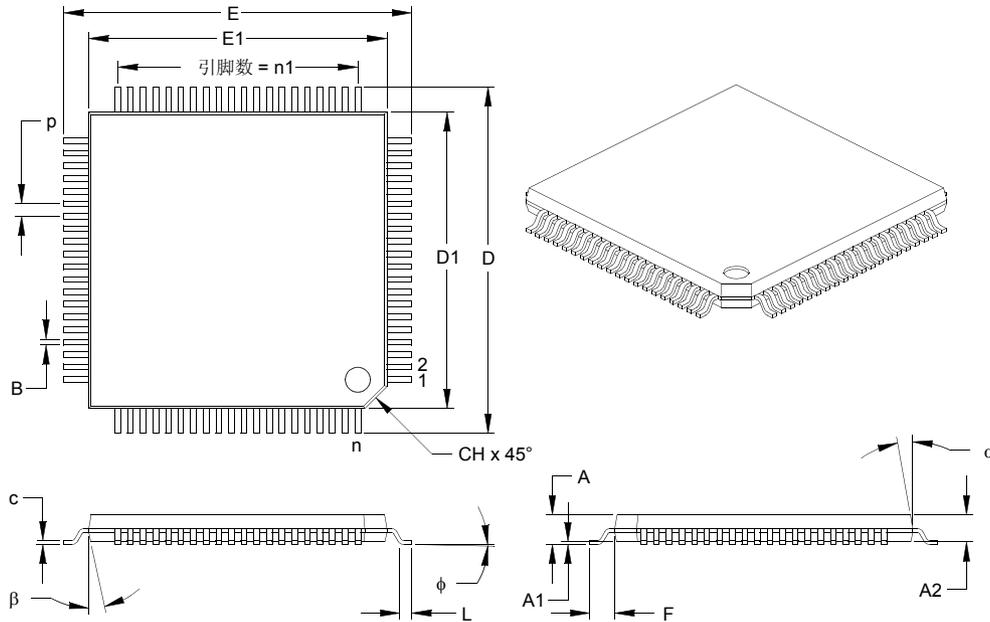
图号: C04-085

修订于 07-22-05

dsPIC30F6010A/6015

80 引脚塑封薄型正方扁平封装 主体 12x12x1 mm， 1.0/0.10 mm 引脚形式 (TQFP)

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



单位		英寸			毫米*		
尺寸范围		最小	正常	最大	最小	正常	最大
引脚数	n		80			80	
引脚间距	p	.020 BSC			.50 BSC		
每侧引脚数	n1	20			20		
总高度	A	.039	.043	.047	1.00	1.10	1.20
塑模封装厚度	A2	.037	.039	.041	0.95	1.00	1.05
悬空间隙	A1	.002	.004	.006	0.05	0.10	0.15
底脚长度	L	.018	.024	.030	0.45	0.60	0.75
底脚占位	F	.039 REF.			1.00 REF.		
底脚倾角	φ	0°	3.5°	7°	0°	3.5°	7°
总宽度	E	.551 BSC			14.00 BSC		
总长度	D	.551 BSC			14.00 BSC		
塑模封装宽度	E1	.472 BSC			12.00 BSC		
塑模封装长度	D1	.472 BSC			12.00 BSC		
引脚厚度	c	.004	.006	.008	0.09	0.15	0.20
引脚宽度	B	.007	.009	.011	0.17	0.22	0.27
引脚 1 处角斜面	CH	.025	.035	.045	0.64	0.89	1.14
塑模顶部锥度	α	5°	10°	15°	5°	10°	15°
塑模底部锥度	β	5°	10°	15°	5°	10°	15°

* 控制参数

注

尺寸 D1 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸 (0.254 毫米)。

BSC: 基本尺寸。给出的是理论上的精确值, 没有公差。

参见 ASME Y14.5M

REF: 参考尺寸, 通常无公差, 仅供参考。

参见 ASME Y14.5M

等同于 JEDEC 号: MS-026

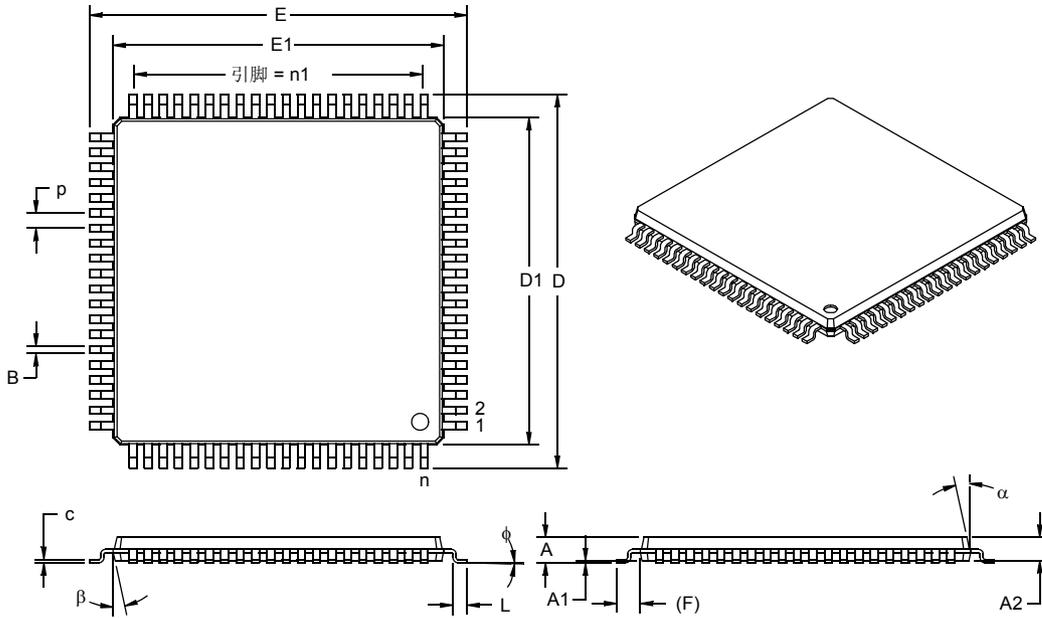
图号: C04-092

修订于 07-22-05

dsPIC30F6010A/6015

80 引脚塑封薄型正方扁平封装 主体 14x14x1 mm， 1.0/0.10 mm 引脚形式 (TQFP)

注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



尺寸范围	单位	英寸			毫米*		
		最小	正常	最大	最小	正常	最大
引脚数	n		80			80	
引脚间距	p		.026			0.65	
每侧引脚数	n1		20			20	
总高度	A			.047			1.20
塑模封装厚度	A2	.037	.039	.041	0.95	1.00	1.05
悬空间隙	A1	.002		.006	0.05		0.15
底脚长度	L	.018	.024	.030	0.45	0.60	0.75
底脚占位	F		.039 REF.			1.00 REF.	
底脚倾角	φ	0°	3.5°	7°	0°	3.5°	7°
总宽度	E		.630 BSC			16.00 BSC	
总长度	D		.630 BSC			16.00 BSC	
塑模封装宽度	E1		.551 BSC			14.00 BSC	
塑模封装长度	D1		.551 BSC			14.00 BSC	
引脚厚度	c	.004		.008	0.09		0.20
引脚宽度	B	.011	.013	.015	0.27	0.32	0.37
塑模顶部锥度	α	11°	12°	13°	11°	12°	13°
塑模底部锥度	β	11°	12°	13°	11°	12°	13°

* 控制参数

注

尺寸 D1 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸 (0.254 毫米)。

BSC: 基本尺寸。给出的是理论上的精确值, 没有公差。

参见 ASME Y14.5M

REF: 参考尺寸, 通常无公差, 仅供参考。

参见 ASME Y14.5M

等同于 JEDEC 号: MS-026

图号: C04-116

修订于 7-20-06

附录 A: 版本历史

版本 A (2005 年 7 月)

dsPIC30F6010A 和 dsPIC30F6015 器件的原始数据手册。

版本 B (2006 年 9 月)

此版本进行了如下更新:

- 数据 RAM 保护功能保证了 RAM 段与引导段和安全代码段同时使用时的安全 (见第 3.2.7 节“数据 RAM 保护功能”)
- 增加了 BSRAM 和 SSRAM SFR 以支持数据 RAM 保护功能 (见表 3-3)
- 移除了基本指令 CP1 (见表 22-2)
- 支持 I²C 从地址 (见表 17-2)
- 修订了电气特性:
 - 工作电流 (IDD) 规范 (见表 24-6)
 - 空闲电流 (I_{IDLE}) 规范 (见表 24-7)
 - 掉电电流 (IPD) 规范 (见表 24-8)
 - I/O 引脚输入规范 (见表 24-9)
 - BOR 电压限制 (见表 24-11)
 - 看门狗定时器超时限制 (见表 24-21)
- 添加了封装图的注。

dsPIC30F6010A/6015

注:

附录 B: 器件比较

本附录列出了 dsPIC30F6010 和 dsPIC30F6010A 之间的差异。表 B-1 突出显示了这些差异。

表 B-1: dsPIC30F6010 和 dsPIC30F6010A 的比较

特性	dsPIC30F6010	dsPIC30F6010A
程序存储器容量 (KB)	144	144
数据存储器容量 (字节)	8192	8192
数据 EEPROM 容量 (字节)	4096	4096
定时器数 (16 位)	5	5
输入捕捉通道数	8	8
输入变更通知通道数	22	22
输出比较 / 标准 PWM 通道数	8	8
电机控制 PWM 通道数	8	8
A/D 10 位 500 ksps	16	16
正交编码器	有	有
UART	2	2
SPI	2	2
I ² C™	1	1
CAN	1	1
PGC (ICSP™ 时钟输入)	RB1/AN1	RB1/AN1
PGC (ICSP 数据输入 / 输出)	RB0/AN0	RB0/AN0
封装	80 引脚 TQFP (14x14x1 mm)	80 引脚 TQFP (14x14x1 mm) 80 引脚 TQFP (12x12x1 mm)
I/O 引脚数 (最大值)	68	68

dsPIC30F6010A/6015

注:

附录 C: 从 dsPIC30F6010 移植到 dsPIC30F6010A

dsPIC30F6010A 器件取代了 dsPIC30F6010 器件。“A”版硅片在各方面都与 dsPIC30F6010 器件基本相同。请查阅 Microchip 网站上发布的最新 dsPIC30F6010A/6015 硅片勘误表文档，了解其针对 *dsPIC30F6010 Rev. B2* (DS80195B) 硅片勘误表文档进行了哪些修正。

本附录列出了 dsPIC30F6010 和 dsPIC30F6010A 器件之间的差异。

C.1 代码兼容性

大多数情况下，使用 MPLAB C30 C 编译器和 MPLAB ASM30 汇编器开发的 dsPIC30F 代码可直接在 dsPIC30F6010 和 dsPIC30F6010A 器件之间移植。但在某些情况下，要根据增强了哪些功能进行一些小的改动。要将源代码移植到 dsPIC30F6010A 器件，需要执行以下两步：

1. 在源代码文件中包含 dsPIC30F6010A 器件的相应头文件 (.h) 和包含文件 (.inc)。
2. 删除源代码或命令行选项中针对 “*dsPIC30F6010 Rev. B2 Silicon Errata*” (DS80195B) 文档中所述勘误实现的任何替代方案。

本部分中未进一步讲述其他差异。有关其他差异的信息，请阅读 Microchip 网站上发布的最新 dsPIC30F6010A/6015 硅片勘误表文档。

C.2 振荡器操作

dsPIC30F6010A 器件新增了一些振荡器模式：

- FRC x4、x8 和 x16 PLL 模式
- HS/2 x4、x8 和 x16 PLL 模式
- HS/3 x4、x8 和 x16 PLL 模式

第 21.0 节 “系统集成” 中描述了这些新增的振荡器模式。为了支持 FRC，增加了 OSCTUN 特殊功能寄存器 (Special Function Register, SFR)。此寄存器包括四个 FRC 调节位 (TUN<5:0>)。关于此新功能，没有代码或操作兼容性问题。

另一个重要变化与 OSCCON SFR 有关。为 COSC 和 NOSC 功能定义增加了位。对于 OSCCON SFR，可使用 dsPIC30F6010A 器件头文件和包含文件来实现代码移植。

注： 需要验证振荡器操作以确保其起振和运行与预期相符。您可能需要调节负载电容值和 / 或振荡器模式以获得适当的值。

C.3 器件配置寄存器

为支持新的振荡器模式，修改了 FOSC 配置寄存器。dsPIC30F6010A 器件头文件和包含文件为器件配置寄存器提供了更新的位域定义和宏。当在 C 和汇编源文件中嵌入特定器件配置信息时，请参考这些新的宏。

C.4 电气特性

对于工业级温度范围，dsPIC30F6010A 器件的额定工作速度是 30 MIPS @ 4.5-5.5 V_{DC} 和 20 MIPS @ 3.0-3.6 V_{DC}。即便兼容器件经过测试表明有相同的电气规范，由于正常的工艺差异，器件特性也可能会有变化。这些差异不应设计于器件规范之内的系统造成影响。对于接近或超出规范限制的系统，生产上的差异可能导致器件工作的不同。

C.5 器件封装

dsPIC30F6010A 器件提供下列 TQFP 封装：

- 80 引脚 TQFP 12x12x1mm (新设计)
- 80 引脚 TQFP 14x14x1mm (支持移植)

欲知这些封装的详细信息，请参见第 25.0 节 “封装信息”。

dsPIC30F6010A/6015

注:

索引

数字

16 位递增 / 递减位置计数器模式	88
错误检查	88
计数方向状态	88

A

A/D

1 Msps 时的配置指南	140
10 位高速模数转换器模块	135
600 ksps 时的配置指南	141
750 ksps 时的配置指南	141
ADCHS	135
ADCON1	135
ADCON2	135
ADCON3	135
ADCSSL	135
ADPCFG	135
通过编程触发转换	138
CPU 空闲模式下的操作	143
CPU 休眠模式中的操作	143
采集要求	142
参考电压原理图	140
掉电模式	143
复位的影响	143
寄存器映射	145
结果缓冲器	137
连接注意事项	144
配置模拟端口引脚	144
输出格式	143
选择转换时钟	138
选择转换顺序	137
中止转换	138
转换操作	137
转换速度	139
转换速率参数	139

B

BOR。见欠压复位。

版本历史	219
备用 16 位定时器 / 计数器	89
备用向量表	43
编程操作	49
擦除程序存储器的一行	49
对闪存程序存储器编程的算法	49
启动编程序列	50
装载写锁存器	50
编程模型	14
图	15
变更通知客户服务	231
边沿对齐 PWM	96

C

CAN

报文发送	127
发送错误	127
发送缓冲器	127
发送中断	128
序列	127
优先级	127
中止	127
报文接收	126
接收错误	126
接收过滤器	126

接收过滤器屏蔽寄存器	126
接收缓冲器	126
接收溢出	126
接收中断	126
波特率设置	128
采样点	129
传播段	129
同步	129
位时序	128
相位段	129
预分频器	129

dsPIC30F6010A/6015 的

CAN1 寄存器映射	130
dsPIC30F6010A 的	
CAN2 寄存器映射	132
概述	123
工作模式	125
初始化	125
错误识别	125
环回	125
监听模式	125
禁止	125
正常	125
帧类型	123

CAN 模块

C 编译器

MPLAB C18	172
MPLAB C30	172

CPU 架构概述

CPU 空闲模式下的 PWM 操作

CPU 休眠模式下的 PWM 操作

操作码描述中使用的符号

程序地址空间

表指令

TBLRDH	23
TBLRDL	23
TBLWTH	23
TBLWTL	23

存储器映射

构成

使用表指令访问程序存储器中的数据

数据访问，地址生成方式

程序计数器

程序空间可视性

程序空间窗口可视操作

程序数据表访问

除法支持

从 dsPIC30F6010 到 dsPIC30F 6010A 的移植

从休眠和空闲模式唤醒

从休眠中唤醒

存储器构成

内核寄存器映射

D

dsPIC30F6010A 端口寄存器映射

dsPIC30F6015 端口寄存器映射

DSP 引擎

乘法器

代码保护

代码示例

擦除程序存储器的一行

擦除数据 EEPROM 块

擦除数据 EEPROM 的一个字

读数据 EEPROM

dsPIC30F6010A/6015

端口写 / 读示例	58	主模式支持	111
启动编程序列	50	I ² C 主模式下的工作	
数据 EEPROM 块写操作	56	发送	111
写数据 EEPROM 的一个字	55	波特率发生器	111
装载写锁存器	50	多主器件通信、总线冲突与总线仲裁	112
单脉冲 PWM 操作	99	接收	111
地址发生器单元	33	时钟仲裁	112
电机控制 PWM 模块	93	J	
8 输出寄存器映射	102	简单捕捉事件模式	
电气特性	175	捕捉缓冲器操作	80
BOR	183	捕捉预分频器	79
绝对最大额定值	175	霍尔传感器模式	80
独立的 PWM 输出	99	Timer2 和 Timer3 选择模式	80
读者反馈表	232	简单 PWM 模式	84
F		输入引脚故障保护	84
防止意外写入 OSCCON	153	周期	85
封装信息	215	简单输出比较匹配模式	84
标识	215	交流特性	184
复位	147, 153	负载电路	184
FSCM 和 PWRT 禁止时的 POR 工作	155	内部 FRC 抗抖动性、准确性和漂移	188
具有长晶振起振时间的 POR	155	内部 LPRC 精度	188
欠压复位 (BOR), 可编程	155	温度和电压规范	184
上电复位 (POR)	154	节能模式	158
复位过程	41	休眠	158
复位源	41	节能模式 (休眠和空闲)	147
G		K	
公式		开发支持	171
A/D 转换时钟	138	看门狗定时器 (WDT)	147, 158
波特率	119	操作	158
串行时钟速率	112	使能和禁止	158
PWM 分辨率	96	勘误表	5
PWM 周期	96	可编程	147
时钟发生器的时间份额	129	可编程数字噪声滤波器	89
工作电流 (IDD)	178	客户通知服务	231
H		客户支持	231
互补 PWM 操作	98	空闲电流 (IDLE)	179
汇编器		框图	
MPASM 汇编器	172	dsPIC30F6010A 的 32 位 Timer2/3	68
I		10 位高速 A/D 功能	136
I/O 端口	57	16 位 Timer4 模块 (B 类定时器)	76
并行 I/O (PIO)	57	16 位 Timer3 (C 类定时器)	71
I ² C 10 位从模式的工作	109	16 位 Timer5 模块 (C 类定时器)	76
10 位地址从模式发送	110	16 位 Timer1 模块 (A 类定时器)	64
10 位地址从模式接收	110	32 位 Timer4/5	75
I ² C 7 位从模式的工作	109	CAN 缓冲器和协议引擎	124
发送	109	dsPIC30F6010A	8
接收	109	dsPIC30F6010A 的 16 位 Timer2 (B 类定时器)	70
I ² C TM 模块	107	dsPIC30F6015	9
编程模型	107	dsPIC30F6015 的 16 位 Timer2 (B 类定时器)	70
地址	109	dsPIC30F6015 的 32 位 Timer2/3	69
各种模式	107	DSP 引擎	17
功能说明	107	复位系统	153
广播呼叫地址支持	111	I ² C	108
寄存器	107	PWM 模块	94
寄存器映射	113	SPI	104
IPMI 支持	111	SPI 主 / 从连接	104
软件控制的时钟延长 (STREN = 1)	110	输出比较模式	83
斜率控制	111	输入捕捉模式	79
引脚配置	107	UART 发送器	115
在 CPU 休眠和空闲模式下的工作	112	UART 接收器	116
中断	111	外部上电复位电路	156
主模式下的工作	111	振荡器系统	149
		正交编码器接口	87

M

Microchip 因特网网站	231
MPLAB ASM30 汇编器、链接器和库管理器	172
MPLAB ICD 2 在线调试器	173
MPLAB ICE 2000 高性能通用在线仿真器	173
MPLAB ICE 4000 高性能通用在线仿真器	173
MPLAB PM3 器件编程器	173
MPLAB 集成开发环境软件	171
MPLINK 目标链接器 / MPLIB 目标库管理器	172
模式寻址	34
操作示例	35
起始地址和结束地址	35
W 地址寄存器选择	35
应用	36

N

内核概述	13
------------	----

P

PICSTART 2 开发编程器	174
PICSTART Plus 开发编程器	174
POR. 见上电复位。	
PWM 故障引脚	100
PWM 更新锁定	101
PWM 故障引脚	
故障状态	100
使能位	100
输入模式	100
锁存	100
逐周期	100
优先级	100
PWM 时基	95
单事件模式	95
后分频器	96
连续递增 / 递减计数模式	95
双更新模式	96
预分频器	96
自由运行模式	95
PWM 输出改写	99
互补输出模式	99
同步	99
PWM 输出和极性控制	100
输出引脚控制	100
PWM 特殊事件触发器	101
后分频器	101
PWM 占空比较单元	97
占空比寄存器缓冲器	97
占空比立即更新	97
PWM 周期	96
配置模拟端口引脚	58

Q

器件比较	221
器件概述	7
器件 ID 单元	147
器件配置	
寄存器映射	161
器件配置寄存器	159
FBORPOR	159
FGS	159
FOSC	159
FWDT	159
欠压复位 (BOR)	147

R

RCON 寄存器的初始化条件	
情形 1	157
情形 2	157
RTSP 控制寄存器	48
NVMADR	48
NVMADRU	48
NVMCON	48
NVMKEY	48
软件堆栈指针 / 帧指针	14
CALL 堆栈帧	29
软件控制的时钟延长 (STREN = 1)	110
软件模拟器 (MPLAB SIM)	172

S

SPI 模块	103
CPU 空闲模式下的操作	105
CPU 休眠模式下的操作	105
功能说明	103
从选择同步	105
SDOx 禁止	103
SPI1 寄存器映射	106
SPI2 寄存器映射	106
帧 SPI 支持	105
字和字节通信	103
闪存程序存储器	47
表指令操作汇总	47
运行时自编程 (RTSP)	47
在线串行编程 (ICSP)	47
上电复位 (POR)	147
上电延时定时器 (PWRT)	147
振荡器起振定时器 (OST)	147
时序规范	
10 位高速 A/D	210
10 位高速 A/D 转换要求	214
CAN I/O 时序要求	209
CLKOUT 和 I/O 特性	189
CLKOUT 和 I/O 要求	189
带隙启动时间要求	191
电机控制 PWM 要求	197
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	191
I ² C 总线数据时序要求 (从模式)	208
I ² C 总线数据时序要求 (主模式)	206
简单 OC/PWM 模式时序要求	196
内部时钟时序示例	187
PLL 抖动	186
PLL 时钟	186
QE1 索引脉冲时序要求	199
QE1 外部时钟要求	194
SPI 从模式 (CKE = 0) 要求	202
SPI 从模式 (CKE = 1) 要求	203
SPI 主模式 (CKE = 0) 要求	200
SPI 主模式 (CKE = 1) 要求	201
输出比较要求	195
Timer2 和 Timer4 外部时钟时序要求	193
Timer3 和 Timer5 外部时钟时序要求	193
Timer1 外部时钟要求	192
外部时钟时序要求	185
正交编码器时序要求	198

dsPIC30F6010A/6015

时序图

10 位高速 A/D 转换 (CHPS = 01, SIMSAM = 0, ASAM = 1, SSR = 111, SAMC = 00001)	213
10 位高速 A/D 转换 (CHPS = 01, SIMSAM = 0, ASAM = 0, SSR = 000)	212
边沿对齐 PWM	96
CAN I/O	209
CAN 位	128
带隙启动时间	191
电机控制 PWM	197
电机控制 PWM 故障	197
复位、看门狗定时器、振荡器起振定时器和上电延时定时器	190
I ² C 总线启动 / 停止位时序特性 (主模式)	205
I ² C 总线启动 / 停止位时序特性 (从模式)	207
I ² C 总线数据 (从模式)	207
I ² C 总线数据 (主模式)	205
OC/PWM	196
PWM 输出	85
QEA/QEB 输入	198
QEI 模块索引脉冲	199
SPI 从模式 (CKE = 0)	202
SPI 从模式 (CKE = 1)	203
SPI 主模式 (CKE = 0)	200
SPI 主模式 (CKE = 1)	201
上电延时序列 (MCLR 连接至 V _{DD})	154
上电延时序列 (MCLR 未连接至 V _{DD}), 情形 1	154
上电延时序列 (MCLR 未连接至 V _{DD}), 情形 2	155
输出比较 (OCx)	195
输入捕捉 (CAPx)	195
死区时间	99
TimerQ (QEI 模块) 外部时钟	194
Timer1、2、3、4 和 5 外部时钟	192
外部时钟	184
中心对齐 PWM	97
时序要求	
输入捕捉	195
使用程序空间可视性访问程序存储器中的数据	24
输出比较模块	83
CPU 空闲模式下的操作	85
CPU 休眠模式下的操作	85
寄存器映射	86
中断	85
数据地址空间	25
存储器映射	25, 26
对齐	28
对齐 (图)	28
空间	28
宽度	28
MCU 和 DSP (MAC 类) 指令示例	27
Near 数据空间	29
软件堆栈	29
无效的存储器访问结果	28
数据 EEPROM 存储器	53
擦除	54
擦除, 块	54
擦除, 字	54
读	53
防止误写	56
写	55
写校验	56
写, 块	56
写, 字	55
数据累加器和加法器 / 减法器	18
回写	19

舍入逻辑	19
数据空间写饱和	20
溢出和饱和	18
输入捕捉模块	79
寄存器映射	81
简单捕捉事件模式	79
休眠和空闲模式下的工作	80
中断	80
输入状态变化通知模块	61
寄存器映射 (bit 15-8)	61
寄存器映射 (bit 7-0, dsPIC30F6010A)	61
寄存器映射 (bit 7-0, dsPIC30F6015)	61
双输出比较匹配模式	84
单脉冲模式	84
连续脉冲模式	84
死区时间发生器	98
范围	98
分配	98
选择位	98

T

Timer2/3 模块	67
32 位定时器模式	67
32 位同步计数器模式	67
ADC 事件触发信号	72
定时器预分频器	72
寄存器映射	73
门控操作	72
休眠模式下的定时器操作	72
中断	72
Timer4/5 模块	75
寄存器映射	77
Timer2 和 Timer3 选择模式	84
Timer1 模块	63
16 位定时器模式	63
16 位同步计数器模式	63
16 位异步计数器模式	63
寄存器映射	66
门控操作	64
实时时钟	65
振荡器工作	65
中断	65
休眠模式下的定时器操作	64
预分频器	64
中断	65
桶形移位器	20
通用异步收发器 (UART) 模块	115

U

UART

波特率发生器 (BRG)	119
地址检测模式	119
发送数据	117
8 位数据模式下	117
9 位数据模式下	117
发送缓冲器 (UxTXB)	117
发送间隔字符	118
中断	118
环回模式	119
接收错误处理	118
接收缓冲器溢出错误 (OERR 位)	118
接收间隔字符	119
空闲状态	119
奇偶校验错误 (PERR)	119
帧错误 (FERR)	119
接收数据	118

接收缓冲器 (UxRXB)	118	其他指令	34
在 8 位或 9 位数据模式下接收	118	文件寄存器指令	33
中断	118	支持的基本寻址模式	33
禁止	117	传送指令和累加器指令	34
模块概述	115	直流特性	176
设定数据、奇偶校验以及停止位选择	117	程序和 EEPROM	183
使能和设置	117	dsPIC30F6010A/6015 的热工作条件	176
UART1 寄存器映射	121	dsPIC30F6010A 的工作速度 (MIPS) 与电压	176
UART2 寄存器映射	121	dsPIC30F6015 的工作速度 (MIPS) 与电压	176
在 CPU 休眠和空闲模式下的工作	120	掉电电流 (IPD)	180
自动波特率支持	120	工作电流 (IDD)	178
W		I/O 引脚输出规范	182
WWW, 在线支持	5	空闲电流 (IDLE)	179
WWW 地址	231	欠压复位	182
外部中断请求	43	热封装特性	176
外设模块禁止 (PMD) 寄存器	160	中断	39
位反转寻址	36	中断过程	43
示例	36	中断堆栈帧	43
实现	36	中断控制器	
修改值表	37	寄存器映射 (dsPIC30F6010A)	44
序列表 (16 个位)	37	寄存器映射 (dsPIC30F6015)	45
位置测量模式	88	中断优先级	40
X		中心对齐 PWM	97
系统集成	147	状态寄存器	14
寄存器映射 (dsPIC30F6010A)	161	自动时钟延长	110
寄存器映射 (dsPIC30F6015)	161	10 位寻址模式 (STREN = 1)	110
陷阱	41	7 位寻址模式 (STREN = 1)	110
向量	42	发送模式	110
硬陷阱和软陷阱	42	接收模式	110
源	41		
Y			
引脚说明	10		
引脚图	3-4		
因特网地址	231		
Z			
在线串行编程 (ICSP)	147		
在线调试器 (ICD 2)	160		
振荡器			
工作模式 (表)	148		
系统概述	147		
振荡器配置	150		
初始时钟源选择	150		
低功耗 RC 振荡器 (LPRC)	152		
故障保护时钟监视器	152		
快速 RC (FRC)	151		
LP 振荡器控制	151		
起振定时器 (OST)	151		
锁相环 (PLL)	151		
振荡器选择	147		
正交编码器接口 (QEI) 模块	87		
CPU 空闲模式下的操作	89		
CPU 空闲模式下的定时器操作	90		
CPU 休眠模式下的操作	89		
CPU 休眠模式下的定时器操作	89		
寄存器映射	91		
逻辑	88		
中断	90		
指令集			
汇总	163, 166		
指令寻址模式	33		
MAC 指令	34		
MCU 指令	33		

dsPIC30F6010A/6015

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、应用笔记和样本程序、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, 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
- 开发系统信息热线

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://support.microchip.com> 获得网上技术支持。

dsPIC30F6010A/6015

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: dsPIC30F6010A/6015 文献编号: DS70150B_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?

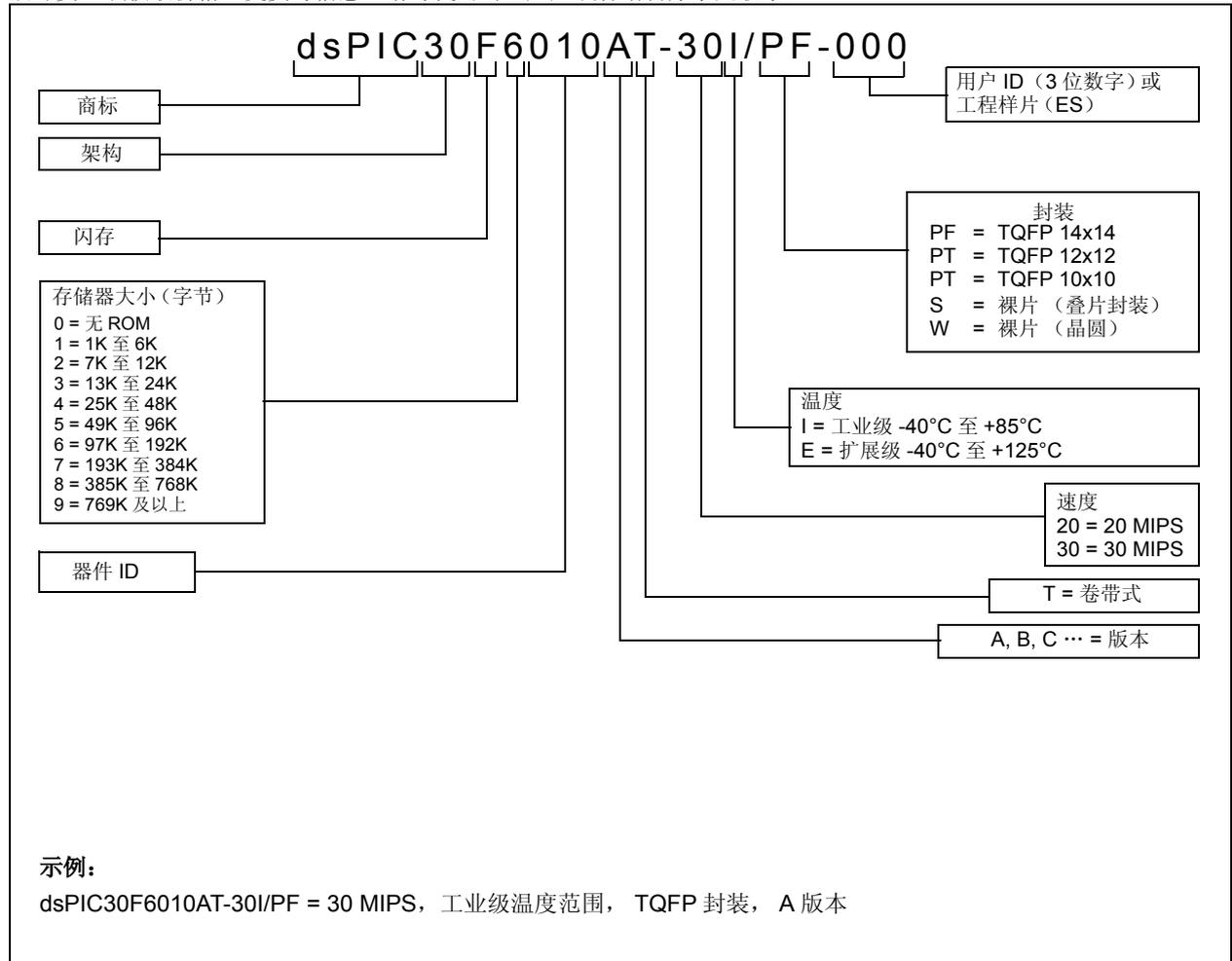
5. 您认为本文档中可以删减哪些内容, 而又不会影响整体使用效果?

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各销售办事处联系。





MICROCHIP

全球销售及服务网点

06/25/07